Information Sciences 375 (2017) 30–47

Contents lists available at ScienceDirect

Information Sciences

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

freepaper.me FREE paper

Multi-objective particle swarm optimization based on global margin ranking

Li Li, Wanliang Wang*, Xinli Xu

College of Computer Science and Technology, Zhejiang University of Technology, Hangzhou, 310023, PR China

ARTICLE INFO

Article history:
Received 15 January 2016
Revised 30 July 2016
Accepted 14 August 2016
Available online 16 August 2016

Keywords:
Global margin ranking
MOPSO
Elitist selection
Swarm intelligence

ABSTRACT

It is prevalent that the Pareto-based dominant framework is inefficient in non-dominated sorting because the performance sharply deteriorates when there are numerous weak dominance relations. In order to address this issue, the paper presents a novel ranking strategy called Global Margin Ranking (GMR) which deploys the position information of individuals in objective space to gain the margin of dominance throughout the population. The method not only considers the distribution of population, but also incorporates the associated information of individuals, without incurring user-defined parameters. Moreover, in view of the challenges faced by Multi-objective Particle Swarm Optimization (MOPSO) in selection of *gBest* and *pBest*, we present an innovative strategy for selecting *gBest* and *pBest* by integrating the GMR and the individual's density information. We compare the GMR method with a variety of other ranking methods in terms of the distribution of ranks, the ranking landscape and convergence of the evolutionary process. The relatively extensive experimental results on some benchmark functions show that MOPSO/GMR performs better than those specialized MOEAs.

© 2016 Published by Elsevier Inc.

1. Introduction

Particle swarm optimization (PSO) [23] is one of the most well-known population-based metaheuristics which imitates the behavior of a flock of birds foraging in nature. In order to locate the regions of promising solutions, the particles in the swarm fly through the search space by interacting with each other to detect the global and personal information called *gBest* and *pBest* that direct their motion. The PSO with a single objective has a simple but a speedy convergence ability applied extensively in optimization and scheduling, and acquires approving results. Whereas there always exist multi-objectives in scientific research and practical engineering [24,38,44], it is usually tough or even unable to get a desired solution with conventional mathematical programming approaches because of the multi-objective problems (MOPs) either mutually restricted or having a discrete or non-convex optimal.

Then, the application of PSO for MOP has received increased attention. Coello [10] proposed MOPSO by bringing the PSO into multi-objective optimization for the first time. By drawing on the concept of decomposition, Al Moubayed [2] developed a D^{2}MOPSO, and Cai et al. [13] presented a decomposition-based MOPSO. Cabrera et al. [8] presented a Micro-MOPSO which characterized for using a very small population size.

* Corresponding author.
E-mail address: wwl@zjut.edu.cn (W. Wang).

http://dx.doi.org/10.1016/j.ins.2016.08.043
0020-0255/© 2016 Published by Elsevier Inc.

However, it will encounter challenges and predicaments when expanding the PSO to solve MOPs. Since the evolutionary mechanism has been altered, we do not obtain a bare single optimal solution but a solution set. There exist some differences between MOPSO and the standard PSO in both updating and preserving the solution set, as well as global best (*gBest*) and history best (*pBest*) particles adoption. In particular, there is no absolute optimal solution for *gBest*.

Nowadays, many researchers tend to study Pareto-based optimal methods [7] in the field of MOP. By constructing as well as updating the Pareto solution set of evolutionary population, the constructed Pareto non-dominated set may climb toward the true optimal front. However, Pareto-based dominant framework will gain approving results when the dominance relation is distinct in objective space, whereas it will correspond to very poor performance when there are lots of weak dominance relations (e.g., the individuals p_2 and p_4 in Fig. 2), or, when multi-modal and deceptive problems have an isolated optimum [15].

In view of this situation, there are several types of Pareto dominance processing techniques listed as follows:

- Randomized schemes, such as roulette wheel selection [1], which would lead to low efficiency and poor caliber of solutions;
- Relaxed Pareto-based dominance methods, such as ϵ-dominance [25], r-dominance [5], grid-based approach [42] etc. This sort of approaches does not address the deficiency of Pareto-based methods;
- Reference-point-based dominance methods [20,36], which utilizes the information of reference point to regulate the solutions to the Pareto front. However, its excessive reliance on the reference point may bring stochastic and probabilistic problems;
- Non-Pareto dominance methods, such as Average Ranking (AR) [6], maxmin fitness [4], Relation Favor (RF) [18], Preference order ranking (PO) [28], Global Detriment (GDR) [19] etc., which also have some flaws that will be discussed in Section 3.1.

In order to overcome the inefficiency of existing Pareto-based dominant framework, this paper proposes a new ranking-based dominant mechanism. The main contributions of this paper are listed as follows.

- A novel sorting scheme called GMR is proposed to simplify and accelerate the process of dominance relation assessment. GMR uses the margin information of particles among the whole population where not only the individual information has been weighed in but also the population distributivity is concerned;
- A density estimator of particle is designed, which takes into account the distance among the particles in the swarm;
- In light of GMR, a succinct and efficient *gBest* and *pBest* selection strategy is conducted for MOPSO.

The remaining part of the paper proceeds as follows: Section 2 surveys the theoretical background of multi-objective optimization problem and multi-objective PSO algorithm. Section 3 proposes and details the proposed Global Margin Ranking along with global best and history best selection strategy. Some empirical results on the complexity of the algorithm implemented are also given. Section 4 lays out the analysis of the convergence. Section 5 compares the efficiency of the ranking methods. Section 6 discusses the numerical experiments. Finally, Section 7 draws the conclusions and proposes the future work.

2. Background details

2.1. Multi-objective optimization problems

Without loss of generality, a general MOOP can be described mathematically as follows:

$$\min_{\mathbf{X}\in\mathbf{\Omega}} \mathbf{F}(\mathbf{X}) = (f_1(\mathbf{X}), f_2(\mathbf{X}), \ldots, f_M(\mathbf{X}))^{\mathrm{T}} \tag{1}$$

$$\begin{cases} \mathbf{X} = (X_1, X_2, \ldots, X_n)^{\mathrm{T}} \\ \mathbf{F} : \mathbf{\Omega} \rightarrow \mathbb{R}^M \\ f_i : \mathbf{\Omega} \rightarrow \mathbb{R}^n \ (i = 1, 2, \ldots, M) \end{cases}$$

where $\mathbf{X}$ denotes to the decision vector, the objective function vector $\mathbf{F(X)}$ includes M ($M \geq 2$) objectives, Ω is the feasible set, $\mathbb{R}^n$ refers to the decision space, $\mathbb{R}^M$ represents the objective space, and $f : \mathbb{R}^n \rightarrow \mathbb{R}^M$ is the objective mapping function.

Pareto dominance: Given two vector $\mathbf{x}, \mathbf{y} \in \mathbb{R}^n$ and their corresponding objective vectors $\mathbf{F(x)}, \mathbf{F(y)} \in \mathbb{R}^M$, $\mathbf{x}$ *dominates* $\mathbf{y}$ (denoted as $\mathbf{x} \succ \mathbf{y}$) if and only if $\forall i \in (1, 2, \ldots, m),\ f_i(\mathbf{x}) \leq f_i(\mathbf{y})$ and $\exists j \in (1, 2, \ldots, m),\ f_i(\mathbf{x}) < f_i(\mathbf{y})$.
Pareto optimal solution: A decision vector $\mathbf{x} \in \mathbb{R}^n$ is said to be *Pareto optimal*, if $\nexists \mathbf{y} \in \mathbb{R}^n : \mathbf{y} \succ \mathbf{x}$.
Pareto set: The set of Pareto optimal solutions (PS) is defined as: $PS = \{\mathbf{x} \in \mathbb{R}^n | \ \nexists \mathbf{y} \in R^n : \mathbf{y} \succ \mathbf{x}\}$.
Pareto front: The *Pareto front* (PF) is defined as: $PF = \{F(\mathbf{x}) | \mathbf{x} \in PS\}$.

2.2. PSO algorithm

In PSO algorithm, each particle represents a potential solution, which utilizes two crucial forms of information in the decision process. The algorithm evolve a swarm of particles towards the optimal solution(s) by iteratively updating the

freepaper.me FREE paper

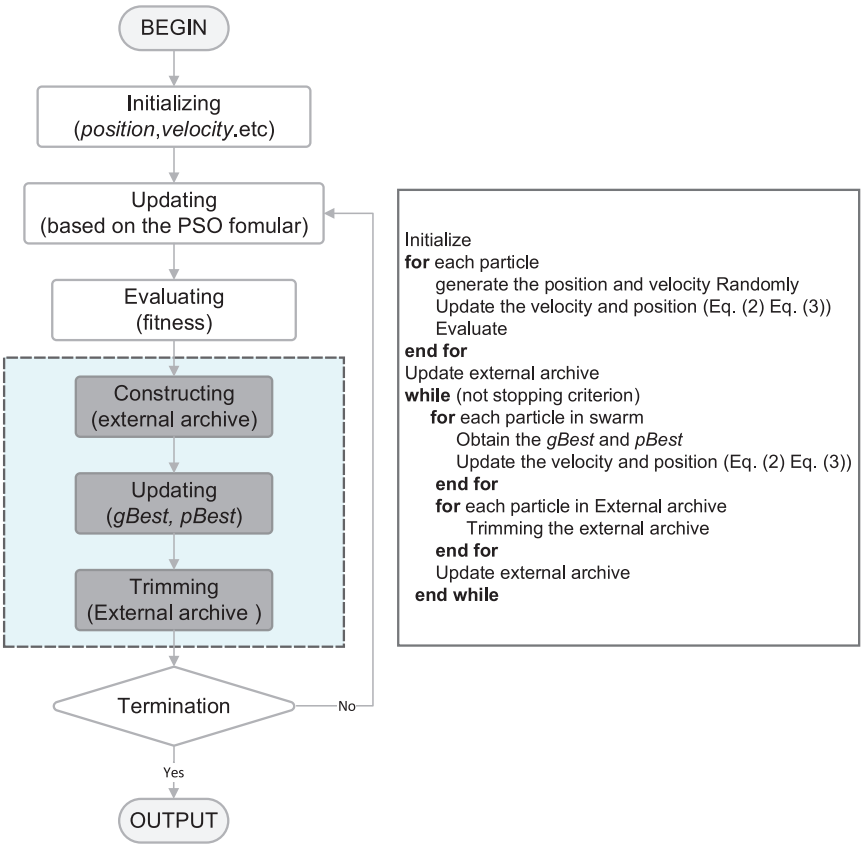

Fig. 1. A framework of multi-objective PSO algorithm.

velocities and positions of the particles based on Eqs. (2) and (3):

$$v_i(k+1) = \omega \cdot v_i(k) + c_1 \cdot r_1 \cdot (P_i - x_i(k)) + c_2 \cdot r_2 \cdot (P_g - x_i(k)) \tag{2}$$

$$x_i(k+1) = v_i(k+1) + x_i(k) \tag{3}$$

where, $i = 1, 2, 3 \ldots N$ is the running index of particles, and N denotes the size of population; $v_i(k)$ represents the velocity of particle i in kth dimension. Similarly, $x_i(k)$ denotes the position of particle i in kth dimension. ω is the inertia weight and the constants $c1$, $c2$ are the acceleration coefficients. $r1$ and $r2$ are random values uniformly distributed in [0, 1]; P_i is the personal best position found so far by particle i while P_g is the global best position found so far in the entire swarm.

