Knee-driven many-objective sine-cosine algorithm

Hongxia Zhao[1*], Yongjie Wang[1], and Maolin Li [1]
[1] School of Information Technology Application Innovation, Yuncheng Vocational and Technical University
Yuncheng, Shanxi, 044000 China
[e-mail: zhaohongxia2022yj@163.com]
[*]Corresponding author: Hongxia Zhao

Received August 22, 2022; revised December 13, 2022; accepted February 6, 2023; published February 28, 2023

Abstract

When solving multi-objective optimization problems, the blindness of the evolution direction of the population gradually emerges with the increase in the number of objectives, and there are also problems of convergence and diversity that are difficult to balance. The many-objective optimization problem makes some classic multi-objective optimization algorithms face challenges due to the huge objective space. The sine cosine algorithm is a new type of natural simulation optimization algorithm, which uses the sine and cosine mathematical model to solve the optimization problem. In this paper, a knee-driven many-objective sine-cosine algorithm (MaSCA-KD) is proposed. First, the Latin hypercube population initialization strategy is used to generate the initial population, in order to ensure that the population is evenly distributed in the decision space. Secondly, special points in the population, such as nadir point and knee points, are adopted to increase selection pressure and guide population evolution. In the process of environmental selection, the diversity of the population is promoted through diversity criteria. Through the above strategies, the balance of population convergence and diversity is achieved. Experimental research on the WFG series of benchmark problems shows that the MaSCA-KD algorithm has a certain degree of competitiveness compared with the existing algorithms. The algorithm has good performance and can be used as an alternative tool for many-objective optimization problems.

Keywords: Evolutionary computations, Many-objective optimization, Knee points, Sine cosine algorithm, Latin square sampling

This work was supported by the Education science Foundation of Shanxi province, China(Grant No. 21290)

http://doi.org/10.3837/tiis.2023.02.003 **ISSN : 1976-7277**

1. Introduction

With the growth of society's demand for production, multi-objective optimization problems in real life become more and more complex, and more and more objective problems need to be considered [1]. Therefore, the current existing evolutionary algorithms face many challenges in solving multi-objective optimization problems [2]. For instance, the blindness of the evolution direction of the population, the inefficiency of the dominance relationship, the imbalance of convergence and diversity, the difficulty of visualizing the set of real solutions, etc [3, 4].

As the number of objective increases, the proportion of non-dominated solutions in the population among feasible solutions will increase explosively, and the selection pressure to promote the evolution of the population decreases, resulting in invalid dominance relations [5, 6]. In response to this problem, some scholars try to increase the selection pressure by changing the dominance relationship, which means using new dominance relationships to increase the selection pressure of the evolutionary population. For example, r dominates, ε dominates [7] and other dominance relations. Although these dominance relationships can select individuals with strong convergence under certain circumstances to maintain the diversity and convergence of the population, such methods involve parameter values, which have great uncertainty [8].

The other method is indicator-based approach: In order to obtain the optimal solution ranking in the objective space, many scholars use multi-objective algorithms based on evaluation indicators to deal with Many-objective Optimization Problems (MaOPs), such as IBEA [9], SMSEMOA [10], HypE [11], DNMOEA/HI [12] and so on. However, it is worth noting that although this type of algorithm has good results when dealing with MaOPs, as the number of targets increases sharply, the computational cost of some of the performance indicators used by these algorithms will also rise sharply.

Another approach is the reference set-based approach: this algorithm mainly uses a set of reference information to assist in measuring the quality of the solution and guide the search process in this way. Praditwong and Wang respectively proposed a new two- archive algorithm [13]. The Convergence Archive can be considered a true reference collection for online updates. In addition, VaEA [14] takes population individuals as the reference set to dynamically guide the evolution process. NSGA-III [15] is a representative algorithm. In this algorithm, a set of predefined reference points are used to strengthen and consolidate the convergence and distribution of candidate solutions. Since then, scholars have also proposed the use of reference points or reference vector-guided evolutionary algorithms for multi-objective optimization methods [16,17] and so on. However, for this type of problem, how to reasonably evaluate and screen candidate solution individuals by combining reference information is one of the key issues in the design and implementation of this type of algorithm. However, they all need to rely on pre-defined reference vectors [18–20].

Sine cosine algorithm (SCA) is a new type of natural simulation optimization algorithm proposed in 2016 [21]. The algorithm generates multiple random candidate solutions and optimizes them using the sine-cosine mathematical model [22]. The problem has the characteristics of simple structure, few parameters, and easy implementation, but there are also problems such as low optimization accuracy, easy to fall into local extremes, and slow convergence speed [23]. The quality of the initial population will affect the convergence speed and accuracy. The initial population of the SCA algorithm is randomly generated, and the diversity of the population and the reasonableness of the distribution in the search space cannot be guaranteed. In other words, random sampling does not do a good job of spreading the sample across the interval when the sample size is small. Unlike random sampling, Latin hypercube sampling [24] has the property of uniform stratification, and has the ability to obtain tail sample values in the case of less sampling. Therefore, the MaSCA-KD generates a more uniform initial point by introducing the Latin hypercube sampling method.

When dealing with MOP, the judgment of the quality of non-dominated solutions is often ambiguous during the evolution process. During evolution, the utilization of knee points in non-dominated solutions can improve the convergence performance of optimization algorithms, because the bias towards knee points is shown to be an approximation of the bias towards large hypervolumes [25]. Knee points are naturally the most popular of the non-dominated solutions if there is no clear decision maker preference [26]. Therefore, in recent years, many scholars have begun to pay attention to the use of knee points. When dealing with dynamic multi-objective optimization problems, Zou et al. [27] introduced knee point sets in the forecasting population, so as to accurately predict the position and distribution of Pareto fronts after environmental changes. Subsequently, they used weights to divide the whole population into several subpopulations in another work, and the knee point in each subpopulation acts as a leader to guide other solutions to search [28]. In literature [29], non-dominated solutions are selected near knee joints and boundary regions, so that the burden of maintaining large and diverse populations throughout evolution is reduced. In addition to knee points, other information in the population can also guide the evolution of the population, such as boundary points. Li et al. introduced a special set of points (such as boundary points and knee points) in the prediction population, which can more accurately track the Pareto front or Pareto set [30]. Through these special solutions, the evolutionary information of the population is reflected. Making full use of this information to guide the evolution process can well balance the convergence and diversity of the population.

The main contributions of this paper are highlighted as follows.

● First of all, in the initial population stage, the Latin hypercube population initialization strategy is used to initialize the population, which can avoid missing part of the valuable search space, so as to improve the diversity of the initial population and the rationality of the distribution in the search space.

● Secondly, the environmental information in the population evolution process is worthy of being used. The ideal point in the population can be used to reduce the search space and increase the selection pressure; the inflection point of the population can also be used to update the global extremum to improve convergence.

