Received January 31, 2018, accepted February 28, 2018, date of publication March 16, 2018, date of current version April 4, 2018.

Digital Object Identifier 10.1109/ACCESS.2018.2812701

A Hybrid Multiobjective Particle Swarm Optimization Algorithm Based on R2 Indicator

LI-XIN WEI, XIN LI, RUI FAN, HAO SUN, AND ZI-YU HU
[1]Key Lab of Industrial Computer Control Engineering of Hebei Province, Yanshan University, Hebei 066004, China
[2]School of Electrical Engineering, Yanshan University, Hebei 066004, China

Corresponding author: Rui Fan (fanfanfr@163.com)

This work was supported by the Natural Science Foundation of Hebei Province under Grant F2016203249.

ABSTRACT When dealing with complex multiobjective problems, particle swarm optimization algorithm is easy to fall into local optimum and lead to uneven distribution. Therefore, this paper presents a hybrid multiobjective particle swarm optimization algorithm based on R2 indicator (R2HMOPSO) for solving multiobjective optimization problem. The proposed algorithm uses the sigmoid function mapping method to adjust the inertia weight and learning factors in order to tradeoffs the exploration and exploitation process effectively. In addition, simulation binary crossover operator is designed to reinitialize the particles to improve the search capability of the algorithm and to prevent particles from falling into local optimum and premature convergence. R2 indicator is incorporated into the R2HMOPSO algorithm so as to deal with the solutions of uneven distribution on the true Pareto front. Besides, polynomial mutation is used to maintain diversity in the external archive. The improved algorithm is evaluated on standard benchmarks. By comparing it with four state-of-the-art multiobjective optimization algorithms, the simulation results show that R2HMOPSO algorithm is competitive and effective in terms of convergence and distribution.

INDEX TERMS Multiobjective optimization problem, R2 indicator, particle swarm algorithm, decomposition method.

I. INTRODUCTION

Multi-objective optimization has been applied in many fields such as science, engineering, economics and logistics, when optimal decision is required to be taken in presence of trade-offs between two or more conflicting objectives. Therefore, it is an important task to address multiple optimization objectives effectively and simultaneously by identifying a set of well-distributed Pareto optimal solutions that generate good values for each objective [23], [29].

In the field of multi-objective optimization, many multi-objective optimization algorithms have been proposed in the literatures which can be broadly categorized under three frameworks [32], [33].

(1) Dominance-based framework - In this framework, a MOP is optimized by optimizing all the objectives simultaneously. The assignment of fitness to solutions is based on Pareto-dominance principle which plays a key role in the convergence of dominance-based optimization algorithm. Furthermore, an explicit diversity preservation scheme is necessary to maintain the diversity of solutions. NSGAII [5] is a classical dominance-based optimization algorithm, which uses Pareto domination and crowding distance to update and maintain external archives. However, when the number of objective increases, selection pressure will be reduced and optimization process will be hampered.

(2) Decomposition-based framework - In this framework, scalarizing functions, for example, the Tchebycheff approach are used to convert a MOP into a set of single-objective optimization subproblems and the subproblems are solved in a single run using an optimization algorithm. The decomposition-based optimization algorithms utilize aggregated fitness value of solutions in the selection. MOEA/D [20] is a classical algorithm based on decomposition, which is updated through the neighborhood of subproblem information. The decomposition method is a milestone in the multi-objective optimization algorithm. However, Zhou *et al.* [23] proposed some recent improvements to MOEA/Ds in order to achieve good results.

(3) Indicator-based framework - In this framework, a performance indicator is used to measure the fitness of a solution by assessing its contribution such as hypervolume indicator or R2 indicator, which can measure convergence

and diversity of a optimization algorithm simultaneously. Some of well-known indicator-based optimization algorithms are R2-IBEA [14], HypE [30]. But the distribution is required to be improved when the algorithms are updated by using indicators alone.

Particle swarm optimization algorithms have been extensively studied in the above three frameworks [3], [18]. Such as NPSO [10], [11], MOPSO/D [13], [28] and R2MOPSO [9]. In addition, Gong *et al.* [7] proposed the essential characteristics of the multi-objective optimization problem, and put forward several viewpoints on the future research of the multi-objective optimization algorithm. Wang *et al.* [18] introduced PSO's present situation of research and application in structure, parameter selection, topology structure, discrete PSO algorithm, PSO algorithm and multi-objective optimization PSO. Tsai *et al.* [17] proposed a multi-objective particle swarm optimizer with the improved operation of ratio assignment and jump improvement to deal with multi-objective problems. Fan *et al.* [36] proposed a multi-objective decomposition particle swarm optimization based on completion-checking to find the true Pareto fronts when tackling some complex multi-objective problems. Monson [34] proposed a simple, effective, computationally cheap, and easily tuned method, which improves PSO's performance by automatically adapting acceleration coefficients. Zhang *et al.* [22] proposed a new adaptive inertia weight adjusting approach based on Bayesian techniques in PSO, which is used to set up a sound tradeoff between the exploration and exploitation. In terms of learning factors, Mohammadi-Ivatloo *et al.* [12] proposed that the acceleration coefficients in PSO algorithm are varied adaptively during iterations to improve solution quality of original PSO and avoid premature convergence. Besides, it would be interesting to use one the of the above frameworks or combine these frameworks to achieve a better preservation of solution diversity, that is, obtain a closer approximation of the Pareto optimal front. Coello [4] proposed a decomposition-based multi-objective particle swarm optimization algorithm that used a set of solutions considered to be global optimal to update the position of each particle according to the decomposition method. Wang *et al.* [35] proposed a weighted Tchebycheff method to convert a non-convex and difficult problem into a set of single objective optimization problems. In addition, an algorithm based on successive convex approximation (SCA) is proposed to solve it effectively. Chaman Garcia *et al.* [2] proposed that hypervolume contribution is used to select global and personal leaders for each particle of archived solutions in the main swarm, and it is a mechanism for pruning the external archive. Li *et al.* [9] proposed that R2 contribution is designed to select global best leaders and update the swarm of archived solutions. Petrovski *et al.* [1] incorporated dominance with decomposition method in multi-objective optimization algorithm and introduced a new archiving technique that facilitates attaining better diversity and convergence in both objective and solution spaces. The above mentioned algorithms are efficient and competitive in dealing with MOPs. However, due to the premature convergence of PSO, these algorithms need to be improved in optimizing some multi-objective problems [17], [18].

Therefore, it is crucial to update global leaders and individual leaders when using PSO to deal with multi-objective problems. In order to obtain a good performance, three demands need to be met:

(1) The algorithm converges as much as possible.

(2) The algorithm is presented from falling into the local optimum and premature convergence.

(3) The solutions obtained are distributed as uniformly as possible on the true Pareto front.

In view of the problems that PSO prone to occur, various strategies are proposed such as improvements of speed formula, selection mechanisms, mutation operators and archive modes.

Existing studies have shown that selection mechanism is classified into three methods, which includes domination-based, decomposition-based and indicator-based. In the process of solving multi-objective problems, the selection mechanism of domination and crowding distance will lead to the decrease of the selective pressure of the particles on account of the increased number of objective. Besides, because the number of particles affects the generation of weight vectors, using the decomposition method to update the particles alone leads to a problem of uneven distribution. Moreover, when the leader particles are updated with R2 indicator alone, the obtained results also perform slightly worse. Therefore, an effective hybrid particle swarm optimization algorithm is proposed by combining these three frameworks.

Based on the existing studies, the main improvements of R2HMOPSO are listed as follows:

(1) For solving multi-objective optimization problems with particle swarm algorithm, the improvement of speed formula is a key to balance the global search and local search during the updating of population. Thus, the Sigmoid function is used to adjust the inertia weight and learning factors adaptively, which tradeoffs the exploration and exploitation process effectively.

(2) In order to avoid the algorithm falling into local optimum and premature convergence, simulation binary crossover operator is designed to re-initialize the particles so as to enhance the search ability and jump out of the local optimum. In addition, polynomial mutation is used in external archive to increase the diversity of population.