The deployment of PSO algorithm for MOP problems has certain advantages:

- Similar to genetic algorithm (GA) and other evolutionary algorithms (EAs), PSO is also based on the sizable population of individuals in solution space to hunt for non-dominated solutions;
- Since the particle swarm trails after the direction of the best position by itself and the global optimal particle which in some sense, has “memory”. In that case, it has a considerable efficiency. A schematic framework of MOPSO is shown in Fig. 1.

As can be seen from Fig. 1, the MOPSO differ from PSO in that it contains the processes of construction and maintenance of the external archive. Also it has a completely different mechanism in terms of the selection on *gBest* and *pBest*.

Due to the inherent characteristics of tracking feature on particle swarm, employing the optimal particle selection mechanism of PSO into MOPSO directly will make particles track the dominated solutions. As a result, the algorithm, which will march towards the local area of Pareto front, will be trapped into local optimum. With regard to this, when expanding the PSO for MOPs, we have to remove the obstacles in selecting optimal guides including *gBest* and *pBest*. On the other hand, for the external archive, a congested and chaotic set of non-dominated solutions would make the output of the algorithm extremely non-uniform, which often makes the results impractical.

Based on these ideas and motivations, a ranking-based MOPSO is investigated and discussed in the following sections.

freepaper.me FREE paper

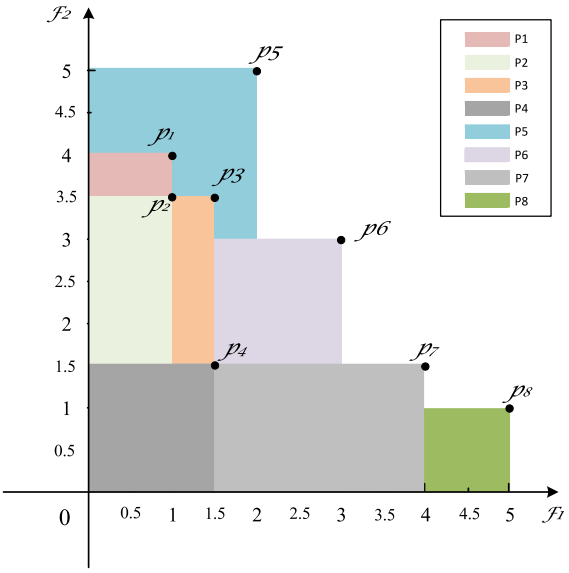

Fig. 2. A sketch map on global margin ranking.

3. The proposed algorithm

3.1. Global margin ranking strategy

As mentioned in Section 1, the current approaches in drawing non-inferior individuals in multi-objective evolutionary algorithms (MOEAs) mainly include Pareto-based dominance strategy [7], relaxed dominance [5,25] which has the essence of the Pareto-based relations, and the up-to-date non-Pareto ranking [4,6,18,19,28]. In a study on ranking methods, Mario et al. [19] proposed a fitness assignment method named GDR which is an acceptable sorting approach to some extent. However, GDR and other non-Pareto methods have general disadvantages in the following aspects:

- The ranking values of extreme points (away from one objective but incline to the other) can be too large and thus these points are likely to be abandoned, even though they have a profound meaning in multi-objective optimization and decision-making [14] because they are important reference points for estimating not only the range of objective values and Pareto front but also the uniformity of the population;
- The calculation of AR [6] is relatively simple, but lacks diversity in ranking, and thus will bring great pressure to the subsequent selection, as well as affect the calculation results, since sorted values obtained are at a concentrated interval or there exist plenty of equal ranking values. In an analysis of preference relations, Lopez [26] revealed that average ranking and preference order relation stress the solutions far from the "knee" region while the generally accepted assumption is that the most interesting solution by decision maker is the knee of the Pareto front (a "knee" region is around the region of maximum bulge on the Pareto curve).

From the definition of dominance, this paper utilizes the individual position information in the objective space to obtain the margin of dominance in the entire population. By using the information of margin, we can get the global dominance value of the individual in the solution space, which takes into account how significant the margins among solutions are. In global margin ranking, we acquire the global general ranking by combining the individual's density information throughout the population for evaluating the quality of individuals to perform optimization. The definition of global margin ranking is given below.

Definition 1 (global margin rank)**.** Define the GMR for individual as the sum of the difference of all objective values, as described in Eq. (4).

$$GMR(X_i) \triangleq \sum_{X_i \neq X_j} \max\left(\left(\prod_{m=1}^{M} f_m(X_i) - \prod_{m=1}^{M} f_m(X_j)\right), 0\right) \tag{4}$$

where, X_i and X_j are two different solutions, M is the number of objectives.

With the concept of Pareto dominance, the smaller $GMR(X_i)$, the more solutions X_i will dominate. According to Eq. (4), for any two individuals in space, X_i is superior to X_j if and only if $GMR(X_i) < GMR(X_j)$. If $GMR(X_i) = 0$, X_i will not be dominated by any other individual in solution space.

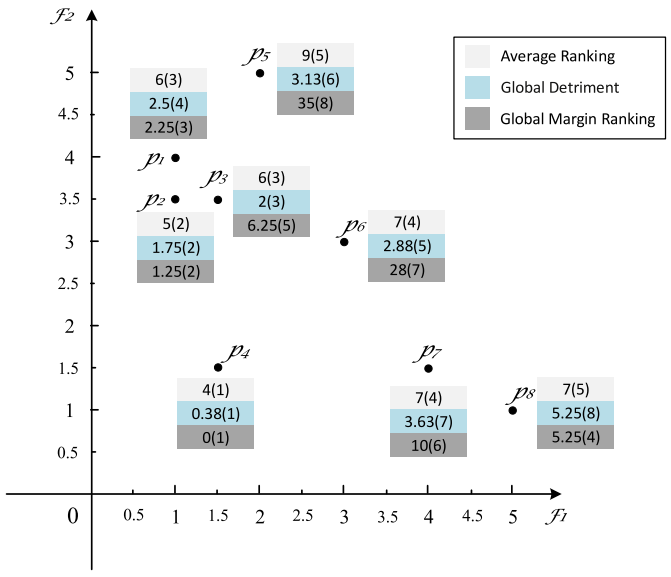

Fig. 3. A graphic illustration on sorting schemes (numbers in the brackets indicate the ranking calculated by AR, GDR and GMR respectively).

It can be seen from Definition 1 that this framework integrates every objective with all individuals' information to get the ranking values. For solution X_i and X_j, $\prod_{m=1}^{M} f_m(X_i) < \prod_{m=1}^{M} f_m(X_j)$ indicates X_i is superior to X_j in every objective. Moreover, it should have$(\prod_{m=1}^{M} f_m(X_j) - \prod_{m=1}^{M} f_m(X_i))$for the value of $GMR(X_j)$, which can gain the part that X_j is inferior to X_i. By comparing with the solutions pairwise in the objective space, we can get the sum of parts whose solutions are superior or inferior to each other. For a more intuitive explanation and comparison, we take GDR and AR as the counterpart, and give a brief illustration through Fig. 2.

Suppose there are two objectives f_1 and f_2, and the eight particles $p_1 - p_8$. Specifically, the global margin ranking for particle p_4 is $GMR(P_4) = \sum \max((\prod_{m=1}^{2} f_m(P_4) - \prod_{m=1}^{2} f_m(P_j)), 0)(j = 1, 2, 3, 5, 6, 7, 8)$, since $\prod_{m=1}^{2} f_m(P_4) < \prod_{m=1}^{2} f_m(P_j)(j = 1, 2, 3, 5, 6, 7, 8)$, from which we can derive $GMR(P_4) = 0$. Apparently, $P4$ dominates all the remaining individuals. It can be calculated easily that the $GMR[1:8] = [2.25, 1.25, 6.25, 0, 35, 28, 10, 5.25]$ with the sort order $p_4 < p_2 < p_1 < p_8 < p_3 < p_7 < p_6 < p_5$; However, $GDR[1:8] = [2.5, 1.75, 2, 0.375, 3.125, 2.875, 3.625, 5.25]$ where the sort order is $p_4 < p_2 < p_3 < p_1 < p_6 < p_5 < p_7 < p_8$, and $AR[1:8] = [6, 5, 6, 4, 9, 7, 7, 7]$ with the sort order $p_4 < p_2 < p_3 = p_1 < p_6 = p_7 = p_8 < p_5$. Fig. 3 gives the graphic illustration upon this scenario.

It can be seen from Fig. 3 that the GDR method grades p_8 (extreme point) in the last seat and will eliminate it while the proposed GMR method will retain it. On the other hand, it can be proved that the AR method generates duplicate ordering value easily which will bring great selection pressure. Moreover, it is apparent that the GMR has a better distribution.

3.2. Distribution preserving combined with GMR

One of the desired targets of MOO is to obtain the candidates as many as possible with a broad and uniform distribution which is a calibration on the merits of the algorithm [12]. Many researchers have carried out the research to achieve this point. In previous studies on MOEAs, the main approach of maintaining the distribution of population was niche technology which drew the conception of "like attracts like" [32]. The recent methods are used mainly in the following ways: clustering technique which uses the dissimilarity or similarity between individuals [29], crowding distance methods [16,30] and cell-based (or grid-based) schemes [41] as well as entropy [40] which are based on density information of individuals. Entropy is a measure of distribution on randomness and dispersion. The more dispersed the solutions, the bigger the entropy is; on the contrary, the more centralized, the smaller the entropy is. Therefore, entropy-based approach can maintain the diversity of population to some extent. However, from the definition of entropy it can be seen that this technology mainly focuses on the whole population without characterizing the relations among individuals, and lacks the ability of regulating in diversity and distribution in evolution process. Crowding distance and grid-based approaches mostly calculate the Euclidean distance pairwise, using the density information to conserve the diversity. This category of methods somewhat takes into account the relationship between individuals.

In this paper, we use the sum of Euclidean distance of the individual to the rest of the items in decision space as a measurement of aggregation of individuals in solution space, and give the definition of the global density:

freepaper.me FREE paper

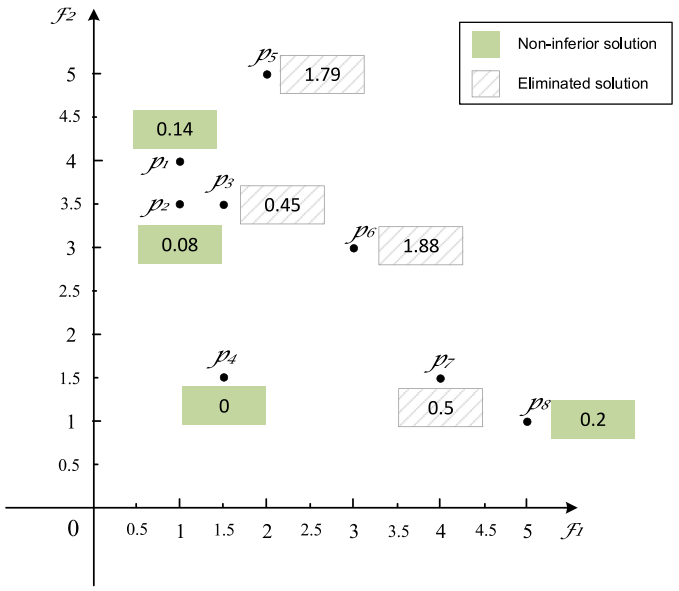

Fig. 4. Schematic diagram of the top 50% strategy with global general ranking.