● When making environmental choices, it is necessary to maintain the diversity of the population. The knee points and extreme points of the population are given priority, and the solution with the largest difference from the selected solution set is selected as the candidate solution from the remaining population.

Following, Section 2 presents the main background knowledge. Section 3 describes MaSCA-KD. Next, the experimental design and results is recorded. Finally, the fifth section is the conclusion and future research directions.

2. Background Knowledge

2.1 Many-objective Optimization Problems

MaOPs refer to MOPs whose objective dimensions exceed 4 [3, 19, 31]. Without loss of generality, taking minimization as an example, the general expression of high-dimensional multi-objectives is as follows:

$$\text{Min} \quad F(X) = \{ f_1(X), f_2(X), f_3(X) \ldots f_m(X) \} \tag{1}$$

where, $f_i(i=1,2,3\ldots m)$ are conflicting objective functions and $m \geq 4$. A solution $X = (x_1, x_2, x_3 \ldots x_D)$, and D is the dimension of decision variables.

2.2 The standard Sine Cosine algorithm

The basic idea of the sine-cosine algorithm is to use the oscillating characteristics of the sine-cosine function to gradually converge to the optimal solution, explore the outward fluctuations globally, and develop locally toward the optimal solution fluctuations. SCA divides the optimization process into two phases: the exploration and the development. The global optimal solution is continuously approached through the search and development stages. The position update equation for the two stages of SCA is as follows:

$$X_{i,k}^{m+1} = X_{i,k}^{g} + r_1 \times \sin(r_2) \times r_3 \times P_{best}^{g} - X_{i,k}^{g} \quad if \ \text{rand} < 0.5, \tag{2}$$

$$X_{i,k}^{g+1} = X_{i,k}^{g} + r_1 \times \cos(r_2) \times r_3 \times P_{best}^{g} - X_{i,k}^{g} \quad otherwise. \tag{3}$$

where $X_{i,k}^{g}$ represents the value of the X_i in the k-th decision space in the g-th generation; the update of r_1 is shown in (4).

$$r_1 = \text{a-a} \frac{g}{G_{\max}} \tag{4}$$

In Eq.(4), a equal to 2; $G_{\max}$ is the maximum number of generation; $r_2 \in [0,\ 2\pi]$, $r_3 \in [-2,\ 2]$ and rand $\in [0,\ 1]$ are random numbers; P_{best}^{t} is the global optimal position at the t-th iteration.

If the SCA algorithm is used for optimization, the following steps need to be performed in sequence. First of all, initialize the population position and related parameters, calculate the individual fitness value, and save it according to the best. Secondly, in the main loop, update the parameter r1 according to (4), and under the control of the switching probability r4, it is determined that the individual chooses the way to update the position: sine or cosine, and then calculates the individual fitness value and updates the global optimum. When the maximum number of iterations is encountered (that is, the termination condition is satisfied), the loop is exited and the optimal solution is output. The location update method performed by individuals is dimension-by-dimension update, and SCA has strong global exploration and local development capabilities. SCA uses the sine-cosine conversion mechanism to make the algorithm have better optimization capabilities when solving optimization functions.

3. MaSCA-KD

The details of the proposed knee-driven many-objective sine-cosine algorithm (MaSCA-KD) are documented in this section.

3.1 Framework of MaSCA-KD

The first part of MaSCA-KD is the initialization phase (lines 1-6 in Algorithm 1). Unlike most many-objective optimization algorithms, N individuals are created based on Latin hypercube sampling rather than random. The knee points in the population is employed to update the global extremum of SCA. At the end of the initialization phase, initialize the convergence point based on the initial population.

The second part of MaSCA-KD is the is the loop iteration stage (lines 7-15 in Algorithm 1). The iteration cycle does not terminate if the maximum number of iterations (T_{max}) is not reached. First, offspring populations are generated by SCA (line 9 in Algorithm 1). After evaluating the progeny population, the knee points of the mixed population (i.e., *UintPOP*) are used to update P_{best} (line 12 in Algorithm 1). Finally, the environment selection operation is performed for the mixed population, and N candidate solutions are selected for output (line 13 in Algorithm 1).

Algorithm 1 Framework of the proposed MaSCA-KD
Input: N, T_{max}
Output: *POP*
1: First part: Initialization
2: t=0
3: Initialize N population (*POP*) based on Latin hypercube sampling
4: Evaluate the fitness value of *POP* according to fitness function
5: Update P_{best} by knee points
6: Initialize convergence point *ConP*
7: The second part: Loop and Iteration
8: while $t < T_{max}$ do
9: Q = Reproduction(*POP*) // according to Eq.(1) and Eq.(2)
10: Evaluate the fitness value of Q according to fitness function
11: *UnitPOP* = *POP* ∪ Q
12: Update P_{best} of SCA by knee points in *UintPOP*
13: (*POP*, *ConP*)=Environmentalsel(*UnitPOP*, *ConP*)
14: t= t +1
15: end while
16: **Output** *POP*

3.2 Initialization based on Latin hypercube sampling

The initial population of the Latin hypercube population initialization strategy will affect the convergence speed and accuracy. Latin hypercube sampling is a method proposed by McKay to approximate random sampling from a multivariate parametric distribution [24]. The initial population of the SCA algorithm is randomly generated, and the diversity of the population and the reasonableness of the distribution in the search space cannot be guaranteed, so the MaSCA-KD generates a more uniformly distributed initial point by introducing the Latin hypercube sampling method. Random sampling obeys a uniform distribution in the interval [0,1]. In the case of a small number of samples, random distribution cannot spread the sample to the entire interval well. The convergence speed and accuracy of the algorithm are affected by the quality of the initial population to a certain extent, and the randomly generated initial population cannot guarantee the rationality of the search space distribution and the diversity of the population.

Unlike random sampling, the characteristics of uniform stratification and equal probability sampling of Latin hypercube sampling can ensure that the variables generated by it cover the entire distribution space [32].

The initial population is to generate a population with a population size of N in the D-dimensional space. Therefore, combined with the Latin hypercube sampling method, the specific steps to obtain the population initialization strategy are as follows:

1. Enter the dimension D of the population and the size N of the population;
2. Define the interval of variable X as [lb, ub], where lb and ub are the lower and upper bounds of the variable, respectively;
3. Divide the [lb,ub] interval into N equal subintervals;
4. In each dimension, a point in each subinterval is selected randomly;
5. Combine the points extracted from each dimension in step 4 to form a set as the initial population.

Through the initialization of the population through the above steps, the individuals of the initial population will cover the entire search space as much as possible, the diversity of the initial population will be improved, and the optimization performance of the algorithm will become better.

