

Research Article

Decomposition-Based Multiobjective Optimization with Invasive Weed Colonies

Yanyan Tan ¹, Xue Lu,¹ Yan Liu,² Qiang Wang ¹ and Huaxiang Zhang¹

¹School of Information Science and Engineering, Shandong Normal University, Jinan 250358, China

²School of Information Science and Technology, Yunnan Normal University, Kunming, China

Correspondence should be addressed to Qiang Wang; wangqiang5169@163.com

Received 14 May 2019; Revised 4 July 2019; Accepted 14 July 2019; Published 6 August 2019

Academic Editor: Piotr Jędrzejowicz

Copyright © 2019 Yanyan Tan et al. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

In order to solve the multiobjective optimization problems efficiently, this paper presents a hybrid multiobjective optimization algorithm which originates from invasive weed optimization (IWO) and multiobjective evolutionary algorithm based on decomposition (MOEA/D), a popular framework for multiobjective optimization. IWO is a simple but powerful numerical stochastic optimization method inspired from colonizing weeds; it is very robust and well adapted to changes in the environment. Based on the smart and distinct features of IWO and MOEA/D, we introduce multiobjective invasive weed optimization algorithm based on decomposition, abbreviated as MOEA/D-IWO, and try to combine their excellent features in this hybrid algorithm. The efficiency of the algorithm both in convergence speed and optimality of results are compared with MOEA/D and some other popular multiobjective optimization algorithms through a big set of experiments on benchmark functions. Experimental results show the competitive performance of MOEA/D-IWO in solving these complicated multiobjective optimization problems.

1. Introduction

Multiobjective optimization problems (MOPs) widely exist in applications [1], such as design [2], scheduling [3–5], path planning [6], retrieval [7], and cloud computing [8]. These problems usually have two or more objectives, which often conflict with each other. Traditional mathematical methods often cannot deal with them well. Evolutionary algorithms present unique superiority in handling this type of problems. Due to the wide application scenes of MOPs, research on multiobjective evolutionary algorithms (MOEAs) remains prosperous [9–11].

Multiobjective evolutionary algorithms that have been proposed in literatures can be classified into three categories [9, 12]: the dominance-based approach, the indicator-based approach, and the decomposition-based approach.

(1) Dominance-based approach: in this type of approach, Pareto-dominance selection principle plays an important role in convergence process, among which the Pareto-based nondominated sorting approach is the most popular, where solutions having better Pareto ranks are selected. Besides, often a diversity maintaining strategy is needed for achieving an

even distribution of the Pareto optimal solutions. Improved strength Pareto EA (SPEA2) [13] and nondominated sorting genetic algorithm II (NSGA-II) [14] are two representative Pareto-based MOEAs, which perform effectively in solving 2-objective or 3-objective MOPs. However, when the number of objectives becomes large, selection pressure will reduce sharply and optimization process will become ineffective [15–17].

(2) Indicator-based approach: in this type of approach, a performance indicator such as hypervolume indicator or R2 indicator is used to measure the fitness of solutions by assessing their contributions. The used indicator needs the capability of measuring both convergence and diversity of an optimization algorithm. R2 indicator based evolutionary algorithm (R2-IBEA) [18], hypervolume-based evolutionary algorithm (HypE) [19], and hybrid Multiobjective Particle Swarm Optimization Algorithm Based on R2 Indicator (R2HMOPSO) [20] are three well-known indicator-based optimization algorithms.

(3) Decomposition-based approach: in this type of approach, an MOP is transformed into a series of single-objective optimization subproblems through decomposition

method, weighted sum approach, for example, and solves these subproblems simultaneously in a single run by an optimization algorithm. Decomposition-based method will utilize aggregated fitness value of solutions in selection process. Multiobjective genetic local search (MOGLS) [21], cellular genetic algorithm for multiobjective optimization (C-MOGA) [22], and MOEA/D [15] etc. are some of well famous representative MOEAs based on decomposition.

MOEA/D first proposed in 2007 [15] is a milestone in the development of MOEAs; it is a classical decomposition-based algorithm. MOEA/D defines a framework of multiobjective optimization; its improved version has won first on CEC 2009 [23]. Since being proposed, MOEA/D and its variants have solved many complex MOPs, which demonstrates that MOEA/D has lower computation complexity and performs better than NSGA-II in dealing with complex MOPs in a sense [24–27]. Therefore, its research is worthy of attention.

Invasive Weed Optimization (IWO) [28] first proposed in 2006, is a derivative-free metaheuristic algorithm mimicking the ecological behavior of colonizing weeds and distribution and is able to efficiently handle general linear, nonlinear, and multidimensional optimization problems [28, 29]. Since its proposal, IWO has been successfully applied in many practical optimization problems, such as developing a recommender system [30], many kinds of antenna configuration optimization [2, 31], and DNA computing [32].

Kundu et al. [33] proposed multiobjective invasive weed optimization (IWO) in 2011 and applied it on solving CEC 2009 MOPs. In their work, fuzzy dominance mechanism, instead of nondominated sorting, was carried out to sort the promising weeds in each iteration. Y. Liu et al. developed multiobjective invasive weed optimization for synthesis of phase-only reconfigurable linear arrays [2]. In addition, as far as we know, there has not much research on the multiobjective invasive weed optimization. Then in this work, we extend the classical IWO algorithm and integrate it into the framework of MOEA/D for well handling multiobjective problems. Based on the smart and distinct features of IWO and MOEA/D, we propose multiobjective invasive weed optimization algorithm based on decomposition (MOEA/D-IWO) and try to combine their excellent features in this extended hybrid algorithm. MOEA/D-IWO decomposes an MOP into a series of single-objective subproblems and solves them in parallel in each generation. The population consists of the best solutions searched so far for each subproblem, and each subproblem utilizes an extended IWO algorithm for evolution in each generation. The performance of the proposed MOEA/D-IWO in both convergence speed and optimality of results are compared with those of NSGA-II, MOEA/D, and some other multiobjective evolutionary algorithms on a big set of MOPs. Comparison results indicate the feasibility of IWO as a very hopeful metaheuristic candidate in the domain of multiobjective optimization.

The remaining parts of this paper are organized as follows. Section 2 formally describes the background knowledge on multiobjective optimization, the basic framework of MOEA/D, and an overview of IWO. Section 3 provides an adaptive modification of IWO and then integrates it into MOEA/D deducing our proposed MOEA/D-IWO.

Experiments are carried out and discussed in Section 4. Finally, Section 5 concludes this paper and prospects our further research.

2. Background

2.1. Multiobjective Optimization Problem. Without loss of generality, an unconstrained continuous multiobjective optimization problem (MOP) can be formally described as:

$$\begin{aligned} & \text{minimize} && F(x) = (f_1(x), f_2(x), \dots, f_m(x))^T \\ & \text{subject to} && x \in \Omega \end{aligned} \quad (1)$$

where $x = (x_1, x_2, \dots, x_n)^T$ is variable vector, n is the number of variables, Ω is the variable (decision) space, $F : \Omega \rightarrow R^m$ is composed of m real-valued objective functions, and R^m is the objective space [34, 35]. We can define the attainable objective set as $\{F(x) \mid x \in \Omega\}$.

Definition 1 ((domination) [35]). $u = (u_1, \dots, u_m)^T, v = (v_1, \dots, v_m)^T \in R^m$ are two vectors, u dominates v if $u_j \leq v_j$ for each $j \in \{1, 2, \dots, m\}$, and there must exist at least one $i \in \{1, 2, \dots, m\}$ satisfying $u_i < v_i$, which can be expressed as $u \succ v$.

Definition 2 ((Pareto optimal solution) [35]). A point $x^* \in \Omega$ is called a Pareto optimal to (1) if and only if there is no point $x \in \Omega$ satisfying $F(x)$ dominates $F(x^*)$.

Definition 3 ((Pareto optimal set (PS)) [35]). The Pareto optimal set (PS) can be defined as $PS = \{x^* \in \Omega \mid \neg \exists x \in \Omega : F(x) \succ F(x^*)\}$; it is the set of all Pareto optimal solutions.

Definition 4 ((Pareto front (PF)) [35]). Consisted with the definition of PS, Pareto front (PF) can be defined as $PF = \{F(x) \mid x \in PS\}$, indicating the set of all the Pareto optimal solutions in the objective space.

There are three goals for MOEAs in handling an MOP: (1) good convergence, obtaining a set of approximations as close as possible to the PF, (2) good diversity, obtaining a set of evenly distributed approximations, and (3) good coverage, which can cover the entire PF.

2.2. MOEA/D: an Overview. Multiobjective evolutionary algorithm based on decomposition (MOEA/D) is a representative of the decomposition-based method, proposed by Zhang and Li in 2007 [15]. Large sets of experiments have illustrated that MOEA/D and its improved versions show superiority over other popular MOEAs on solving MOPs with complicated Pareto set shapes [16, 36]. The basic idea behind MOEA/D is to transform an MOP into a series of single-objective optimization subproblems through decomposition method and coevolve these subproblems in each generation.