Definition 2 (global density)**.**

$$GD(X_i) = \sum_{\substack{j=1 \\ i \neq j}}^{pop} d_{i,j} \tag{5}$$

Here, $GD(X_i)$ represents the global density of particle X_i, $d_{i,j}$ is the Euclidean distance between X_i and X_j. The greater the value of $GD(X_i)$ is, the fewer the particles around X_i are, with a greater difference among individuals and a better distribution.

Definition 3 (global general ranking)**.**

$$GGR(X_i) \triangleq \frac{GMR(X_i)}{GD(X_i)} \tag{6}$$

Here $GGR(X_i)$ denotes the global general ranking of X_i. The smaller the $GGR(X_i)$ is, the more dominance X_i has, which also indicates that the decent GD of X_i has a better distribution. According to Global General Ranking obtained by Eq. (6), we adopt a strategy that puts the first half into the external archive and eliminate the second half, in order to get the optimal front of the population. The process continues until the algorithm terminates.

As we can see from Fig. 2, by using the Eqs. (5) and (6) to gain the distribution information and global general ranking of each particle $D(X_i) = [16.31, 15.25, 13.87, 18.30, 19.60, 14.87, 20.16, 26.50]$, $GGR(X_i) = [0.14, 0.08, 0.45, 0, 1.79, 1.88, 0.50, 0.20]$, if we adopt the strategy of eliminating the later 50%, the p_4, p_2, p_1, p_8 will be retained, which have both distribution and uniformity. A schematic illustration is shown in Fig. 4.

3.3. Elitist selection and external archive maintenance

3.3.1. Elitist selection

The selection of particle optimal position and global optimal position is the leading aspects in PSO, and particles in the swarm are tracking these positions with iterating to optimize. In single-objective PSO, *gBest* and *pBest* can be determined simply and uniquely, whereas in MOPSO there are numerous potential feasible solutions which are unable to be distinguished through fitness values. Adopting which selection mechanism is the key to find or approach the optimal boundary. The choosing of *pBest* is relatively simple, and the mainstream research is Pareto-based dominance [3], which adopts non-dominated particle between current position and previous best position as *pBest*. If do not dominate each other, then select them randomly.

The selection of *gBest* is relatively complicated, and many researchers have studied it, mainly in the following categories.

- Stochastic selection [10]. For non-dominant individuals stored in external archive, random picking is the simplest way. But this will be prone to raise a greater selection probability in regions where particles are concentrated, which is not conducive to the distribution among optimal front, and also drops the diversity of the population;

freepaper.me FREE paper

- Crowding density [13,30]. Calculate the Euclidean distance between current particles and non-inferior solutions in external archive, and then choose a particle with minimum distance as *gBest*.

In this paper, we present a novel and efficient method for selecting *gBest* and *pBest* based on the GMR value and global density of particle.

gBest: As can be seen from the analysis in Sections 3.1 and 3.2, for the particle with the smallest global general ranking value which both have domination and distribution, picking it as the *gBest* can be more close to the real optimal boundary to speed convergence, and avoid being trapped into local optimum.

pBest: Compare the GGR value of current particle with previous optimal one. If GGR value of current particle is smaller than previous optimal one, then select the current one as *pBest*; otherwise, remain the previous one.

The optimal particles selection strategy using GMR and global density information has a high efficiency which is comparable with standard PSO, and only needs to evaluate the ranking value and density information, so the proposed algorithm can have a considerable convergence speed similar with PSO.

3.3.2. External archive maintenance

Preserving and maintaining non-dominated solutions in a fixed size is an important part of the MOPSO, the general approach is to establish an external archive to lodge the non-dominated solutions. When the number of non-inferior solutions reach the predetermined value, it needs to be trimmed to improve the search efficiency of the algorithm and maintain the diversity of solutions so as to offer a greater search space. Adopting a suitable *gBest* can diversify the external archive, which can push the algorithm to obtain a more uniform optimal front. There are few approaches focusing on archiving, such as clustering-based schemes [21,29], niching-based (or crowding) techniques [33] etc.

This paper sets the size of external archive to N which is also the size of the population, by using the global general ranking in definition 3, we fetch the top 50% solutions ($N/2$) in ascending order into the external archive. Since the distribution of entire population has changed in external archive, it is necessary to re-calculate the global general ranking to get N to gain the optimal front.

3.4. Procedure of the proposed MOPSO/GMR algorithm

According to the framework of general PSO, combined with the global margin ranking strategy, MOPSO/GMR algorithm is described as Algorithm 1.

3.5. Computational complexity analysis

Algorithmic complexity signifies the execution time as the order of magnitude with the increasing of scale on input. Here, we detail the time complexity on some major scenarios.

For MOPSO/GMR, the major computational costs lie in the calculation of GMR which depicted in Definition 1. The complexity of GMR is $O(MN^2)$, the computation of GGR requires $O(N^2)$ computations. To obtain *pBest* and *gBest* requires $O(N)$ computations. External archive maintenance requires $O(MN)$ computations. Therefore, the overall complexity of the MOPSO/GMR is $O(MN^2)$.

4. Convergence analysis

Generally, the convergence of an MOEA can be considered from two aspects. One is convergence in a finite time which is the ideal situation; the other is convergence with iteration $t \to \infty$ which has a significant guidance for the design of MOEA.

Rudolph [31] gives a definition of convergence which is shown in Eq. (7).

$$\lim_{t\to\infty}\left(|PF_{know}(t)| - |PF_{true} \cup PF_{know}(t)|\right) = 0 \tag{7}$$

with probability 1 as $t \to \infty$.

The convergence analysis of PSO has been studied by researchers. By using the discrete time dynamic system theory, Trelea [37] discussed the dynamic behavior and the convergence of the PSO algorithm. Jiang [22] analyzed the PSO algorithm determined by parameter tuple $\{\omega, c1, c2\}$ using stochastic process theory, and offered the convergent condition and corresponding parameter selection to guarantee its convergence. Sun [35] proved the fixed point theorem of the quantum-behaved PSO converges to the global optimum in probability. In addition, Chakraborty [9] first analyzed how the control parameters $\{\omega, c1, c2\}$ impact the convergence of general Pareto-based MOPSO, and deduced the conditions which ensure the algorithm converge to the center of Pareto front.

In single-objective optimization, since the fitness of each individual can be obtained specifically and definitely, it is possible to compare their values, and then acquire their certain numerical relations. Therefore, their set of feasible solutions is a totally ordered set, that is, the relation between individuals is totally ordered relation. Whereas in MOP, the relations of individuals is not a totally ordered set, and there does not exist the numerical relation but the dominance. So their feasible solution is a partially ordered set.

freepaper.me FREE paper

Algorithm 1 Pseudo code of MOPSO/GMR.

INPUT: pop. size N, external archive size N,
obj. number M, max iteration t
OUTPUT: optimal front

BEGIN
Step 1: Initialization:
$t \leftarrow 0$
randomly generate N particles in domain;
initialize algorithm parameters;
initialize the external archive, set null;
set the particles' current position as *pBest*;
Step 2: Evolutionary Computation:
i. Global Margin Ranking calculation:
using the Definition 1 to get the ranking values;
ii. Global Density calculation:
using the Definition 2 to gain the density
information of each particle;
iii. Global General Ranking calculation:
using Definition 3 to get the general ranking of each particle;
Step 3: Non-inferior solution selection:
using the Elitist selection strategy aforementioned in Section 3.3;
add the obtained non-inferior solutions into external archive;
Step 4: Updating:
using the Eq. (2) to update velocity, Eq. (3) to update position;
Step 5: external archive maintenance
using the strategy above-mentioned in Section 3.3
to maintenance the external archive;
Step 6: Meet the terminal conditions?
No, $t \leftarrow t+1$, goto Step 2;
Yes, end, output optimal front in external archive.
END

Specifically, the proposed MOPSO/GMR has a quantitative characteristic both on optimal particle selection and iteration, which signifies that the MOPSO/GMR has a totally ordered feasible solutions.

Theorem 1. *A non-inferior set with Global General Ranking is a totally ordered set.*

Proof. Consider a MOOP with objectives of M, population size is N, the codomain on each objective is $[0, \Phi_1]$, $[0, \Phi_2]$... $[0, \Phi_M]$, respectively.

For any particle X_i, it can be quantified specifically according to Definition 1 $GMR(X_i) = \sum_{X_i \neq X_j} \max((\prod_{m=1}^{M} f_m(X_i) - \prod_{m=1}^{M} f_m(X_j)), 0)$, we have $f_m(X_i) \in [0, \Phi_i]$.

Therefore, $\prod_{m=1}^{M} f_m(X_i) \in [0, \Phi_1\Phi_2\cdots\Phi_M)$

$$\left(\prod_{m=1}^{M} f_m(X_i) - \prod_{m=1}^{M} f_m(X_j)\right) \in \left(-\Phi_1\Phi_2\cdots\Phi_M, \Phi_1\Phi_2\cdots\Phi_M\right)$$

$$\max\left(\left(\prod_{m=1}^{M} f_m(X_i) - \prod_{m=1}^{M} f_m(X_j)\right), 0\right) \in [0, \Phi_1\Phi_2\cdots\Phi_M)$$

$$\therefore GMR(X_i) \in [0, (N-1)\Phi_1\Phi_2\cdots\Phi_M) \tag{8}$$

Similarly, according to Definition 2, it follows

$$d_{i,j} = \sqrt{\sum_{\substack{m=1 \\ i \neq j}}^{M} (f_m(X_i) - f_m(X_j))^2}$$

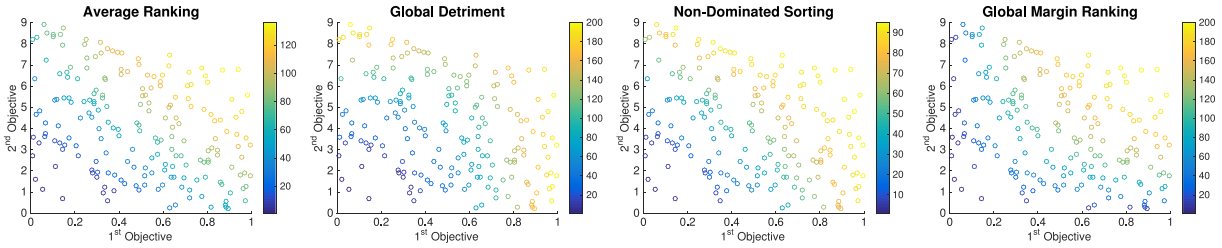

Fig. 5. Ranking landscape in 1st generation[1]. (For interpretation of the references to color in the text, the reader is referred to the web version of this article.)