3.3 Convergence Point Driven Strategy

Although MOEAs are considered to be the most effective method for dealing with MOPs, most of the researches are limited to the MOPs with 2 or 3 objectives. The increase in the number of targets not only makes the problem itself more complicated and difficult to solve. Therefore, the difficulty in solving the super multi-objective optimization problem is the problem of the convergence of the huge search space.

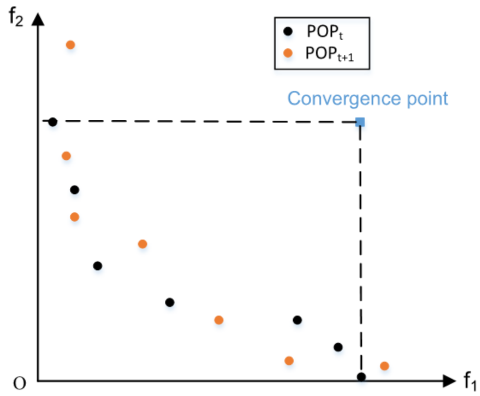


Fig. 1. Schematic diagram of convergence point

In this paper, the convergence point is introduced to increase the convergence pressure of the population. The convergence point can be regarded as the nadir point, that is, $Conp = [f_1^{\max}, f_2^{\max}, \ldots, f_m^{\max}]$. The convergence points of the previous generation are used in the next-generation environmental selection mechanism to ensure the full use of historical information.

As shown in **Fig. 1**, the population for two consecutive generations is plotted. The convergence area formed by the convergence points selected in the *t*-th generation guarantees the improvement of convergence in the next iteration. In other words, in the (*t*+1)-th generation, all individuals whose population overflows the convergence area are eliminated.

3.4 Knee point drive strategy

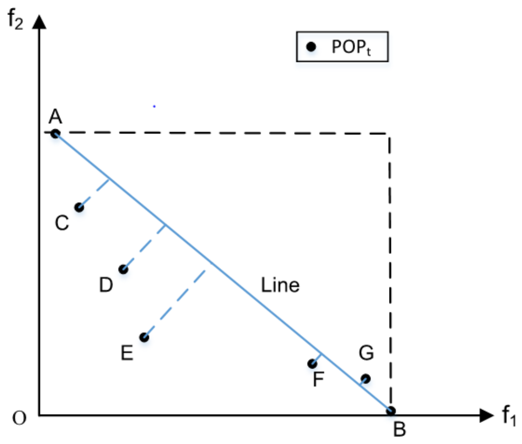


Fig. 2. Schematic diagram of knee point determination

In the multi-objective optimization algorithm, knee points are the solutions in a subset of the Pareto optimal solution set. This type of solution is intuitively represented as the most ”concave” part of the Pareto front surface. Near these knee points, a slight decrease in the target value in any one dimension results in a large increase in the target value in the other dimensions. Therefore, while the pre-Pareto solutions do not dominate each other, the knee joints outperform the other solutions to some extent.

As shown in **Fig. 2**, the black dots in the figure represent individuals in the current population. If you want to find the knee point, you need to follow the steps below.

1. Find the two end points of the population, namely A and B in **Fig. 2**.

2. Link the end points to form a straight line and judge the knee point, that is the solid line Line in **Fig. 2**.

3. Calculate the vertical distance from all individuals to the Line, and the point with the largest vertical distance is the knee point of the population, that is, point E in the figure.

It can be observed from **Fig. 2** that the convergence of point E is obviously better than that of other individuals. Therefore, using knee points to define the global extremum can improve the convergence of the population.

3.5 Environmental selection mechanism

When designing the solution selection mechanism, two aspects need to be considered: convergence and diversity. In this paper, the convergence point and the inflection point are used to improve the convergence of the current population, and then the candidate solutions are screened by the diversity difference method to achieve a balance between convergence and diversity.

Algorithm 2 (*POP*, *ConP*) = Environmentalsel(*UnitPOP*,*ConP*)
Input: *UnitPOP*, *ConP*
Output: *POP*
1: $UnitPOP = \Pr essure(UnitPOP, ConP)$ // according to Section 3.3
2: $K = \text{Findknee}(UnitPOP)$ // according to Section 3.4
3: $POPsel = K \cup \{EX_1, EX_2, \ldots, EX_M\}$
4: $UnitPOP = UnitPOP - POPsel$
5: while $POPsel < N$ do // supplement to the set of candidate solutions
6: L=$Maxangle(UnitPOP, POPsel)$
7: $POPsel$=$POPsel \cup L$
8: Remove *L* from *UnitPOP*
9: end while
10: *POP*=*POPsel*
11: **Return** *POP*

Algorithm 2 gives the pseudocode of the environment selection process. For the mixed population *UnitPOP*, pruning is carried out according to the convergence point driving strategy. According to the description in Section 3.3, the individuals outside the convergence range in the population are removed to ensure the convergence of the population (line 1 in Algorithm 2). Then, select the knee points in the population for preservation. The purpose of this is to preserve the population's astringent individuals and further enhance the population's convergence. Next, record and save the extreme solution ($\{EX_1, EX_2, \ldots, EX_M\}$ of the population (line 2 in Algorithm 2).

Boundary points and knee points are extremely important special points, and the evolution of the population is guided by them. Therefore, they are preferentially selected as candidate solutions (line 3 in Algorithm 2). Then, in order to ensure the diversity, a solution with the greatest diversity in the remaining population is selected into the candidate solution set. If the number of candidate solutions in the current candidate solution set is less than N, find the solution L with the largest difference from the current candidate solution set (that is, the solution L with the largest angle with all individuals in the current candidate solution set) as one of the next generation candidate solution set individual, and then add it to the candidate solution set (lines 5-9 in Algorithm 2). Repeat the above steps until N individuals are selected to form a new round of candidate solution sets, and finally output candidate solution sets.

In Algorithm 2, driven by the convergence point, retaining the knee points further enhances the convergence of the population, and the most differentiated solution selection method is used to ensure the diversity of the solution set. Therefore, the convergence and diversity of the final solution set reach a balance.

4. Experimental Result

To verify the performance of MaSCA-KD, the WFG series test set [33] was selected for experimental research. **Table 1** shows the test problems and their characteristics. The most representative algorithms in recent years are selected as comparison algorithms, including NSGA-III [15], MOEA/DD [34], MaPSO [3], KnEA [25] and RVEA [16]. **Table 2** records the parameters of each comparison algorithm.

4.1 Experimental settings

Suppose a D-dimensional optimization problem with M objectives, the value of M is 3, 5 or 7 for each test function chosen to be used, and $D = M + 9$. The value of the position-related variable k is $M - 1$, and the value of the distance-related variable l is $D - k$.

