

Many-Objective Particle Swarm Optimization Algorithm Based on New Fitness Allocation and Multiple Cooperative Strategies

Weiwei Yu, Beihang University, China

Li Zhang, Beihang University, China

Chengwang Xie, Nanning Normal University, China

ABSTRACT

Many-objective optimization problems (MaOPs) refer to those multi-objective problems (MOPs) with more than three objectives. In order to solve MaOPs, a multi-objective particle swarm optimization algorithm based on new fitness assignment and multi cooperation strategy (FAMSHMPSO) is proposed. Firstly, this paper proposes a new fitness allocation method based on fuzzy information theory to enhance the convergence of the algorithm. Then a new multi-criteria mutation strategy is introduced to disturb the population and improve the diversity of the algorithm. Finally, the external files are maintained by the three-point shortest path method, which improves the quality of the solution. The performance of FAMSHMPSO algorithm is evaluated by evaluating the mean value, standard deviation, and IGD+ index of the target value on dtlz test function set of different targets of FAMSHMPSO algorithm and other five representative multi-objective evolutionary algorithms. The experimental results show that FAMSHMPSO algorithm has obvious performance advantages in convergence, diversity, and robustness.

KEYWORDS

Fitness Allocation, High-Dimensional Multi-Objective Optimization, Multi-Criteria Variation, Particle Swarm Optimization

1. INTRODUCTION

In today's scientific research and engineering practice, the problems faced by decision makers are becoming more and more complex, and often need to deal with multiple objectives at the same time. Usually, this type of problem that requires multiple objectives to achieve the optimal design is called Multi-objective Optimization Problem(Wang, Zhang, Li, Zhao et al, 2018; Wu et al., 2017)(MOP). Generally, the goals of the MOP problem conflict with each other, and it is often difficult to obtain the optimal solution to the problem. As the number of goals increases, this defect becomes more obvious. Therefore, there is no unique optimal solution for the MOP problem so that each target can obtain the optimal value at the same time, but a solution set consisting of a set of compromise solutions, that is, a Pareto solution set. Because the MOP problem model is highly complicated, it is difficult to make general analysis methods effective, So many researchers have done a lot of research(Nayyar, Garg, Gupta et al, 2018; Nayyar, Le, & Nguyen, 2018; Nayyar & Nguyen, 2018a), most of them use

DOI: 10.4018/IJCINI.20211001.0a29

This article published as an Open Access article distributed under the terms of the Creative Commons Attribution License (<http://creativecommons.org/licenses/by/4.0/>) which permits unrestricted use, distribution, and production in any medium, provided the author of the original work and original publication source are properly credited.

multi-objective optimization algorithm(Nayyar & Nguyen, 2018b; Nayyar & Singh, 2016) to obtain the approximate value of Pareto solution set. Particle Swarm Optimization(Kennedy & Eberhart, 1994)(PSO) is a heuristic group intelligent random algorithm for simulating the evolution process of natural biological groups. It has the advantages of fast convergence, simple parameter setting and easy programming. Because it has the characteristics of independent problem model, the adaptability of optimization process, the implicit parallelism and the robustness of solving complex nonlinear problems in single-objective optimization, some scholars have begun to apply it to Multi-objective optimization problems and widespread attention in the field of multi-objective optimization. So far, researchers have proposed a variety of multi-objective particle swarm optimization algorithms based on different research backgrounds and perspectives. Among them, D2MOPSO(Moubayed et al., 2014) and OMOPSO(Kaur & Kadam, 2018) algorithms based on decomposition ideas are some of the more classical algorithms. But most articles about particle swarms deal with low-dimensional multi-objective problems, that is the number of targets is generally 2-3. In reality, problems that require more and more targets to be optimized at the same time are constantly emerging. Researchers generally call the problems of simultaneously optimizing 4 or more targets as Many-objective optimization problem(Li, Liang, Yang et al, 2019)(MaOP). Compared with the MOP problem, the MaOP problem is more difficult to solve. The reason is as follows: For many-objective optimization problem, the search ability of these traditional algorithms is greatly reduced. As the target spatial dimension increases, the number of non-dominated solutions in the population will increase exponentially, even close to the entire Pareto frontier. For example, for the MaOP problem of m targets, the k solutions are distributed on each target, and mk^{m-1} solutions are needed to represent the Pareto frontier. Pareto dominance is a strong sorting relationship. When the dimension of the optimization problem is high, it will cause the evolutionary algorithm to generate a large number of non-dominated solutions in a finite-scale group. There is a lack of certain comparison criteria between these non-dominated solutions, making it difficult to choose from them, and the search ability of the algorithm will be greatly weakened. Visualization of non-dominated solution sets for many-objective optimization is also a problem. The commonly used Cartesian coordinate system can only represent 3D at most, and it is not suitable for high-dimensional multi-target problems, which creates obstacles for the final choice of decision makers. Because the Pareto dominance relationship is degraded in the high-dimensional target space, the distribution retention mechanism becomes the dominant factor of the algorithm selection individual, but the selection mechanism dominated by the individual density information may not be able to effectively drive the approximate Pareto front to approximate the real Pareto front. It may even have a negative impact on the optimization process. many-objective optimization needs to calculate more Pareto optimal solutions to approximate the Pareto frontier, so the complexity of the algorithm is also higher. Recent studies have also shown that the traditional Pareto-dominated PSO algorithm performs even worse than the random search algorithm when the number of optimization problems increases to 10 or more.

In the past few decades, PSO algorithms have developed rapidly, and they are not only widely used in the field of multi-objective optimization, but also have a good performance on many-objective optimization problems. In summary, it can be divided into the following categories: 1)Combining effective strategies to improve the performance of particle swarm algorithm(Li et al., 2018; Li, Wang, & Li, 2019; Wang et al., 2017). This type of method is to increase the diversity and convergence of the particle swarm algorithm by using effective elite particle selection, local search, and Gaussian chaos mutation strategies. For example: Wang *et al.*(2018) proposed a hybrid PSO algorithm using adaptive learning strategy (ALPSO) in order to balance the global search ability and convergence of the PSO algorithm. The algorithm designed a tolerance-based search direction adjustment mechanism It improves the convergence accuracy and convergence speed of the PSO algorithm. 2) Add new particle speed and position update methods(Wang et al., 2017; Zhang et al., 2008). This type of method is an innovation of the original particle update formula of the PSO algorithm, so that the new particle speed and position update formula can greatly improve the optimization speed and

accuracy of particles. For example: Kumar *et al.*(2016) proposed a self-learning MOPSO algorithm to deal with multi-objective optimization problems. This algorithm proposed 4 kinds of calculation methods on particle speed and position update, which effectively improved the effectiveness of each particle search. 3) A method of adding user preferences(Wang *et al.*, 2019; Zhou *et al.*, 2018). The preference of such methods in multi-objective optimization means that the decision maker assigns different values to the goal to indicate its importance or priority order. It enables the search process or optimization results to be in the area of interest of the decision maker, but requires the user to specify preferences in advance or requires interaction with the search process, thereby increasing the difficulty for users to use the algorithm. For example, Ki-Baek Lee *et al.*(2011) proposed a preference-based solution selection algorithm (PSSA) in order to solve the defect that the number of non-dominated solutions increases exponentially with the increase in the number of MOPSO algorithms. The user's preference is added to the MOPSO algorithm, so that the user can choose the required solution from it, which greatly improves the efficiency of the algorithm. 4) Develop new fitness allocation strategies(Jaszkiewicz, 2002; Kaliszewski *et al.*, 2012). This type of method uses the individual's fitness allocation to guide the evolution of the algorithm, because the fitness allocation mechanism determines the performance of the algorithm. A good fitness allocation strategy can not only reduce the calculation time complexity of the algorithm, but also improve the algorithm's solution performance. For example, Wei *et al.*(2006) proposed an allocation strategy based on Pareto priority relationship ordering and an allocation strategy based on random weight summation. Later, more and more scholars carried out more in-depth research on multi-objective particle swarm optimization (MOPSO)(Figueiredo *et al.*, 2016; Lin *et al.*, 2018; Zhao, 2018). For example, Raquel *et al.*(2005) used crowding distance as the fitness value to sort particles; Li *et al.*(2017) used the method of integrating GMR (Global margin ranking) and particle density information of particles to sort particles, which can efficiently and quickly select pbest and gbest Particle; Lin *et al.*(2015) use aggregate function to process multiple objective functions; Dai *et al.*(2015) use the method of objective space decomposition to ensure the diversity of particle optimal solution; In addition, the MOPSO and other optimization algorithms are cross used to carry out collaborative optimization through a certain proportion, such as teaching and learning method(Cheng *et al.*, 2017), differential evolution method(Su & Chi, 2017), Gaussian mutation method(Jordehi, 2015; Wei, 2012), etc. Among the above types of methods, the preference method and the combination of effective strategies are commonly used methods, and each has its own advantages and disadvantages. The development of new particle speed and position update methods is not only poorly versatile and the effect cannot be guaranteed, and new adaptations are developed. The degree allocation strategy works well and the method is simple. In addition, the PSO algorithm based on the new fitness allocation strategy does not essentially change the original particle swarm algorithm but simply starts with fitness allocation, so it does not affect the integration of other algorithm update strategies or methods. For example, a decomposition strategy can be used to simplify the optimization problem. A new fitness allocation strategy can be used to enhance the performance of the algorithm. At the same time, a better particle update formula can be added to further improve the performance of the algorithm. Therefore, the new fitness allocation PSO algorithm has the potential to solve complex optimization problems.