We have $d_{i,j} \in [0, \sqrt{\Phi_1^2 + \Phi_2^2 + \cdots \Phi_M^2})$

$$\left[0, \sqrt{\Phi_1^2 + \Phi_2^2 + \cdots \Phi_M^2}\right) \subset \left[0, \sqrt{M}\max(\Phi_1, \Phi_2 \cdots, \Phi_M)\right)$$

Let $\Phi_{\max} = \max(\Phi_1, \Phi_2 \cdots, \Phi_M)$

$$d_{i,j} \in \left[0, \sqrt{M}\Phi_{\max}\right)$$

$$\therefore\ GD(X_i) \in \left[0, (N-1)\sqrt{M}\Phi_{\max}\right) \tag{9}$$

According to Eqs. (8) and (9), we obtain

$$GGR(X_i) \in [0, +\infty) \tag{10}$$

namely, for any particle X_i in non-inferior set, it has a certain corresponding value within the range of real numbers. □

According to the specialized literature and Theorem 1, it can be verified that the proposed MOPSO/GMR has a similar convergence as PSO.

5. Ranking efficiency analysis

One of the eligible criterion to assess the quality of a ranking approach is the ranges and shapes of the obtained solutions. A satisfying ranking approach should depend upon the optimal front as much as possible with a uniform distribution in decision space. In [19] the authors used relative entropy [11] to measure the distribution of approaches including Pareto-based one. However, the metric of relative entropy just detects the range of ranks but not the value or distribution in the population.

To measure the capability of the proposed ranking scheme guiding the population towards the optimal front, we make a visual comparison with other three frequently used ranking methods: GDR [19], AR [6] and non-dominated sorting (NDS) [16].

Similar to the fitness distribution used in mono-objective optimization, we visualize the procedure of how a ranking method guiding the stage and process of evolution during the iteration in objective space. Here, we investigate the distribution of ranks for different ranking schemes with a tangible population upon a specific problem. By integrating the GDR, AR and NDS into MOPSO, incorporate with the proposed *gBest* and *pBest* selection strategy, we implement these candidates to ZDT1 and ZDT3 which are described in Appendix A. We employ the ZDT1 and ZDT3 mainly for the demonstrability of the ranking landscapes.

In order to eliminate the impact of randomization, as well as ensuring the homogeneity, a preserved population, which is generated by a randomized algorithm, was adopted as all the initial populations for ranking candidates. Then we generate the ranking landscape by plotting decision variables against the ranks assigned by the ranking schemes in search space. Due to the similarity of the results earned on test instances, and the identical convergence for each algorithms, the demonstrations are only in 1st and 5th generation. The obtained ranking landscapes are shown in Figs. 5 and 6. Since we adopt the same initial population, the landscapes in 1st iteration are virtually identical on ZDT1 and ZDT3 which are shown in Fig. 5. The deeper the color is, the more chances the particles will have to be drawn to reproduce.

It can be seen from Fig. 5 that GMR has obtained a relatively uniform sort in the first iteration. Particles with lower ranking values which are more likely to be chosen for the next generation, have been comparatively drawn near the True Pareto front, while GDR, AR and NDS almost concentrate in the left corner within interval (0, 0.6) in the 1st objective. Whereas in Fig. 6, the MOPSO/GMR is approaching convergence at the 5th generation, and particles with lower values of ranking are distributed evenly in the front as well.[1]

[1] The landscape only represents the sort values of ranking methods, not the final optimum values. The algorithm will perform to obtain optimal front in the light of density information and ranking values.

freepaper.me FREE paper

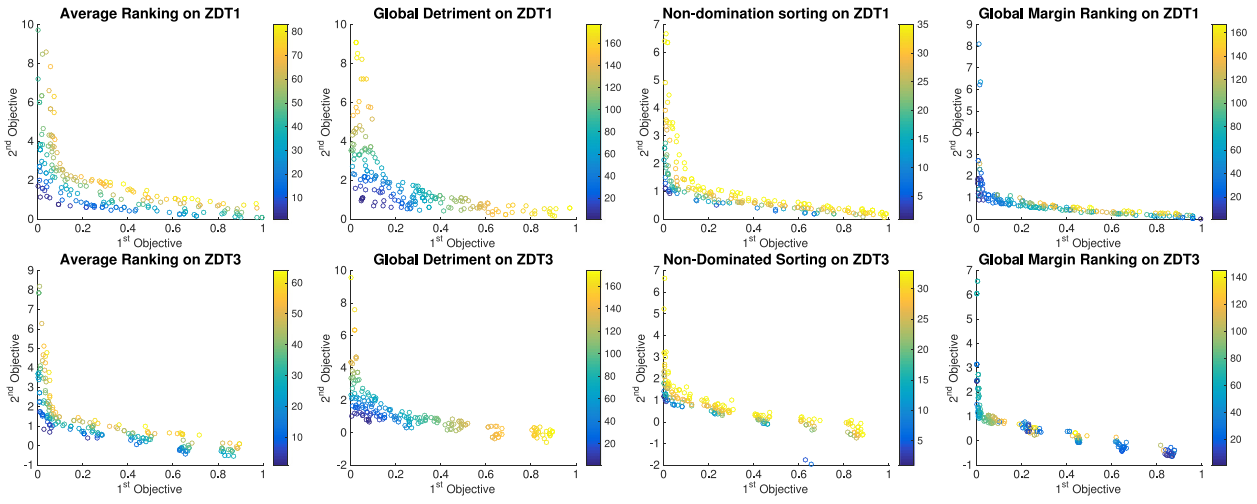

Fig. 6. Ranking landscape in 5th generation[1]. (For interpretation of the references to color in the text, the reader is referred to the web version of this article.)

It is noticeable that in latter iterations the AR approach has limited range of ranks both in Figs. 5 and 6, which will decrease the pressure upon selecting the optimal individuals, affecting the performance of algorithm. This can be drawn from the definition of AR. For GDR, as mentioned in the Section 3.1, since extreme points would have a comparatively large sort value to be struck, it is scarcely able to obtain the complete Pareto optimal front. The NDS approach performs well in these two-dimensional issues, but it has a narrow range of ranks, the same as AR scheme. However, in Section 6.3.4, we benchmark the properties of ranking-based schemes against the Pareto-based algorithms in terms of graphic comparison on convergence and distribution, where the performance of NDS approach will deteriorate when the dominance relations are complicated. For MOPSO/GMR in Fig. 6, the dark blue points scatter uniformly in decision space, which implies that the GMR gains better ability to guide the population convergence. Moreover, from the Colorbar it can be seen that the GMR achieves a richer variety of ranks, which can enhance the selection pressure.

For a more intuitive description of each ranking schemes, a metric called weighted generational distance (WGD) which borrows from the concept of Generational Distance [39] is introduced. Generational Distance is used to measure how far the evolved solution set is from the true Pareto front. Here, we incorporate the ranking of particle X_i with Generational Distance to evaluate the performance of ranking efficiency from the evolutionary perspective.

Consider a population of N ranked solutions ([1, N]). The weighted generational distance is defined as the product of a ranking value with the corresponding distance between each solutions and the nearest neighbor in the Pareto optimal set, which is formulated as Eq. (11):

Definition 4 (weighted generational distance)**.**

$$WGD \triangleq \frac{\sqrt{\sum_{i=1}^{N}(d_i^2 \times r_i)}}{N} \tag{11}$$

where, d_i refers to the Euclidean distance between the found non-dominated solution and the nearest member of the true Pareto set. r_i denotes the ranking of particle X_i.

The WGD provides a metric of how a ranking scheme guides the search process. Lower WGD value means that the ranking scheme can obtain a wider range of ranks with an ideal distribution.

As given in Fig. 7, the GMR has the lowest value of WGD while AR and GDR have relatively high values, which is identical with the phenomenon in Fig. 6.

Therefore, as can be seen from the above analysis, GMR has outperformed its rivals both in maintaining the solution in terms of diversity and convergence.

6. Evolutionary analyses of MOPSO/GMR

6.1. Parameters and instances

In this section, some more experiments with some classic and representative MOEAs and MOPSOs (NSGA-II [16], SPEA2 [46], MOEA/D [43], sMOPSO [27], D^{2}MOPSO [2]) have been adopted in order to evaluate the performance of the proposed MOPSO/GMR algorithm. Standard benchmark test problems include ZDT{1,2,3,4} suites [45] and DTLZ{1,3,4,7} suites [17]. The selected test problems cover various MOPs with linear, convex, concave, connected, and disconnected PFs, with two

freepaper.me FREE paper

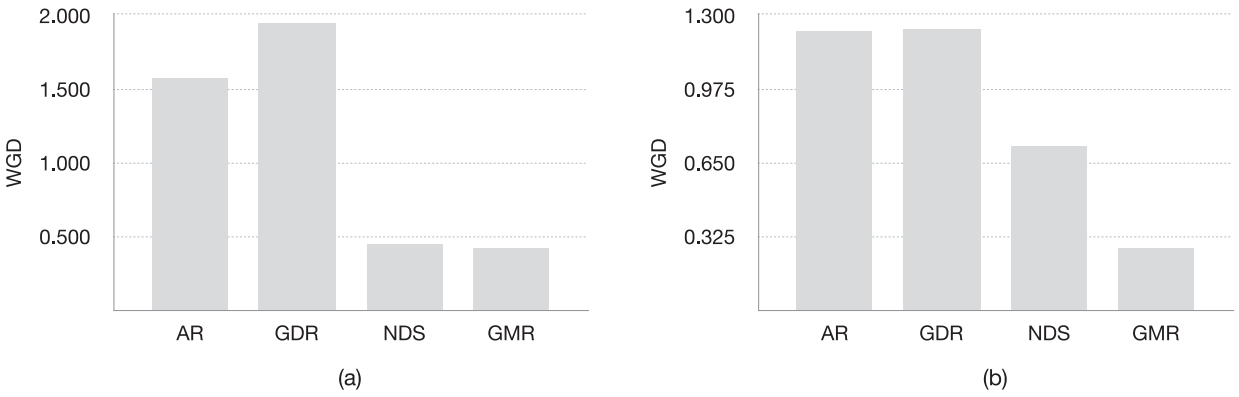

Fig. 7. Histogram of WGD metric results on (a) ZDT1. (b) ZDT3.