The WFG series of test functions is one of the most widely used test function sets for multi-objective and many-objective optimization problems. The objective of these test functions and the number of decision variables can be given by the user. Furthermore, different test functions have different characteristics, such as convex, concave, mixed of convex and concave, biased, separable and so on. Hence, we employed WFG series of test problems to test the performance of the proposed algorithm. The characteristics and general parameters of these test functions are presented in **Table 1** and **Table 2**. For the mathematical representation of these test functions, see the original reference.

Table 1. Characteristics and setting of the WFG series of test functions

Function	Characteristic	M	D
WFG1	Mixed, Biased, Separable, Biased	3, 5, 7	12, 14, 16
WFG2	Convex, Unimodal, Disconnected		
WFG3	Degenerate, linear		
WFG4	Separable, Multi-modal, Concave		
WFG5	Deceptive, Separable, Concave		
WFG6	Unimodal, Concave		
WFG7	Biased, Separable, Concave		
WFG8	Biased, Non-separable, Concave		
WFG9	Multi-modal, Biased, Deceptive, Concave		

Table 2. Parameter settings

Name	Parameter
MaSCA-KD	$T_{\max}$=300, p_c=1.0, p_m=1/D, $\eta_c = \eta_m = 20$
NSGA-III	$T_{\max}$=300, p_c=1.0, p_m=1/D, $\eta_c = \eta_m = 20$
MOEA/DD	$T_{\max}$=300, θ=5, T=0.1N, δ=0.9
MaPSO	$T_{\max}$=300, K=3, $\theta_{\max}$=0.5
KnEA	$T_{\max}$=300, K=0.5,
RVEA	$T_{\max}$=300, α=2, $f_r = 0.1$

The general parameters of each algorithm are set as follows. M=3, N=200; M=5, N=300; M=7, N=300. All algorithms are independently run 30 times on each test function.

Inverted generational distance (IGD) is employed in this work to evaluate the performance of all compared algorithms [35]. It is used as a metric to evaluate the performance of the algorithm as it provides a joint measure of the overall performance (convergence and diversity are included) of the population obtained. In the process of calculating the index, the number of reference points is 10,000.

4.2 Analysis of results

Table 3-Table 5 records and analyzes the average results of all comparison algorithms in the WFG test function under the IGD index.

Table 3. IGD results on the 3-objective WFG Functions

Function	MaSCA-KD	NSGA-III	MOEA/DD	MAPSO	KnEA	RVEA
WFG1	**0.0859**	0.2883	0.1206	0.8296	0.1469	0.1268
WFG2	**0.2815**	0.3489	0.2933	0.6121	0.3178	0.3145
WFG3	0.0738	0.0561	0.0891	0.3915	0.2182	**0.0519**
WFG4	0.1636	**0.1439**	0.1512	0.8428	0.1511	0.1492
WFG5	**0.1676**	0.1695	0.1739	0.3768	0.1742	0.1708
WFG6	0.1872	0.1818	0.1961	0.2970	0.1817	**0.1641**
WFG7	**0.1468**	0.1773	0.1485	0.3749	0.1799	0.1477
WFG8	**0.2618**	0.3029	0.4255	0.6595	0.4736	0.3057
WFG9	**0.2080**	0.2758	0.2861	0.3339	0.2924	0.2824
Optimal rate	66.67%	11.11%	0.00%	0.00%	0.00%	22.22%

For the 3-objective WFG functions, MaSCA-KD reaches the optimal value on the 6 test functions, i.e., WFG1, WFG2, WFG5, WFG7-WFG9. The optimal rate of MaSCA-KD reaches 66.67%. NSGA-III obtains the best performance on WFG4, and RVEA is the best algorithm on WFG3 and WFG6. The optimal rates obtained by the MaSCA-KD, NSGA-III, MOEA/DD, MAPSO, KnEA and RVEA algorithms are 66.67%, 11.11%, 0, 0, 0, and 22.22%. Overall, MaSCA-KD is the best performing algorithm in terms of its probability of achieving optimal performance.

Table 4. IGD results on the 5-objective WFG Functions

Function	MaSCA-KD	NSGA-III	MOEA/DD	MAPSO	KnEA	RVEA
WFG1	**0.4969**	0.5367	4.3426	0.6184	0.5345	1.7840
WFG2	**0.8124**	0.8992	5.8929	4.4284	0.8622	1.5897
WFG3	0.8099	0.4704	5.4327	0.8518	0.5398	**0.4539**
WFG4	1.1964	1.3237	6.4498	1.3266	**1.1477**	1.1676
WFG5	1.1965	1.3301	8.2129	1.2894	1.2379	**1.1785**
WFG6	**1.1354**	1.3381	3.5797	1.2972	1.2362	1.1845
WFG7	**1.1434**	1.3158	6.1295	1.3121	1.1931	1.1775
WFG8	**1.2383**	1.3847	3.4435	1.3226	1.3466	1.2738
WFG9	1.1796	1.2376	5.5684	1.2769	1.2940	**1.1509**
Optimal rate	55.56%	0.00%	0.00%	0.00%	11.11%	33.33%

As shown in **Table 4**, for the 5-objective test function, MaSCA-KD obtains the best performance on WFG1-WFG2, WFG6-WFG8. The performance of NSGA-III, MOEA/DD and MAPSO are poor than that of other competitors. The optimal rate of MaSCA-KD is 56.67%, and RVEA is a competitive algorithm whose optimal rate of MaSCA-KD is 33.33%. KnEA only achieved the optimal value of WFG4. The optimal rates obtained by the MaSCA-KD, NSGA-III, MOEA/DD, MAPSO, KnEA and RVEA algorithms are 55.56%, 0, 0, 0, 11.11%, and 33.33%. In the 5-objective test function, RVEA becomes more competitive, but A is still the best performing algorithm.

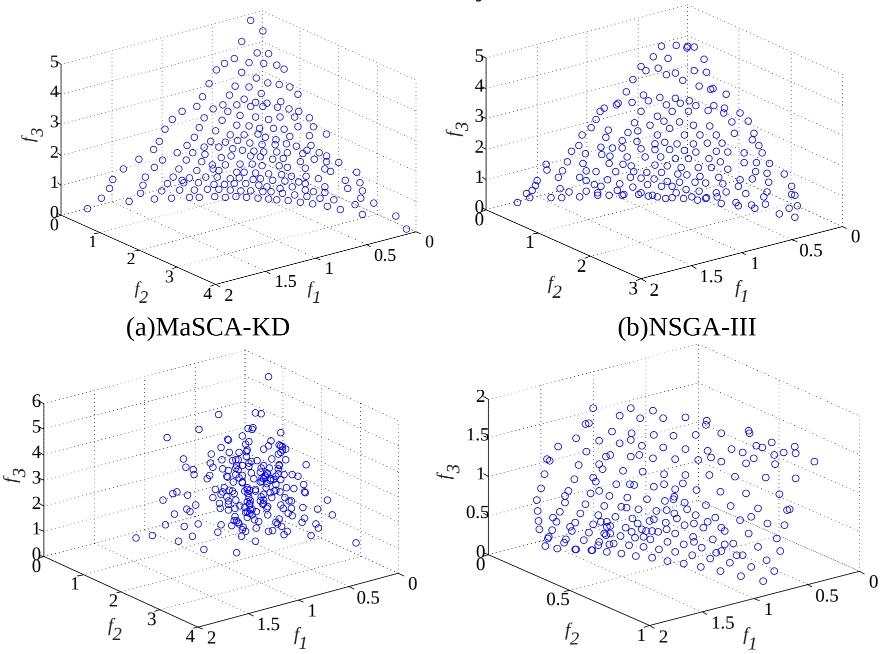
Table 5. IGD results on the 7-objective WFG Functions