Based on the existing PSO algorithm and its variants, this paper proposes a new high-dimensional particle swarm evolution algorithm based on the characteristics of many-objective optimization problems. This algorithm is different from the existing high-dimensional particle swarm evolution algorithms in that: 1) Combined with fuzzy information theory, a fitness allocation method based on fuzzy correlation entropy is proposed, which objectively allocates fitness values, which increases the pressure of population selection, eliminates the influence of external uncertain factors on the algorithm, simplifies the algorithm process, effectively avoids the disadvantage of insufficient population selection pressure caused by traditional fitness allocation, and makes the algorithm having better convergence. 2) A new multi criteria mutation strategy is introduced, which can effectively disturb the process of particle updating, enhance the diversity of the algorithm, and prevent the algorithm

from falling into local optimum. 3) Using three-point shortest path method to maintain the external files, further improve the quality of the whole solution set. The above three strategies constitute the main characteristics of FAMSHMPSO algorithm. They are implemented in different stages of the algorithm, taking into account the convergence and diversity of the algorithm, and improving the overall performance of the algorithm to solve many objective optimization problems. Section 1 of the paper introduces the theory of many-objective optimization problems. Section 2 describes the important components of the FAMSHMPSO algorithm and the algorithm flow. It is the key chapter of this article. Section 3 is the experimental results and analysis, and the last section is the conclusion of this article.

2. RELATED WORK OF FAMSHMPSO ALGORITHM

2.1. Basic Knowledge

Without loss of generality, a multi-objective optimization problem with n decision variables and m objective functions, taking the minimization problem as an example, can be expressed as the form of equation (1).

$$\begin{cases} \min y = F(x) = (f_1(x), f_2(x), \dots, f_m(x)) \\ x = (x_1, x_2, \dots, x_n) \in X \subset R^n \\ y = (y_1, y_2, \dots, y_m) \in Y \subset R^m \end{cases} \quad (1)$$

In equation (1), x is called decision vector, X is n -dimensional decision space; y is called objective function, Y is m -dimensional target space; objective function F defines mapping function and m targets that need to be optimized simultaneously. When $m \geq 4$, the formula (1) is called the many-objective optimization problem. If X is a connected and closed region in space R^n , and the objective function $f_i (i = 1, 2, \dots, m)$ is continuous with respect to x , then equation (1) is a continuous many-objective optimization problem. Unless otherwise stated, this article only discusses this type of optimization problem.

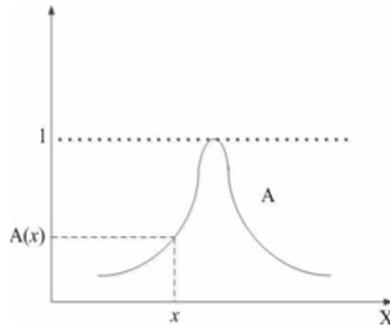
For any two points $x_1, x_2 \in X$ in the decision space, when the objective function of x_1 is not greater than and there is at least one objective function smaller than x_2 , x_1 Pareto is said to dominate x_2 , which is recorded as $x_1 \prec x_2$. If $x^* \in X$ is not subject to other individuals in the population, then x^* is called Pareto non-dominated solution, and the set of all non-dominated solutions in the population is called Pareto Set (PS), and the corresponding set of objective functions is called Pareto Front (PF).

2.2. FUZZY SET THEORY

Fuzzy Theory refers to the theory of using the basic concept of fuzzy set or continuous membership function, and also the method of studying fuzzy phenomena and inaccurate concepts. In the classical set theory of accurate mathematics in real life, there are only two possibilities for the relationship between an object and a set: belonging or not. The relationship between an object and a fuzzy set in fuzzy set theory of fuzzy mathematics is expressed as the degree to which the object belongs to the fuzzy set. Its scientific definition is:

Given a given domain U and any mapping $\mu_A : U \rightarrow [0, 1]$ of U to $[0, 1]$, then μ_A determines a fuzzy sub set A of U .

Figure 1. Fuzzy set schematic.



Where μ_A is called the membership function of A , and $\mu_A(u)(u \in U)$ is the degree to which u belongs to fuzzy sub set A . It is called the membership degree of u to A , and can also be directly recorded as $A(u)$. For the sake of easy understanding, give a simple example.

There are five classmates x_1, x_2, x_3, x_4, x_5 in a class group, that is, the domain is $U = \{x_1, x_2, x_3, x_4, x_5\}$. Now we assign a score to each student’s personality stability. Dividing the score by the percentage system and dividing it by 100 is actually a mapping from the to the closed interval, Table 1 shows the fuzzy set of this example.

This determines a fuzzy subset A , which indicates how well the group’s classmates fit into the vague concept of “stable character”.

Entropy is a measure of the uncertainty of things, and fuzzy information entropy is a measure of fuzzy sets on fuzzy information(Dadid & Cihan, 2014; Li et al., 2015). The fuzzy association entropy is proposed based on the fuzzy set theory. Its coefficient represents the similarity of the fuzzy information of the two fuzzy sets, and the uncertainty of the fuzzy association entropy mainly comes from internal rather than external factors. At present, some researchers have used fuzzy correlation entropy to solve problems in decision analysis, image processing, and fuzzy recognition. The results are good. These methods can be roughly divided into two categories: the association entropy method of rough fuzzy sets(Meng et al., 2014) and the association entropy method of intuitionistic fuzzy sets(Moubayed et al., 2014). These two kinds of association entropy methods require upper and lower

Table 1. Fuzzy set instance

U		$\mu_A : U \rightarrow [0, 1]$	
x_1	88	$\mu_A(x_1)$	0.88
x_2	77	$\mu_A(x_2)$	0.77
x_3	99	$\mu_A(x_3)$	0.99
x_4	44	$\mu_A(x_4)$	0.44
x_5	66	$\mu_A(x_5)$	0.66

approximation of two groups of memberships, but the relational entropy method of rough fuzzy sets is a very harsh condition in the application of equivalence relations (Qiu et al., 2015). But the binary relationship in the domain of many-objective optimization is not equivalent. The association entropy method of intuitionistic fuzzy sets needs to determine the attribute weight of the target first when calculating the correlation entropy coefficient. However, it is very difficult to calculate the attribute weight of the target in the many-objective optimization problem, and it is essentially affected by external uncertain factors.

2.3. FITNESS ALLOCATION METHOD BASED ON FUZZY ASSOCIATION ENTROPY

This article analyzes the difficulties in applying fuzzy association entropy method to many-objective optimization problems. It is found that if different sets are mapped to fuzzy sets based on fuzzy information entropy theory and membership function, By comparing the correlations of different fuzzy sets, calculating the similarity of the two fuzzy sets, and classifying different fuzzy sets, they can just make up for the shortcomings of the above two methods of association entropy.

Therefore, based on the above ideas, this paper proposes a fitness allocation method (FAMOF AE) based on fuzzy association entropy to solve many-objective optimization problems. The core idea of this method is to use the membership function of the fuzzy set to map the ideal solution in the many-objective optimization problem and the Pareto solution obtained by the many-objective optimization algorithm into a membership fuzzy set containing all the information. Then use the fuzzy information entropy theory to process the information contained in the fuzzy set, that is, based on the membership fuzzy sets corresponding to the ideal solution and the Pareto solution, to analyze the problem by determining how much the Pareto solution contains the ideal solution information (fuzzy association entropy coefficient). Finally, the fuzzy association entropy coefficient of the membership fuzzy sets is used as the fitness value of the Pareto solution to dominate the selection. It is considered that the Pareto solution with a large fuzzy association entropy coefficient is closer to the ideal solution, thereby guiding the algorithm evolution. The key of FAMOF AE to achieve many-objective optimization is to use the membership function of fuzzy association entropy to map the Pareto solution and ideal solution of many-objective optimization algorithm to membership fuzzy sets. The fuzzy sets corresponding to the Pareto solution and the ideal solution are effectively correlated, and the association between the Pareto solution and the ideal solution is fully mined, thereby solving the basic problem of applying fuzzy information association entropy to many-objective optimization.

The advantage of FAMOF AE is to use the fuzzy set theory to take the similarity between the ideal solution and the Pareto solution (fuzzy association entropy coefficient) as the fitness to simplify the evolution process of the algorithm. Improve the quality of the Pareto solution set, thereby avoiding the disadvantage of insufficient population selection pressure caused by traditional fitness allocation, and better guide the algorithm evolution.