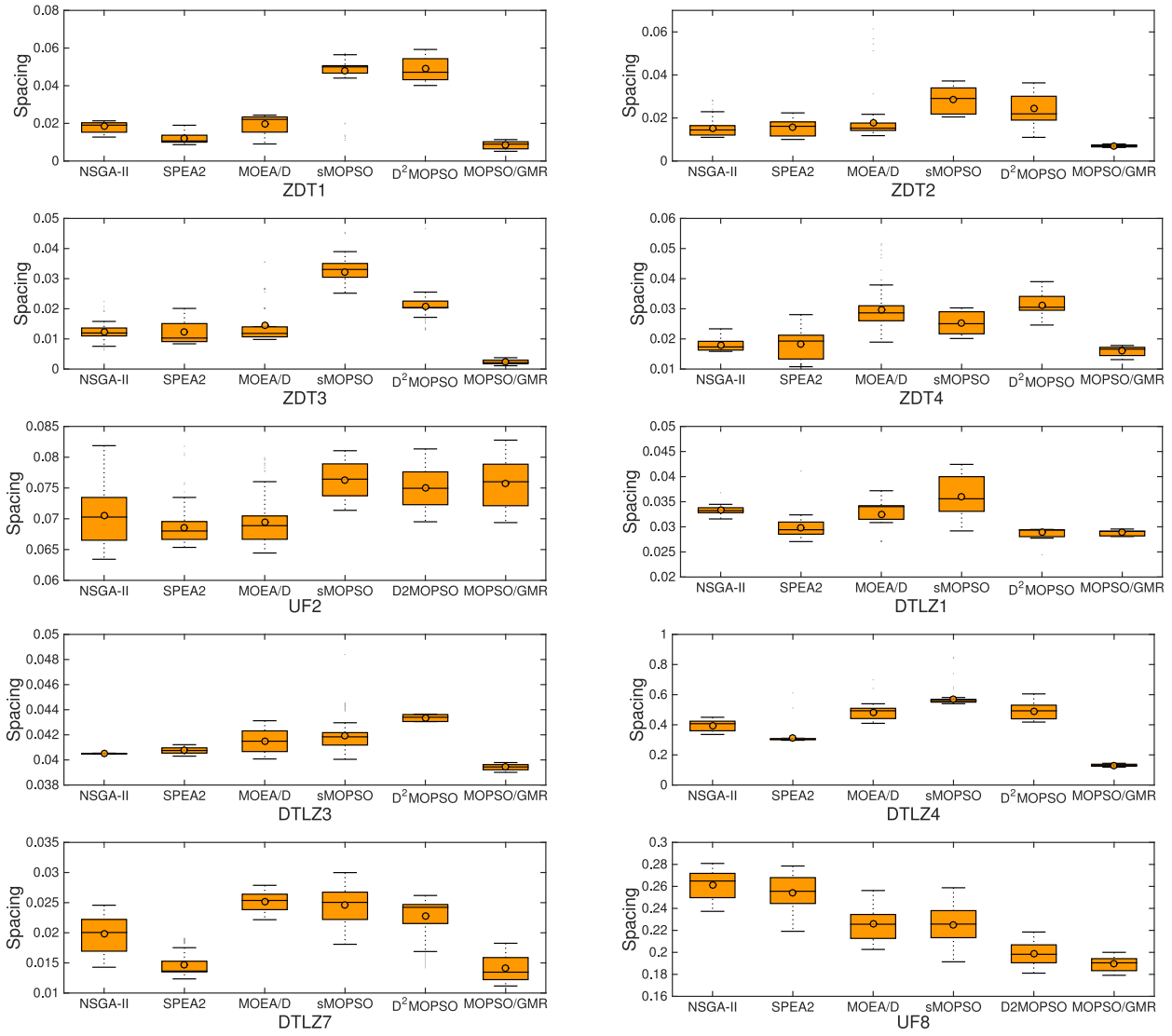

Fig. 8. Spacing metric on ZDT{1,2,3,4}, DTLZ{1,3,4,7} and UF{2,8} series problems.

Table 1
Parameters for NSGA-II, SPEA2, MOEA/D, D^{2}MOPSO, sMOPSO and MOPSO/GMR.

Algorithm	Parameter	Value	
NSGA-II	Crossover probability	1	
SPEA2	Mutation probability	0.1	
MOEA/D	Probability	1	
	Differential weight	0.5	
	Mutation distribution index	20	
	Neighborhood size	30	
sMOPSO	Turbulence probability	0.01 (ZDT4:0.05)	Pop. size:100
	Inertia weight	0.4	Gen. :200
	c1	2	
	c2	2	
D^{2}MOPSO	Inertia weight	0.4	
MOPSO/GMR	c1	2	
	c2	2	

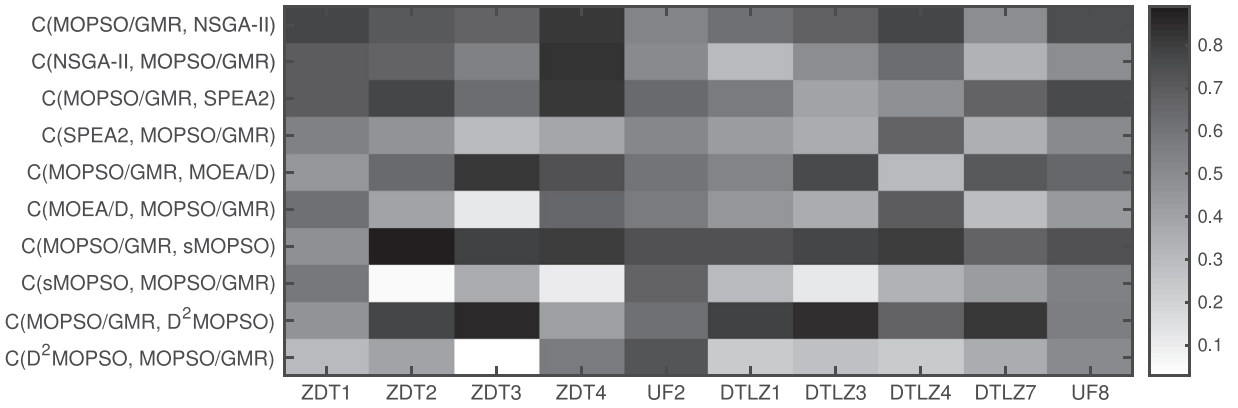

Fig. 9. Two set coverage metric between each two algorithms.

or three optimization objectives. The parameters of related algorithms and benchmark functions are shown in Table 1 and Appendix A respectively. Besides, as we mentioned in Section 4, in [9] the authors analyzed and validated that for any practical problems with ω=0.4 the swarm will never converge if $(c1 + c2)/2 > 2(1 + \omega)$, so in our implementation we adopt the parameter tuple $\{\omega, c1, c2\} = \{0.4, 2, 2\}$ for the convergence, which also recommended by the authors in compared methodologies.

6.2. Performance metrics

Generally speaking, the assessment of an algorithm is mainly from the following two aspects: a) the proximity to the real Pareto front, the closer, the better; b) the diversity among solutions, the wider the range is, the better. Taking all the above factors into consideration, the following three widely used metrics are adopted to perform the proposed approaches: inverted generational distance (IGD) [47], spacing (S) [34] and two set coverage (C) [45], brief explanations of the metrics are shown as follows:

- **Inverted generational distance** (IGD): measures the distance between true Pareto front and the obtained Pareto front. Let P^* denotes a set of uniformly distributed solutions in the objective space along the Pareto front. P is an approximation to the PF which is obtained by the algorithm. The IGD is described as:

$$IGD(P, P^*) \triangleq \frac{\sum_{i=1}^{|P^*|} dist(P_i^*, P)}{|P^*|} \tag{12}$$

where, $dist(P_i^*, P)$ is the Euclidean distance between a point $x^* \in P^*$ and its nearest neighbor in P, and $|P^*|$ is the cardinality of P^*. It can be seen from the definition of IGD that for a large $|P^*|$ it can approximately cover the entire Pareto front, which is another aspect of metric in terms of diversity. Generally, the lower the IGD value is, the better the quality of P for approximating the whole Pareto front is. The IGD equals 0 indicates the obtained Pareto front contains every point of the true Pareto front.

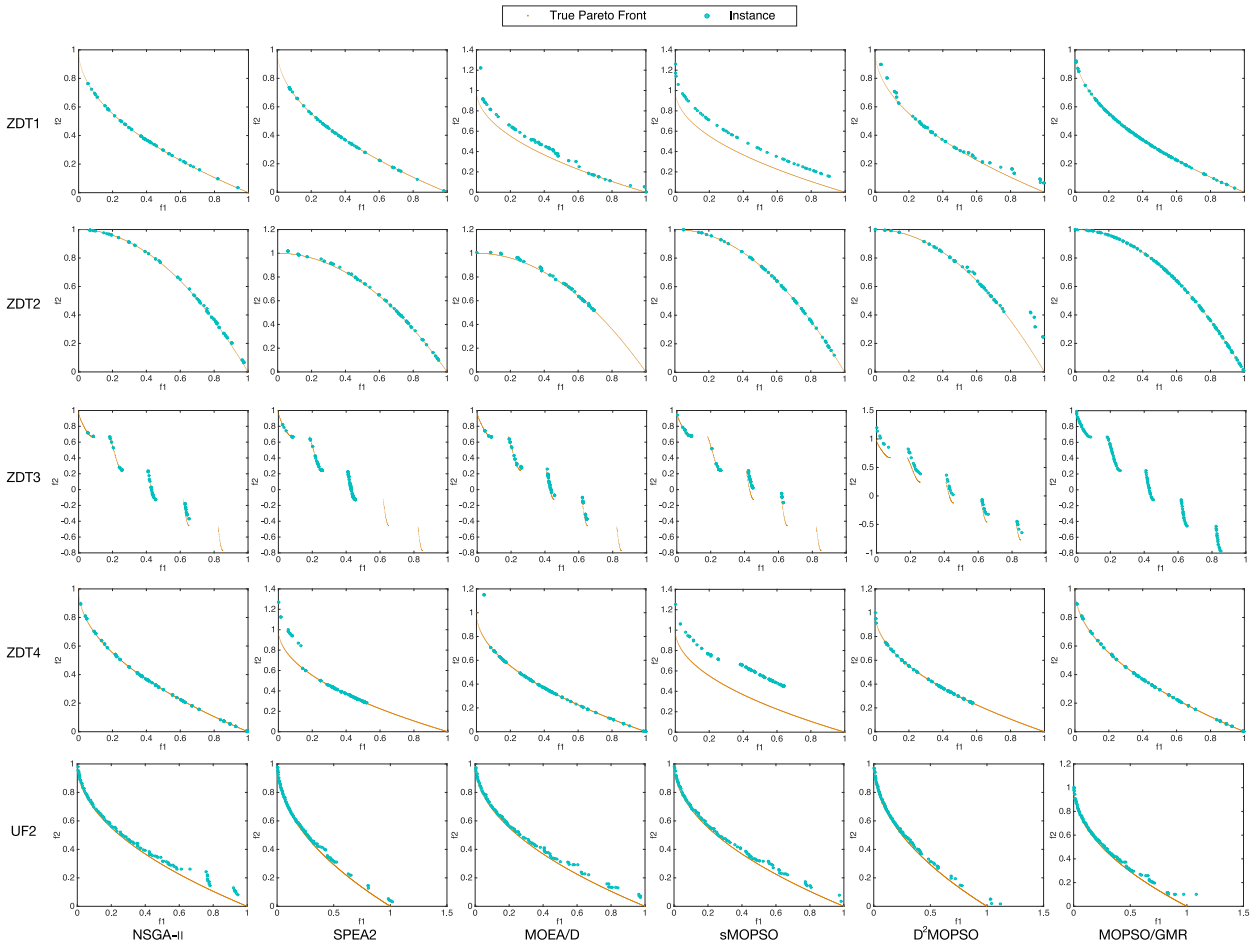

Fig. 10. Non-inferior solutions obtained by candidates on bi-objective instances. (For interpretation of the references to color in the text the reader is referred to the web version of this article.)

- **Spacing** (S): measures the range variance of neighboring solutions in the non-dominated set by comparison with the solutions converged to the true Pareto front. This metric is defined as:

$$S \triangleq \sqrt{\frac{1}{n-1}\sum_{i=1}^{n}(\bar{d}-d_i)^2} \tag{13}$$

where, $d_i = \min_j\left(\left|f_1(X_i) - f_1(X_j)\right| + \left|f_2(X_i) - f_2(X_j)\right|\right)$ is the minimal distance between two solutions and $\bar{d} = \frac{1}{n}\sum_{i=1}^{n} d_i$ is the mean value for all d_i, n is the number of vectors in the Pareto front. A smaller value of this metric indicates a uniform distribution of solution in the non-dominated set as the solutions will be equally spaced. If S equals 0, it shows that all solutions on Pareto front are equidistantly placed.