The framework of MOEA/D is formally described in Algorithm 1. Tchebycheff method is used for decomposing an MOP into N subproblems in this framework; g^{te} is the aggregated scalar function after decomposition. There have been other decomposition methods, such as weighted

```

input:  $N$ : population size;
          $T$ : the number of neighbors for each weight vector,  $0 < T < N$ ;
          $\lambda^1, \dots, \lambda^N$ : a set of  $N$  evenly distributed weight vectors;
output: Pareto solutions on the objective space:  $\{F(x^1), \dots, F(x^N)\}$ 
1 Initialization:
2  $x^1, \dots, x^N$  are randomly sampled from  $[l, u]$ ,  $FV^j = F(x^j)$ ;
3 foreach  $j = 1$  to  $N$  do  $neb(j) = \{a, b, \dots, t\}$ ;          /*  $\lambda^a, \lambda^b, \dots, \lambda^t$  are the  $T$  closest weight vectors to  $\lambda^j$  */
4 reference point  $z = (z_1, \dots, z_m)^T$ .                      /*  $z_i = \min_{1 \leq j \leq N} f_i(x^j)$  */
5 while stop criteria are not met do
6   for  $j = 1$  to  $N$  do
7      $\bar{y} \leftarrow$  reproduce  $(x^j, x^s, x^k)$ ;          /*  $s$  and  $k$  are selected from  $neb(j)$  */
8      $y \leftarrow$  mutate  $(\bar{y})$ ;
9     if  $y \notin [l, u]$  then
10       $y \leftarrow$  repair  $(y)$ ;
11    end
12    foreach  $i = 1$  to  $m$  do
13      if  $z_i > f_i(y)$  then
14         $z_i = f_i(y)$ ;
15      end
16    end
17    foreach  $i \in neb(j)$  do
18      if  $g^{te}(y | \lambda^i, z) < g^{te}(x^i | \lambda^i, z)$  then
19         $x^i = y$ ;
20         $FV^i = F(y)$ ;
21      end
22    end
23  end
24 end

```

ALGORITHM 1: MOEA/D.

sum (WS) and penalty-based boundary intersection (PBI) and can be used for decomposing. Detailed descriptions of these decomposition methods can refer to [15]. Just as the optimal solution of each subproblem has been proved to be Pareto optimal to the MOP under consideration, then the solutions set of all subproblems can be considered as a good approximation of PF.

Compared with other MOEAs, MOEA/D has the following three important features.

(1) MOEA/D transforms an MOP into a series of single-objective optimization subproblems through decomposition and solves these subproblems simultaneously; it does not directly solve the MOP as a whole. Different decomposition methods often have different effects on solving problems. Furthermore, many kinds of optimization strategies used in single-objective optimization algorithm can also be integrated into MOEA/D.

(2) MOEA/D implements the coevolution of subproblems. With the solutions information of adjacent subproblems, multiple subproblems can be optimized simultaneously. Then the computational complexity of MOEA/D is lower than that of NSGA-II.

(3) MOEA/D can solve the MOPs with complicated Pareto set shapes very well, which we often encounter in practical engineering optimization, synthesis of phase-only reconfigurable linear arrays for example [2]. In addition, MOEA/D can solve the problem with multiple objects

(especially when the number of objects m is greater than four). When the number of objects is large, the performance of MOEA tends to decline, which requires a larger population for optimization. However, the performance of MOEA/D does not significantly decrease.

2.3. Invasive Weed Optimization (IWO): an Overview. Weeds are plants whose vigorous and invasive habits of growth make them very robust and adaptive to changes in environment. Thus, capturing their properties and imitating their behaviors would lead to a powerful optimization method. This is the main idea behind IWO, which was originally proposed in [28]; it is a simple but effective meta-heuristic algorithm. To fulfill the IWO process, the following steps are needed.

Step 1 (initialization). A number of weeds are uniformly generated in the feasible decision space, where each weed represents a trial solution of the optimization problem under consideration.

Step 2 (fitness evaluation and ranking). Each weed will grow to a plant. Besides, fitness evaluation function will assign each plant a fitness and rank these plants based on their fitness values.

Step 3 (reproduction). Every plant produces seeds based on its rank or assigned fitness value. In other words, the number

of seeds each plant is permitted to produce, s_i , is decided by its fitness or rank, f_i , and the permissible maximum and minimum numbers of seeds s_{\max} and s_{\min} . s_i is formulated as

$$s_i = \text{floor} \left(\frac{f_i - f_{\min}}{f_{\max} - f_{\min}} (s_{\max} - s_{\min}) + s_{\min} \right) \quad (2)$$

where f_{\max} and f_{\min} are the highest and the lowest fitness of the population. Generally speaking, high fitness or rank will have the chance of producing more seeds. This step also provides an important property that allows all plants to participate in the reproduction contest; i.e., it gives all plants the chance of surviving and reproducing based on their rank or fitness.

Step 4 (spatial distribution). The produced seeds are designed to randomly distribute on the search space by Gaussian distribution with mean zero but varying variance. This step can ensure that the produced seeds are generated around their parent plant conducting a local search around each plant. However, the standard deviation of the random function is designed to decrease with iterations. At the current iteration ' $iter$ ', the standard deviation is described as

$$\sigma_{\text{iter}} = \sigma_{\text{final}} + \left(\frac{\text{iter}_{\max} - \text{iter}}{\text{iter}_{\max}} \right)^{\text{pow}} \cdot (\sigma_{\text{initial}} - \sigma_{\text{final}}) \quad (3)$$

where iter_{\max} is the upper limit of iterations and pow is a nonlinear regulatory factor. σ_{initial} and σ_{final} are presented as the initial and final standard deviations, respectively. It can be observed from (3) that the probability of dropping a seed in a remote area reduces nonlinearly with iterations leading to group fitter plants and elimination of inappropriate plants. Therefore, this step can be considered as the selection mechanism of IWO.

Step 5 (repeat and terminate). After the above steps carry out for all of the plants, the process will be repeated at Step 2 until stop conditions are met. It should be noted that weeds with lower fitness have a high probability of being eliminated after all plants reproduce to the maximum number pop_{\max} in colony process.

3. Multiobjective Invasive Weed Optimization Algorithm Based on Decomposition

In this part, we present a multiobjective invasive weed optimization algorithm based on decomposition, abbreviated as MOEA/D-IWO. We first adapt IWO for multiobjective optimization and then integrate it into MOEA/D providing a decomposition-based multiobjective optimization algorithm with invasive weed colonies.

The main aspects of our motivation are as follows: IWO is a population-based stochastic optimization technique in solving continuous optimization problems. In case of nonlinear multidimensional continuous optimization problems, IWO outperforms PSO, GA, memetic algorithms, and shuffled frog leaping [28]. However, in conventional IWO, fitness is used not only to compare two solutions but also in

the reproduction process unlike in PSO, GA, etc. Comparing to other EAs, the fitness assignment of each solution in IWO is more difficult in solving MOPs than that in single objective optimization. Kundu et al. developed multiobjective invasive weed optimization [33], where fuzzy dominance mechanism, instead of nondominated sorting, is carried out to sort the promising weeds in each iteration. However, with the number of objectives becoming large, selection pressure will reduce sharply and optimization process becomes ineffective. To avoid this difficulty, the framework of decomposition-based multiobjective algorithm [15] can be considered as a reliable candidate. The mentioned advantages and disadvantages of IWO motivate us to propose a new hybrid version of IWO with MOEA/D framework to solve MOPs.

3.1. Adaptive Modification of IWO. In IWO, only individuals with high fitness values are permitted to reproduce offsprings, and the number of offsprings is determined by the normalized fitness value. Therefore, IWO is able to avoid wasting time on searching the less feasible region in a constrained optimization problem. However, as a local search algorithm, IWO is sensitive to the initial values of the parameters and easily gets trapped into local optima.

An adaptive modification of IWO in this study is for the aim of acquiring the balance between effective exploration and efficient exploitation utilizing neighborhood information for multiobjective optimization. The original IWO leads to a coarse-grained local search because the offsprings have the same dispersal degree in all dimensions at a certain iteration. In detail, it can be clearly seen from (3) that σ_{iter} decreases with the increase of iterations; however, the value of σ_{iter} for each parent seed in one iteration is the same, which is not conducive to exploration and efficient exploitation. Furthermore, we plan to integrate IWO into MOEA/D for multiobjective optimization. MOEA/D decomposes an MOP into a big set of scalar subproblems and coevolves these subproblems through neighborhood relationship. In the process of coevolution, we plan to utilize IWO fulfilling optimization for each subproblem. However, the current best solution and its neighbors have obviously different fitness for each subproblem; then the same setting of σ_{iter} for them is not proper. In other words, σ_{iter} influences the distance between parents and their produced children weeds, though they are under the same iteration. Different parent should have its own σ_{iter} differing from those of other parent weeds. Thus, in this study we improve IWO and propose an adaptive standard deviation std_{iter} , where the value of σ_{iter} varies not only with the iteration but also with the rank of the individual's fitness in the subproblem, as described in

$$\text{std}_{\text{iter}} = \begin{cases} \left(1 + Q \frac{g^{\text{te}} - g_{\text{mean}}^{\text{te}}}{g_{\text{max}}^{\text{te}} - g_{\text{mean}}^{\text{te}}} \right) \cdot \sigma_{\text{iter}} & \text{if } g^{\text{te}} \geq g_{\text{mean}}^{\text{te}} \\ \left(1 - Q \frac{g_{\text{mean}}^{\text{te}} - g^{\text{te}}}{g_{\text{mean}}^{\text{te}} - g_{\text{min}}^{\text{te}}} \right) \cdot \sigma_{\text{iter}} & \text{otherwise} \end{cases} \quad (4)$$

where g^{te} is the aggregated scalar function value of the weed (Tchebycheff method is used for example), $g_{\text{min}}^{\text{te}}$, $g_{\text{max}}^{\text{te}}$,

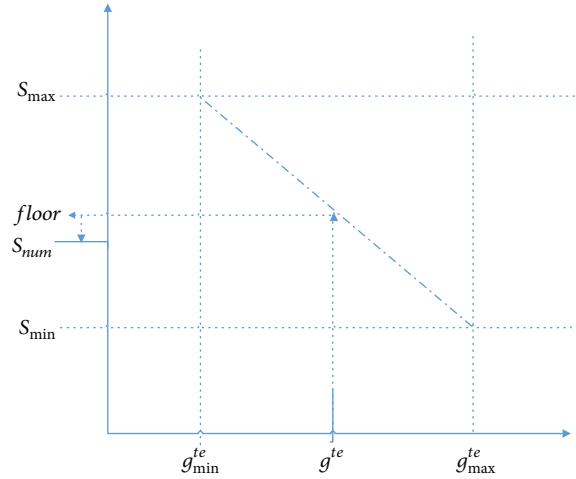


FIGURE 1: Seed production procedure in a colony of weeds.

and g_{mean}^{te} , respectively, represent the minimum, maximum, and average scalar function value among all weeds (current solution and its neighbors) in current iteration for each subproblem, and Q is a regulatory factor of adjusting the variation range of standard deviation; its value is generally set from 0 to 0.5.