(3) For the purpose of getting the solutions converged to the true Pareto front and uniformly distributed, R2 indicator contribution value is designed to select and delete the particles of external archive rather than crowding distance. This selection mechanism improves convergence and distribution of the algorithm and it combines dominance-based with indicator-based effectively.

Finally, the obtained experimental results show that R2HMOPSO displays a better performance than other algorithms when dealing with complex problems such as convex, non-convex, discontinuous and multi-modality .

The remainder of this paper is organized as follows. Section II describes the multi-objective problems, decomposition method, R2 indicator and particle swarm optimization. Section III explains the details of a hybrid multi-objective particle swarm algorithm based on the R2 indicator. Section IV presents a comparative results with respect to other algorithms. Section V provides conclusions and some possible paths for future work.

II. BACKGROUND

A. MULTI-OBJECTIVE OPTIMIZATION PROBLEM

In general, multi-objective problems can be described as:

$$\begin{aligned} &min\ F(x) = (f_1(x), f_2(x), \ldots, f_m(x))^T \\ &s.t\ x \in \Omega^n \end{aligned} \tag{1}$$

Where $x = (x_1, x_2, , x_n)$ is a n-dimensional decision variable, Ω^n is a feasible solution space for decision variables, $F : \Omega^n \rightarrow R^m$ is the mapping from the decision space to the target space, m is the dimension of the target space.

B. DECOMPOSITION METHOD

The decomposition method is used to transform the multi-objective problem into a set of single-objective optimization problems. There are three conversion approaches: Weighted Sum approach, Tchebycheff approach and Boundary Intersection approach. In this paper, the Tchebycheff method is adopted. The Tchebycheff aggregate function with a non-negative weight vector w and reference point $z^* = min\{f_i(x)) \mid x \in \Omega\}$ is written as follows:

$$g^{te}(x \mid w, z^*) = max\left\{w_i \mid f_i(x) - z^* \mid\right\} \tag{2}$$

Where $z^* = (z_1^*, \ldots, z_m^*)$ is reference point, set $z^* = min\{f_i(x)) \mid x \in \Omega\}, i = 1, 2, \ldots, m$.

In R2HMOPSO algorithm, for each Pareto optimal solution x^*, there is a weight vector w for the corresponding problem. Then, the optimal solution of the scalar problem also corresponds to the Pareto solution of a multi-objective optimization problem. Thus, different Pareto optimal solutions can be obtained by modifying the weight vector.

C. R2 INDICATOR METHOD

R2 indicator [25] is presented to evaluate the relative quality of the two groups of individuals initially. Assuming the standard weighted Tchebycheff function with a specific reference point z^*, R2 indicator evaluates the mass of A through a set of individuals (set A) and the reference point is z^*:

$$R2(A, W, z^*) = \frac{1}{\mid W \mid}\sum_{w \in W} \min_{x \in A}\{\max_{i=1,2,\ldots,m} w_i(f_i(x) - z^*)\} \tag{3}$$

Where W is a set of weight vectors in m target spaces, each weight vector $w = (w_1, w_2, \ldots, w_m) \in W$, the weight vector is evenly distributed in the target space, and $1/(\mid W \mid)$ denotes a probability distributed on W.

The contribution value of R2 indicator is used to evaluate the quality of a solution. The R2 contribution value (CR2) of $x \in A$ is defined as:

$$CR2(x, A, W, z^*) = R2(A, W, Z^*) - R2(A \setminus \{x\}, W, z^*) \tag{4}$$

For simplity, this article uses the absolute value method in the original R2 indicator contribution value formula. An two-dimensional example of the section of particles is shown in Fig.1, which uses CR2 value selection strategy.

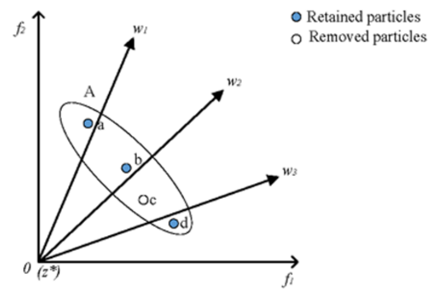

FIGURE 1. The example of CR2 value election strategy.

Let $w_1 = (0.3, 0.7)$, $w_2 = (0.5, 0.5)$, $w_3 = (0.7, 0.3)$ are the weight vectors, $z^* = (0, 0)$ is the reference point. particles $\{a, b, c, d\}$ are the non-dominated solutions in A, and set $a = (0.3, 0.8)$, $b = (0.5, 0.5)$, $c = (0.6, 0.4)$, $d = (0.7, 0.2)$. Then, calculating the CR2 values of the particles according to Eq.(4), $CR2(a) = 0.02$, $CR2(b) = 0.0167$, $CR2(c) = 0$ and $CR2(d) = 0.0233$.

The magnitude of R2 indicator contribution value determines the mass of a solution. The lower R2 indicator contribution value, the less selected probability to update the population during the search. By calculating R2 indicator contribution values, particles $\{a, b, d\}$ can be selected into the next generation.

Thus, in R2HMOPSO algorithm, R2 indicator contribution value can be used as a selection mechanism effectively. The details will be shown in Section III.

D. PARTICLE SWARM OPTIMIZATION

Particle Swarm Optimization (PSO) is a group of intelligent optimization algorithms proposed by Kennedy and Eberhart [8]. Shi and Eberhart [26] introduced the inertia weight, and it got the velocity v^t and the position x^t update formula of the universal particle swarm algorithm.

$$v_i^{t+1} = wv_i^t + c_1r_1(x_{pb,i}^t - x_i^t) + c_2r_2(x_{gb,i}^t - x_i^t) \tag{5}$$

$$x_i^{t+1} = x_i^t + v_i^{t+1} \tag{6}$$

Where w is inertia vector, c_1 and c_2 are learning factors, r_1 and r_2 are uniformly distributed between (0, 1) respectively, v_i^t and x_i^t are the velocity and position of the particle i in the t-th generation respectively. $x_{pb,i}^t$ and $x_{gb,i}^t$ are individual optimal positions and global optimal positions of particle i respectively.

III. A HYBRID MUTIL-OBJECTIVE PARTICLE SWARM ALGORITHM BASED ON R2 INDICATOR

The proposed R2HMOPSO algorithm includes domination, decomposition and R2 indicator methods. Decomposition method is designed to update individual optimal particles. While domination method and R2 indicator are designed to update the external archive and the global optimal particles. In the following paragraphs, the implementation details of each component in R2HMOPSO algorithm will be explained step-by-step.

A. DYNAMIC ADJUSTMENT OF SPEED FORMULA BASED ON SIGMOID FUNCTION

In the process of multi-objective particle swarm optimization, the inertia weight and learning factors directly affect the convergence of the population [26]. Dynamic inertia weight and learning factors can adjust the search direction during the search process, so dynamic w, c_1 and c_2 are designed to enhance global search and local search capability.

In this paper, w, c_1 and c_2 are adjusted with the number of evaluations, which use the Sigmoid function [16] $y = 1/(1 + a\exp(-bf))$. w, c_1 and c_2 are adjusted adaptively by using the linear mapping function $f = (GEN - t)/GEN$ ($t = (FEAS/N)$; $GEN = FEASUM/N$). $FEAS$ is the number of evaluation; $FEASUM$ is the total number of evaluation; N is the population size; a is a rounding parameter. Reference to the literature in the parameter setting method, the use of Sigmoid function mapping makes w, c_1 and c_2 with f changes [31].

The inertia weight w is decremented from 0.9 to 0.4 and the w change formula is described as follow:

$$w = \frac{1}{1 + 1.5\exp(-2.6f)} \tag{7}$$

The individual factor c_1 is decremented from 2.5 to 0.5 and the c_1 change formula is described as follow:

$$c_1 = \frac{5}{1 + 9\exp(-2.18f)} \tag{8}$$

The social factor c_2 set from 0.5 to 2.5 and the c_2 change formula is described as follow:

$$c_2 = \frac{5}{1 + \exp(-2.2f)} \tag{9}$$

This is an example that the inertia weight and the learning factors are adjusted with the number of evaluation. The algorithm is critical in exploration phase and needed to enhance the global search capability. As the algorithm gradually converges to a steady state, local search capability need to be strengthened. As shown in Fig.2, with the increase of evaluations, inertia weight w becomes greater so as to improve the global search capability. In the exploitation search process, learning factors are also adjusted correspondingly.