- **Two set coverage** (C): this metric is adopted to compare two sets of non-dominated solutions. Let P' and P'' be two sets of non-dominated solutions obtained by two algorithms respectively. The percentage of solutions in P'' dominated by that of P' is calculated with $C(P', P'')$ which is defined as:

$$C(P', P'') \triangleq \frac{|\{a'' \in P'' | \exists a' \in P', a' \succ a'' or\ a' = a''\}|}{|P''|} \tag{14}$$

If all points in P' dominate or equal all points in P'', this implies that $C(P', P'') = 1$. Otherwise, if no point in P' dominates any points in P'' then $C(P', P'') = 0$. Since the Pareto dominance relation is not symmetric, we will consider both $C(P', P'')$ and $C(P'', P')$. This metric relates to the relative coverage comparison of two sets.

6.3. Assessment results

6.3.1. IGD metric results

This section provides the results obtained from the IGD metric described in Section 6.2. Tables 2 and 3 show the results obtained by the candidates upon the test instances. Thirty independent runs are implemented for each test problem to

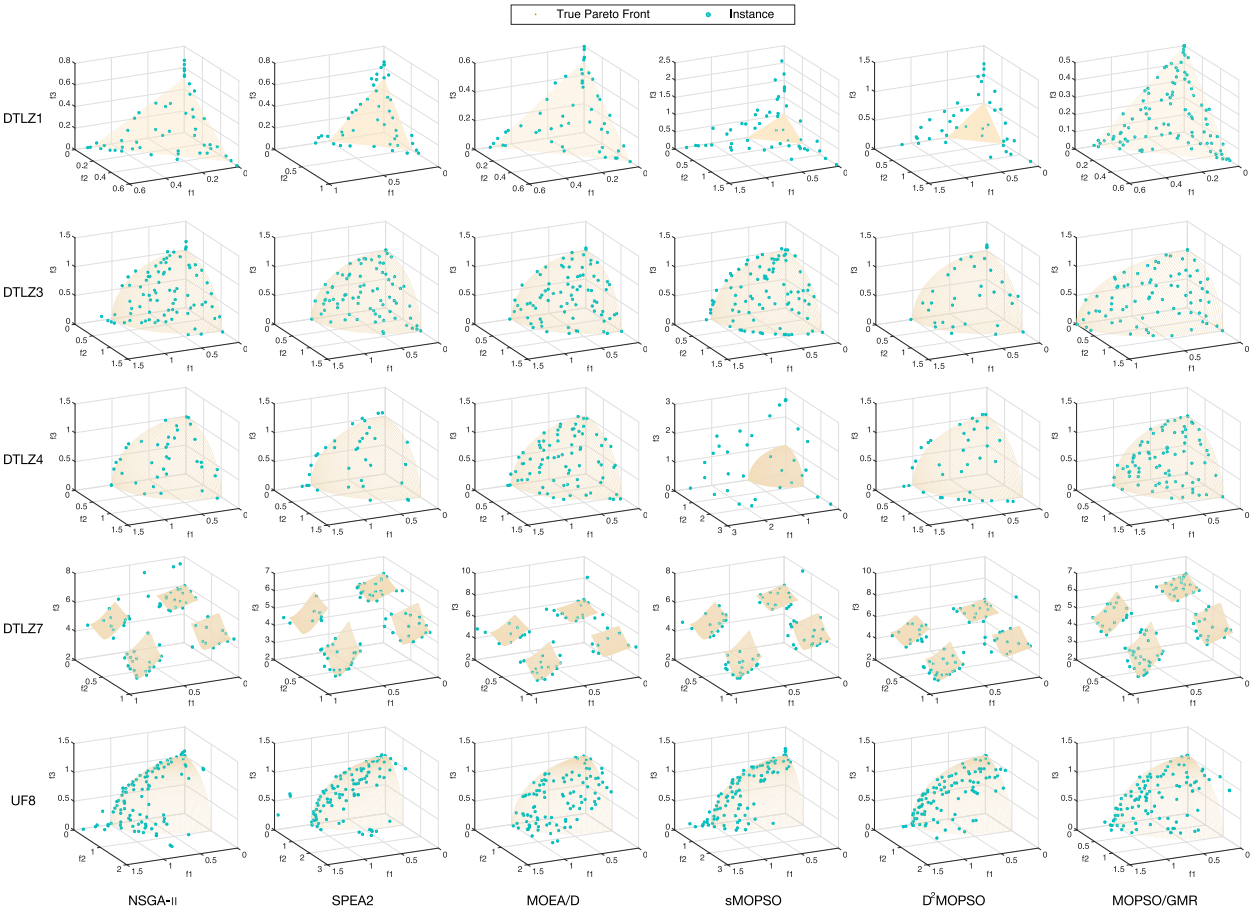

Fig. 11. Non-inferior solutions obtained by candidates on tri-objective instances. (For interpretation of the references to color in the text, the reader is referred to the web version of this article.)

Table 2
IGD metric results on bi-objective instances.

		NSGA-II	SPEA2	MOEA/D	sMOPSO	D^2MOPSO	MOPSO/GMR
ZDT1	Mean	7.34E−03	2.69E−04	5.64E−03	3.80E−02	4.10E−02	**8.20E−04**
	Std.	9.12E−05	8.16E−05	4.74E−04	7.94E−03	5.90E−03	**2.10E−05**
		+	=	+	+	+	
ZDT2	Mean	6.10E−03	2.78E−03	2.54E−03	4.92E−02	3.70E−02	**2.04E−03**
	Std.	7.70E−05	3.37E−04	1.97E−04	5.44E−03	2.24E−03	**4.12E−05**
		=	=	=	+	+	
ZDT3	Mean	5.03E−03	4.62E−03	7.66E−03	4.60E−03	5.40E−02	**3.90E−03**
	Std.	8.90E−05	5.22E−04	8.93E−04	3.40E−04	3.18E−03	**5.22E−05**
		=	=	+	=	+	
ZDT4	Mean	**6.10E−03**	9.23E−03	1.96E−02	6.10E−02	9.70E−02	6.73E−03
	Std.	**7.08E−04**	8.94E−04	9.20E−03	7.09E−03	3.30E−03	8.19E−04
		−	=	+	+	=	
UF2	Mean	3.55E−02	5.27E−02	**1.24E−02**	3.02E−02	4.42E−02	4.29E−02
	Std.	2.45E−03	4.22E−03	**2.94E−03**	4.91E−03	5.93E−03	5.14E−03
		=	+	=	+	=	
$w/t/l$		1/3/1	1/4/0	3/2/0	4/1/0	3/2/0	

avoid randomness. The statistics values including mean value (Mean) and standard deviations (Std.). In addition, the paired Wilcoxon signed-rank test is used to compare the significance between two algorithms. In Tables 2 and 3, "+","−","=" indicate the proposed algorithm is better than, worse than or similar to its competitor respectively according to the Wilcoxon signed-ranked test at significance level $\alpha = 0.05$. The results are summarized as "$w/t/l$", which denotes that MOPSO/GMR wins on w functions, ties on t functions, and loses on l functions, compared with its corresponding competitor.

freepaper.me FREE paper

Table 3
IGD metric results on tri-objective instances.

		NSGA-II	SPEA2	MOEA/D	sMOPSO	D^2MOPSO	MOPSO/GMR
DTLZ1	Mean	2.85E−02	7.43E−03	5.82E−03	1.29E+00	3.07E−01	**4.23E−03**
	Std.	3.60E−03	2.31E−04	4.12E−04	2.80E−01	6.71E−02	**2.90E−04**
		+	=	=	+	+	
DTLZ3	Mean	6.36E−03	**2.91E−03**	7.92E−03	7.80E−01	5.30E−02	5.91E−03
	Std.	8.95E−04	**1.80E−04**	6.40E−04	2.71E−03	4.31E−03	6.53E−04
		=	−	=	+	+	
DTLZ4	Mean	3.52E−03	6.29E−03	7.65E−03	3.11E−02	5.20E−01	**5.95E−04**
	Std.	5.40E−04	4.11E−04	5.61E−04	5.72E−04	4.59E−03	**4.55E−05**
		=	+	+	+	+	
DTLZ7	Mean	7.21E−03	2.37E−02	6.32E−02	3.98E−01	3.94E−01	**5.63E−03**
	Std.	6.43E−04	5.64E−03	3.87E−03	4.46E−02	3.82E−02	**5.46E−05**
		=	+	+	+	+	
UF8	Mean	6.87E−02	6.18E−02	5.64E−02	6.62E−02	5.59E−02	**5.29E−02**
	Std.	5.43E−03	4.94E−03	3.81E−03	6.94E−03	3.95E−03	**2.43E−03**
		+	+	=	+	=	
w/t/l		2/3/0	3/1/1	2/3/0	5/0/0	4/1/0	

In the case of the problem ZDT4, UF2 and DTLZ3, the NSGA-II, MOEA/D and SPEA2 achieve the best results. Although the results of MOPSO/GMR slightly deviate from the norm of other candidates, it is very close to.

However, the MOPSO/GMR outperforms others for the remaining instances, especially to the MOPSOs. This reveals that MOPSO/GMR provides better approximation Pareto fronts when both convergence and diversity are considered. In addition, the Std. values of MOPSO/GMR are smaller than most of the other candidates in test problems, demonstrating the stability of our method.

From the two tables we can clearly see that MOPSO/GMR gain the best results for the IGD metric in most scenes of this metric.

6.3.2. Spacing metric results

Boxplot is useful for identifying outliers as well as comparing distributions on discrete data, and it can specifically present the form of Spacing metric since it measures the distribution of obtained solutions in the approximated Pareto front, so we adopt this methodology to visualize this metric. Fig. 8 displays the boxplots generated by eight algorithms on benchmark functions. The whiskers in all figures represent the upper and lower borders in each data set.

This figure is quite revealing in several ways. First, the NSGA-II performs well on all test functions except DTLZ7, which indicates the NSGA-II has an incomplete convergence since the DTLZ7 has some different Pareto fronts. Second, sMOPSO produces lots of outliers on ZDT1. This indicates the poor stability, which can be observed from standard deviation in IGD metric in Table 2. Last but not least, the proposed MOPSO/GMR has a notable distribution duo to its incorporation of margin ranking and general density.

6.3.3. Two set coverage metric results

Fig. 9 shows the Two Set Coverage metric between each set of candidates. The performance is indicated by grey-scale rectangles, where darker ones refer to the best achieved C values.

From Fig. 9 we can see that the MOPSO/GMR is drawn in the darkest color which means the solutions obtained by MOPSO/GMR dominates a higher ratio of solutions produced by others. Notice that on ZDT4 both NSGA-II and MOPSO/GMR get the similar darker colors which indicates both algorithms achieve the nearly same part of solutions. This can also be observed in the Fig. 10 in Section 6.3.4.