The general process of FAMOF AE in the many-objective optimization process is as follows: First, each sub-objective of the many-objective problem is solved as a single objective to obtain the optimal solution, and then these optimal solutions correspond to the decision variables to form the ideal solution. Then, a multi-objective PSO algorithm is used to generate a set of Pareto solutions. The membership function of the fuzzy set theory is used to map the ideal solution and the Pareto solution into a membership fuzzy set containing all the information, and then the information contained in the fuzzy set is processed by the fuzzy information entropy theory. That is, based on the membership fuzzy set of the ideal solution and the Pareto solution, the problem is analyzed by determining how much of the ideal solution information the Pareto solution contains. Finally, the fuzzy correlation coefficient is used as the fitness value. It is considered that the Pareto solution with a large fuzzy correlation coefficient is closer to the ideal solution, and this solution is added to the external elite file to guide the evolution of the high-dimensional multi-target PSO algorithm. All in all, the fuzzy information entropy theory and membership function are used to fully mine the similarity between the Pareto solution and the ideal solution. The specific implementation process of FAMOF AE is as follows:

- Step 1: Determine the ideal solution F_B and Pareto solution F_i . The many-objective optimization problem contains M sub-goals. The single-objective PSO algorithm is used to obtain the k -sub-objective optimal solution function value corresponding to the decision variable so that the ideal solution $X^k = (x_1^k, x_2^k, x_3^k, \dots, x_N^k)$, so that the ideal solution $F_B = (X^1, X^2, X^3, \dots, X^M)$.
- Step 2: Similarly, any Group i Pareto solution generated by the multi-objective PSO algorithm is $F_i = \{X^1(i), X^2(i), X^3(i), \dots, X^M(i)\}$, and $X^k(i)$ is the decision variable corresponding to the k th sub-object of the Group i Pareto solution and $X^k(i) = (x_1^k(i), x_2^k(i), x_3^k(i), \dots, x_N^k(i))$.
- Step 3: Using the fuzzy set theory, calculate the membership degree of each sub-goal and establish the fuzzy set of membership degree. The membership function is used to extract the uncertainty of each sub-objective, and the normalized membership function is used to map each to the membership fuzzy set, eliminating the influence of each sub-target dimension and magnitude on the algorithm. The normalized membership function is used to calculate the membership degree of each sub-object as follows:

$$U_N(i) = \begin{cases} 0 & f_k(i) \leq f_k^{\min}(i) \text{ or } f_k(i) > f_k^{\max}(i) \\ \frac{f_k(i) - f_k^{\min}(i)}{f_k(B) - f_k^{\min}(i)} & f_k^{\min}(i) < f_k(i) \leq f_k(B) \\ \frac{f_k^{\max}(i) - f_k(i)}{f_k^{\max}(i) - f_k(B)} & f_k(B) < f_k(i) \leq f_k^{\max}(i) \end{cases} \quad (2)$$

Where $f_k(B)$ is the optimal solution function value of the k th sub-object of the ideal solution F_B , $f_k^{\min}(i)$ is the minimum function value of the k th sub-object of the Pareto solution of the i -group, and $f_k^{\max}(i)$ is the largest function of the k -th sub-object of the Pareto solution of the i -group Value, $U_N(k)$ is the k th target membership.

Step 4: Using the mapping result of equation (2), the membership degrees of the ideal solution F_B and the Pareto solution F_i sub-objects are respectively composed into the ideal solution membership degree fuzzy set FS_{F_B} and the Pareto solution membership degree fuzzy set FS_{F_i} , where $FS_{F_B} = \{U_{F_B}(1), U_{F_B}(2), U_{F_B}(3), \dots, U_{F_B}(k), \dots, U_{F_B}(M)\}$ and $FS_{F_i} = \{U_{F_i}(1), U_{F_i}(2), \dots, U_{F_i}(k), \dots, U_{F_i}(M)\}$. The fuzzy set of membership degree is equivalent to the rich information in the solution, which is the basis of the application of fuzzy information entropy theory to many-objective optimization.

Step 5: The uncertainty of the fuzzy set is calculated. Based on the fuzzy set of membership degree, the fuzzy entropy $E(F_B)$ and $E(F_i)$ corresponding to the ideal solution F_B and the Pareto solution F_i are calculated by using the fuzzy entropy formula.

$$E(A) = -K \sum_{k=1}^M [U_N(k) \ln U_N(k) + (1 - U_N(k)) \ln(1 - U_N(k))] \quad (3)$$

Where K is the normalization factor and $K > 0, K = 1 / M \ln 2$ and M are the number of sub-targets. After obtaining the fuzzy entropy $E(F_B)$ and $E(F_i)$ corresponding to the ideal solution F_B and the Pareto solution F_i , the fuzzy entropy of the two needs to be correlated to determine the similarity between the Pareto solution F_B and the ideal solution F_i . Then the fuzzy deviation entropy using the fuzzy set theory is very suitable, because the fuzzy deviation entropy is to determine the degree of deviation between the two fuzzy entropies, the formula is as follows:

The fuzzy deviation entropy formula of the fuzzy entropy of the Pareto solution F_i for the fuzzy entropy of the ideal solution F_B is as follows.

$$E_{F_B}(F_i) = -\sum_{k=1}^M [U_{F_B}(k) \ln U_{F_i}(k) + (1 - U_{F_B}(k)) \ln(1 - U_{F_i}(k))] \quad (4)$$

The fuzzy entropy formula of the fuzzy entropy of the ideal solution F_B for the fuzzy entropy of the Pareto solution F_i is as follows.

$$E_{F_i}(F_B) = -\sum_{k=1}^M [U_{F_i}(k) \ln U_{F_B}(k) + (1 - U_{F_i}(k)) \ln(1 - U_{F_B}(k))] \quad (5)$$

Fuzzy entropy and fuzzy deviation entropy in fuzzy set theory are both a measure of the uncertainty of fuzzy sets. It can be seen from the calculation process of fuzzy entropy and fuzzy deviation entropy that the fuzzy correlation coefficient method (FCCM) eliminates the problem that the algorithm is sensitive to the number of sub-targets and its given order in the multi-objective optimization process.

Step 6: After obtaining the partial entropy of Pareto solution F_i and ideal solution F_B , we can determine the similarity between Pareto solution F_i and ideal solution F_B , that is, fuzzy correlation coefficient. Fuzzy correlation coefficient can be expressed by fuzzy set theory association entropy formula.

$$C(F_i : F_B) = \frac{M \ln 2 [E(F_i) + E(F_B)]}{E_{F_B}(F_i) + E_{F_i}(F_B)} \quad (6)$$

Equation (3) considers the interaction between the sub-objects of the Pareto solution, and equations (4) and (5) consider the relationship between the Pareto solution and the sub-objects of the ideal solution. Formula (6) starts from the combination of the two, considers the relationship between the Pareto solution and the ideal solution, and makes full use of various information. In fact, the fuzzy correlation coefficient $C(F_i : F_B)$ is a measure of the similarity between the Pareto solution F_i and the ideal solution F_B . It can be seen from the implementation process of FCCM that it is a method using fuzzy set theory to calculate the amount of information that the Pareto solution contains the ideal solution. This method of fitness allocation starts from the Pareto solution and the ideal solution and its internal data of each sub-objective. It has no relationship with each sub-target weight, and has no relationship with the number of sub-objects and its given order, and only relates to the ideal solution and the Pareto solution itself. Different from other fitness allocation mechanisms, FCCM not only

compares the advantages and disadvantages between sub-goals, but also considers the similarity between Pareto solution and ideal solution. Taking multiple sub-goals as a unified mutual constraint, the population selection pressure is increased, the fitness value is objectively distributed, the influence of external uncertain factors on the algorithm is eliminated, and the algorithm process is simplified.

2.4. Multi-Criteria Mutation Strategy

From the whole particle algorithm, we can know that the role of individual optimal value and global optimal particle plays an important role in the whole population. In the update of particle individual optimal value and global optimal value, the standard MOPSO algorithm particles will often appear “premature” phenomenon, which makes the population converge prematurely and fall into local optimum. Many studies (Bin, 2010; Guo & Wang, 2017; Qing et al., 2017; Yang et al., 2017) have found that adding some disturbances when the particle position is updated can easily jump out of the local optimum to avoid “premature”. Therefore, this paper proposes a method of adding multi-criteria criteria to mutate the location update to make the particles dynamically update. The number of iterations is introduced into the mutation factor, so that the mutation factor works in a dynamic form. According to the change of the number of iterations, the particles are dynamically mutated according to different criteria. First, we must analyze the variation ability of each particle in each generation, and determine the variation factor of each generation of particles as shown in the following formula:

$$P_d(t) = 0.05 + \frac{\exp\left(\frac{5t-5}{T_{\max}-1}\right) - 1}{2(\exp(5) - 1)} \quad (7)$$

Where t is the current number of iterations, and T_{\max} is the maximum number of iterations of the algorithm. The idea of particle variation in the algorithm uses the following algorithm 1, where $pop(i, j)$, $pbest(i, j)$, $gbest(i, j)$ represent the current particle, the individual optimal particle, and the global optimal particle, respectively. $P_d(t)$ is a variation factor that is related to the number of iterations and varies with t . $normrnd(\mu, \sigma)$ is a normal distribution function formula in which the average value is μ and the standard deviation is σ in the Matlab function library. For example, $unifrnd(0, 1, 1, 1)$ refers to a function formula for generating a random number of 1 row and 1 column 0 to 1 in the Matlab function library.

Algorithm 1 Multi-criteria mutation strategy in HMOPSO

Input. pop (Initial population), N (Population size), M (Target dimension), μ (Mean value), σ (Standard deviation), $P_d(t)$ (Variation factor), T_{\max} (Maximum number of iterations).

Output. New position after particle variation.

1: The number of iterations is from 1 to the maximum number of iterations, and P_d is calculated in equation (7)

2: if $unifrnd(0,1,1,1) < 0.5$

3:

$pop(i, j) = normrnd\left(\frac{(1 - rand) * gbest(1, j) + rand * pbest(1, j)}{2}, |gbest(1, j) - pbest(i, j)|\right)$

4: else

5: $pop(i, j) = pbest(i, j)$

6: if $unifrnd(0,1,1,1) < P_d(t)$

7: $pop(i, j) = (1 + rand) * pbest(i, j) + normrnd(\mu_1, \sigma_1)$

8: else

```
9: pop(i, j) = normrnd( $\mu_2, \sigma_2$ ) * pop(i, j)
10: end
```

Through the above algorithmic mutation operation, the particles can be more quickly converge to the real Perato frontier, and the local optimum is better. Especially for such “multi-peak” functions, there are many pseudo-precursions, and the speed and position update of the standard particle swarm optimization algorithm often suffer from local optimum. After introducing the above variation at the location update, the drawbacks of “premature” convergence are well overcome, the convergence is accelerated, and the true frontier of Perato is well covered.