The compared experiment of R2HMOPSO algorithm and R2HMOPSO1 is described in Section IV. Fixed w, c_1 and c_2 are used in R2HMOPSO1, and other mechanisms are identical with R2HMOPSO algorithm.

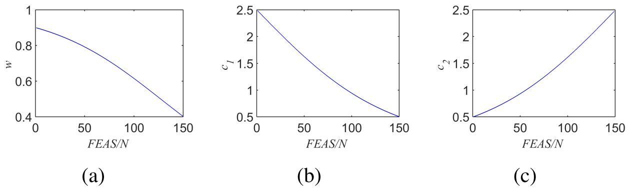

FIGURE 2. Variation of inertia weight and learning factors. (a) Variation of inertia weight (w). (b) Variation of individual factor (c_1). (c) Variation of social factor (c_2).

B. POPULATION EVOLUTION STRATEGY

1) SELECTION OF LEADER PARTICLES

In the process of multi-objective particle swarm optimization, the selections of leader particles include individual particles and global particle (*pbest* and *gbest*) are essential for the updating process of population. The reasonable selection of *pbest* helps to improve local search space for population, while the reasonable selection of *gbest* helps to global search space for population. The procedure of update for *pbest* is listed in Algorithm 1.

Algorithm 1 Update for *pbest*

Input:
  $pbest, pop$;
Output:
  $pbest$;

for $i = 1$ to N **do**
  if $g^{te}(x_i|w_i, z^*) < g^{te}(pbest_i|w_i, z^*)$
    $pbest_i = x_i$
    $age_i = 0$
  else
    $age_i = age_i + 1$
  end if
end for

The selections of individual optimal particles contribute to the development of local regions. In this paper, *pbest* is updated by calculating Tchebycheff aggregate value. The particle depends on its historical merit to the front flight. The principle of selecting the optimal particle is: the optimal particle $x_{pb,i}$ of the i-th particle, i represents the optimal position of the i-th subproblem of the particle, set $x_{pb,i} = x_i$. Then the individual optimal particles are updated. If the Tchebycheff aggregate value of the new particle is better than the value of the previous particle, the previous particle will be replaced by the new particle. Otherwise, the individual optimal particle remains unchanged. The selection of the global optimal particle is very important for population guidance. In this paper, we choose one of the non-dominated particles in external archive as the global optimal particle. The principle of selecting the global optimal particle is that the non-dominated particles in external archives are sorted according to the R2 indicator contribution value. Then the particle with the largest R2 indicator contribution value is taken as global optimal particle.

2) RE-INITIALIZE THE PARTICLES

In order to prevent the proposed algorithm from falling into the local optimum and sticking into premature convergence, the simplest method of increasing the perturbation is to re-initialize the particles. During the particle updating process, if a particle is not updated, the age of the particle increases by one. But if the age of the particle reaches its maximum value, the particle will be re-initialized. Therefore, this paper uses simulation binary crossover operator to re-initialize the particle position. The re-initialization formula is as follows:

$$x_i^{t+1}(j) = 0.5 * ((1-gabeta) * x_{gb}^t(j) + (1+gabeta) * x_i^t(j)) \tag{10}$$

$$gabeta = \begin{cases} (2r)^{\frac{1}{miu+1}}, & r \le 0.5 \\ (\frac{1}{2(1-r)})^{\frac{1}{miu+1}}, & otherwise \end{cases} \tag{11}$$

Where r is a random number between (0, 1), $miu = 2$ is the distribution probability.

3) USING R2 INDICATOR CONTRIBUTION VALUE TO UPDATE AND MAINTAIN EXTERNAL ARCHIVES

The updating and maintenance of the external archive can reflect the convergence and distribution of the algorithm. As the number of iteration of the particle swarm algorithm increases, the obtained non-dominated solutions will increase and even exceed the default quantity of external archive. Therefore, it is essential to update and maintain external archive.

As we all known, crowding distance is used to update and maintain external archive widely in optimization algorithms [1], [18]. In recent years, R2 indicator contribution value is being designed as an update mechanism gradually [9], [14].

In order to verify R2 indicator contribution value is better than crowding distance in the selection of the non-dominated solutions, the ZDT1 problem is taken as an example in this paper. A set of non-dominated solutions ($N = 100$) is selected from the true front of ZDT1 randomly. R2 indicator contribution value and crowding distance are used to select 20 solutions respectively. Then, these 20 solutions are calculated for Spacing [15] (Sp) performance. The strategy of the two methods are shown in Fig.3.

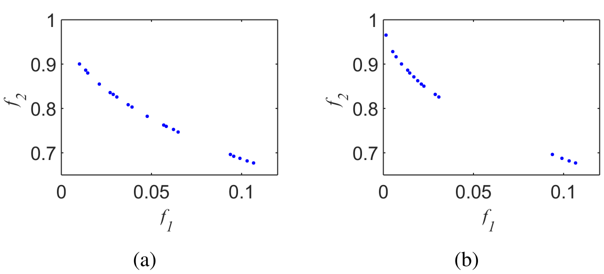

FIGURE 3. The comparison of R2 indicator contribution strategy and crowding distance strategy. (a) R2 indicator strategy. (b) crowding distance strategy.

The results of the two strategies are calculated by using the Spacing performance respectively:

(1) The strategy of R2 indicator contribution value is used to select particles and $Sp = 0.0233$;

(2) The strategy of crowding distance is used to select particles and $Sp = 0.1483$.

The Spacing performance value is more lower, which shows the solutions are more uniformly spread. From the Sp values obtained above, it can be seen that the R2 indicator contribution value strategy shows better performance than the crowding distance in the distribution of solutions. Hence, R2 indicator contribution value is proposed to update and maintain external archive in this paper.

The process of combining dominance-based method and CR2 contribution value selection is shown in Fig.4.

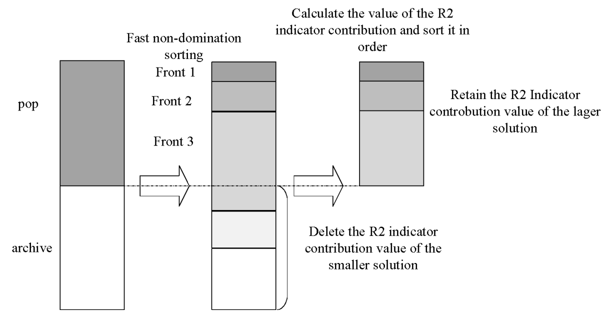

FIGURE 4. The process of fast non-dominated and CR2 value screening.

In order to clearly demonstrate the update process of external archive, the procedure of update for archive is listed in Algorithm 2.

Algorithm 2 Update for *archive*

Input:
 $archive, pop, A = \emptyset, arch = \emptyset, rank = \emptyset, i = 1$
Output:
 $archive$
$archive = archive \bigcup pop$
execute non-dominated sorting in archive
calculate CR2 value on each rank
sort particles according to CR2 on each rank
while $|A| < \frac{|archive|}{2}$
 $|A| = |A| + |rank(i)|$
 $i = i + 1$
end
$arch = arch \bigcup rank(1) \bigcup, \ldots, \bigcup rank(i-1)$
$rank(i) = rank(i)_{\frac{|archive|}{2} - |arch|}$
$archive = arch \bigcup rank(i)$

The program of particle selection in external archives is as follows:

(1) The non-domination sorting is designed to classify the population, then R2 indicator contribution value for each particle is calculated on each rank according to Eq.(4). Those particles will be selected into the next external

archive according to the rank sort until the number of selected particles is equal to the preset number of external archive.

(2) When all particles are non-dominated solutions in external archive and when the number of particles exceeds the archive size, external archive needs to be pruned. Particles have a smaller R2 indicator contribution value would be removed until the number of particles meets the default size of external archive.