6.3.4. Convergence results

To visually demonstrate the convergence and distribution of the proposed algorithm, this section will deliver a graphic illustration on convergence and distribution after quantitative analysis in the above sections. Figs. 10 and 11 express how the obtained non-inferior solutions distribute in the objective space through the eight candidates. Orange lines and surfaces represent the standard optimal fronts.

On ZDT test suites, each algorithm can obtain a front close to the PF from the overall perspective, but there is a difference in distribution, which can be seen from Fig. 10. Compared with the other two MOPSO algorithms, MOPSO/GMR has obvious advantages in both distribution and convergence. Fig. 11 shows the visualization by candidates on DTLZ test suites. As for the DTLZ suites which have multiple local PF and traps, the results are slightly different. It is clear that the MOPSO/GMR is able to find a uniform distribution of solutions on each test problem. In Fig. 11, results of sMOPSO on DTLZ3 outperform those of DTLZ4, since the DTLZ3 is mainly used to examine the ability to converge to the global Pareto front while DTLZ4 mainly focuses on the ability to maintain the distribution of solutions, which validates the effectiveness of sigma method [27] from another aspect.

freepaper.me FREE paper

For those well-known MOEAs, such as NSGA-II, PESA2 and MOEA/D, can obtain better convergence, whereas the other MOPSOs are unstable in face of complex test functions, which is the same as ZDT suites. However, the proposed MOPSO/GMR has a better convergence and diversity among these candidates.

freepaper.me FREE paper

7. Conclusions and future work

This paper presents a framework of non-dominant sorting strategy called GMR. Unlike those Pareto-based techniques on the basis of pairwise comparison, GMR employs the margin information of particles from the entire population to obtain the ranking values in solution space. By integrating the GMR of particles with density information, we propose a novel *gBest* and *pBest* selection approach. A new multi-objective particle swarm optimization algorithm based on GMR is proposed, combined with the aforementioned sorting and updating mechanism. The effectiveness and efficiency of GMR strategies are verified by the complexity and convergence analysis as well as experimental studies. The experimental results indicate the algorithm equipped with the proposed GMR can not only earn a richer variety of ranks but also attain a well-distributed solution set.

Although the convergence analysis has been made, an extensive mathematic assessment should be implemented as a part of our future work. Furthermore, further research needs to be conducted on incorporating the ranking strategy into some other swarm intelligence and evolutionary algorithms. Moreover, it would be a promising vision to expand the ranking methods into high-dimensional issues.

Acknowledgments

This work was supported by the National Science & Technology Pillar Program during the Twelfth Five-year Plan Period under Grant 2012BAD10B01 and National Natural Science Foundation of China under Grant 61379123.

Appendix A. Related benchmark functions.

Instance	Expression	Dim.	Obj.	Domain	Properties
ZDT1	$\begin{cases} f_1(x) = x_1, \quad f_2(x) = g(1-\sqrt{f_1/g}) \\ g(x) = 1 + 9\sum_{i=2}^{m} x_i/(m-1) \end{cases}$	30	2	$0 \le x_i \le 1$	Convex PF
ZDT2	$\begin{cases} f_1(x) = x_1, \quad f_2(x) = g(1-(f_1/g)^2) \\ g(x) = 1 + 9\sum_{i=2}^{m} x_i/(m-1) \end{cases}$	30	2	$0 \le x_i \le 1$	Non-convex PF
ZDT3	$\begin{cases} f_1(x) = x_1, \quad f_2(x) = g(1-\sqrt{f_1/g} - (f_1/g)\sin(10\pi f_1)) \\ g(x) = 1 + 9(\sum_{i=2}^{m} x_i)/(m-1) \end{cases}$	30	2	$0 \le x_i \le 1$	Disconnected PF
ZDT4	$\begin{cases} f_1(x) = x_1, \quad f_2(x) = g(1-\sqrt{f_1/g}) \\ g(x) = 1 + 10(m-1) + \sum_{i=2}^{m}(x^2 - 10\cos(4\pi x_i)) \end{cases}$	30	2	$0 \le x_1 \le 1 - 5 \le x_i \le 5 (i = 2, 3, \cdots, m)$	Non-convex, Multimodal PF
UF2	$\begin{cases} f_1(x) = x_1 + \frac{2}{\lvert J_1\rvert}\sum_{j\in J_1} y_j^2 \\ f_2(x) = 1 - \sqrt{x_1} + \frac{2}{\lvert J_2\rvert}\sum_{j\in J_2} y_j^2 \\ J_1 = \{j \mid j \text{ is odd and } 2 \le j \le n\} \\ J_2 = \{j \mid j \text{ is enev and } 2 \le j \le n\} \\ y_i = \begin{cases} x_j - [0.3x_1^2\cos(24\pi x_1 + \frac{4j\pi}{n}) + 0.6x_1]\cos(6\pi x_1 + \frac{j\pi}{n}) & j \in J_1 \\ x_j - [0.3x_1^2\cos(24\pi x_1 + \frac{4j\pi}{n}) + 0.6x_1]\sin(6\pi x_1 + \frac{j\pi}{n}) & j \in J_2 \end{cases} \end{cases}$	30	2	$0 \le x_i \le 1$ $(i = 1, 2, \cdots, m)$	Convex, Complex PF
DTLZ1	$\begin{cases} f_1(x) = 0.5x_1x_2\cdots x_{m-1}[1+g(x_m)] \\ f_2(x) = 0.5x_1x_2\cdots(1-x_{m-1})[1+g(x_m)] \\ \vdots \\ f_{m-1}(x) = 0.5x_1(1-x_2)[1+g(x_m)] \\ f_m(x) = 0.5(1-x_1)[1+g(x_m)] \\ g(x_m) = 100[\lvert x_m\rvert + \sum_{x_i\in X_m}(x_i-0.5)^2 - \cos(20\pi(x_i-0.5))] \end{cases}$	10	3	$0 \le x_i \le 1$ $(i = 1, 2, \cdots, m)$	Linear, Multimodal PF

Continued

Instance	Expression	Dim.	Obj.	Domain	Properties
DTLZ3	$\begin{cases} f_1(x) = [1+g(x_m)]\cos(x_1{}^{\pi}/_2)\cdots\cos(x_{m-2}{}^{\pi}/_2)\cos(x_{m-1}{}^{\pi}/_2) \\ f_2(x) = [1+g(x_m)]\cos(x_1{}^{\pi}/_2)\cdots\cos(x_{m-2}{}^{\pi}/_2)\sin(x_{m-1}{}^{\pi}/_2) \\ f_3(x) = [1+g(x_m)]\cos(x_1{}^{\pi}/_2)\cdots\sin(x_{m-2}{}^{\pi}/_2) \\ \quad\vdots \\ f_m(x) = [1+g(x_m)]\sin(x_1{}^{\pi}/_2) \\ g(x_m) = 100[\lvert x_m\rvert + \sum_{x_i\in x_m}(x_i-0.5)^2 - \cos(20\pi(x_i-0.5))] \end{cases}$	10	3	$0 \le x_i \le 1$ $(i = 1, 2, \cdots, m)$	Non-convex, Multimodal PF
DTLZ4	$\begin{cases} f_1(x) = [1+g(x_m)]\cos(x_1^{\alpha}{}^{\pi}/_2)\cdots\cos(x_{m-2}^{\alpha}{}^{\pi}/_2)\cos(x_{m-1}^{\alpha}{}^{\pi}/_2) \\ f_2(x) = [1+g(x_m)]\cos(x_1^{\alpha}{}^{\pi}/_2)\cdots\cos(x_{m-2}^{\alpha}{}^{\pi}/_2)\sin(x_{m-1}^{\alpha}{}^{\pi}/_2) \\ f_3(x) = [1+g(x_m)]\cos(x_1^{\alpha}{}^{\pi}/_2)\cdots\sin(x_{m-2}^{\alpha}{}^{\pi}/_2) \\ \quad\vdots \\ f_m(x) = [1+g(x_m)]\sin(x_1^{\alpha}{}^{\pi}/_2) \\ g(x_m) = \sum_{x_i\in x_m}(x_i-0.5)^2 \end{cases}$	10	3	$0 \le x_i \le 1$ $(i = 1, 2, \cdots, m)$	Non-convex, Biased PF
DTLZ7	$\begin{cases} f_1(x) = x_1 \\ f_2(x) = x_2 \\ \quad\vdots \\ f_{m-1}(x_{m-1}) = x_{m-1} \\ f_m(x) = [1+g(x_m)]h(f_1, f_2, \cdots f_{m-1}, g) \\ g(x_m) = 1 + \frac{9}{\lvert x_m\rvert}\sum_{x_i\in x_m} x_i \\ h(f_1, f_2, \cdots f_{m-1}, g) = m - \sum_{i=1}^{m-1}\left[\frac{f_i}{1+g}(1+\sin(3\pi f_i))\right] \end{cases}$	10	3	$0 \le x_i \le 1$ $(i = 1, 2, \cdots, m)$	Mixed, Discontinuous, Multimodal PF
UF8	$\begin{cases} f_1(x) = \cos(0.5x_1\pi)\cos(0.5x_2\pi) + \frac{2}{\lvert J_1\rvert}\sum_{j\in J_1}(x_j - 2x_2\sin(2\pi x_1 + \frac{j\pi}{n}))^2 \\ f_2(x) = \cos(0.5x_1\pi)\sin(0.5x_2\pi) + \frac{2}{\lvert J_2\rvert}\sum_{j\in J_2}(x_j - 2x_2\sin(2\pi x_1 + \frac{j\pi}{n}))^2 \\ f_3(x) = \sin(0.5x_1\pi) + \frac{2}{\lvert J_3\rvert}\sum_{j\in J_3}(x_j - 2x_2\sin(2\pi x_1 + \frac{j\pi}{n}))^2 \\ J_1 = \{j \mid 3 \le j \le n, \textit{and } j-1 \textit{ is a multiplication of } 3\} \\ J_2 = \{j \mid 3 \le j \le n, \textit{and } j-2 \textit{ is a multiplication of } 3\} \\ J_3 = \{j \mid 3 \le j \le n, \textit{and } j \textit{ is a multiplication of } 3\} \end{cases}$	30	3	$0 \le x_i \le 1$ $(i = 1, 2, \cdots, m)$	Non-convex, Complex PS

References

[1] S. Agrawal, Y. Dashora, M. Tiwari, Y.J. Son, Interactive particle swarm: a Pareto-adaptive metaheuristic to multiobjective optimization, IEEE Trans. Syst. Man Cybern. Part A 38 (2) (2008) 258–277, doi:10.1109/TSMCA.2007.914767.

[2] N. Al Moubayed, A. Petrovski, J. McCall, D^{2}MOPSO: MOPSO based on decomposition and dominance with archiving using crowding distance in objective and solution spaces, Evol. Comput. 22 (1) (2014) 47–77, doi:10.1162/EVCO_a_00104.

[3] J.E. Alvarez-Benitez, R.M. Everson, J.E. Fieldsend, A MOPSO algorithm based exclusively on Pareto dominance concepts, in: Evolutionary Multi-Criterion Optimization, vol. 3410, Springer, 2005, pp. 459–473, doi:10.1007/978-3-540-31880-4_32.