2.5. External File Update Method

The FAMSHMPSO algorithm sets an archive set outside the population to preserve the non-dominated solutions obtained by the algorithm during the search process. In this paper, the external file is updated using the progressive method, and algorithm 2 gives the update strategy for the external file.

Algorithm 2 Updation of external archive in HMOPSO

Input. N (Population size), M (Target dimension), C (The maximum capacity of the external file), x (The resulting non-inferior solution), T_{\max} (Maximum number of iterations).

Output. All non-inferior solutions in the external file.

```
1: if The external file size does not reach maximum capacity  $C$ 
2: Add individual  $x$  to the external file
3: Use the Pareto dominance relationship to update the file and
   remove the dominated individuals in the file
4: else
5: if Individual  $x$  dominates some individuals in the external
   file
6: Exclude all individuals in the file that are dominated by  $x$ ,
   add the new solution  $x$  to the external file
7: else if Individual  $x$  is non-dominated in relation to all
   members of the external file
8: Perform individual density estimates with individual  $x$  and
   archival members, and eliminate the worst-diverse individuals
9: else
10: Individual  $x$  is dominated by archival members, discarding
   individuals  $x$ 
11: end if
12: end if
13: end if
14: Output updated external file members
```

The eighth step of Algorithm 2 performs the individual density estimation using the three-point shortest path strategy. The specific method is as follows: the point i_1 in the target space is the solution point to be evaluated, wherein the point closest to the i_1 point is set to i_2 , the distance between the two is recorded as D_{i_1, i_2} , and the point closest to the i_2 point is i_3 . The distance between them is denoted as D_{i_2, i_3} , and if $i_1 \neq i_2 \neq i_3$ is satisfied, the three-point shortest path value of i_1 point can be defined as:

$$D_{i_1} = D_{i_1, i_2} + D_{i_2, i_3} - |D_{i_1, i_2} - D_{i_2, i_3}| \#(8)$$

Using equation (8) to evaluate the solution, the greater the distance value obtained by the individual, the more sparse the individual distribution. Moreover, the third term on the right side of (8) makes the distance values obtained by the uniformly distributed solution points larger than the distance values of the solution points whose distribution is not uniform. In addition, the three-point shortest path method emphasizes that the solution points involved in the distribution evaluation are three different points in the target space, which can eliminate the problem that the distances in the space are close to each other because they are regarded as the nearest neighbors, and the distribution cannot be accurately evaluated.

Figure 2 takes the 2-target space as an example to illustrate the effectiveness of the three-point shortest path method to evaluate the degree of solution. In Figure 2, b and c are close together and belong to a denser solution point. The path value of point c is calculated according to the three-point shortest path method as $D_c = D_{bc} + D_{ab} - |D_{ab} - D_{bc}|$. However, if the position of point b is moved to the point b' , the distribution of the three points a , b' and c is better, and the path value of point c should be $D'_c = D_{b'c} + D_{b'a} - |D_{b'a} - D_{b'c}|$. Because $|D_{b'a} - D_{b'c}| < |D_{ab} - D_{bc}|$, and $(D_{bc} + D_{ab}) = (D_{b'c} + D_{b'a})$, so $D'_c > D_c$. It is shown that the three-point shortest path value obtained by the uniformly distributed solution point is larger than the path value of those distributed uneven solution points.

Since the shortest path reflects the distance relationship between the individual and its surrounding individuals, it is reasonable to use the three-point shortest path value of the solution to represent the density of the solution point distribution in the target space. It is worth mentioning that using the three-point shortest path method to evaluate the distribution of the solution group does not need to introduce additional parameters, which saves the inconvenience of repeated adjustments.

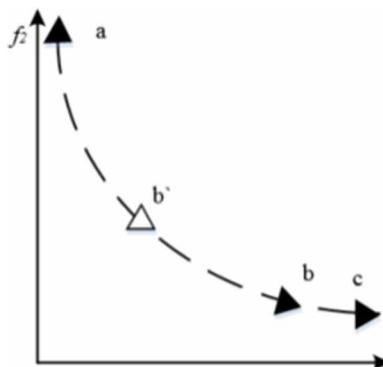
2.6. FAMSHMPSO Algorithm Flow

Algorithm 3 is the flow of the FAMSHMPSO algorithm based on the descriptions in Sections 2.1~2.5 above.

Algorithm 3 many-objective Particle Swarm Algorithm Based on New Fitness Allocation and Multiply Cooperative Strategies

Input. pop (Initial population), N (Population size), M (Target dimension), C (The maximum capacity of the external file), η (Variation distribution index), μ (Mean value), σ (Standard deviation), $P_d(t)$ (Variation factor), $c_{1,2}$ (Mutation probability), P (Mutation probability), ω (Inertia weight), T_{max} (Maximum number

Figure 2. Schematic diagram of three-point shortest path method



of iterations)

Output. Approximate Pareto solution set $PS: \{x_1, x_2, x_3, \dots, x_N\}$,
 approximate Pareto frontier $PF: \{F(x_1), F(x_2), F(x_3), \dots, F(x_N)\}$

1: Initialization:

1) The PSO algorithm is used as the single objective optimization algorithm to find the optimal solution of each target, and the ideal solution is formed. The ideal solution F_B is transformed into the membership fuzzy set FS_{F_B} by using formula (2)

2) When calculating the membership degree, the PSO algorithm is used as the single-objective optimization algorithm to optimize the f_k -subfunction 30 times. The minimum value is f_k^{\min} , and the average value is $f_k(B)$. The maximum value of all the first generations is f_k^{\max}

3) Randomly generate the initial population of N particles X_i^t , randomly select a group of particles to establish an external elite file

4) Initialization iteration counter $t = 0$

2: Update:

for $i = 1, 2, 3, \dots, N$ do

1) Demap Pareto to membership degree fuzzy set: For the current particle of the t th generation, the sub-objective function value of each particle is calculated by the many-objective particle algorithm, and the Pareto solution F_B is obtained. The F_i is mapped to the membership fuzzy set FS_{F_i} by using the formula (2)

2) Solving fitness values: Calculate the fuzzy correlation coefficient $C(F_i: F_B)$ of F_i with respect to F_B using equations (3) to (6) as the fitness value. For all particles, compare their individual historical optimal fitness values and population global optimal fitness values, and select the global optimal particles and the best position of the particles

3) Update current population particle position: Algorithm 1 is used to dynamically update the current group particle position

4) Update current population particle speed: The current population particle velocity is updated using the formula

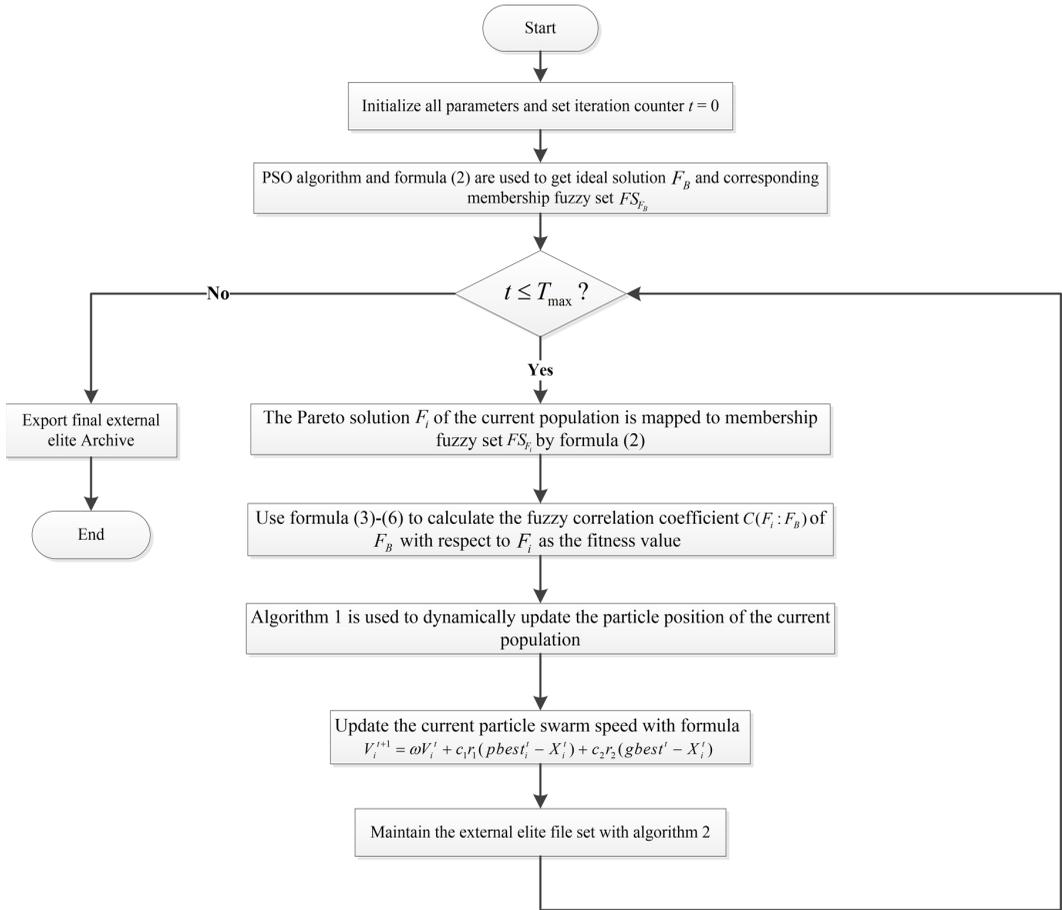
$$V_i^{t+1} = \omega V_i^t + c_1 r_1 (pbest_i^t - X_i^t) + c_2 r_2 (gbest^t - X_i^t)$$

5) Maintenance and dynamic update of external elite files: The non-inferior solution generated by FAMSHMPSO is compared with the particles in the external file. If the external file size is less than the current external file size less than C , the generated non-inferior solution is added to the external file, otherwise the external file is updated by algorithm 2

3: Judging:

Judging the iteration termination condition: When the number of contemporary iterations reaches, the algorithm ends, and the final external elite archives $\{x_1, x_2, x_3, \dots, x_N\}$ and $\{F(x_1), F(x_2), F(x_3), \dots, F(x_N)\}$ are output, otherwise the iteration counter is set. Go to step 2

Figure 3. Flow chart of FAMSHMPSO algorithm



to continue the update operation.