CR2 value is used as a deletion mechanism in external archive. However, this selection mechanism also needs to add a disturbance to increase the diversity of the population. Therefore, this paper uses polynomial mutation proposed by Deb [5] to compensate for this deficiency and increase the diversity of the population in external archives.

C. THE PROCEDURE OF R2HMOPSO ALGORITHM

The above content has described the improved speed update formula and population evolution strategy, which compose the main components of R2HMOPSO algorithm. The procedure of R2HMOPSO algorithm is shown in Algorithm 3.

Algorithm 3 R2HMOPSO Algorithm

Input:

popuation size(N), number of evaluation ($FEAS$), scale of archive($ARCH$), weight vectors(W), agemax of particle(T_a);

Output:

archive;

Initialization;

```
initialize the velocity and position of the population(pop)
initialize the reference point z* = (z*_1, z*_2, ..., z*_m), z*_j = minf_j(x) and j = 1, 2, .., m
set pbest = pop
archive = pop, then execute Algorithm 2
update gbest according to CR2 value
Iterations and updates
while evaluation <FEAS do
    for i = 1 to N do
        if age_i <T_a
            update the particle according to Eq.(5) and Eq.(6)
        else
            re-initialize the particle according to Eq.(10) and Eq.(11)
        end if
    end for
update the reference point z* according to fitness
update pbest according to Algorithm 1
update archive according to Algorithm 2
perform polynomial mutation in archive
update gbest according to CR2 value
end while
return archive
```

IV. PERFORMANCE TESTING AND EXPERIMENTAL RESULTS ANALYSIS

A. TEST PROBLEMS

This paper uses nineteen test problems with different characteristics of the Pareto front, which includes convex, concave, discontinuous and multi-modality. Among them are two-objective test suites of Zitzler-Deb Thiele (ZDT) [24], three-objective problems of Deb-Thiele-Laumanns Zitzler (DTLZ) [6] and CEC'09 benchmark problems (UF1-UF10) [21].

B. PERFORMANCE METRIC

The inverse generation distance (IGD) [27] is a comprehensive metric. IGD measures the average Euclidean distance from uniformly distributed points along the whole Pareto front to their closest solution in the obtained solution set. So, it avoids the situation that all obtained solutions concentrate to one point which may lead the convergence well while the diversity is not satisfied. Thus, IGD can measure convergence and diversity of a algorithm simultaneously. The lower value shows that the algorithm gains a better performance. The formula is as follows:

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

Where P is the number of groups on the true front; $P^*$is the Pareto solutions set for the multi-objective algorithm; $|P|$ is the population size of P; $d(P_i, P^*)$ represents the minimum between P_i and P^*.

C. EXPERIMENTAL SETTINGS

In the R2HMOPSO algorithm, the parameters are as follows: the inertia weight $w_{start} = 0.9$, $w_{end} = 0.4$; the social learning factor $c_{1start} = 2.5$, $c_{1end} = 0.5$; the individual learning factor $c_{2start} = 0.5$, $c_{2end} = 2.5$.

TABLE 1. Parameter settings for each algorithm.

Algorithm	Parameter setting
R2HMOPSO	$T_a = 2, p_c = 0.5, p_m = 1/V$
MOEA/D	$T = 20, p_c = 0.5, p_m = 1/V, \eta_c = \eta_m = 20$
NSGAII	$p_c = 0.9, p_m = 1/V, \eta_c = \eta_m = 20$
dMOPSO	$T = 20, T_a = 2$
R2MOPSO	$T = 20, T_a = 2, p_m = 1/V$

The parameter setting of the each algorithm in Table 1 is consistent with the parameter setting in the original paper. In order to display the effectiveness of the proposed strategy, the parameters of the R2HMOPSO algorithm and the parameters of the compared algorithms are set to be the same. Parameters include neighborhood size T, particle of agemax T_a, probability of crossover p_c, probability of mutation p_m, distribution index $\eta_c = \eta_m$.

TABLE 2. Statistics on IGD performance indicators on ZDT and DTLZ test functions.

Problem		R2HMOPSO	R2HMOPSO1	MOEA/D	NSGAII	dMOPSO	R2MOPSO
ZDT1	mean	3.904E-03	3.943E-03	7.544E-03	4.929E-03	**3.899E-03**	3.928E-03
	std	6.209E-05	8.593E-05	8.473E-04	2.043E-04	**3.839E-05**	1.091E-04
	p		2.068E-02	3.020E-11	3.020E-11	***7.172E-01***	1.273E-02
ZDT2	mean	3.834E-03	2.260E-01	2.503E-02	4.900E-03	6.442E-02	**3.825E-03**
	std	4.199E-05	2.969E-01	1.104E-01	2.074E-04	1.849E-01	**3.336E-05**
	p		4.574E-04	6.526E-07	3.020E-11	***8.766E-01***	***1.373E-01***
ZDT3	mean	6.123E-02	8.460E-03	1.212E-02	**7.254E-03**	1.064E-02	1.017E-02
	std	1.583E-01	**6.058E-04**	5.585E-03	7.634E-03	7.098E-04	6.508E-04
	p		***1.120E-01***	5.091E-06	4.998E-09	1.680E-03	4.515E-02
ZDT4	mean	**4.518E-03**	5.093E-01	2.452E-01	8.420E-03	5.972E+00	8.374E-02
	std	3.733E-03	2.356E-01	1.571E-01	**2.731E-03**	4.477E+00	9.787E-02
	p		3.020E-11	4.504E-11	5.573E-10	2.800E-11	3.690E-11
ZDT6	mean	1.888E-03	1.868E-03	1.892E-03	2.696E-03	1.879E-03	**1.865E-03**
	std	4.334E-05	2.227E-05	1.179E-05	4.111E-05	**8.549E-06**	3.484E-05
	p		4.676E-02	***9.823E-01***	3.020E-11	***6.145E-02***	3.917E-02
DTLZ1	mean	1.994E-02	1.724E+00	**1.842E-02**	1.935E-02	1.529E+01	1.745E+01
	std	2.080E-03	9.906E-01	**4.773E-04**	6.011E-04	1.180E+01	3.133E+00
	p		3.020E-11	4.459E-04	***9.000E-01***	3.020E-11	3.020E-11
DTLZ2	mean	**4.368E-02**	5.555E-02	4.597E-02	5.315E-02	4.716E-02	2.250E-01
	std	**1.203E-03**	3.302E-03	1.218E-03	2.373E-03	1.489E-03	1.526E-02
	p		***1.000E+00***	9.833E-08	3.020E-11	1.070E-09	3.020E-11
DTLZ4	mean	5.198E-02	2.505E-01	**5.160E-02**	5.417E-02	1.175E-01	4.343E-01
	std	2.011E-03	1.262E-01	1.948E-03	**1.727E-03**	7.627E-02	5.111E-02
	p		3.020E-11	***4.119E-01***	9.792E-05	3.020E-11	3.020E-11
DTLZ7	mean	7.036E-02	**6.459E-02**	1.271E-01	7.129E-02	1.265E-01	8.729E-02
	std	8.439E-03	9.521E-03	**5.106E-03**	5.101E-02	6.728E-03	8.431E-03
	p		***7.730E-02***	5.084E-03	5.573E-10	2.324E-02	9.919E-11
+/=/-			5/3/1	5/2/2	7/1/1	5/3/1	6/1/2

In order to verify that the Sigmoid function is used to dynamically adjust w, c_1 and c_2 are better than the convergence and distribution of the same algorithm using fixed weights, let $w = 0.925$, $c_1 = c_2 = 2.0$ in the R2HMOPSO1 algorithm.

ZDT: $m = 2$, $N = 100$, $FEAS = 30000$; DTLZ: $m = 3$, $N = 210$, $FEAS = 210000$; UF: $N = 300$, $FEAS = 300000$. For each test function, all algorithms run 30 times independently.