It can be found from (4) that std_{iter} of weed consists with its scalar function value; the lower the scalar function value is, the smaller the standard deviation of the weed std_{iter} will be, which ensures the children seeds produced by better parents distribute relatively near around their parents, and the children seeds produced by worse parents distribute relatively far away from their parents. Moreover, the variable range of std_{iter} is extended to $[1 - Q, 1 + Q]\sigma_{\text{iter}}$ strengthening the diversity of the seeds, and the standard deviation of producing weeds decreases with iterations on the whole. This will accelerate the convergence rate and meanwhile can escape from local optimum. Global and local search capabilities can be well balanced through this mechanism.

On the other hand, the number of seeds produced by parent plant is described as

$$s_{\text{num}} = \text{floor} \left(\frac{g_{\text{max}}^{te} - g^{te}}{g_{\text{max}}^{te} - g_{\text{min}}^{te}} (s_{\text{max}} - s_{\text{min}}) + s_{\text{min}} \right) \quad (5)$$

where s_{max} and s_{min} in (5) are the largest and smallest number of seeds each parent is permitted to produce, respectively, $\text{floor}(\ast)$ means the floor function of \ast . It is very evident that better individual will produce more seeds. Figure 1 visually illustrates the procedure.

Let $x^i = (x_1^i, x_2^i, \dots, x_n^i)^\top$ be the i -th current parent individual, and each new seed produced by x^i is $y = (y_1, y_2, \dots, y_n)^\top$, where each element y_j is generated as follows:

$$y_j = x_j^i + N(0, std_{\text{iter}}^2), \quad j = 1, 2, \dots, n. \quad (6)$$

Finally, s_{num} new solutions of y are generated by (3)–(6) and applied for updating the i -th subproblem.

The dispersal degree of offsprings in adaptive IWO variant is determined by the estimation of the neighborhood information around their parents based on the neighborhood topology, which is more powerful in subproblem local search compared with the original IWO.

3.2. MOEA/D-IWO-Algorithm Description. From the previous two sections we conclude that MOEA/D provide a good framework for multiobjective optimization while adaptive IWO novelly offers good exploration and diversity. In this part, we combine the two algorithms and present a novel algorithm: MOEA/D-IWO for handling multiobjective optimization problems. Based on the smart and distinct features of IWO and MOEA/D, we propose MOEA/D-IWO and try to combine their excellent features in this extended hybrid algorithm.

Under the framework of MOEA/D, MOEA/D-IWO decomposes a multiobjective problem into a big set of scalar optimization subproblems and solves them simultaneously. In each subproblem, adaptive IWO is adopted for search, where the objective is to minimize the aggregation function of all the objects under consideration. Each subproblem has its own aggregation weight vector constructing its aggregation function, which is different from any of the others; i.e., all these aggregation weight vectors of the decomposed subproblems differ with each other. At each generation, the population is composed of the best solutions searched so far for each subproblem; then the number of the decomposed subproblems is also the population size. If the population size is set to N , then, we need to optimize these N subproblems simultaneously.

An MOP can be transferred into a series of scalar optimization subproblems through decomposition [34]. Tchebycheff decomposition approach is mainly employed in our experiments. Let $\lambda^1, \dots, \lambda^N$ be a set of uniformly distributed weight vectors, and $z^* = (z_1^*, \dots, z_m^*)^\top$, $z_i^* = \min\{f_i(x) \mid x \in [l, u]\}$, $i = 1, \dots, m$ is the reference point, with the Tchebycheff decomposition method; the objective function

```

Input:  $N$ : population size;
          $T$ : the number of neighbors for each weight vector,  $0 < T < N$ ;
          $\lambda^1, \dots, \lambda^N$ : a set of  $N$  evenly distributed weight vectors;
Output: Pareto solutions on the objective space:  $\{F(x^1), \dots, F(x^N)\}$ 
1 Initialization:
2  $x^1, \dots, x^N$  are randomly sampled from  $[l, u]$ ,  $FV^j = F(x^j)$ ;
3 foreach  $j = 1$  to  $N$  do  $neb(j) = \{a, b, \dots, t\}$ ;    /*  $\lambda^a, \lambda^b, \dots, \lambda^t$  are the  $T$  closest weight vectors to  $\lambda^j$  */
4 reference point  $z = (z_1, \dots, z_m)^T$ .                /*  $z_i = \min_{1 \leq j \leq N} f_i(x^j)$  */
5 while stop criteria are not met do
6 for  $j = 1$  to  $N$  do
7    $U \leftarrow$  IWO ( $x^j, std_{iter}^j$ );
8    $V \leftarrow$  IWO ( $x^k, std_{iter}^k$ );
9                                     /*  $k$  is selected from  $neb(j)$  */
10  foreach  $\bar{y} \in U \cup V$  do
11    if  $\bar{y} \notin [l, u]$  then
12       $y \leftarrow$  Repair ( $\bar{y}$ );
13    end
14    foreach  $i = 1$  to  $m$  do
15      if  $z_i > f_i(y)$  then
16         $z_i = f_i(y)$ ;
17      end
18    end
19    foreach  $i \in neb(j)$  do
20      if  $g^{te}(y | \lambda^i, z) < g^{te}(x^i | \lambda^i, z)$  then
21         $x^i = y$ ;
22         $FV^i = F(y)$ ;
23      end
24    end
25  end
26 end
27 end

```

ALGORITHM 2: MOEA/D-IWO.

of the i -th subproblem can be described as the following [34]:

$$g^{te}(x | \lambda^i, z^*) = \max_{1 \leq j \leq m} \{\lambda_j^i | f_j(x) - z_j^* | \} \quad (7)$$

where $\lambda^i = (\lambda_1^i, \dots, \lambda_m^i)^T$, m is the number of objects. MOEA/D-IWO optimizes all those N objective functions simultaneously. Each subproblem is optimized by adaptive IWO using information only from its neighbors. Neighborhood relations among subproblems here are defined based on the distance between their aggregation coefficient vectors. Detailed description of MOEA/D-IWO is provided in Algorithm 2.

In the line labeled 7 of the Algorithm 2, $U \leftarrow$ IWO (x^i, std_{iter}^i) describes the procedure of x^i producing seeds. U consists of all children seeds produced by x^i . Suppose $U = \{y^1, y^2, \dots, y^k\}$; then k is the total number of children seeds produced by x^i , its value is determined by (5), where g^{te} , i.e., $g^{te}(x^i | \lambda^i, z^*)$, and $g_{\min}^{te}, g_{\max}^{te}$ are obtained by the following equations, respectively:

$$g_{\min}^{te} = \min \{g^{te}(x^k | \lambda^i, z^*) | x^k \in B(i)\} \quad (8)$$

$$g_{\max}^{te} = \max \{g^{te}(x^k | \lambda^i, z^*) | x^k \in B(i)\} \quad (9)$$

std_{iter}^i stands for the adaptive standard deviation of x^i ; its value can be got through (3) and (4), where g_{mean}^{te} is calculated by

$$g_{\text{mean}}^{te} = \frac{\sum \{g^{te}(x^k | \lambda^i, z^*) | x^k \in B(i)\}}{|B(i)|} \quad (10)$$

where $|B(i)|$ is the number of neighbors for subproblem i .

Likewise, the same computing model is applied on the neighbors of x^i in the line (labeled 8) of Algorithm 2.

4. Experiments

For illustrating the performance of MOEA/D-IWO in handling MOPs, in this part MOEA/D-IWO is experimented on a big set of benchmark test instances. Firstly, MOEA/D-IWO is tested on nine problems with complex Pareto set shapes chosen from [16] and compared with other two classical algorithms: NSGA-II and MOEA/D on these problems. This set of nine complex functions was proposed by professors Zhang and Li [16]. Many experimental results have shown that this kind of complicated PSs as well as PFs could seriously affect the performance of MOEAs [23].

TABLE 1: IGD-metric comparison results of MOEA/D-IWO, NSGA-II, and MOEA/D on F1-F9.

Problems		MOEA/D-IWO	MOEA/D	ss	NSGA-II	ss
F1	mean	0.0014	0.0015	+	0.0031	+
	min	0.0014	0.0014		0.0029	
	std	1.0051E-05	3.4533E-05		1.0912E-04	
F2	mean	0.0396	0.0596	+	0.0437	+
	min	0.0107	0.0161		0.0341	
	std	0.0287	0.0462		0.0055	
F3	mean	0.0343	0.0492	+	0.0434	+
	min	0.0054	0.0073		0.0368	
	std	0.0221	0.0369		0.0037	
F4	mean	0.0023	0.0215	+	0.0431	+
	min	0.0021	0.0144		0.0357	
	std	1.4079E-04	0.0049		0.0042	
F5	mean	0.0246	0.0327	+	0.0416	+
	min	0.0025	0.0149		0.0356	
	std	0.0201	0.0217		0.0029	
F6	mean	0.0270	0.0330	+	0.0750	+
	min	0.0257	0.0303		0.0622	
	std	0.0011	0.0015		0.0061	
F7	mean	0.0569	0.2407	+	0.4302	+
	min	0.0025	0.1687		0.1942	
	std	0.0579	0.1703		0.0258	
F8	mean	0.1441	0.2569	+	0.4184	+
	min	0.0927	0.1479		0.3094	
	std	0.0414	0.0800		0.0397	
F9	mean	0.0322	0.0440	+	0.0446	+
	min	0.0127	0.0148		0.0346	
	std	0.0202	0.0291		0.0060	

Besides, MOEA/D-IWO is also tested on ten of CEC 2009 problems UF1-UF10 in this part for further comparing with other hybrid or outstanding algorithms including MOEA/D-DE [16], MOEA/D-PSO, dMOPSO [37], I-MOEA/D [17], and R2HMOPSO [20]. Among these ten problems UF1-UF10, the first seven UF1-UF7 are problems with two objectives while the last three UF8-UF10 are problems with three objectives. Each of the test problems UF1-UF10 has a decision space composed of 30 variables. Detailed descriptions of these ten test problems can be found in [23].

All those nineteen test problems are for minimization of the objectives.