[4] R. Balling, S. Wilson, The maxi-min fitness function for multi-objective evolutionary computation: application to city planning, in: Proceedings of the Genetic and Evolutionary Computation Conference (GECCO'2001), 2001, pp. 1079–1084.

[5] L. Ben Said, S. Bechikh, K. Ghedira, The r-Dominance: anew dominance relation for interactive evolutionary multicriteria decision making, IEEE Trans. Evol. Comput. 14 (5) (2010) 801–818, doi:10.1109/TEVC.2010.2041060.

[6] P.J. Bentley, J.P. Wakefield, Finding acceptable solutions in the Pareto-optimal range using multiobjective genetic algorithms, in: Soft Computing in Engineering Design and Manufacturing, Springer, 1998, pp. 231–240.

[7] S. Biswas, S. Das, P.N. Suganthan, C.A.C. Coello, Evolutionary multiobjective optimization in dynamic environments: a set of novel benchmark functions, in: 2014 IEEE Congress on Evolutionary Computation (CEC), IEEE, 2014, pp. 3192–3199, doi:10.1109/CEC.2014.6900487.

[8] J.C.F. Cabrera, C.A.C. Coello, Micro-MOPSO: amulti-objective particle swarm optimizer that uses a very small population size, in: Multi-Objective Swarm Intelligent Systems, Springer, 2010, pp. 83–104, doi:10.1007/978-3-642-05165-4_4.

[9] P. Chakraborty, S. Das, G.G. Roy, A. Abraham, On convergence of the multi-objective particle swarm optimizers, Inf. Sci. 181 (8) (2011) 1411–1425, doi:10.1016/j.ins.2010.11.036.

[10] C.A.C. Coello, M.S. Lechuga, MOPSO: a proposal for multiple objective particle swarm optimization, in: Evolutionary Computation, 2002. CEC'02. Proceedings of the 2002 Congress on, vol. 2, IEEE, 2002, pp. 1051–1056.

[11] D.W. Corne, J.D. Knowles, Techniques for highly multiobjective optimisation: some nondominated points are better than others, in: Proceedings of the 9th Annual Conference on Genetic and Evolutionary Computation - GECCO '07, ACM, ACM Press, New York, New York, USA, 2007, p. 773, doi:10.1145/1276958.1277115.

[12] M. Črepinšek, S.-H. Liu, M. Mernik, Exploration and exploitation in evolutionary algorithms: a survey, ACM Comput. Surv. 45 (3) (2013) 1–33, doi:10.1145/2480741.2480752.

[13] C. Dai, Y. Wang, M. Ye, A new multi-objective particle swarm optimization algorithm based on decomposition, Inf. Sci. 325 (2015) 541–557, doi:10.1016/j.ins.2015.07.018.

[14] D. Dasgupta, G. Hernandez, A. Romero, D. Garrett, A. Kaushal, J. Simien, On the use of informed initialization and extreme solutions sub-population in multi-objective evolutionary algorithms, in: 2009 IEEE Symposium on Computational Intelligence in Milti-Criteria Decision-Making, IEEE, 2009, pp. 58–65, doi:10.1109/MCDM.2009.4938829.

[15] K. Deb, Multi-objective genetic algorithms: problem difficulties and construction of test problems., Evol.Comput. 7 (3) (1999) 205–230, doi:10.1162/evco.1999.7.3.205.

[16] K. Deb, S. Pratab, S. Agarwal, T. Meyarivan, A fast and elitist multiobjective genetic algorithm: NGSA-II, IEEE Trans. Evol. Comput. 6 (2) (2002) 182–197, doi:10.1109/4235.996017.

[17] K. Deb, L. Thiele, M. Laumanns, E. Zitzler, Scalable multi-objective optimization test problems, in: Proceedings of the 2002 Congress on Evolutionary Computation, CEC 2002, vol. 1, IEEE, 2002, pp. 825–830, doi:10.1109/CEC.2002.1007032.

freepaper.me FREE paper

[18] N. Drechsler, R. Drechsler, B. Becker, Multi-objective optimisation based on relation favour, in: Evolutionary Multi-Criterion Optimization, Springer, 2001, pp. 154–166.
[19] M. Garza-Fabre, G.T. Pulido, C.A.C. Coello, Ranking methods for many-objective optimization, in: MICAI 2009: Advances in Artificial Intelligence, in: LNAI, vol. 5845, Springer, 2009, pp. 633–645, doi:10.1007/978-3-642-05258-3_56.
[20] F. Goulart, F. Campelo, Preference-guided evolutionary algorithms for many-objective optimization, Inf. Sci. 329 (2016) 236–255, doi:10.1016/j.ins.2015.09.015.
[21] U. Halder, S. Das, D. Maity, A cluster-based differential evolution algorithm With external archive for optimization in dynamic environments, IEEE Trans. Cybern. 43 (3) (2013) 881–897, doi:10.1109/TSMCB.2012.2217491.
[22] M. Jiang, Y. Luo, S. Yang, Stochastic convergence analysis and parameter selection of the standard particle swarm optimization algorithm, Inf. Process. Lett. 102 (1) (2007) 8–16, doi:10.1016/j.ipl.2006.10.005.
[23] J. Kennedy, R. Eberhart, Particle swarm optimization, in: Proceedings of ICNN'95 - International Conference on Neural Networks, vol. 4, IEEE, 1995, pp. 1942–1948, doi:10.1109/ICNN.1995.488968.
[24] S. Kundu, S. Das, A.V. Vasilakos, S. Biswas, A modified differential evolution-based combined routing and sleep scheduling scheme for lifetime maximization of wireless sensor networks, Soft Comput. 19 (3) (2015) 637–659, doi:10.1007/s00500-014-1286-9.
[25] M. Laumanns, L. Thiele, K. Deb, E. Zitzler, Combining convergence and diversity in evolutionary multiobjective optimization, Evol. Comput. 10 (3) (2002) 263–282, doi:10.1162/106365602760234108.
[26] A. López Jaimes, C.A. Coello Coello, Study of preference relations in many-objective optimization, in: Proceedings of the 11th Annual conference on Genetic and evolutionary computation - GECCO '09, volume Montreal, ACM, ACM Press, New York, New York, USA, 2009, p. 611, doi:10.1145/1569901.1569986.
[27] S. Mostaghim, J. Teich, Strategies for finding good local guides in multi-objective particle swarm optimization (MOPSO), in: Proceedings of the 2003 IEEE Swarm Intelligence Symposium. SIS'03 (Cat. No.03EX706), IEEE, 2003, pp. 26–33, doi:10.1109/SIS.2003.1202243.
[28] F. di Pierro, S.-T. Khu, D.A. Savi, An investigation on preference order ranking scheme for multiobjective evolutionary optimization, IEEE Trans. Evol. Comput. 11 (1) (2007) 17–45, doi:10.1109/TEVC.2006.876362.
[29] G.T. Pulido, C.A.C. Coello, Using clustering techniques to improve the performance of a multi-objective particle swarm optimizer, in: Genetic and Evolutionary Computation–GECCO 2004, Springer, 2004, pp. 225–237, doi:10.1007/978-3-540-24854-5_20.
[30] C.R. Raquel, P.C. Naval Jr, An effective use of crowding distance in multiobjective particle swarm optimization, in: Proceedings of the 2005 Conference on Genetic and Evolutionary Computation, ACM, 2005, pp. 257–264, doi:10.1145/1068009.1068047.
[31] G. Rudolph, Some Theoretical Properties of Evolutionary Algorithms under Partially Ordered Fitness Values, Technical Report, Universit{ä}t Dortmund, 2001.
[32] B. Sareni, L. Krahenbuhl, Fitness sharing and niching methods revisited, IEEE Trans. Evol.Comput. 2 (3) (1998) 97–106.
[33] I. Schoeman, A. Engelbrecht, Niching for dynamic environments using particle swarm optimization, in: Simulated Evolution and Learning, Springer, 2006, pp. 134–141.
[34] J.R. Schott, Fault tolerant design using single and multicriteria genetic algorithm optimization. Ph.D. thesis, Massachusetts Institute of Technology, 1995.
[35] J. Sun, X. Wu, V. Palade, W. Fang, C.-H. Lai, W. Xu, Convergence analysis and improvements of quantum-behaved particle swarm optimization, Inf. Sci. 193 (2012) 81–103, doi:10.1016/j.ins.2012.01.005.
[36] L. Thiele, K. Miettinen, P.J. Korhonen, J. Molina, A preference-based evolutionary algorithm for multi-objective optimization, Evol. Comput. 17 (3) (2009) 411–436, doi:10.1162/evco.2009.17.3.411.
[37] I.C. Trelea, The particle swarm optimization algorithm: convergence analysis and parameter selection, Inf. Process. Lett. 85 (6) (2003) 317–325, doi:10.1016/S0020-0190(02)00447-7.
[38] A. Trivedi, D. Srinivasan, S. Biswas, T. Reindl, Hybridizing genetic algorithm with differential evolution for solving the unit commitment scheduling problem, Swarm Evol. Comput. 23 (2015) 50–64, doi:10.1016/j.swevo.2015.04.001.
[39] D.A. Van Veldhuizen, Multiobjective evolutionary algorithms: classifications, analyses, and new innovations. Ph.D. thesis, Air Force Institute of Technology, Wright Patterson AFB, OH, USA, 1999.
[40] Y.-N. Wang, L.-H. Wu, X.-F. Yuan, Multi-objective self-adaptive differential evolution with elitist archive and crowding entropy-based diversity measure, Soft Comput. 14 (3) (2010) 193–209, doi:10.1007/s00500-008-0394-9.
[41] W.-F. Leong, G. Yen, PSO-based multiobjective optimization with dynamic population size and adaptive local archives, IEEE Trans. Syst. Man Cybern.Part B (Cybern.) 38 (5) (2008) 1270–1293, doi:10.1109/TSMCB.2008.925757.
[42] S. Yang, M. Li, X. Liu, J. Zheng, A grid-based evolutionary algorithm for many-objective optimization, IEEE Trans. Evol. Comput. 17 (5) (2013) 721–736, doi:10.1109/TEVC.2012.2227145.
[43] Q. Zhang, H. Li, MOEA/D: a multiobjective evolutionary algorithm based on decomposition, IEEE Trans. Evol. Comput. 11 (6) (2007) 712–731, doi:10.1109/TEVC.2007.892759.
[44] Z. Zhu, J. Xiao, S. He, Z. Ji, Y. Sun, A multi-objective memetic algorithm based on locality-sensitive hashing for one-to-many-to-one dynamic pickup-and-delivery problem, Inf. Sci. 329 (2016) 73–89, doi:10.1016/j.ins.2015.09.006.
[45] E. Zitzler, K. Deb, L. Thiele, Comparison of multiobjective evolutionary algorithms: empirical results, Evol.Comput. 8 (2) (2000) 173–195, doi:10.1162/106365600568202.
[46] E. Zitzler, M. Laumanns, L. Thiele, SPEA2: improving the strength Pareto evolutionary algorithm, 2001, doi:10.1.1.28.7571
[47] E. Zitzler, L. Thiele, M. Laumanns, C. Fonseca, V. da Fonseca, Performance assessment of multiobjective optimizers: an analysis and review, IEEE Trans. Evol. Comput. 7 (2) (2003) 117–132, doi:10.1109/TEVC.2003.810758.

freepaper.me FREE paper