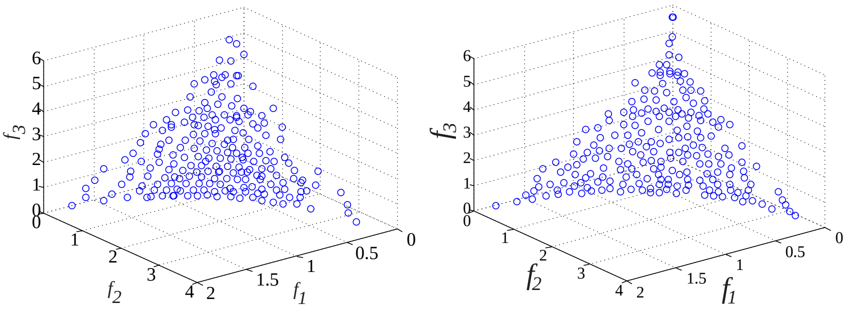
Function	MaSCA-KD	NSGA-III	MOEA/DD	MAPSO	KnEA	RVEA
WFG1	**0.8671**	0.8734	9.5437	1.2959	1.0373	0.8981
WFG2	1.8900	**1.0414**	9.3436	7.5562	1.8847	4.1476
WFG3	**0.9262**	1.4316	1.2107	2.0279	1.0241	0.9646
WFG4	3.1677	3.4788	9.0226	4.1858	3.1451	**3.0288**
WFG5	**3.0259**	3.3470	12.9001	3.8838	3.3719	3.2672
WFG6	3.1823	3.4298	14.9064	4.1827	3.5592	**3.1280**
WFG7	**3.0285**	3.4111	11.5963	3.8063	3.1546	3.0430
WFG8	**3.2317**	3.5560	9.3742	4.2838	3.5706	3.2491
WFG9	3.2326	3.3110	9.4924	3.9084	3.4044	**3.0248**
Optimal rate	55.56%	11.11%	0.00%	0.00%	0.00%	33.33%

As shown in **Table 5**, MaSCA-KD obtains the best performance on WFG1, WFG3, WFG5, WFG7 and WFG8 on the WFG series test functions for 7 objectives. The optimal performance of WFG2 is reflected in NSGA-III. The optimal rate of MaSCA-KD, NSGA-III, MOEA/DD, MAPSO, KnEA and RVEA is 55.56%, 11.11%, 0, 0, 0, 33.33%, respectively. The performance of Algorithm MaSCA-KD is still in the leading position among all the comparison algorithms.

Therefore, in most of the test functions, MaSCA-KD can obtain the best performance, indicating that MaSCA-KD can obtain an approximate population with good convergence and diversity.

Pareto front on 3-objective WFG1

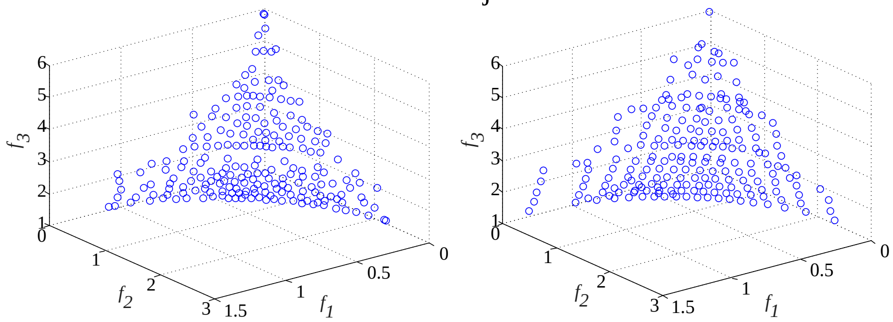




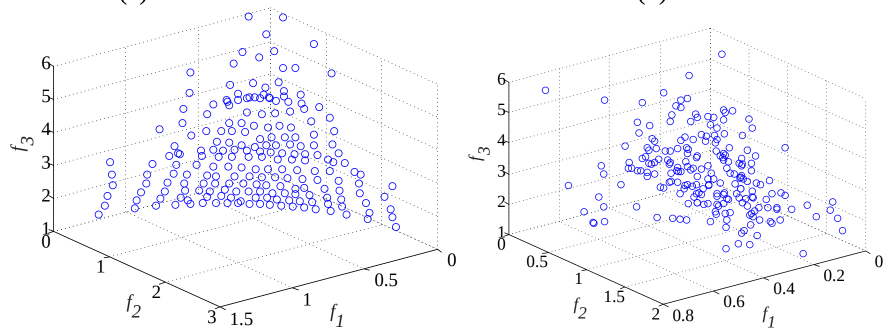
(e) KnEA (f) RVEA

Fig. 3. The approximate front of MaSCA-KD, NSGA-III, MOEA/DD, MAPSO, KnEA, RVEA on 3-objective WFG1