D. RESULTS ANALYSIS

In order to evaluate the performance of the algorithm, the results of the R2HMOPSO algorithm are compared with MOEA/D [22], NSGAII [6], dMOPSO [5] and R2MOPSO [10].

Table 2 and Table 3 show the mean and standard deviation (std) of the IGD performance metrics of the five algorithms. The bold font is the optimal value obtained for each algorithm on the same test function. In addition, Table 2 and Table 3 also record the Wilconxon [19] rank sum test of p value, which is the IGD statistic of R2HMOPSO algorithm and the other five algorithms on the same test function. The significant difference level $a = 0.05$, that is, if the value of p is greater than 0.05, there is no significant difference between the two algorithms (In italics in the table). R2HMOPSO algorithm and the compared algorithms are tested in the same rank test function of the IGD value, *better/similar/inferior*$(+/=/-)$ show that R2HMOPSO algorithm is superior to, similar to, and inferior to the number of the compared algorithm.

First, Table2 and Table3 show that the IGD performance of R2HMOPSO algorithm is better than the performance of R2HMOPSO1 algorithm. The IGD value of R2HMOPSO algorithm is inferior to R2HMOPSO1 algorithm's only on ZDT6 and UF5. For these data show that the Sigmoid function mapping method is used to improve the convergence and distribution of the algorithm.

As shown in Table2 and Fig.5, R2HMOPSO algorithm produces a better approximation along the Pareto front on ZDT functions except ZDT3 and ZDT6. However, the difference between R2HMOPSO algorithm and dMOPSO algorithm is not significant difference on ZDT1, ZDT2 and ZDT6. While R2HMOPSO algorithm programs a better IGD performance than other four algorithms on ZDT4. Meanwhile, Table 2 and Fig.6 show that R2HMOPSO algorithm obtains a better distribution and convergence than other four algorithms, but R2HMOPSO algorithm performs slightly worse than MOEA/D on DTLZ1. In addition, the std value of R2HMOPSO algorithm is worse than other algorithms except on DTLZ4, because there are some bad IGD value during the 30 times run. It can be seen from the above figures and data, R2HMOPSO algorithm performs better than other algorithms except on ZDT3 and DTLZ1, because R2HMOPSO algorithm still needs to be improved in dealing with discontinuous and multi-modality problems.

As can be seen in Table3 and Fig.7, R2HMOPSO algorithm performs worse than other algorithms on UF3, UF8, and UF9. UF3 is two-objective instance and its PS shape is nonlinear spiral type in the decision space. UF8 is three-objective

TABLE 3. Statistics on IGD performance indicators on UF test functions.

Problem		R2HMOPSO	R2HMOPSO1	MOEA/D	NSGAII	dMOPSO	R2MOPSO
UF1	mean	**6.081E-02**	1.007E-01	8.155E-02	8.653E-02	9.159E-02	2.124E-01
	std	1.770E-02	1.661E-02	3.326E-02	2.873E-02	**1.151E-02**	2.008E-02
	p		4.183E-09	2.068E-02	2.531E-04	2.602E-08	3.020E-11
UF2	mean	**1.844E-02**	1.628E-01	2.821E-02	2.828E-02	4.729E-02	9.984E-02
	std	7.145E-03	4.565E-02	5.200E-03	8.947E-03	8.118E-03	**3.533E-03**
	p		3.020E-11	2.390E-08	3.646E-08	3.159E-10	3.020E-11
UF3	mean	6.203E-01	7.520E-01	4.724E-01	**2.367E-01**	5.012E-01	4.614E-01
	std	1.766E-01	**5.794E-02**	7.183E-02	8.829E-02	2.563E-01	9.913E-02
	p		8.120E-04	6.736E-06	2.439E-09	***1.296E-01***	5.462E-06
UF4	mean	**3.758E-02**	7.234E-02	6.793E-02	5.173E-02	9.213E-02	7.179E-02
	std	2.644E-03	2.852E-03	3.347E-03	1.771E-03	4.959E-03	**1.377E-03**
	p		8.120E-04	3.020E-11	3.020E-11	3.020E-11	3.020E-11
UF5	mean	1.748E-01	**7.234E-02**	4.526E-01	3.233E-01	7.095E-01	1.720E+00
	std	2.725E-02	**2.852E-03**	1.281E-01	1.159E-01	1.731E-01	1.369E-01
	p		3.020E-11	4.504E-11	2.034E-09	3.020E-11	3.020E-11
UF6	mean	**1.748E-01**	4.349E-01	5.271E-01	3.690E-01	4.394E-01	8.412E-01
	std	6.451E-02	**5.192E-02**	1.828E-01	1.334E-01	2.098E-01	1.123E-01
	p		3.020E-11	8.101E-10	3.825E-09	5.092E-08	3.020E-11
UF7	mean	**3.137E-02**	8.292E-02	2.704E-01	1.478E-01	1.372E-01	2.094E-01
	std	7.625E-03	**6.702E-03**	1.901E-01	1.576E-01	1.296E-01	1.827E-02
	p		3.020E-11	9.792E-05	***2.458E-01***	4.616E-10	3.020E-11
UF8	mean	4.144E-01	3.707E-01	**1.170E-01**	1.601E-01	1.968E-01	2.336E-01
	std	5.216E-02	8.129E-02	1.198E-01	4.457E-02	**3.136E-02**	5.568E-02
	p		***3.871E-01***	5.573E-10	3.690E-11	3.020E-11	2.610E-10
UF9	mean	2.362E-01	4.521E-01	1.434E-01	2.425E-01	**1.142E-01**	4.385E-01
	std	5.908E-02	3.225E-02	5.087E-02	8.280E-02	2.881E-02	**1.792E-02**
	p		3.020E-11	3.081E-08	***8.303E-01***	3.197E-09	3.020E-11
UF10	mean	**2.384E-01**	4.572E-01	5.945E-01	7.823E-01	1.011E+00	4.196E-01
	std	**5.693E-02**	2.010E-01	2.187E-01	1.550E-01	1.742E-01	2.376E-01
	p		***4.290E-01***	1.680E-03	4.975E-11	2.310E-10	***7.978E-02***
+/=/-			7/2/1	7/0/3	6/2/2	7/1/2	7/1/2
total			12/5/2	12/2/5	13/3/3	12/4/3	13/2/4

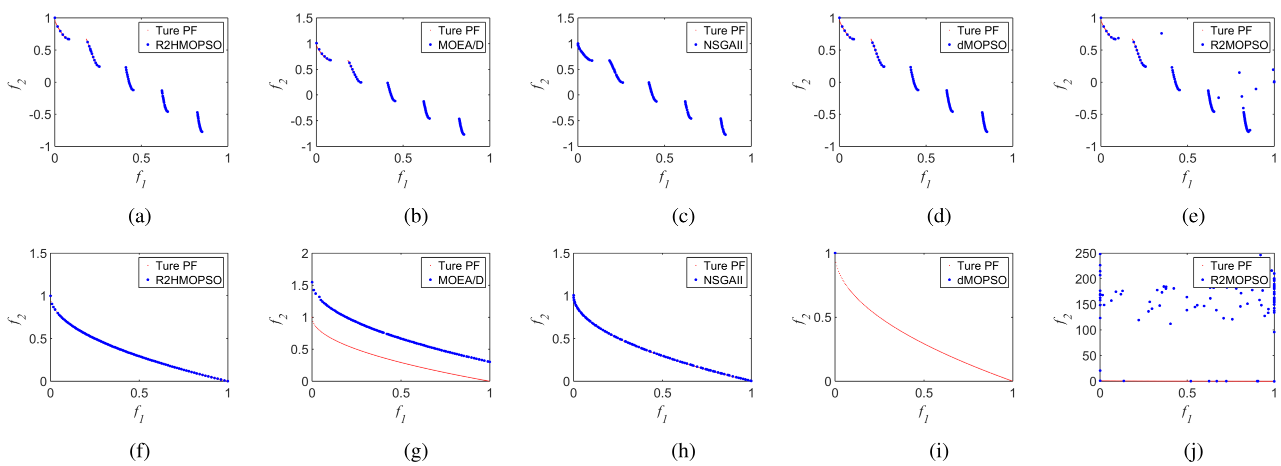

FIGURE 5. The solutions set of each compare algorithm on the PF of ZDT3, ZDT4. (a)-(e) represent the compare algorithms on ZDT3. (f)-(j) represent the compare algorithms on ZDT4.

instance and its PS shape is parabolic type in the decision space. Besides, UF9 is three-objective instance and its PS shape is planar type in the decision space. R2HMOPSO algorithm obtains worse IGD value when dealing with such complicated problems. Overall, R2HMOPSO algorithm is superior to the other five compared algorithms in the most of the nineteen test problems.