4.1. Performance Metric. In multiobjective optimization, there are two basic aims that all the multiobjective algorithms pursue; i.e., the obtained solutions set must be as close as possible to the Pareto front, while the diversity of the solutions set needs to be maintained. In order to evaluate and compare the different algorithms quantitatively, we use the following performance metrics in experiments.

- (i) Inverted generational distance (IGD) [38]: suppose S^* is a large set of uniformly distributed points along the PF representing it well, and P is the solutions set obtained by multiobjective algorithm. $IGD(S^*, P)$

represents the average distance from S^* to P described as

$$IGD(S^*, P) = \frac{\sum_{u \in S^*} d(u, P)}{|S^*|} \quad (11)$$

where $d(u, P)$ is the minimum Euclidean distance between u and the points in P . If $|S^*|$ is large enough to represent the PF very well, $IGD(S^*, P)$ could measure both the diversity and convergence of P in a sense. To have a low value of $IGD(S^*, P)$, P must be very close to the PF and cannot miss any part of the whole PF.

- (ii) Spacing (S): [39] proposed the spacing metric which measures the variance of distance of each solution in P to its closest neighbour:

$$S(P) = \sqrt{\frac{1}{|P| - 1} \sum_{x \in P} (\bar{d} - d_x)^2}$$

$$d_x = \min_{\substack{x^* \in P \\ x^* \neq x}} \left\{ \sum_{l=1}^m |f_l(x) - f_l(x^*)| \right\} \quad (12)$$

$$\bar{d} = \frac{1}{|P|} \sum_{x \in P} d_x$$

TABLE 2: S-metric comparison results of MOEA/D-IWO, NSGA-II, and MOEA/D on F1-F9.

Problems		MOEA/D-IWO	MOEA/D	ss	NSGA-II	ss
F1	mean	0.0026	0.0035	+	0.0017	-
	min	0.0025	0.0033		0.0016	
	std	1.0811E-04	1.3405E-04		1.1012E-04	
F2	mean	0.0017	0.0124	+	0.0058	+
	min	9.5090E-04	0.0034		0.0043	
	std	4.8412E-04	0.0249		0.0021	
F3	mean	0.0029	0.0081	+	0.0083	+
	min	0.0024	0.0038		0.0037	
	std	0.0015	0.0061		0.0091	
F4	mean	0.0028	0.0324	+	0.0059	+
	min	0.0025	0.0056		0.0038	
	std	4.0495E-04	0.0210		0.0027	
F5	mean	0.0025	0.0064	+	0.0049	+
	min	0.0023	0.0037		0.0030	
	std	1.2618E-04	0.0048		0.0016	
F6	mean	0.0275	0.0401	+	0.0698	+
	min	0.0267	0.0323		0.0294	
	std	5.0778E-04	0.0082		0.0584	
F7	mean	0.0111	0.0320	+	0.0124	+
	min	0.0034	0.0031		0.0020	
	std	0.0057	0.0345		0.0069	
F8	mean	0.0239	0.0263	+	0.0345	+
	min	0.0052	5.7510E-04		7.1083E-04	
	std	0.0178	0.0256		0.0385	
F9	mean	0.0029	0.0085	+	0.0108	+
	min	0.0015	0.0030		0.0026	
	std	0.0010	0.0093		0.0128	

A lower variance is preferred as this indicates a better distribution of solutions in the Pareto set. The idea value is 0 as this indicates that the distances from one solution to its closest neighbour is the same for every solution in the Pareto set which means a uniform distribution of solutions in the Pareto set.

- (iii) Hypervolume (HV) [40]: the hypervolume metric measures the size of the region which is dominated by the solutions in P . Therefore a higher value of the HV-metric is preferred. Mathematically, the HV-metric is described as

$$HV(P) = vol\left(\bigcup_{v \in P} [f_1(v), z_1^r] \times \dots \times [f_m(v), z_m^r]\right) \quad (13)$$

where $vol(\cdot)$ is the Lebesgue measure, and $z^r = (z_1^r, \dots, z_m^r)^T$ is an antioptimal reference point in the objective space that is dominated by all Pareto-optimal objective vectors.

4.2. Parameter Setting. Experiments are implemented on a personal computer (Intel (R) Core (TM) i7-6700 CPU @3.40 GHz, 16 GB of RAM). Programming language is Visual C++ 6.0.

Parameters used in algorithms are set as follows.

(1) *Population Size and Number of Evaluations*

- (i) For F1 - F9, population size N is set to 300 and 595 for the problems with two objectives and three objectives, respectively, in all compared algorithms. The maximal number of generations is set to 250 for F1 - F9.
- (ii) For UF1 - UF10, population size N is set to 300 and 600 for the test instances with two objectives and three objectives, respectively. The total number of evaluations $FEAS = 300000$ for all UF1 - UF10.
- (iii) Each algorithm runs 30 times independently on each test instance F1 - F9 and UF1 - UF10. All those algorithms stop running after getting a given maximal number of function evaluations or generations.

(2) *Parameters in Reproduction Operators*

- (i) $F = 0.5$ and $CR = 1.0$ for DE operator;
- (ii) $\eta_c = 20$ and $p_c = 1$ for SBX crossover;
- (iii) $\eta_m = 20$ and $p_m = 1/n$ for mutation operator.

TABLE 3: HV-metric comparison results of MOEA/D-IWO, NSGA-II, and MOEA/D on F1-F9.

Problems		MOEA/D-IWO	MOEA/D	ss	NSGA-II	ss
F1	mean	3.3355	3.3276	+	3.2934	+
	min	3.335	3.3241		3.2883	
	std	4.06E-04	0.003		0.0039	
F2	mean	3.8218	3.4647	+	2.6804	+
	min	3.4908	3.0233		2.1773	
	std	0.4572	0.6302		0.1966	
F3	mean	3.7243	3.3438	+	3.0782	+
	min	3.4504	2.9318		2.9247	
	std	0.2889	0.3294		0.0937	
F4	mean	5.1184	3.8813	+	3.3103	+
	min	3.5836	3.4647		3.2994	
	std	0.8417	0.3691		0.0226	
F5	mean	3.7202	3.318	+	3.125	+
	min	3.4415	2.9702		2.9583	
	std	0.2503	0.3592		0.1128	
F6	mean	15.6539	13.597	+	8.8261	+
	min	10.9886	9.3974		8.8202	
	std	5.4213	3.6458		0.004	
F7	mean	5.0462	4.6764	+	3.5387	+
	min	2.8393	3.2214		3.0697	
	std	1.5641	1.9123		0.2888	
F8	mean	6.3312	4.2083	+	3.8939	+
	min	3.5082	2.6899		3.1358	
	std	2.0242	1.1118		0.8123	
F9	mean	4.1681	3.9585	+	3.6627	+
	min	3.7747	3.7037		3.3224	
	std	0.3954	0.3093		0.1091	

(3) Other Control Parameters in MOEA/D, dMOPSO, R2HMOPSO, and MOEA/D-IWO

- (i) Neighborhood size: $T = 10$ for F1 - F9 and $T = 20$ for UF1 - UF10;
- (ii) $\eta_r = 2$ for F1-F9, and $\eta_r = 0.01N$ for UF1-UF10;
- (iii) $T_a = 2$;
- (iv) consist with [20], $w_{start} = 0.9$, $w_{end} = 0.4$, $c_{1start} = 2.5$, $c_{1end} = 0.5$, $c_{2start} = 0.5$, $c_{2end} = 2.5$ in R2HMOPSO;
- (v) $\sigma_{ini} = 0.1$, $\sigma_{final} = 0.01$, $s_{max} = 3$, $s_{min} = 1$, $pow = 3$, $Q = 0.5$ in MOEA/D-IWO.

4.3. Experimental Analysis

4.3.1. MOEA/D-IWO Is Compared with NSGA-II [14] and MOEA/D [16] on F1 - F9. MOEA/D-IWO is compared with MOEA/D and NSGA-II in terms of performance metrics values. The statistical results of performance metrics obtained by MOEA/D-IWO and the other two algorithms are summarized in Tables 1–3. The three statistical results are based on 30 independent runs for each test problem, including the mean, the minimum, and the standard deviation (std) of the performance metrics values. The best performance on the

same test problem is highlighted by bold font. Besides, in the fifth and seventh columns of Tables 1–3, the statistical significance (ss) of the advantage of MOEA/D-IWO in the mean IGD-metric, S-metric, and HV-metric value is reported. +/=-/, respectively, represents that MOEA/D-IWO is statistically superior to, equal to, and inferior to MOEA/D and NSGA-II in terms of mean performance metric value.

As listed in Table 1, for almost all the test problems, the mean and the best IGD-metric values obtained by MOEA/D-IWO are smaller than those obtained by MOEA/D and NSGA-II, respectively, which demonstrates that MOEA/D-IWO behaves better than MOEA/D and NSGA-II in pursuing PF on both the convergence and diversity.

The spacing-metric numerically describes the spread of the solutions on the objective space. Table 2 clearly shows that the spacing-metric values obtained by MOEA/D-IWO are smaller than other two algorithms for almost all the test problems, which indicates that the solutions obtained by MOEA/D-IWO are spaced more evenly than those obtained by MOEA/D and NSGA-II in general.