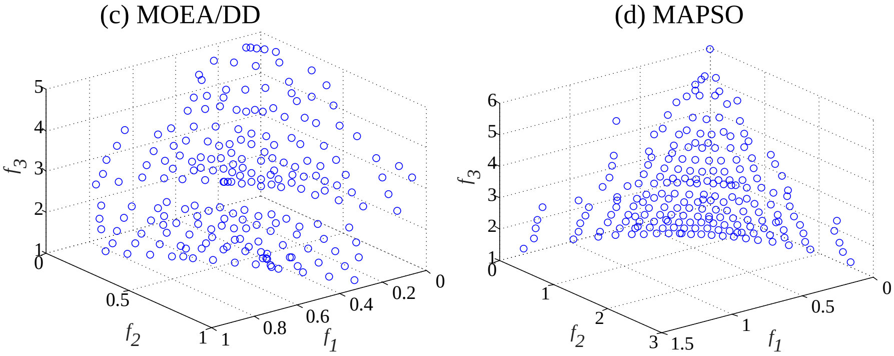
Pareto front on 3-objective WFG2



(a)MaSCA-KD (b)NSGA-III

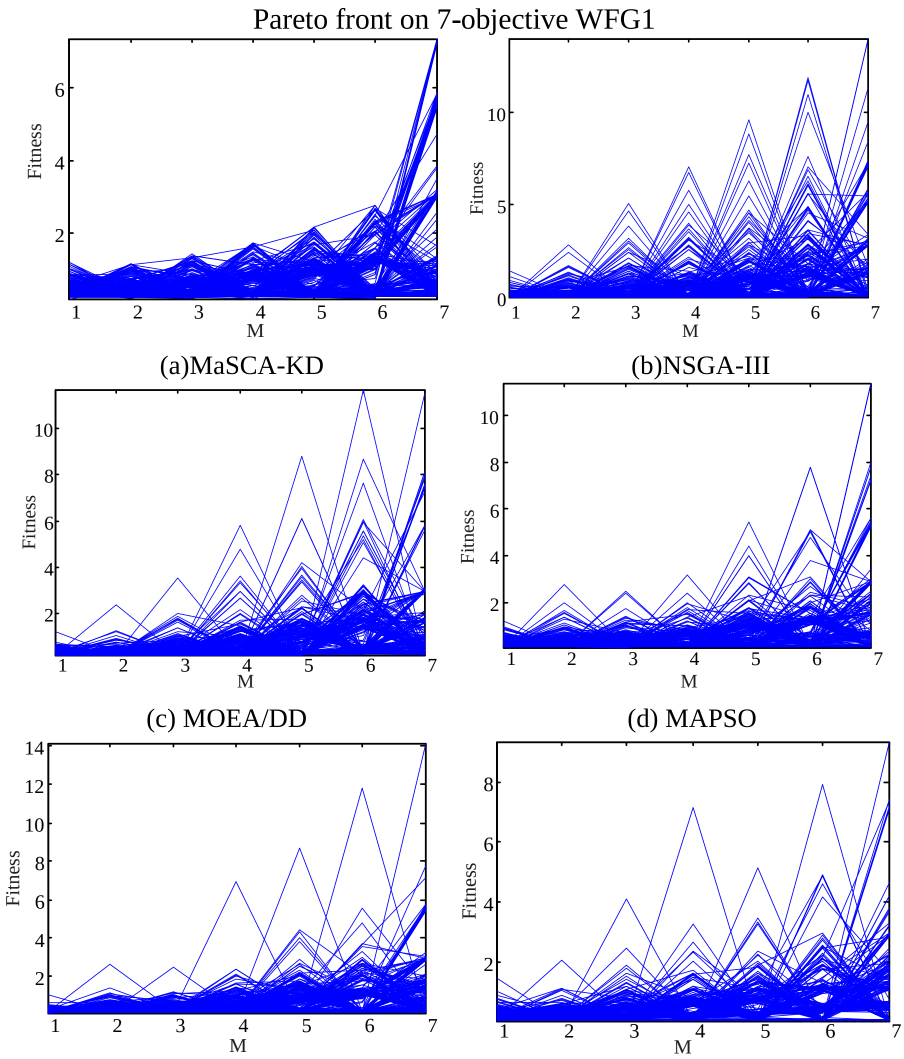


(c) MOEA/DD (d) MAPSO



(e) KnEA (f) RVEA

Fig. 4. The approximate front of MaSCA-KD, NSGA-III, MOEA/DD, MAPSO, KnEA, RVEA on 3-objective WFG2



(a)MaSCA-KD (b)NSGA-III
(c) MOEA/DD (d) MAPSO
(e) KnEA (f) RVEA

Fig. 5. The approximate front of MaSCA-KD, NSGA-III, MOEA/DD, MAPSO, KnEA, RVEA on 7-objective WFG1

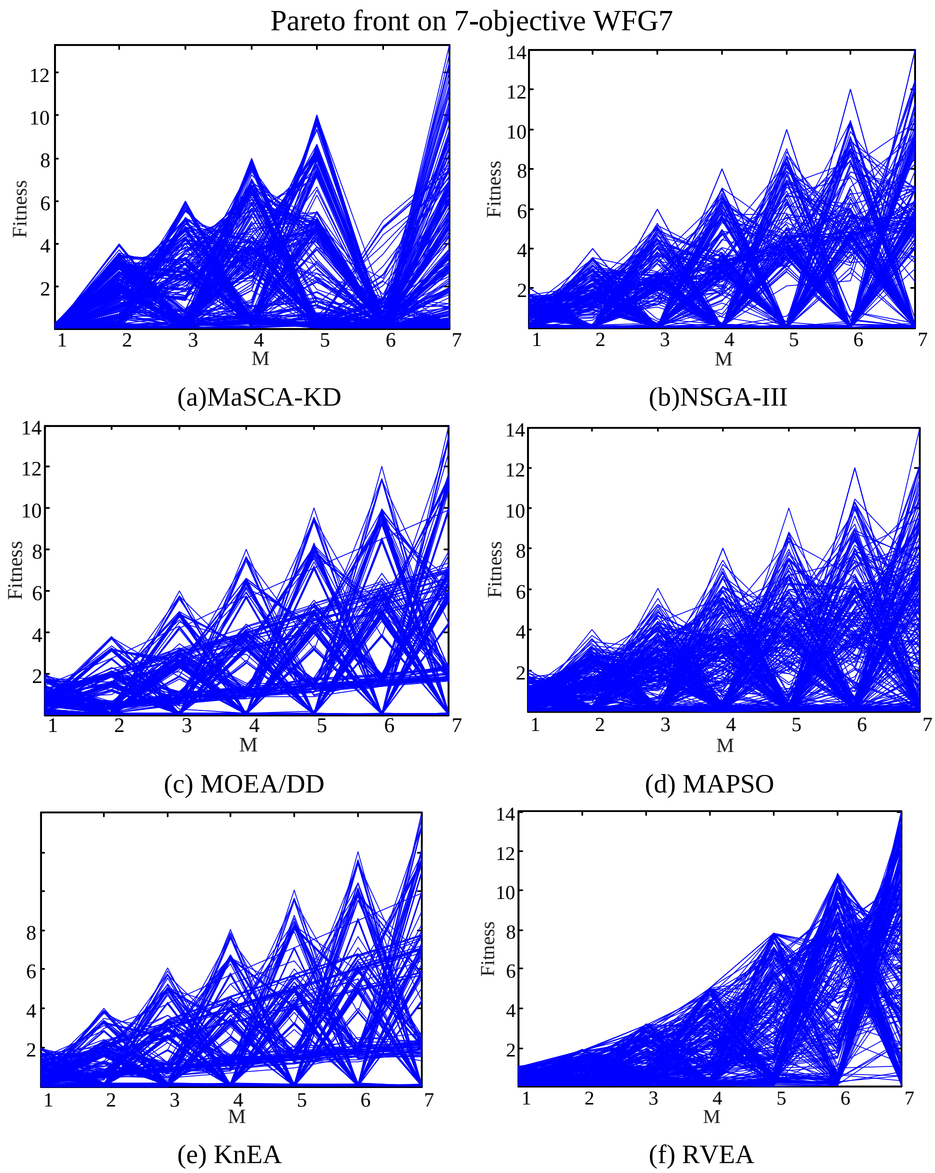


Fig. 6. The approximate front of MaSCA-KD, NSGA-III, MOEA/DD, MAPSO, KnEA, RVEA on 7-objective WFG7