Fig.5, Fig.6 and Fig.7 show the approximate Pareto fronts obtained by R2HMOPSO, MOEA/D, NSGAII, dMOPSO and R2MOPSO in some clearer identified test problems. As can be seen from the above figures, R2HMOPSO algorithm generated the approximated Pareto fronts which demonstrate the better distribution on ZDT, DTLZ and most CEC'09 test problems.

Fig.8 shows the IGD descent tendency graph for the five algorithms with the recorded data being the experimental average of 30 runs independently. The abscissa is the total number of evaluation per operation and it is notes that

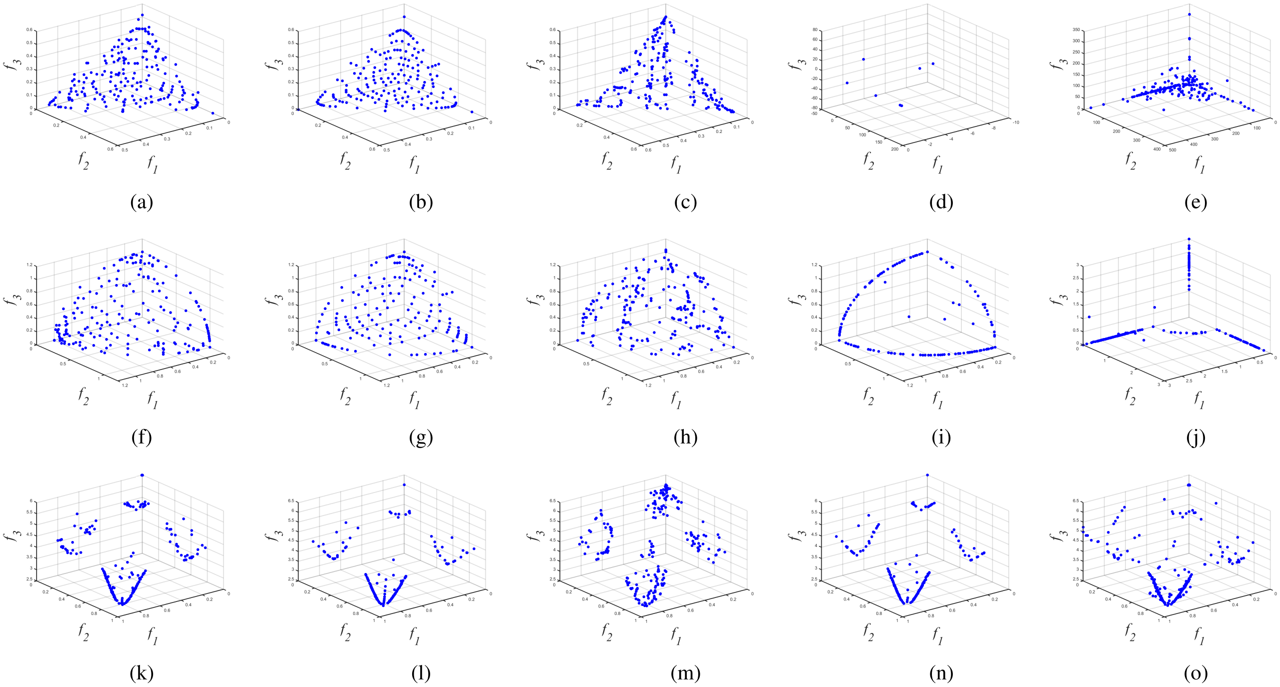

FIGURE 6. The solutions set of each compare algorithm on the PF of DTLZ1, DTLZ4, DTLZ7. (a)-(e) represent the compare algorithms on DTLZ1. (f)-(j) represent the compare algorithms on DTLZ4. (k)-(o) represent the compare algorithms on DTLZ7.

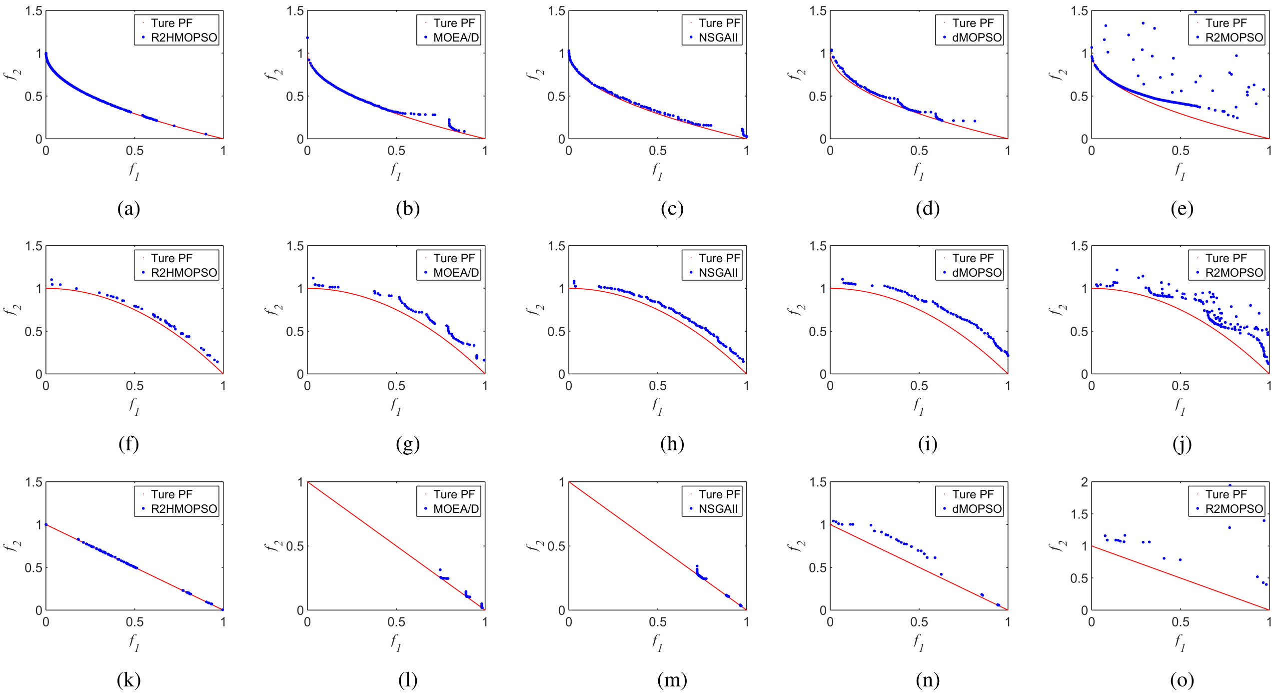

FIGURE 7. The solutions set of each compare algorithm on the PF of UF2, UF4, UF7. (a)-(e) represent the compare algorithms on UF2. (f)-(j) represent the compare algorithms on UF4. (k)-(o) represent the compare algorithms on UF7.

the ordinate is *log*(IGD) for ease of observing expediently. It can be seen from Fig.8 that the IGD of the R2HMOPSO algorithm is the fastest on UF1, UF2, UF4, UF5, UF6 and UF7 problems, and most of other functions are in the suboptimal position. However, R2HMOPSO algorithm does not converge quickly on ZDT2, ZDT4 and DTLZ7 problems.