The HV-metric measures the size of the region which is dominated by the obtained Pareto front, i.e., the region of coverage of the obtained Pareto front. Therefore the higher value of the HV-metric is preferred. As described in Table 3,

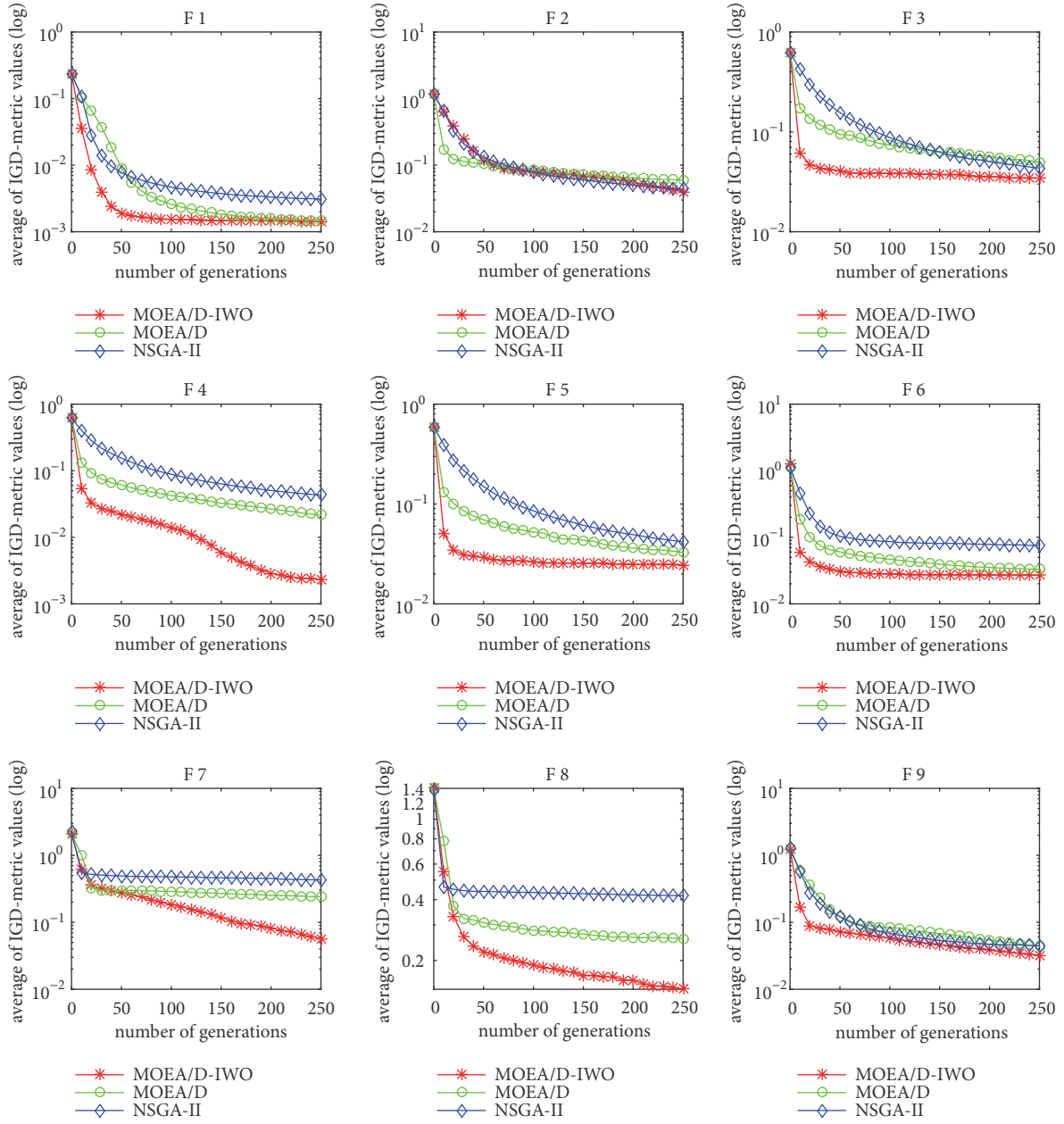


FIGURE 2: Convergence curves of IGD metric values on F1 - F9.

for almost all the test problems MOEA/D-IWO has better performance than MOEA/D and NSGA-II in terms of HV-metric.

To verify the convergence trend of the proposed algorithm, convergence graphs of the three algorithms on F1 - F9 are shown in Figure 2 plotting the evolution of the average IGD-metric values. It can be clearly seen from Figure 2 that MOEA/D-IWO converges much faster than NSGA-II and MOEA/D in minimizing the IGD-metric values for almost all the problems, which indicates that in most cases the adaptive IWO is effective in accelerating the convergence, and the proposed hybrid MOEA/D-IWO is feasible in improving the accuracy of the Pareto solutions.

Figures 3–5 plot the distribution of the final population in the objective space obtained by three algorithms on F1 - F9. It can be observed from these three figures that MOEA/D-IWO can obtain good approximations to F1, F3, F4 - F6. However, it fails within the given number of generations, to approximate the PFs of the problems F8 and F9 satisfactorily, perhaps for the reason that incorporating IWO (using the random Gaussian reproduction mechanism for optimization) into an MOEA would, in some sense, spoil the diversity of the algorithm, and the current mechanism is not good enough for well solving the concave or problems with many local Pareto solutions. However, as evidenced from Tables 1–3 and Figures 2–5, in general MOEA/D-IWO performs well and preferably.

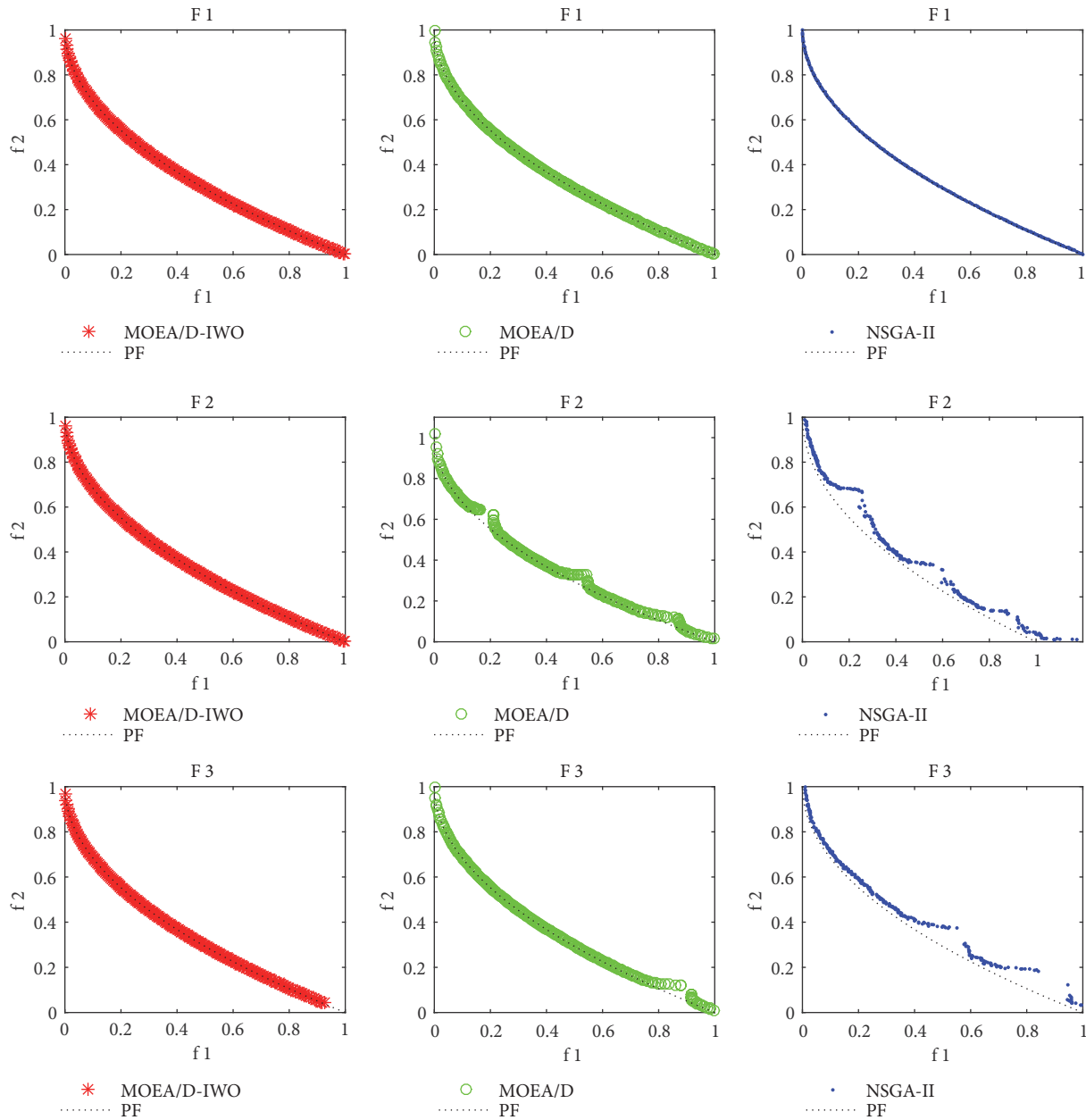


FIGURE 3: Pareto fronts obtained by MOEA/D-IWO, MOEA/D, and NSGA-II on F1 - F3.

4.3.2. MOEA/D-IWO Is Compared with Some Other MOEAs on UF1-UF10. In order to see whether MOEA/D-IWO prevails superiority over other state-of-the-art algorithms or other hybrid MOEA/D algorithms, a comprehensive comparison is executed on ten of the CEC 2009 problems UF1 - UF10. Compared algorithms consist of six algorithms including MOEA/D-IWO (MOEA/D hybrids with IWO), MOEA/D-DE (MOEA/D hybrids with DE), MOEA/D-PSO (MOEA/D hybrids with PSO), I-MOEA/D, dMOPSO, and R2HMOPSO. Experimental results of I-MOEA/D, dMOPSO, and R2HMOPSO are taken from the literatures [17, 20]. MOEA/D-DE and MOEA/D-PSO are run on PlatEMO V1.1 [41]. For fair comparisons, we run MOEA/D-IWO using the same set of parameters and stop conditions.

Table 4 provides the mean and the standard deviation (std) of the IGD performance metrics of all compared algorithms, where the best performance on each problem is highlighted by bold font. In order to validate the statistical significance of the advantages of MOEA/D-IWO over other algorithms, *t*-test is carried out on the obtained IGD performance metric values and the results are shown in the rows labeled 'ss'. +/- shows that MOEA/D-IWO is superior to, similar to, or inferior to the compared algorithm, respectively. Total comparing results are summed up in the last row.