4: end

The flow chart of FAMSHMPSO algorithm is shown in Figure 3.

3. EXPERIMENTAL RESULTS AND ANALYSIS

3.1. Experiment Setup

3.1.1 Peer Comparison Algorithm

In order to test the performance of FAMSHMPSO algorithm, five algorithms such as PSO classical algorithm OMOPSO, PSO improved algorithm SMPSO, decomposition based D²MOPSO, improved non-inferior classification genetic algorithm NSGA-III and file-based hybrid decentralized search algorithm AbYSS are selected as peer-to-peer comparison algorithms. In this way, on the one hand, the performance of the algorithm and other algorithm examples of the same framework can be detected. On the other hand, the performance between the FAMSHMPSO algorithm and the well-known algorithms outside the framework can be examined.

3.1.2. High-Dimensional Multi-Target Test Function Set

In order to test the effectiveness of the proposed algorithm, the FAMSHMPSO algorithm is compared with the above five comparison algorithms on the DTLZ test function set of the 4, 10, and 30 targets. Use the generic test function DTLZ_{1,2,4,5} that can be extended to any target dimension and independent dimension. All experiments were run on a hardware configuration Intel I7-9700 CPU, 3.0GHz clocked, 8.0G RAM, Windows 10 64-bit operating system in Lenovo Tianyi Pro. 1) When the number of targets is 4, the parameters of the DTLZ function set are set as follows: the number of targets is 4, the dimension of the decision space is 10, and the number of iterations of the algorithm is 2000; 2) When the number of targets is 10, the parameters of the DTLZ function set are set as: the number of targets is 10, the dimension of the decision space is 20, and the number of algorithm iterations is 5000; 3) When the number of targets is 30, the DTLZ function set parameters are set to: the target number is 30, the decision space dimension is 50, and the number of iterations of the algorithm is 10000.

3.1.3. Performance

In the study of many-objective optimization, Hypervolume (HV) is often used to evaluate the performance of the algorithm because the HV indicator is a known one-dimensional evaluation indicator compatible with the Pareto relationship. However, the disadvantage is that the calculation of the HV value will consume a large amount of computing resources as the number of optimization targets increases (for example, the number of targets exceeds 10) and the number of non-dominated solutions is large (for example, the number of solutions exceeds 1000). The Inverted Generational Distance (IGD) evaluates the performance of the algorithm by measuring the distance between the real Pareto frontier and the approximate Pareto frontier obtained by the algorithm. The IGD indicator is more computationally efficient than the HV indicator because the IGD indicator is obtained by calculating the distance between two points, and the distance calculation is generally more efficient, even in high-dimensional multi-target spaces. However, it must be pointed out that the biggest drawback of the IGD indicator compared to the HV indicator is that it does not have Pareto compatibility. In order to use the IGD index to calculate high efficiency while overcoming its incompatibility with Pareto optimality, the literature(Liguo & Weidong, 2017) uses the improved IGD index, the IGD⁺ indicator, to evaluate the performance of the algorithm. IGD⁺ not only calculates the distance between two points, but also considers the Pareto relationship between them. Therefore, the IGD⁺ index has weak Pareto compatibility, which improves the accuracy of the performance of the measurement algorithm.

For a minimized multi-objective optimization problem that satisfies equation (1), let m be the number of optimization targets, Z is the set of reference points obtained from the uniform sampling of the known real Pareto front, and A is the approximate Pareto obtained by the algorithm. $z_j = (z_1, z_2, z_3, \dots, z_m) \in Z, j = 1, 2, 3, \dots, |Z|$, $a_i = (a_1, a_2, a_3, \dots, a_m) \in A, i = 1, 2, 3, \dots, |A|$ For the solution set, the IGD⁺ indicator can be calculated by the formula (9).

$$IGD^+(A) = \frac{1}{|Z|} \sum_{j=1}^{|Z|} \min_{a_i \in A} d_{IGD^+}(a_i, z_j) \quad (9)$$

Where $d_{IGD^+}(a_i, z_j) = \sqrt{\sum_{k=1}^m (\max\{\frac{a_k - z_k}{f_k^{\max} - f_k^{\min}}, 0\})^2}$ is the normalized Euclidean distance, f_m^{\max} and f_m^{\min} and respectively represent the maximum and minimum values obtained by the set Z on the m th target. In general, the smaller the IGD⁺ index value, the better the convergence and diversity of the approximate Pareto frontier. The experiment calculates IGD⁺ values by sampling 10,000 evenly

Table 2. Parameter settings of Six peer algorithms

Algorithm	Parameter settings
OMOPSO	$N = 100, M \in \{4, 10, 30\}, C_1 \in [1.5, 2.5], C_2 \in [1.5, 2.5], p_m = 1 / n, \eta_m = 20$
SMP SO	$N = 100, M \in \{4, 10, 30\}, C_1 \in [1.5, 2.5], C_2 \in [1.5, 2.5], p_m = 1 / n, \eta_m = 20$
D ² MOPSO	$N = 100, M \in \{4, 10, 30\}, C_1 \in [1.5, 2.5], C_2 \in [1.5, 2.5], p_m = 1 / n, \eta_m = 20$
NSGA-III	$N = 100, M \in \{4, 10, 30\}, p_c = 0.9, p_m = 1 / n, \eta_m = 20$
AbYSS	$N = 100, M \in \{4, 10, 30\}, N_{Re fSet1,2} = 10, p_c = 0.9, p_m = 1 / n, \eta_c = 20, \eta_m = 20$
FAMSHMPSO	$N = 100, M \in \{4, 10, 30\}, C_1 \in [1.5, 2.5], C_2 \in [1.5, 2.5], P_d(t), p = 1 / n, \eta_m = 20, \mu = 0, \sigma = 1, \omega \in [0.8, 1.2]$

distributed Pareto optimal solution points for each test function as an approximation of the real Pareto front.

Experimental parameters. The main parameters of the five comparison algorithms are based on the suggested values in the corresponding references, as shown in Table 2.

3.2. Experiment And Analysis

Tables 3 to 6 respectively calculate the IGD⁺ mean and standard deviation (std) obtained when the function DTLZ{1, 2, 4, 5} has 4, 10, and 30 targets. The “+”, “-” and “≈” in the table indicate that the results obtained by the algorithm are significantly better than, inferior to, and similar to the results obtained by the FAMSHMPSO algorithm for the Wilcoxon rank sum test (Saha et al., 2016).

It can be seen from Table 3 that the FAMSHMPSO algorithm obtains the optimal IGD⁺ mean on the DTLZ1 problem of 4-target, 10-target and 30-target, respectively. None of the other comparison algorithms OMOPSO, SMP SO, D²MOPSO, NSGA-III and AbYSS can get the best IGD⁺ mean. Therefore, the FAMSHMPSO algorithm is superior to the five comparison algorithms in general. Then, by ranking the IGD⁺ values obtained by each algorithm in the three test cases of DTLZ1(4, 10, 30), it is found that the best performing is the FAMSHMPSO algorithm, followed by NSGA-III, OMOPSO, D²MOPSO, SMP SO and AbYSS. From the results of “better/worst/similar” in Table 3, the results of the comparison algorithms on the three test functions of DTLZ(4, 10, 30) are “0/3/0”. This indicates that the proposed algorithm has significant IGD⁺ performance advantages over the other comparison algorithms on the DTLZ1 problem of the three high-dimensional multi-targets. In addition, because the Pareto front of the DTLZ1 problem has linear and multi-modal features, the FAMSHMPSO algorithm is more advantageous than other algorithms in solving the MaOP problem of such problem features.