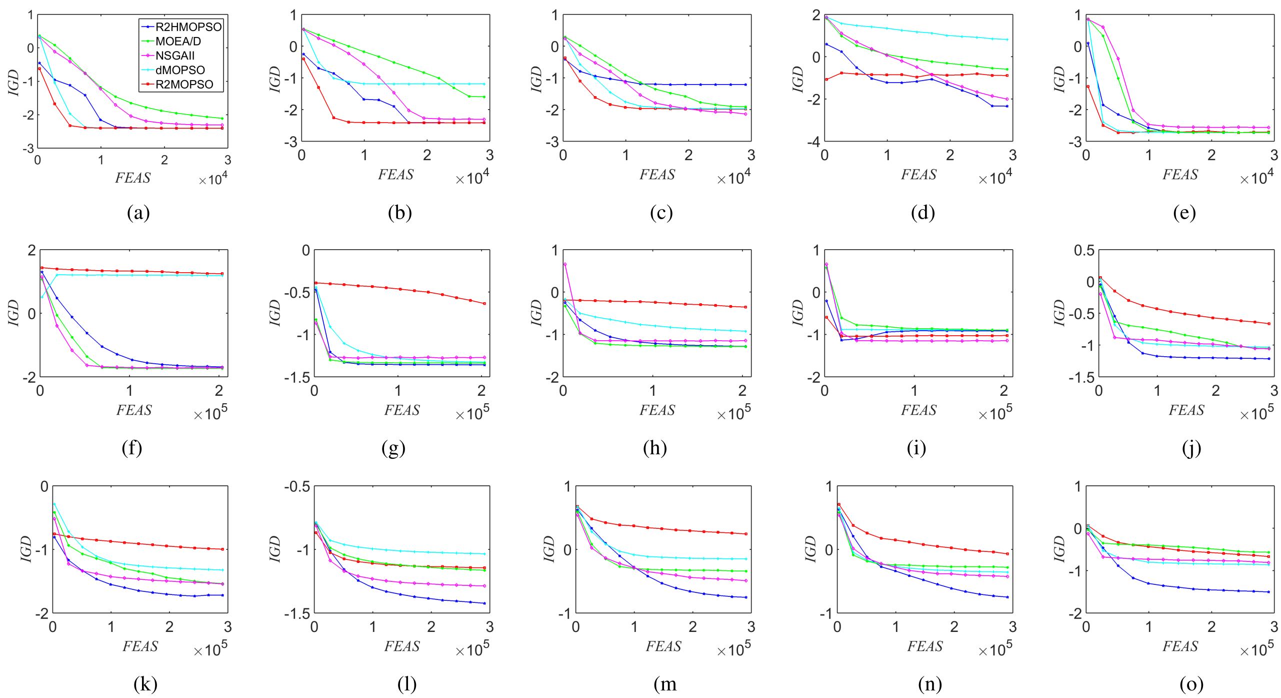

FIGURE 8. **Convergence curve of IGD metric value. (a) ZDT1. (b) ZDT2. (c) ZDT3. (d) ZDT4. (e) ZDT6. (f) DTLZ1. (g) DTLZ2. (h) DTLZ4. (i) DTLZ7. (j) UF1. (k) UF2. (l) UF4. (m) UF5. (n) UF6. (o) UF7.**

In particular, DTLZ7 firstly converged to the lowest and then increased because there is no good retention of the elite solutions. In short, R2HMOPSO algorithm obtains a good distribution and convergence, which indicates R2HMOPSO algorithm is competitive and effective algorithm.

According to the law of no free lunch, the algorithm is proposed in this paper, which cannot guarantee all test problems are superior to other algorithms. The above data and graphs display that R2HMOPSO algorithm is better than the other four algorithms in most of test problems. It is not difficult to find that R2HMOPSO algorithm obtains a well-distributed solutions, which indicates the great potential of R2HMOPSO to deal with complex MOPs.

E. COMPLEXITY ANALYSIS

The complexity analysis is based on the size of the population and the size of the external archive. The complexity of MOEA/D algorithm is $O(TN)$, which is obtained by the neighborhood size T and population size N. The complexity of NSGAII algorithm is $O(N^2)+O(NlogN)$, which is obtained by using non-dominated ordering and crowding distance calculations. Due to $O(N^2) > O(NlogN)$, the complexity of NSGAII algorithm is $O(N^2)$. The dMOPSO algorithm introduces the re-initialization of particles on the basis of decomposition and the complexity of dMOPSO algorithm is $O(N^2)$. Based on the decomposition and re-initialization of the particles, the R2MOPSO algorithm updates *gbest* with the calculation of R2 indicator. The complexity of R2MOPSO algorithm is $O(N^2) + O(NlogN)$, because $O(N^2) > O(NlogN)$, the complexity of R2MOPSO is $O(N^2)$. Based on the decomposition and re-initialization of the particles, R2HMOPSO algorithm uses non-dominated sorting and R2 indicator contribution value. The complexity of R2HMOPSO algorithm is $O(N^2) + O(N^2) + O(NlogN)$, because $O(N^2) > O(NlogN)$, the complexity of R2HMOPSO algorithm is $O(N^2)$. Through the above data, R2HMOPSO algorithm does not increase the complexity of the algorithm, and has the same level of complexity as NSGAII, dMOPSO and R2MOPSO.

V. CONCLUSION AND PROSPECT

This paper presents a hybrid multi-objective particle swarm optimization algorithm based on R2 indicator. In external archive, R2 indicator contribution value is designed to select particles instead of crowding distance value, which enhances the distribution of the population. In addition, R2HMOPSO algorithm combines dominance-based method and indicator-based method effectively. Besides, simulation binary crossover operator is designed to re-initialize the particles so as to prevent particles from falling into the local optimum. Furthermore, external archive uses polynomial mutations to increase the diversity of the population. And in order to balance the global search and the local search processing, this paper proposes to improve the particle swarm optimization algorithm updating formula by using Sigmoid function mapping, which improves the convergence of the algorithm effectively.

Actually, R2HMOPSO algorithm is faced with some challenging and promising research direction. Although the R2HMOPSO algorithm is improved in performance, the convergence rate needs to be improved and the effectiveness of solving discontinuous problems needs to be enhanced. Furthermore, since the calculation of R2 indicator is not affected by the spatial dimension, the algorithm for exploring many-objective will be the direction of future research.

ACKNOWLEDGMENT
The authors would like to appreciate the editor and reviewers for their helpful comments and suggestions to improve the quality of this paper.