From Table 4 we can observe that MOEA/D-IWO performs the best in all those six multiobjective algorithms. Among those ten test problems UF1-UF10, MOEA/D-IWO behaves better than MOEA/D-PSO on all of them;

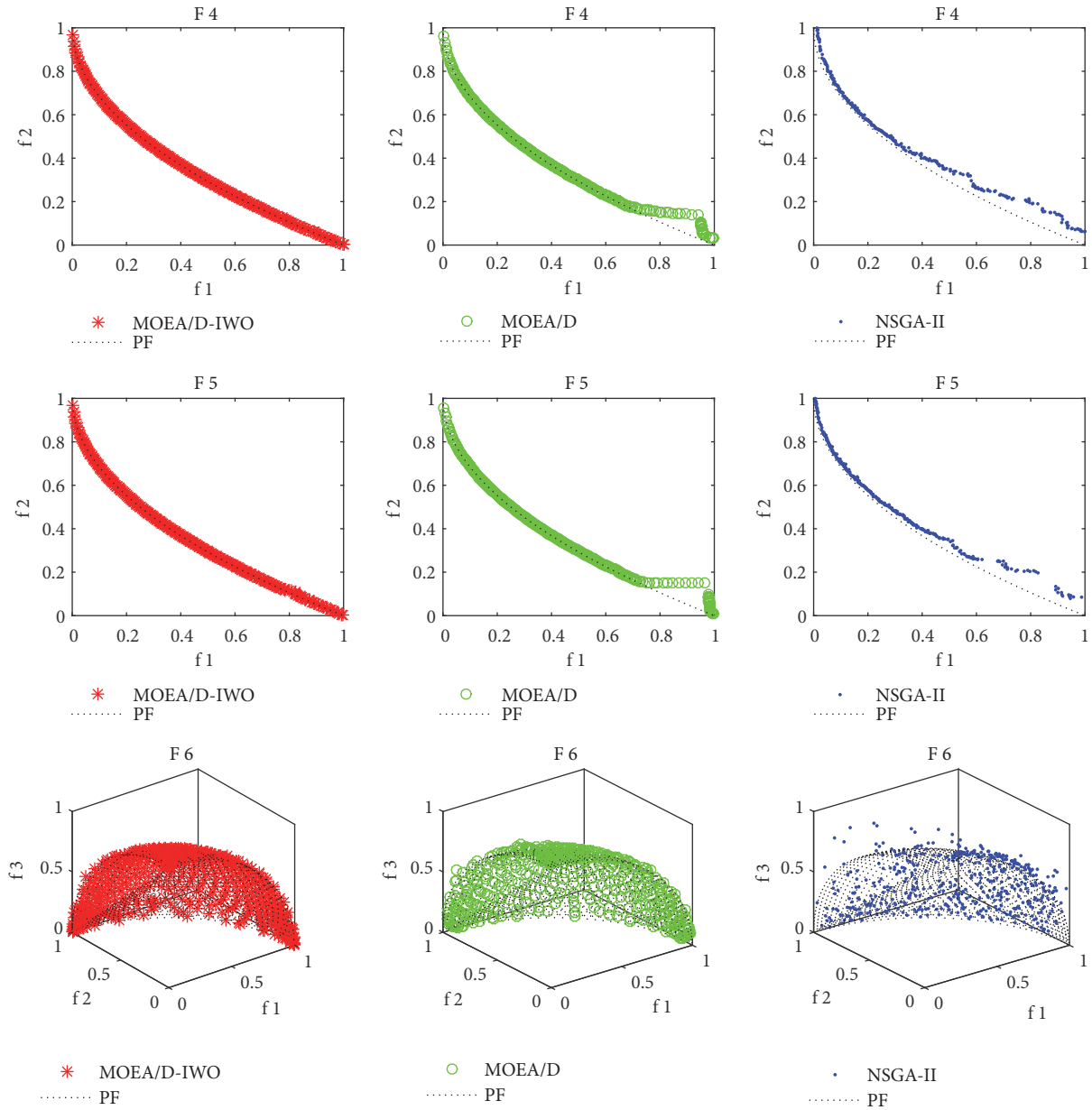


FIGURE 4: Pareto fronts obtained by MOEA/D-IWO, MOEA/D, and NSGA-II on F4 - F6.

MOEA/D-IWO behaves better than dMOPSO on nine problems; MOEA/D-IWO behaves better than MOEA/D-DE and I-MOEA/D on eight problems and better than R2HMOPSO on seven problems. All these results demonstrate the preferable performance of MOEA/D-IWO.

Figure 6 visually plots the approximate Pareto fronts of UF1 - UF10 searched by MOEA/D-IWO. It can be seen from the figure that MOEA/D-IWO could find good approximations to UF1, UF2, UF3, and UF7. Its approximations to UF6, UF8, and UF9 are acceptable. However, it fails to approximate satisfactory PFs of UF4, UF5, and UF10 under the present given stop conditions. For well solving these problems with discontinuous or concave PFs, MOEA/D-IWO needs to be further improved. Nevertheless, as evidenced from Tables

1–4, the hybrid of MOEA/D and IWO can generally improve the performance of MOEA/D. Compared with other hybrid MOEA/D algorithms, MOEA/D-IWO is efficient and competitive.

In a word, through comparing with other popular MOEAs we validate the great potential of MOEA/D-IWO in dealing with this kind of complicated multiobjective problems.

4.3.3. Additional Experimental Discuss. MOEA/D-IWO decomposes an MOP into a big set of scalar subproblems and coevolves these subproblems simultaneously. In the process of coevolution, an adaptive IWO is proposed and utilized for each subproblem. As described in Section 3.1,

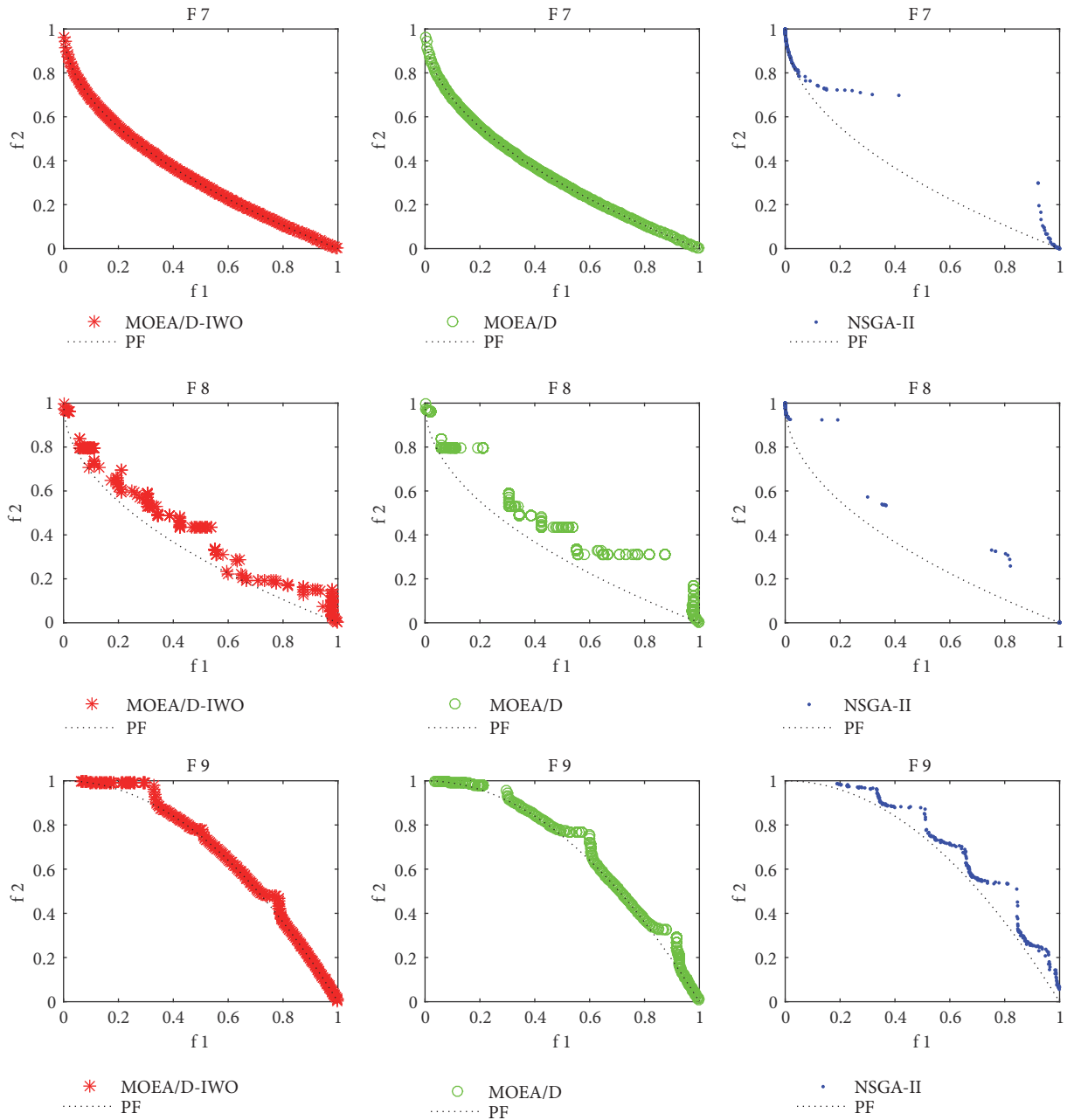


FIGURE 5: Pareto fronts obtained by MOEA/D-IWO, MOEA/D, and NSGA-II on F7 - F9.

Q is an important regulatory factor in adaptive IWO for balancing effective exploration and efficient exploitation. Trial experiments observed that its value is properly set from 0 to 0.5. To investigate the impact of Q on the performance of MOEA/D-IWO, different settings of Q have been tested in this part.

F7 and F8 are two complicated MOPs with many local Pareto solutions; MOEA/D-IWO performs not well on them in experiments. We choose the two problems as examples to test the impact of Q . Different settings of Q in the implementation of MOEA/D-IWO for UF7 and UF8 have been tested. All the other parameters settings are the same as in Section 4.2 except the setting of Q . For each setting

of Q , MOEA/D-IWO runs 30 times independently. Figures 7(a) and 7(b) box plot the IGD-metric values of the obtained solutions for UF7 and UF8 based on those 30 independent runs, while Figure 7(c) depicts the variation trend of the mean IGD-metric values under different Q . As clearly shown in Figure 7, MOEA/D-IWO performs relatively stable with Q from 0 to 0.5 on the two problems, and it will deteriorate when Q is greater than 0.5 on UF7. It is evident that MOEA/D-IWO is not very sensitive to the setting of Q under the range considered $[0, 0.5]$. When Q is relatively large, the reason for the poor performance of MOEA/D-IWO may be that the search diversity of the algorithm is enhanced but the exploration ability is weakened.