It can be seen from Table 4 that the FAMSHMPSO algorithm obtains the optimal IGD⁺ mean value for the 10-target and 30-target DTLZ2 problems, respectively. The NSGA-III algorithm obtains the optimal IGD⁺ mean on the 4-target DTLZ2 problem, so the FAMSHMPSO algorithm is superior to the three comparison algorithms in general. Then, the IGD⁺ values are ranked in the three test cases of DTLZ2(4, 10, 30) by statistical algorithms. The first is FAMSHMPSO algorithm, followed by NSGA-III, D²MOPSO, OMOPSO, SMP SO and AbYSS. Finally, from the “better/worst/similar” results in Table 4, only the result of NSGA-III is “1/1/1”, and the results of the other four comparison algorithms are all “0/3/0”. This indicates that the proposed algorithm has significant IGD⁺ performance

Table 3. Comparison of IGD⁺ on DTLZ1 instance for six algorithms

Target number / Algorithm		OMOPSO	SMP SO	D ² MOPSO	NSGA-III	AbYSS	FAMSHMPSO
4-target	<i>mean</i>	1.7157e-03	3.5836e-03	2.9147e-03	8.3356e-04	4.3211e-03	4.3336e-04
	<i>std</i>	1.3176e-03	4.6666e-03	5.5514e-03	3.2300e-03	3.2121e-03	1.7768e-04
	<i>rank</i>	3-	5-	4-	2-	6-	1
10-target	<i>mean</i>	7.8741e-03	1.0410e-02	1.1016e-02	5.9321e-03	2.7916e-02	1.1935e-03
	<i>std.</i>	5.1616e-03	3.5171e-03	4.6144e-03	3.0403e-03	3.6488e-03	2.5443e-03
	<i>rank</i>	3-	4-	5-	2-	6-	1
30-target	<i>mean</i>	3.9021e-02	5.9768e-02	4.8112e-02	3.4005e-02	9.5716e-02	1.1171e-02
	<i>std</i>	7.4942e-03	4.5141e-03	7.9218e-03	6.4342e-03	3.2343e-03	1.5141e-02
	<i>rank</i>	3-	5-	4-	2-	6-	1
<i>rank suam</i>		9	14	13	6	18	3
<i>final rank</i>		3	5	4	2	6	1
<i>better/worst/similar</i>		0/3/0	0/3/0	0/3/0	0/3/0	0/3/0	/

Table 4. Comparison of IGD⁺ on DTLZ2 instance for six algorithms

Target number / Algorithm		OMOPSO	SMP SO	D ² MOPSO	NSGA-III	AbYSS	FAMSHMPSO
4-target	<i>mean</i>	6.1241e-02	6.9877e-02	5.1514e-02	5.9836e-03	8.9126e-02	8.7818e-03
	<i>std</i>	4.7318e-03	2.1453e-02	3.7435e-03	1.1951e-03	6.5144e-03	7.1352e-03
	<i>rank</i>	4-	5-	3-	1+	6-	2
10-target	<i>mean</i>	7.9101e-02	8.6387e-02	7.1005e-02	5.2553e-02	1.2019e-01	3.0265e-02
	<i>std.</i>	6.1212e-03	4.3237e-02	5.1505e-03	4.3036e-04	6.6401e-02	3.1223e-03
	<i>rank</i>	4-	5-	3-	2-	6-	1
30-target	<i>mean</i>	3.1112e-01	5.6624e-01	3.6076e-01	1.6683e-01	6.5447e-01	1.4127e-01
	<i>std</i>	5.5123e-02	3.2631e-02	6.6014e-02	7.4441e-02	9.1387e-02	2.1544e-02
	<i>rank</i>	3-	5-	4-	2≈	6-	1
<i>rank suam</i>		11	15	10	5	18	4
<i>final rank</i>		4	5	3	2	6	1
<i>better/worst/similar</i>		0/3/0	0/3/0	0/3/0	1/1/1	0/3/0	/

advantages over the other high-dimensional multi-target DTLZ2 problems compared with other comparison algorithms. In addition, since the DTLZ2 problem has a concave Pareto front, this also shows that the FAMSHMPSO algorithm can better solve the MaOP problem of such problem features.

It can be seen from Table 5 that the FAMSHMPSO algorithm obtains the best IGD⁺ mean on the 10-

target DTLZ4 problem, and the NSGA-III algorithm obtains the optimal IGD⁺ mean on the 4-target and 30-target DTLZ4 functions. So overall, although the IGD⁺ mean of the FAMSHMPSO algorithm is worse than the NSGA-III algorithm, it is better than the other four comparison algorithms.

Table 5. Comparison of IGD⁺ on DTLZ4 instance for six algorithms

Target number / Algorithm	OMOPSO	SMPSO	D ² MOPSO	NSGA-III	AbYSS	FAMSHMPSO	
4-target	<i>mean</i>	1.1026e-02	9.8532e-03	5.2025e-03	2.1401e-03	7.7504e-03	4.2549e-03
	<i>std</i>	3.7543e-03	4.1915e-03	1.8091e-02	2.0702e-03	1.8342e-03	7.1978e-04
	<i>rank</i>	6-	5-	3≈	1+	4-	2
10-target	<i>mean</i>	5.9273e-02	4.6712e-02	1.3102e-02	8.2166e-03	2.3381e-02	5.3742e-03
	<i>std.</i>	3.6101e-03	4.5291e-03	4.2633e-03	2.1165e-03	4.1736e-03	4.1525e-03
	<i>rank</i>	6-	5-	3-	2-	4-	1
30-target	<i>mean</i>	4.1155e-01	2.6824e-01	8.8388e-02	5.1026e-02	9.5523e-02	7.1031e-02
	<i>std</i>	5.0329e-02	8.3274e-02	6.5771e-02	2.1512e-02	8.6212e-03	5.9161e-02
	<i>rank</i>	6-	5-	3≈	1+	4-	2
<i>rank suam</i>	18	15	9	4	12	5	
<i>final rank</i>	6	5	3	1	4	2	
<i>better/worst/similar</i>	0/3/0	0/3/0	0/1/2	2/1/0	0/3/0	/	

Table 6. Comparison of IGD⁺ on DTLZ5 instance for six algorithms

Target number / Algorithm	OMOPSO	SMPSO	D ² MOPSO	NSGA-III	AbYSS	FAMSHMPSO	
4-target	<i>mean</i>	7.3013e-03	8.1382e-03	5.6332e-03	4.1688e-03	6.9742e-03	3.4491e-03
	<i>std</i>	9.5611e-04	7.1461e-04	8.2194e-04	8.4131e-03	7.8432e-03	5.8178e-04
	<i>rank</i>	5-	6-	3-	2≈	4-	1
10-target	<i>mean</i>	4.9896e-02	4.7126e-02	3.0211e-02	1.0228e-02	2.2117e-02	8.6621e-03
	<i>std.</i>	6.1816e-03	3.3478e-03	6.1755e-03	4.4174e-03	5.8814e-03	2.4125e-03
	<i>rank</i>	5-	6-	4-	2-	3-	1
30-target	<i>mean</i>	3.0236e-01	2.9641e-01	9.3868e-02	6.9677e-02	1.3267e-01	3.372e-02
	<i>std</i>	3.4495e-02	4.2524e-02	5.1150e-02	5.2323e-02	2.7875e-02	9.8781e-03
	<i>rank</i>	6-	5-	3-	2-	4-	1
<i>rank suam</i>	16	17	10	6	9	3	
<i>final rank</i>	5	6	3	2	4	1	
<i>better/worst/similar</i>	0/3/0	0/3/0	0/3/0	0/2/1	0/3/0	/	

Then, by counting the ranking of IGD⁺ values obtained by each algorithm on DTLZ4(4,10,30), NSGA-III algorithm ranks first, followed by FAMSHMPSO algorithm, followed by D²MOPSO, AbYSS, SMPSO, and OMOPSO. Finally, from the “better/worst/similar” results in Table 5, except for the result of NSGA-III is “2/1/0”, the results of other comparison algorithms are not as good as the FAMSHMPSO algorithm. This indicates that the proposed algorithm is only worse than NSGA-III in the three test problems of DTLZ4(4, 10, 30), but it is better than other comparison algorithms. In addition, because the Pareto front of the DTLZ4 problem is concave and irregular, it also shows that the FAMSHMPSO algorithm has certain advantages in solving the MaOP problem of this problem.

It can be seen from Table 6 that the FAMSHMPSO algorithm in this paper obtains the best IGD^+ mean on the three test questions of DTLZ5(4, 10, 30). Therefore, the FAMSHMPSO algorithm is superior to the three comparison algorithms in general. Then, by statistically calculating the final order of the IGD^+ values obtained by the three algorithms in DTLZ5(4, 10, 30), the best performance is the FAMSHMPSO algorithm, followed by NSGA-III, D²MOPSO, AbYSS, OMOPSO and SMPSO. Finally, from the results of “better/worst/similar” in Table 6, only the result of NSGA-III is “0/2/1”, and the results of the other two comparison algorithms are “0/3/0”. This indicates that the FAMSHMPSO algorithm has significant IGD^+ performance advantages on these three high-dimensional multi-target DTLZ5 problems. In addition, since the DTLZ5 problem has a concave and degenerate Pareto frontier, this also shows that the FAMSHMPSO algorithm can better solve the MaOP problem of such problem features.

From Tables 3 to 6, it can be clearly seen that the FAMSHMPSO algorithm has a good IGD^+ value and ranking in the DTLZ{1,2,4,5} test set, whether it is a 4-target, 10-target test function, or a 30-target test function. In addition, since the Pareto front of the DTLZ{1,2,4,5} test set has different shapes, for example, the DTLZ1 test problem has a convex Pareto front, and the DTLZ2 test problem has a concave Pareto front. Therefore, the FAMSHMPSO algorithm has good robustness when solving MaOP problems with different problem characteristics.

Based on the experimental results in Tables 3 to 6, the FAMSHMPSO algorithm has significant convergence and diversity advantages over the other five representative algorithms. In order to further investigate the stability of each algorithm’s problem solving, Table 7 shows the statistics results of the IGD^+ values obtained by independently running 30 times on each of the 4, 10, and 30 target problems on the DTLZ{1, 2, 4, 5} series functions in the form of a box plot. For the convenience of illustration, OMOPSO, SMPSO, D²MOPSO, NSGA-III, AbYSS and FAMSHMPSO are simply abbreviated as A1, A2, A3, A4, A5 and A6.