Fig. 3- Fig. 6 show the plan views of candidate solution sets obtained by each comparison algorithm on some test functions, so as to observe the distribution of approximate PF in the objective space more intuitively. For WFG1 with a mixed PF, it can be seen from **Fig. 5**, both the convergence and diversity of MaSCA-KD get the best performance. NSGAIII did not converge to the complete PF; MOEA/DD and MAPSO did not get a solution set similar in shape to the PF, that is, did not find the true PF; the diversity of KnEA and RVEA is not as good as MaSCA-KD. When dealing with the discontinuity problem WFG2, it can be obtained by observing **Fig. 4** that MAPSO and KnEA did not converge to the true PF shape, and in all comparison algorithms, MaSCA-KD can not only get an approximate frontier close to the frontier, but also form an even distribution in the objective space.

As shown in **Fig. 5** and **Fig. 6**, the approximate PF obtained by MaSCA-KD is also better than other comparison algorithms for 7-objective WFG function.

5. Conclusion

When dealing with multi-objective optimization problems, the object space expands dramatically as the object dimension increases. Therefore, it is crucial to maintain the convergence and diversity balance of the Pareto solution set. In this paper, a knee-driven many-objective sine-cosine algorithm (MaSCA-KD) has been proposed. First of all, a Latin hypercube sampling strategy is adopted to generate a uniform initial population and fully search the objective space in the stage of initializing the population to increase the diversity of the population. Secondly, in order to increase the convergence pressure, special points are used for environmental selection design. Among them, the convergence point is used to increase the selection pressure of the environment, and the knee point has two functions, namely screening the global extreme value and giving priority to the candidate solution. After the convergence of the population is guaranteed, the diversity of the candidate solutions with the largest difference is improved during environment selection. The above strategies work at the same time, so that the convergence and diversity of the population reach an equilibrium state. Finally, compare the proposed algorithm with the excellent algorithms from this year. Through the simulation experiments on the 3-objective, 5- objective and 7-objective WFG series of test functions, the results show that MaSCA-KD can achieve the best performance on most functions in a total of 27 functions. In particular, the MaSCA-KD algorithm has a significant improvement in the balance of convergence and diversity compared with other similar algorithms.

In further research work, testing problems with more objective will be considered. For complex PF, more suitable environmental selection mechanisms need to be designed, especially for irregular frontiers. In addition, we will use the designed algorithm in practical problems to increase the practicality of the algorithm.

References

[1] Z. Hu, Y. Lan, Z. Zhang and X. Cai, "A Many-objective Particle Swarm Optimization Algorithm Based on Multiple Criteria for Hybrid Recommendation System," *KSII Transactions on Internet and Information Systems*, vol. 15, no. 2, pp. 442-460, 2021. Article (CrossRef Link)

[2] D. Gong, Y. Liu and G. G. Yen, "A Meta-Objective Approach for Many-Objective Evolutionary Optimization," *Evolutionary Computation*, vol. 28, no. 1, pp. 1-25, March 2020. Article (CrossRef Link)

[3] Y. Xiang, Y. Zhou, Z. Chen and J. Zhang, "A Many-Objective Particle Swarm Optimizer With Leaders Selected From Historical Solutions by Using Scalar Projections," *IEEE Transactions on Cybernetics*, vol. 50, no. 5, pp. 2209-2222, May 2020. Article (CrossRef Link)

[4] L. Zhen, M. Li, D. Peng, X. Yao, "Objective reduction for visualising many-objective solution sets," *Information Sciences*, vol. 512, pp. 278-294, 2020. Article (CrossRef Link)

[5] H. Wang, C. Sun, G. Zhang, J. E. Fieldsend, Y. Jin, "Non-dominated sorting on performance indicators for evolutionary many-objective optimization," *Information Sciences*, vol. 551, pp.23-38, 2021. Article (CrossRef Link)

[6] Y. Lan, L. Xie, X. Cai and L. Wang, "A many-objective evolutionary algorithm based on integrated strategy for skin cancer detection," *KSII Transactions on Internet and Information Systems*, vol. 16, no. 1, pp. 80-96, 2022. Article (CrossRef Link)