REFERENCES
[1] N. Al Moubayed, A. Petrovski, and J. Mccall, "D^{2}MOPSO: MOPSO based on decomposition and dominance with archiving using crowding distance in objective and solution spaces," *Evol. Comput.*, vol. 22, no. 1, pp. 47–77, 2014.
[2] I. C. García, C. A. C. Coello, and A. Arias-Montaño, "MOPSOhv: A new hypervolume-based multi-objective particle swarm optimizer," in *Proc. Evol. Comput.*, Jul. 2014, pp. 266–273.
[3] C. A. C. Coello and M. S. Lechuga, "MOPSO: A Proposal for multiple objective particle swarm optimization," in *Proc. IEEE Congr. Evol. Comput. (CEC)*, May 2002, pp. 1051–1056.
[4] C. A. C. Coello and S. W. Martínez, "A multi-objective particle swarm optimizer based on decomposition,"in *Proc. Conf. Gen. Evol. Comput. ACM*, 2011, pp. 69–76.
[5] K. Deb, A. Pratap, S. Agarwal, and T. Meyarivan, "A fast and elitist multiobjective genetic algorithm: NSGA-II," *IEEE Trans. Evol. Comput.*, vol. 6, no. 2, pp. 182–197, Apr. 2002.
[6] K. Deb, L. Thiele, M. Laumanns, and E. Zitzler, "Scalable multi-objective optimization test problems," in *Proc. IEEE Congr. Evol. Comput. (CEC)*, May 2002, pp. 825–830.
[7] M. G. Gong *et al.*, "Research on evolutionary multi-objective optimization algorithms," *J. Softw.*, vol. 20, no. 20, pp. 271–289, 2009.
[8] J. Kennedy and R. Eberhart, "Particle swarm optimization," in *Proc. IEEE Int. Conf. Neural Netw.*, vol. 4. Nov./Dec. 1995, pp. 1942–1948.
[9] F. Li, J. Liu, S. Tan, and X. Yu, "R2-MOPSO: A multi-objective particle swarm optimizer based on R2-indicator and decomposition," in *Proc. Evol. Comput.*, 2015, pp. 3148–3155.
[10] S. K. Saha, R. Kar, D. Mandal, and S. P. Ghoshal, "IIR filter design and identification using NPSO technique," in *Proc. IEEE Int. Conf. Knowl. Smart Technol.*, Jan./Feb. 2013, pp. 128–133.
[11] A. Man-Im, W. Ongsakul, J. G. Singh, and C. Boonchuay, "Multi-objective optimal power flow using stochastic weight trade-off chaotic NSPSO," in *Proc. IEEE Smart Grid Technol.-Asia*, Nov. 2016, pp. 1–8.
[12] B. Mohammadi-Ivatloo, M. Moradi-Dalvand, and A. Rabiee, "Combined heat and power economic dispatch problem solution using particle swarm optimization with time varying acceleration coefficients," *Electr. Power Syst. Res.*, vol. 95, pp. 9–18, Feb. 2013.
[13] P. Hu, L. Rong, C. Liang-Lin, and L. Li-Xian, "Multiple swarms multi-objective particle swarm optimization based on decomposition," *Procedia Eng.*, vol. 15, no. 2, pp. 3371–3375, 2011.
[14] D. H. Phan and J. Suzuki, "R2-IBEA: R2 indicator based evolutionary algorithm for multiobjective optimization," in *Proc. IEEE Congr. Evol. Comput.*, Jun. 2013, pp. 1836–1845.
[15] J. R. Schott, "Fault tolerant design using single and multicriteria genetic algorithm optimization," M.S. thesis, Dept. Aeronautics Astron., Massachusetts Inst. Technol., Cambridge, MA, USA, 1995, vol. 37.
[16] A. P. Meagher, F. A. Frizelle, and S. Janes, "Practice parameters for sigmoid diverticulitis," *Diseases Colon Rectum*, vol. 50, no. 5, pp. 683–685, 2007.
[17] S.-J. Tsai, T.-Y. Sun, C.-C. Liu, S.-T. Hsieh, W.-C. Wu, and S.-Y. Chiu, "An improved multi-objective particle swarm optimizer for multi-objective problems," *Expert Syst. Appl.*, vol. 37, no. 8, pp. 5872–5886, 2010.
[18] D. Wang, D. Tan, and L. Liu, "Particle swarm optimization algorithm: An overview," *Soft Comput.*, vol. 22, no. 2, pp. 387–408, 2017.
[19] F. Wilcoxon, "Some rapid approximate statistical procedures," *Ann. New York Acad. Sci.*, vol. 52, no. 6, pp. 808–814, 2010.
[20] Q. Zhang and H. Li, "MOEA/D: A multiobjective evolutionary algorithm based on decomposition," *IEEE Trans. Evol. Comput.*, vol. 11, no. 6, pp. 712–731, Dec. 2007.
[21] Q. Zhang *et al.*, "Multiobjective optimization test instances for the CEC 2009 special session and competition," Univ. Essex, Colchester, U.K. and Nanyang Technol. Univ., Singapore, Tech. Rep., 2008, p. 264.
[22] L. Zhang, Y. Tang, C. Hua, and X. Guan, "A new particle swarm optimization algorithm with adaptive inertia weight based on Bayesian techniques," *Appl. Soft Comput.*, vol. 28, pp. 138–149, Mar. 2015.
[23] A. Zhou, B.-Y. Qu, H. Li, S.-Z. Zhao, P. N. Suganthan, and Q. Zhang, "Multiobjective evolutionary algorithms: A survey of the state of the art," *Swarm Evol. Comput.*, vol. 1, no. 1, pp. 32–49, 2011.
[24] E. Zitzler, K. Deb, and L. Thiele, "Comparison of multiobjective evolutionary algorithms: Empirical results," *Evol. Comput.*, vol. 8, no. 2, pp. 173–195, 2000.
[25] D. Brockhoff, T. Wagner, and H. Trautmann, "2 indicator-based multiobjective search," *Evol. Comput.*, vol. 23, no. 3, pp. 369–395, 2015.
[26] Y. Shi and R. Eberhart, "Modified particle swarm optimizer," in *Proc. IEEE ICEC Conf.*, vol. 6. Anchorage, AK, USA, 1998, pp. 69–73.
[27] F. Neri and C. Cotta, "Memetic algorithms and memetic computing optimization: A literature review," *Swarm Evol. Comput.*, vol. 2, pp. 1–14, Feb. 2012.
[28] W. Peng and Q. Zhang, "A decomposition-based multi-objective particle swarm optimization algorithm for continuous optimization problems," in *Proc. IEEE Int. Conf. Granular Comput.*, Aug. 2008, pp. 534–537.
[29] A. Jabri, A. El Barkany, and A. El Khalfi, "Multi-objective optimization using genetic algorithms of multi-pass turning process," in *Proc. IEEE Int. Conf. Ind. Eng. Syst. Manage.*, Oct. 2014, pp. 601–610.
[30] J. Bader and E. Zitzler, "HypE: An algorithm for fast hypervolume-based many-objective optimization," *Evol. Comput.*, vol. 19, no. 1, pp. 45–76, 2011.
[31] M. Hu, T. Wu, and J. D. Weir, "An adaptive particle swarm optimization with multiple adaptive methods," *IEEE Trans. Evol. Comput.*, vol. 17, no. 5, pp. 705–720, Oct. 2013.
[32] A. Trivedi, D. Srinivasan, A. Ghosh, and K. Sanyal, "A survey of multiobjective evolutionary algorithms based on decomposition," *IEEE Trans. Evol. Comput.*, vol. 21, no. 3, pp. 440–462, Jun. 2017.
[33] K. Li, Q. Zhang, S. Kwong, M. Li, and R. Wang, "Stable matching-based selection in evolutionary multiobjective optimization," *IEEE Trans. Evol. Comput.*, vol. 18, no. 6, pp. 909–923, Dec. 2014.
[34] C. K. Monson, "Simple adaptive cognition for PSO," in *Proc. IEEE Evol. Comput.*, Jun. 2011, pp. 1657–1664.
[35] Y. Wang, Y. Wu, F. Zhou, Z. Chu, Y. Wu, and F. Yuan, "Multi-objective resource allocation in a NOMA cognitive radio network with a practical non-linear energy harvesting model," *IEEE Access*, to be published.
[36] R. Fan, L. Wei, X. Li, and Z. Hu, "A novel multi-objective PSO algorithm based on completion-checking," *J. Intell. Fuzzy Syst.*, vol. 34, no. 1, pp. 321–333, 2018.

LI-XIN WEI received the B.Eng. degree in manufacturing and automation, and the M.Eng. and Ph.D. degrees in control theory and control engineering from Yanshan University in 2000, 2002, and 2005, respectively. He is currently a Professor of control theory and control engineering with Yanshan University. His current research interests include system optimization theory and application, metallurgical integrated automation, and intelligent control theory and application.

XIN LI is currently pursuing the M.Eng. degree in control theory and control engineering with Yanshan University, China. Her current research interests include system optimization theory and applications.

RUI FAN is currently pursuing the Ph.D. degree in control theory and control engineering with Yanshan University, China. Her current research interests include system optimization theory and applications.

HAO SUN received the B.Eng. and Ph.D. degrees in control science and engineering from Yanshan University, Qinhuangdao, China, in 2009 and 2015, respectively. He is currently a Lecturer with the Department of Automation, Yanshan University. His current research interests include the control of metallurgical automation system and evolutionary algorithms.

ZI-YU HU received the B.Eng. degree in engineering from the North China Institute of Aerospace Engineering, Langfang, China, in 2012. He is currently pursuing the Ph.D. degree in control science and engineering with Yanshan University, Qinhuangdao, China. His current research interests include multi-objective optimization evolutionary algorithm and the simulation and control of complex system.