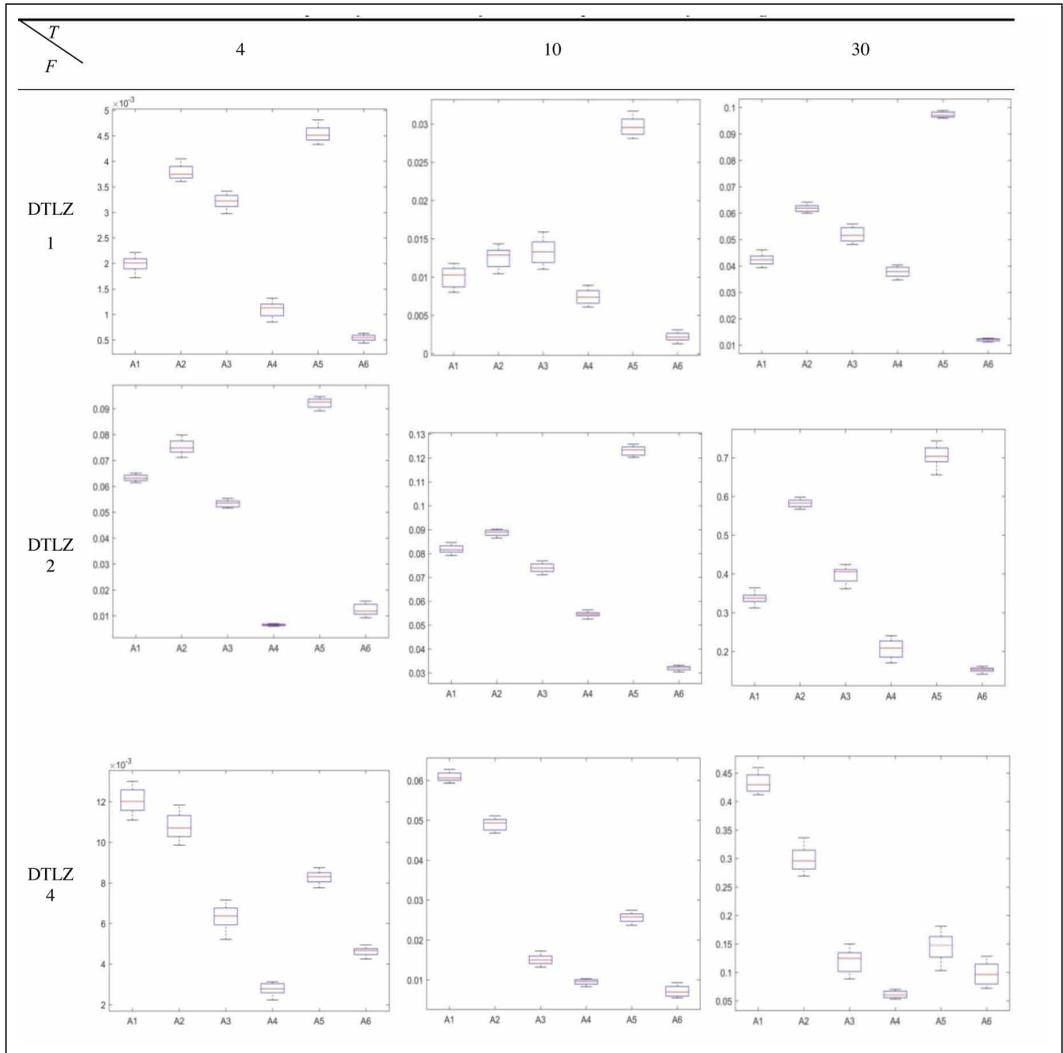
It can be seen from Table 7 that the FAMSHMPSO algorithm has the best stability of the IGD^+ values obtained on the DTLZ1, DTLZ4 and DTLZ5 test cases for the 4-target DTLZ{1, 2, 4, 5} problem. The NSGA-III algorithm achieves the best stability on the DTLZ2 problem. The stability of the proposed algorithm on the DTLZ2 problem is second only to the NSGA-III algorithm, but better than several other comparison algorithms. For the 10-target DTLZ{1, 2, 4, 5} problem, the algorithm has the best stability of IGD^+ values on the three test problems of DTLZ1, DTLZ2 and DTLZ5. On the DTLZ4 problem, the stability of the IGD^+ value of the FAMSHMPSO algorithm is slightly inferior to the NSGA-III algorithm, and is equivalent to the D2MOPSO algorithm. For the 30-target DTLZ{1, 2, 4, 5} problem, the FAMSHMPSO algorithm has the best stability for the IGD^+ values obtained on the three functions DTLZ1, DTLZ2 and DTLZ5. The IGD^+ value of the AbYSS algorithm on DTLZ4 has the best stability, and the stability of the IGD^+ value obtained by the algorithm on the DTLZ4 problem is slightly inferior to that of the AbYSS, and is equivalent to the D2MOPSO algorithm. It is not difficult to see from Table 7 that the FAMSHMPSO algorithm achieves the best IGD^+ value stability on most MaOP problems.

4. CONCLUSION

In order to solve MaOPs with satisfactory convergence, diversity, and efficiency at the same time, we proposed a new Multi-objective evolutionary algorithm--FAMSHMPSO. Through extensive comparative experiments on the DTLZ problems with varied numbers of objectives, FAMSHMPSO is shown to be good at coping with MaOPs in all aspects. The main novelties of this paper include the following.

- **A new fitness allocation method:** The core idea of this method is to use the membership function of the fuzzy set to map the ideal solution in the many-objective optimization problem and the Pareto solution obtained by the many-objective optimization algorithm into a membership

Table 7. Box plots of IGD* derived from 30 independent runs for six algorithms



fuzzy set containing all the information. Then the fuzzy association entropy coefficient of the membership fuzzy sets is used as the fitness value of the Pareto solution to dominate the selection. It is considered that the Pareto solution with a large fuzzy association entropy coefficient is closer to the ideal solution, thereby guiding the algorithm evolution.

- **A new multi-criteria mutation strategy:** The number of iterations is introduced into the mutation factor, so that the mutation factor works in a dynamic form. According to the change of the number of iterations, the particles are dynamically mutated according to different criteria., effectively avoiding the premature phenomenon of the algorithm.
- **Three-point shortest path method:** The 3-point shortest path method is used to maintain the external file. Since the shortest path reflects the distance relationship between the individual and its surrounding individuals, it is reasonable to use the three-point shortest path value of the solution to represent the density of the solution point distribution in the target space. It is worth mentioning that using the three-point shortest path method to evaluate the distribution of the

solution group does not need to introduce additional parameters, which saves the inconvenience of repeated adjustments.

Although FAMSHMPSO performs very well on MaOPs, it has several disadvantages that needs further improvement in the future. The performance of FAMSHMPSO on several DTLZ problems is not quite satisfactory, because of less-well extents on PFs. In the future, we will mainly study the following aspects: 1) Improve the new fitness allocation method of the article under constraints or dynamic environment; 2) Study the new dominance relationship method under high-dimensional goals; 3) Apply FAMSHMPSO to solve practical problems in a certain field; 4) FAMSHMPSO did not keep extreme points, in contrast to NSGA-III. We can borrow the extreme point maintenance scheme in NSGA-III to improve our FAMSHMPSO further.

REFERENCES

- Bin, J. B. J. (2010). *Job Shop Scheduling Based on an Improved Cooperative Particle Swarm Optimization*. Academic Press.
- Cai, D., Wang, Y., & Miao, Y. (2015). A new multi-objective particle swarm optimization algorithm based on decomposition. *Information Sciences*, 325, 541–557. doi:10.1016/j.ins.2015.07.018
- Cheng, T., Chen, M., & Fleming, P. J. (2017). A novel hybrid teaching learning based multi-objective particle swarm optimization. *Neurocomputing*, 222(26), 11–25.
- Dadid, M. C., & Cihan, H. D. (2014). Computational Complexity Measures for Many-objective Optimization Problem. *Procedia Computer Science*, 36(3), 185–191.
- Figueiredo, E. M. N., Luderemir, T. B., & Bastos-Filho, C. J. A. (2016). Many Objective Particle Swarm Optimization. *Information Sciences*, 374, 115–134. doi:10.1016/j.ins.2016.09.026
- Guo, J., & Wang, B. (2017). Particle Swarm Optimization with Gaussian Disturbance. In *2017 International Conference on Industrial Informatics - Computing Technology, Intelligent Technology, Industrial Information Integration (ICIICII)*. IEEE Computer Society. doi:10.1109/ICIICII.2017.81
- Jaszkiwicz, A. (2002). Genetic local search for multi-objective combinatorial optimization. *European Journal of Operational Research*, 137(1), 50–71. doi:10.1016/S0377-2217(01)00104-7
- Jordehi, R. A. (2015). Enhanced leader PSO (ELPSO): A new PSO variant for solving global optimisation problems. *Applied Soft Computing*, 26, 401–417. doi:10.1016/j.asoc.2014.10.026
- Kaliszewski, I., Miroforidis, J., & Podkopaev, D. (2012). Interactive Multiple Criteria Decision Making Based on Preference Driven Evolutionary Multi-objective Optimization with Controllable Accuracy. *European Journal of Operational Research*, 216(1), 188–199.
- Kaur, M., & Kadam, S. (2018). A novel multi-objective bacteria foraging optimization algorithm (MOBFOA) for multi-objective scheduling. *Applied Soft Computing*, 66, 66. doi:10.1016/j.asoc.2018.02.011
- Kennedy, J., & Eberhart, R. (1944). Particle swarm optimization. Particle swarm optimization. *Neural Networks, 1995 Proceedings, IEEE International Conference on*, 4, 1942–1948.
- Kumar, R. S., Kondapaneni, K., Dixit, V., Goswami, A., Thakur, L., & Tiwari, M. (2016). Multi-objective modeling of production and pollution routing problem with time window: A self-learning particle swarm optimization approach. *Computers & Industrial Engineering*, 99, 29–40. doi:10.1016/j.cie.2015.07.003
- Lee, K. B., & Kim, J. H. (2011). Multi-objective particle swarm optimization with preference-based sorting. *Proceedings of the IEEE Congress on Evolutionary Computation*. doi:10.1109/CEC.2011.5949929
- Li, H., Ding, M., & Deng, J. (2015). On the use of random weights in MOEA/D. In *2015 IEEE Congress on Evolutionary Computation (CEC)*. IEEE. doi:10.1109/CEC.2015.7256996
- Li, K., Chen, Y., Li, W., He, J., & Xue, Y. (2018). Improved gene expression programming to solve the inverse problem for ordinary differential equations. *Swarm and Evolutionary Computation*, 38, 231–239. doi:10.1016/j.swevo.2017.07.005
- Li, K., Liang, Z., Yang, S., Chen, Z., Wang, H., & Lin, Z. (2019). Performance Analyses of Differential Evolution Algorithm Based on Dynamic Fitness Landscape. *International Journal of Cognitive Informatics and Natural Intelligence*, 13(1), 36–61. doi:10.4018/IJCINI.2019010104
- Li, K., Wang, H., & Li, S. (2019). A mobile node localization algorithm based on an overlapping self-adjustment mechanism. *Information Sciences*, 481, 635–649. doi:10.1016/j.ins.2018.12.006
- Li, L., Wang, W., & Xu, X. (2017). Multi-objective Particle swarm optimization based on global margin ranking. *Information Sciences*, 375, 30–47. doi:10.1016/j.ins.2016.08.043
- Liguo, L. Y. U., & Weidong, J. I. (2017). Improved particle swarm optimization algorithm combined centroid and Cauchy mutation. *Jisuanji Yingyong*.