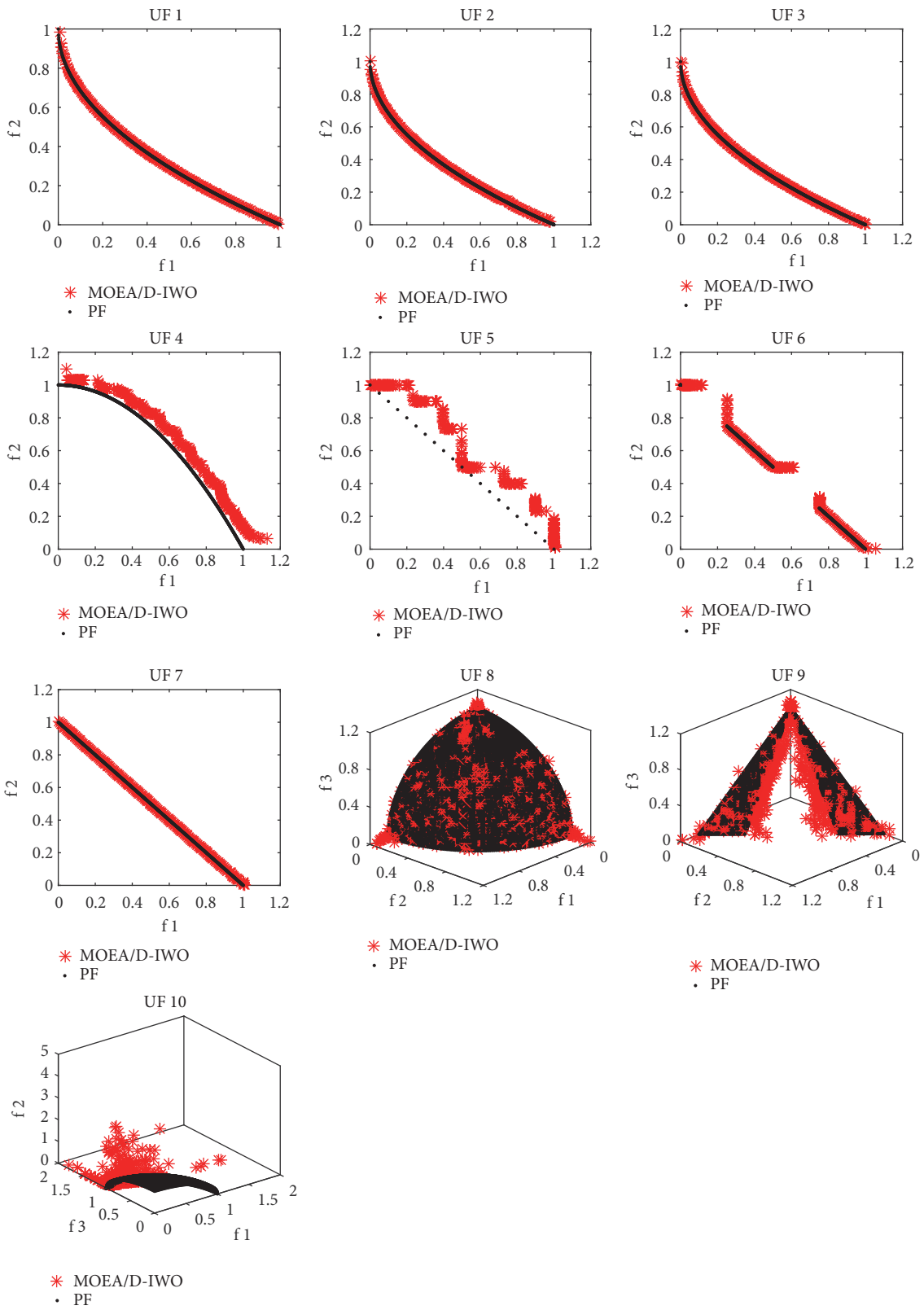


FIGURE 6: Pareto fronts obtained by MOEA/D-IWO on UF1 - UF10.

TABLE 4: Statistics of IGD performance metric on UF1 - UF10.

Problems	MOEA/D-IWO	MOEA/D-DE[16]	MOEA/D-PSO	dMOPSO[37]	I-MOEA/D[17]	R2HMOPSO[20]
UF1	mean	0.0019	0.0588	0.0916	0.0057	0.0608
	std	0.0001	0.0944	0.0115	0.0021	0.0177
	ss	+	+	+	+	+
UF2	mean	0.0076	0.1125	0.0473	0.0098	0.0184
	std	0.0020	0.0536	0.0081	0.0035	0.0071
	ss	=	+	+	+	+
UF3	mean	0.0047	0.0050	0.7075	0.4614	0.6203
	std	0.0032	0.0044	0.2440	0.0591	0.1766
	ss	+	+	+	+	+
UF4	mean	0.0591	0.0625	0.1103	0.0921	0.0376
	std	0.0053	0.0058	0.0056	0.0014	0.0026
	ss	+	+	+	-	-
UF5	mean	0.2074	0.2500	0.3894	0.7095	0.1748
	std	0.0769	0.0519	0.1929	0.1731	0.0273
	ss	+	+	+	+	-
UF6	mean	0.0113	0.1049	0.2951	0.4394	0.1748
	std	0.0066	0.0519	0.0671	0.2098	0.0645
	ss	+	+	+	+	+
UF7	mean	0.0058	0.0026	0.1745	0.1372	0.0314
	std	0.0040	0.0004	0.0843	0.1296	0.0076
	ss	-	-	+	+	+
UF8	mean	0.0671	0.0875	0.2322	0.1968	0.4144
	std	0.0123	0.0140	0.0391	0.0314	0.0522
	ss	+	+	+	+	+
UF9	mean	0.1244	0.1515	0.4861	0.1142	0.2362
	std	0.0453	0.0667	0.1744	0.0288	0.0591
	ss	+	+	+	0.0179	+
UF10	mean	0.4196	0.5545	0.6882	0.4196	0.2384
	std	0.0611	0.0854	0.2435	0.2376	0.0569
	ss	+	+	+	=	-
+/-/-	8/1/1	8/1/1	10/0/0	9/0/1	8/1/1	7/0/3

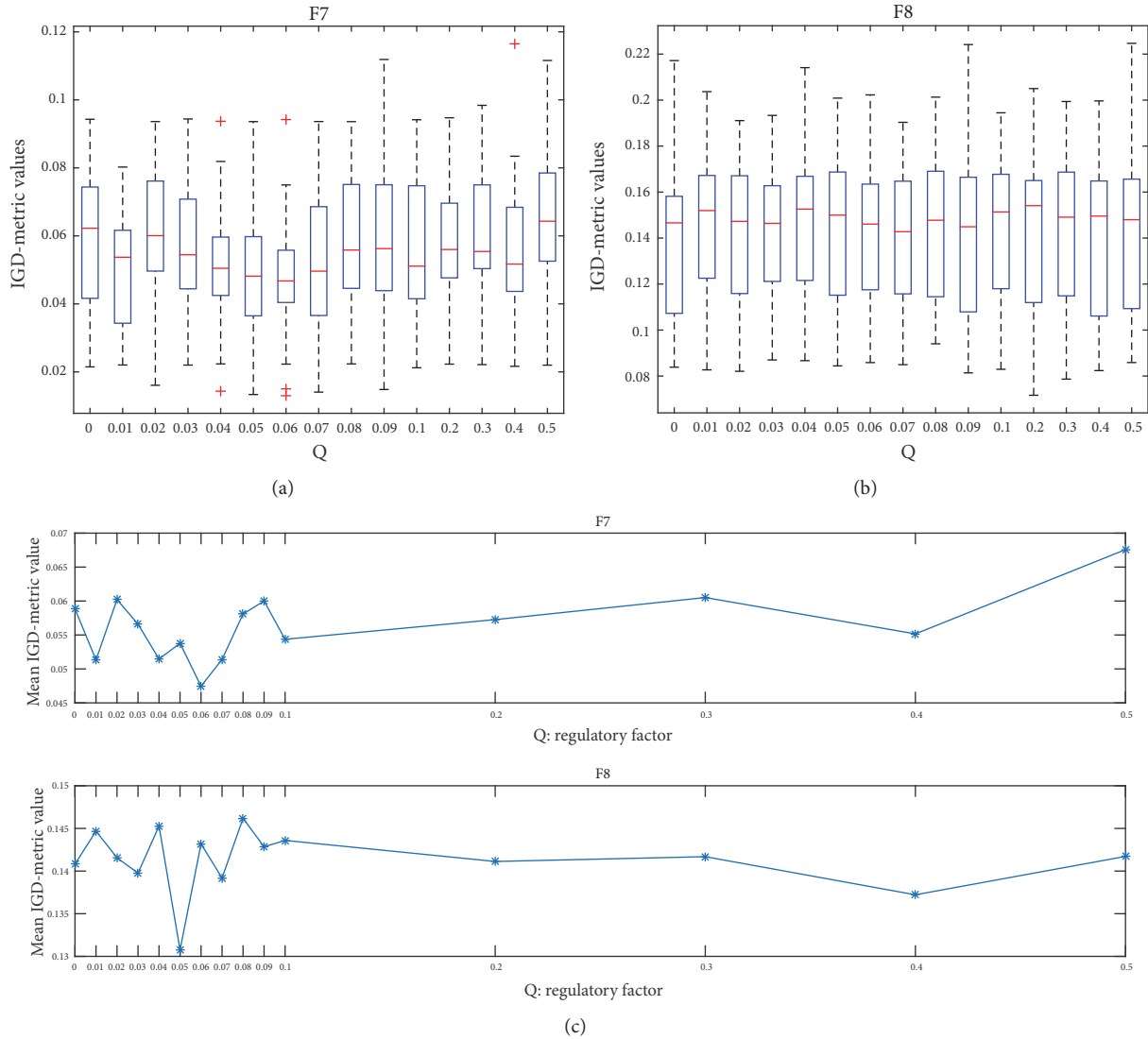


FIGURE 7: Performance evaluation with different Q values.