[7] H. Aguirre, K. Tanaka, "Space partitioning with adaptive ε-ranking and substitute distance assignments: a comparative study on many-objective mnk-landscapes," in *Proc. of the 11th Annual conference on Genetic and evolutionary computation*, pp. 547–554, 2009. Article (CrossRef Link)
[8] P. Zhang, J. Li, T. Li and H. Chen, "A New Many-Objective Evolutionary Algorithm Based on Determinantal Point Processes," *IEEE Transactions on Evolutionary Computation*, vol. 25, no. 2, pp. 334-345, April 2021. Article (CrossRef Link)
[9] E. Zitzler and S. Künzli, "Indicator-based selection in multiobjective search," in *Proc. of International conference on parallel problem solving from nature*, Springer, Berlin, Heidelberg, pp. 832-842, 2004. Article (CrossRef Link)
[10] N. Beume, B. Naujoks and M. Emmerich, "SMS-EMOA: Multiobjective selection based on dominated hypervolume," *European Journal of Operational Research*, vol. 181, no. 3, pp. 1653-1669, 2007. Article (CrossRef Link)
[11] P. Zhang, J. Li, T. Li and H. Chen, "A New Many-Objective Evolutionary Algorithm Based on Determinantal Point Processes," *IEEE Transactions on Evolutionary Computation*, vol. 25, no. 2, pp. 334-345, April 2021. Article (CrossRef Link)
[12] K. Li, S. Kwong, J. Cao, M. Li, J. Zheng and R. Shen, "Achieving balance between proximity and diversity in multi-objective evolutionary algorithm," *Information Sciences*, vol. 182, no. 1, pp. 220-242, 2012. Article (CrossRef Link)
[13] K. Praditwong and X. Yao, "A New Multi-objective Evolutionary Optimisation Algorithm: The Two-Archive Algorithm," in *Proc. of 2006 International Conference on Computational Intelligence and Security*, vol. 1, pp. 286–291, 2006. Article (CrossRef Link)
[14] Y. Xiang, Y. Zhou, M. Li and Z. Chen, "A Vector Angle-Based Evolutionary Algorithm for Unconstrained Many-Objective Optimization," *IEEE Transactions on Evolutionary Computation*, vol. 21, no. 1, pp. 131-152, Feb. 2017. Article (CrossRef Link)
[15] K. Deb and H. Jain, "An Evolutionary Many-Objective Optimization Algorithm Using Reference-Point-Based Nondominated Sorting Approach, Part I: Solving Problems With Box Constraints," *IEEE Transactions on Evolutionary Computation*, vol. 18, no. 4, pp. 577-601, Aug. 2014. Article (CrossRef Link)
[16] R. Cheng, Y. Jin, M. Olhofer and B. Sendhoff, "A Reference Vector Guided Evolutionary Algorithm for Many-Objective Optimization," *IEEE Transactions on Evolutionary Computation*, vol. 20, no. 5, pp. 773-791, Oct. 2016. Article (CrossRef Link)
[17] Y. Liu, D. Gong, X. Sun, and Y. Zhang, "Many-objective evolutionary optimization based on reference points," *Applied Soft Computing*, vol. 50, pp. 344-355, 2017. Article (CrossRef Link)
[18] L He, K. Shang, and H. Ishibuchi, "Simultaneous use of two normalization methods in decomposition-based multi-objective evolutionary algorithms," *Applied Soft Computing*, vol. 92, 106316, 2020. Article (CrossRef Link)
[19] J. Luo, X. Huang, Y. Yang, X. Li, Z. Wang and J. Feng, "A many-objective particle swarm optimizer based on indicator and direction vectors for many-objective optimization," *Information Sciences*, 514, 166-202, 2020. Article (CrossRef Link)
[20] S. Qin, C. Sun, G. Zhang, X. He and Y Tan, "A modified particle swarm optimization based on decomposition with different ideal points for many-objective optimization problems," *Complex & Intelligent Systems*, vol. 6, no. 2, pp. 263-274, 2020. Article (CrossRef Link)
[21] S. Mirjalili, "SCA: a sine cosine algorithm for solving optimization problems," *Knowledge-based systems*, vol. 96, pp. 120-133, 2016. Article (CrossRef Link)
[22] A. B. Gabis, Y. Meraihi, S. Mirjalili and A. Ramdane-Cherif, "A comprehensive survey of sine cosine algorithm: variants and applications," *Artificial Intelligence Review*, vol. 54, no. 7, pp.5469-5540, 2021. Article (CrossRef Link)
[23] W. Y. Guo, Y. Wang, F. Dai and P. Xu, "Improved sine cosine algorithm combined with optimal neighborhood and quadratic interpolation strategy," *Engineering Applications of Artificial Intelligence*, vol. 94, 103779, 2020. Article (CrossRef Link)
[24] M. D. Mckay, R. J. Beckman and W. J. Conover, "A Comparison of Three Methods for Selecting Values of Input Variables in the Analysis of Output From a Computer Code, " *Technometrics*, vol. 42, no.1, pp. 55-61, 2000. Article (CrossRef Link)

[25] X. Zhang, Y. Tian and Y. Jin, "A Knee Point-Driven Evolutionary Algorithm for Many-Objective Optimization," *IEEE Transactions on Evolutionary Computation*, vol. 19, no. 6, pp. 761-776, Dec. 2015. Article (CrossRef Link)
[26] X. Yu, Y. Duan and W. Luo, "A knee-guided algorithm to solve multi-objective economic emission dispatch problem," *Energy*, vol.259, no.124876, 2022. Article (CrossRef Link)
[27] J. Zou, Q. Li, S. Yang, H. Bai and J. Zheng, "A prediction strategy based on center points and knee points for evolutionary dynamic multi-objective optimization, " *Applied soft computing*, vol. 61, pp. 806-818, 2017. Article (CrossRef Link)
[28] J. Zou, C. Ji, S. Yang, Y. Zhang, J. Zheng and K. Li, "A knee-point-based evolutionary algorithm using weighted subpopulation for many-objective optimization," *Swarm and Evolutionary Computation*, vol. 47, pp. 33-43, 2019. Article (CrossRef Link)
[29] F. Zou, G. G. Yen, L. Tang, "A knee-guided prediction approach for dynamic multi-objective optimization," *Information Sciences*, vol. 509, pp. 193-209, 2020. Article (CrossRef Link)
[30] Q. Li, J. Zou, S. Yang, J. Zheng and G. Ruan, "A predictive strategy based on special points for evolutionary dynamic multi-objective optimization," *Soft Computing*, vol. 23, no. 11, pp. 3723-3739, 2019. Article (CrossRef Link)
[31] X. Li, X. Li and K Wang, "A many-objective evolutionary algorithm based on vector angle distance scaling," *Journal of Intelligent & Fuzzy Systems*, vol. 40, no. 5, pp. 10285-10306, 2021. Article (CrossRef Link)
[32] Z. He, Y. Pan, K. Wang, L. Xiao and X, Wang, "Area optimization for MPRM logic circuits based on improved multiple disturbances fireworks algorithm," *Applied Mathematics and Computation*, vol. 399, no. 126008, 2021. Article (CrossRef Link)
[33] S. Huband, P. Hingston, L. Barone and L. While, "A review of multiobjective test problems and a scalable test problem toolkit," *IEEE Transactions on Evolutionary Computation*, vol. 10, no. 5, pp. 477-506, Oct. 2006. Article (CrossRef Link)
[34] K. Li, K. Deb, Q. Zhang and S. Kwong, "An Evolutionary Many-Objective Optimization Algorithm Based on Dominance and Decomposition," *IEEE Transactions on Evolutionary Computation*, vol. 19, no. 5, pp. 694-716, 2015. Article (CrossRef Link)
[35] P. A. N. Bosman and D. Thierens, "The balance between proximity and diversity in multiobjective evolutionary algorithms," *IEEE Transactions on Evolutionary Computation*, vol. 7, no. 2, pp. 174-188, April 2003. Article (CrossRef Link)

Hongxia Zhao received M.S. degree in Taiyuan University of Technology. She is a Lecturer in Yuncheng Vocational and Technical University, Yuncheng, Shanxi, China. Her research interests include data analysis and data mining.

Yongjie Wang is an Associate Professor in Yuncheng Vocational and Technical University, Yuncheng, Shanxi, China. His research interests include machine learning and data mining.

Maolin Li received M.S. degree in Dalian University of Technology. He is a Lecturer in Yuncheng Vocational and Technical University, Yuncheng, Shanxi, China. His research interests include intelligent optimization algorithm and computer software.