- Lin, Q., Li, J., Du, Z., Chen, J., & Ming, Z. (2015). A novel multi-objective particle swarm optimization with multiple search strategies. *European Journal of Operational Research*, 247(3), S0377221715006256. doi:10.1016/j.ejor.2015.06.071
- Lin, Q., Liu, S., Zhu, Q., Tang, C., Song, R., Chen, J., Coello, C. A. C., Wong, K.-C., & Zhang, J. (2018). Particle Swarm Optimization With a Balanceable Fitness Estimation for Many-Objective Optimization Problems. *IEEE Transactions on Evolutionary Computation*, 22(1), 32–46. doi:10.1109/TEVC.2016.2631279
- Meng, Y., Yang, S., & Shi, P. (2014). Multi-objective optimization of emergency evacuation using improved genetic algorithm. *Wuhan Daxue Xuebao*, 39(2), 201–205.
- Moubayed, N. A., Petrovski, A., & Mccall, J. (2014). D2MOPSO: MOPSO Based on Decomposition and Dominance with Archiving Using Crowding Distance in Objective and Solution Spaces. *Evolutionary Computation*, 22(1), 47–77. doi:10.1162/EVCO_a_00104 PMID:23614775
- Nayyar, A., Garg, S., Gupta, D., & Khanna, A. (2018). Evolutionary computation: theory and algorithms. In *Advances in Swarm Intelligence for Optimizing Problems in Computer Science* (pp. 1–26). Chapman and Hall/CRC. doi:10.1201/9780429445927-1
- Nayyar, A., Le, D. N., & Nguyen, N. G. (Eds.). (2018). *Advances in swarm intelligence for optimizing problems in computer science*. CRC Press. doi:10.1201/9780429445927
- Nayyar, A., & Nguyen, N. G. (2018a). Introduction to swarm intelligence. *Advances in Swarm Intelligence for Optimizing Problems in Computer Science*, 53-78.
- Nayyar, A., & Nguyen, N. G. (2018b). Introduction to swarm intelligence. In *Swarm Intelligence and Evolutionary Algorithms in Healthcare and Drug Development*. CRC Press. doi:10.1201/9780429445927-3
- Nayyar, A., & Singh, R. (2016, March). Ant colony optimization—computational swarm intelligence technique. In *2016 3rd International conference on computing for sustainable global development (INDIACom)* (pp. 1493-1499). IEEE.
- Qing, G., Zhu, S., & Juan, C. (2017). Parameter tuning of linear active disturbance rejection controller based on chaotic quantum behaved particle swarm optimization. In *2017 29th Chinese Control And Decision Conference (CCDC)*. IEEE.
- Qiu, F.Y., Mo, L.P., Jian, B., & Wang, L.P. (2015). Multi-Objective Particle Swarm Optimization Algorithm Using Large Scale Variable Decomposition. *Chinese Journal of Computers*, 38.
- Raquel, C. R. (2005). An effective use of crowding distance in multiobjective particle swarm optimization. *Genetic and Evolutionary Computation Conference*. doi:10.1145/1068009.1068047
- Saha, S., Seal, D. B., Ghosh, A., & Dey, K. N. (2016). A novel gene ranking method using Wilcoxon rank sum test and genetic algorithm. *International Journal of Bioinformatics Research and Applications*, 12(3), 263–279. doi:10.1504/IJBRA.2016.078236
- Su, Y. X., & Chi, R. (2017). Multi-objective particle swarm-differential evolution algorithm. *Neural Computing & Applications*, 28(2), 407–418. doi:10.1007/s00521-015-2073-y
- Wang, F., Li, Y., Zhang, H., Hu, T., & Shen, X. (2019). An adaptive weight vector guided evolutionary algorithm for preference-based multi-objective optimization. *Swarm and Evolutionary Computation*, 49, 220–233. doi:10.1016/j.swevo.2019.06.009
- Wang, F., Zhang, H., Li, K., Lin, Z., Yang, J., & Shen, X.-L. (2018). A Hybrid Particle Swarm Optimization Algorithm Using Adaptive Learning Strategy. *Information Sciences*, 436-437, 162–177. doi:10.1016/j.ins.2018.01.027
- Wang, F., Zhang, H., Li, Y., Zhao, Y., & Rao, Q. (2018). External archive matching strategy for MOEA/D. *Soft Computing*, 22(23), 7833–7846. doi:10.1007/s00500-018-3499-9
- Wang, F., Zhang, Y., Rao, Q., & Zhang, H. (2017). Exploring Mutual Information based Sentimental Analysis with Kernel based Extreme Learning Machine for Stock Prediction. *Soft Computing*, 21(12), 3193–3205. doi:10.1007/s00500-015-2003-z

- Wei, Z. (2012). San Ye. Quantum-Behaved Particle Swarm Optimization Algorithm with Adaptive Mutation Based on q-Gaussian Distribution. *Chinese Journal of Electronics*, 21(3), 449–452.
- Wu, H., Kuang, L., Wang, F., Rao, Q., Gong, M., & Li, Y. (2017). A Multiobjective Box-Covering Algorithm for Fractal Modularity on Complex Networks. *Applied Soft Computing*, 61, 294–313. doi:10.1016/j.asoc.2017.07.034
- Yang, Mu, & Che. (2017). Improved multi-objective particle swarm optimization algorithm based on multiple strategies. *Kongzhi Yu Juece/Control & Decision*, 32(3), 435-442.
- Zhang, Y. F., Xue, Q. S., & Xiong, Z. Y. (2008). Dynamic particle swarm optimization based on tabu search. *Computer Engineering and Applications*, 44(24), 56–58.
- Zhao, Y. (2018). Many objective particle swarm optimization algorithm based on preference. *Proceedings of the 37th China Control Conference*, 1456-1462.
- Zhong, W., Xu, X., & Deng, S. (2006). Evolutionary Algorithm for Multi-objective Hybrid Flow Shop Job Scheduling. *Jisuanji Jicheng Zhizao Xitong*, 12(08), 1227–1234.
- Zhou, J., Wang, F., Xu, J., Yan, Y., & Zhu, H. (2018). A Novel Character Segmentation Method for Serial Number on Banknotes with Complex Background. *Journal of Ambient Intelligence and Humanized Computing*, 1–15.

Weiwei Yu is a computer doctoral student of Beihang University, Beijing, China. His current research interests include Swarm Intelligence, and multi-objective optimization.

Zhang Li, female, doctor, Professor, doctoral supervisor. She received his doctorate in 1996. Professor in 2002. At present, she is vice president of software college, director of Software Engineering Research Institute of computer college, member of software engineering professional teaching steering committee of Ministry of education, and leader of graduate education cooperation group of national software engineering professional degree. National Computer Engineering Education Certification expert. Member of the software engineering committee of the computer society, deputy director of the education committee, and member of the cloud computing Expert Committee of the Chinese Electronics Society. Executive director of the National Institute of computer education. Member of the international information processing Union (IFIP), member of the European Virtual Laboratory for international enterprise interoperability (v-lab).

Chengwang Xie received the M.Sc. degree from Wuhan University of Technology, Wuhan, China, in 2005, and the Ph.D. degree from Wuhan University, Wuhan, China, in 2010, respectively. He is currently a Professor with the School of Computer and Information Engineering, Nanning Normal University, Nanning, China. His current research interests include Swarm Intelligence, and multi-objective optimization.