5. Conclusion

IWO is a smart algorithm mimicking the ecological behavior of colonizing weeds and distribution and is able to efficiently handle general linear, nonlinear, and multidimensional optimization problems. Since its proposal, IWO has been successfully applied in many practical optimization problems. However, to our knowledge, there has been little research on IWO for multiobjective optimization. For well solving the complex multiobjective problems, in this work, we broaden the use of classical IWO and integrate it into the frame of MOEA/D for handling MOPs. Based on the smart and distinct features of IWO and MOEA/D, we introduce MOEA/D-IWO and try to combine their excellent features in this extended hybrid algorithm. MOEA/D-IWO decomposes an MOP into a big set of single-objective subproblems and handles them simultaneously in each generation. The

population consists of the best solutions found so far for each subproblem, and each subproblem adopts an adaptive IWO for evolution in each generation. The performance of the proposed MOEA/D-IWO in both convergence speed and optimality of results are compared with those of NSGA-II, MOEA/D, and some other MOEAs on a big set of MOPs. Comparison results indicate the feasible and competitive performance of MOEA/D-IWO in the field of multiobjective optimization.

Actually, MOEA/D-IWO is still faced with some challenges in solving the MOPs with discontinuous or concave PFs. Strengthening the performance of the algorithm remains to be studied further. MOEA/D-IWO may be improved by using better and newer variants of IWO in future. Meanwhile, we also intend to study the ability of MOEA/D-IWO in solving high-dimensional multiobjective optimization problems in the future.

Data Availability

The DAT data used to support the findings of this study are available from the corresponding author upon request.

Conflicts of Interest

The authors declare no conflict of interest.

Acknowledgments

The work is partially supported by the National Natural Science Foundation of China (No. 61401260, 61602283, 61702310, 61572298) and the Social Science Planning Fund Program, Shandong Province (No. 18BXWJ03).

References

- [1] C. A. Coello, D. A. Van Veldhuizen, and G. B. Lamont, *Evolutionary Algorithms for Solving Multi-objective Problems*, vol. 242, Kluwer Academic, New York, NY, USA, 2002.
- [2] Y. Liu, Y.-C. Jiao, Y.-M. Zhang et al., "Synthesis of phase-only reconfigurable linear arrays using multiobjective invasive weed optimization based on decomposition," *International Journal of Antennas and Propagation*, vol. 2014, Article ID 630529, 11 pages, 2014.
- [3] H. Sang, P. Duan, and J. Li, "An effective invasive weed optimization algorithm for scheduling semiconductor final testing problem," *Swarm and Evolutionary Computation*, vol. 38, pp. 42–53, 2018.
- [4] K. Gao, F. Yang, M. Zhou, Q. Pan, and P. N. Suganthan, "Flexible job-shop rescheduling for new job insertion by using discrete jaya algorithm," *IEEE Transactions on Cybernetics*, vol. 49, no. 5, pp. 1944–1955, 2019.
- [5] K. Z. Gao, P. N. Suganthan, Q. K. Pan, T. J. Chua, T. X. Cai, and C. S. Chong, "Pareto-based grouping discrete harmony search algorithm for multi-objective flexible job shop scheduling," *Information Sciences*, vol. 289, pp. 76–90, 2014.
- [6] H. Liu, B. Xu, D. Lu, and G. Zhang, "A path planning approach for crowd evacuation in buildings based on improved artificial bee colony algorithm," *Applied Soft Computing*, vol. 68, pp. 360–376, 2018.
- [7] D. Shi, L. Zhu, Z. Cheng, Z. Li, and H. Zhang, "Unsupervised multi-view feature extraction with dynamic graph learning," *Journal of Visual Communication and Image Representation*, vol. 56, pp. 256–264, 2018.
- [8] X. W. Zheng, B. Hu, D. J. Lu, and H. Liu, "A multi-objective virtual network embedding algorithm in cloud computing," *Journal of Internet Technology*, vol. 17, no. 4, pp. 633–642, 2016.
- [9] A. Trivedi, D. Srinivasan, K. Sanyal, and A. Ghosh, "A survey of multiobjective evolutionary algorithms based on decomposition," *IEEE Transactions on Evolutionary Computation*, vol. 21, no. 3, pp. 440–462, 2017.
- [10] X. Zheng and H. Liu, "A scalable coevolutionary multi-objective particle swarm optimizer," *International Journal of Computational Intelligence Systems*, vol. 3, no. 5, pp. 590–600, 2010.
- [11] X.-W. Zheng, X.-M. Yu, Y. Li, and H. Liu, "An enhanced multi-objective group search optimizer based on multi-producer and crossover operator," *Journal of Information Science and Engineering*, vol. 33, no. 1, pp. 37–50, 2017.
- [12] K. Li, Q. Zhang, S. Kwong, M. Li, and R. Wang, "Stable matching-based selection in evolutionary multiobjective optimization," *IEEE Transactions on Evolutionary Computation*, vol. 18, no. 6, pp. 909–923, 2014.
- [13] E. Zitzler, M. Laumanns, and L. Thiele, "SPEA2: improving the strength pareto evolutionary algorithm for multiobjective optimization," *Evolutionary Methods for Design Optimization and Control with Applications to Industrial Problems*, pp. 95–100, 2001.
- [14] K. Deb, A. Pratap, S. Agarwal, and T. Meyarivan, "A fast and elitist multiobjective genetic algorithm: NSGA-II," *IEEE Transactions on Evolutionary Computation*, vol. 6, no. 2, pp. 182–197, 2002.
- [15] Q. Zhang and H. Li, "MOEA/D: a multiobjective evolutionary algorithm based on decomposition," *IEEE Transactions on Evolutionary Computation*, vol. 11, no. 6, pp. 712–731, 2007.
- [16] H. Li and Q. Zhang, "Multiobjective optimization problems with complicated pareto sets, MOEA/D and NSGA-II," *IEEE Transactions on Evolutionary Computation*, vol. 13, no. 2, pp. 284–302, 2009.
- [17] W. Zheng, Y. Tan, L. Meng et al., "An improved MOEA/D design for many-objective optimization problems," *Applied Intelligence*, vol. 48, no. 10, pp. 3839–3861, 2018.
- [18] D. H. Phan and J. Suzuki, "R2-IBEA: R2 indicator based evolutionary algorithm for multiobjective optimization," in *Proceedings of the IEEE Congress on Evolutionary Computation (CEC '13)*, pp. 1836–1845, June 2013.
- [19] J. Bader and E. Zitzler, "HypE: an algorithm for fast hypervolume-based many-objective optimization," *Evolutionary Computation*, vol. 19, no. 1, pp. 45–76, 2011.
- [20] L. Wei, X. Li, R. Fan, H. Sun, and Z. Hu, "A hybrid multiobjective particle swarm optimization algorithm based on R2 indicator," *IEEE Access*, vol. 6, pp. 14710–14721, 2018.
- [21] H. Ishibuchi and T. Murata, "A multi-objective genetic local search algorithm and its application to flowshop scheduling," *IEEE Transactions on Systems, Man, and Cybernetics, Part C: Applications and Reviews*, vol. 28, no. 3, pp. 392–403, 1998.
- [22] T. Murata and M. Gen, "Cellular genetic algorithm for multi-objective optimization," in *Proceedings of the Fourth Asian Fuzzy System Symposium*, pp. 538–542, 2000.
- [23] Q. Zhang, W. Liu, and H. Li, "The performance of a new version of MOEA/D on CEC09 unconstrained MOP test instances," in *Proceedings of the IEEE Congress on Evolutionary Computation (CEC '09)*, pp. 203–208, May 2009.
- [24] S.-Z. Zhao, P. N. Suganthan, and Q. Zhang, "Decomposition-based multiobjective evolutionary algorithm with an ensemble of neighborhood sizes," *IEEE Transactions on Evolutionary Computation*, vol. 16, no. 3, pp. 442–446, 2012.
- [25] H.-L. Liu, F. Gu, and Q. Zhang, "Decomposition of a multi-objective optimization problem into a number of simple multi-objective subproblems," *IEEE Transactions on Evolutionary Computation*, vol. 18, no. 3, pp. 450–455, 2014.
- [26] Z. Wang, Q. Zhang, A. Zhou, M. Gong, and L. Jiao, "Adaptive replacement strategies for MOEA/D," *IEEE Transactions on Cybernetics*, vol. 46, no. 2, pp. 474–486, 2016.
- [27] X. Zheng, Y. Li, H. Liu, and H. Duan, "A study on a cooperative character modeling based on an improved NSGA II," *Multimedia Tools and Applications*, vol. 75, no. 8, pp. 4305–4320, 2016.
- [28] A. R. Mehrabian and C. Lucas, "A novel numerical optimization algorithm inspired from weed colonization," *Ecological Informatics*, vol. 1, no. 4, pp. 355–366, 2006.

- [29] H. Sang, Q. Pan, J. Li et al., "Effective invasive weed optimization algorithms for distributed assembly permutation flowshop problem with total flowtime criterion," *Swarm and Evolutionary Computation*, vol. 44, pp. 64–73, 2019.
- [30] H. S. Rad and C. Lucas, "A recommender system based on invasive weed optimization algorithm," in *Proceedings of the 2007 IEEE Congress on Evolutionary Computation, CEC 2007*, pp. 4297–4304, Singapore, September 2007.
- [31] A. R. Mallahzadeh, H. Oraizi, and Z. Davoodi-Rad, "Application of the invasive weed optimization technique for antenna configurations," *Progress in Electromagnetics Research*, vol. 79, pp. 137–150, 2008.
- [32] X. Zhang, Y. Wang, G. Cui, Y. Niu, and J. Xu, "Application of a novel IWO to the design of encoding sequences for DNA computing," *Computers & Mathematics with Applications*, vol. 57, no. 11-12, pp. 2001–2008, 2009.
- [33] D. Kundu, K. Suresh, S. Ghosh, S. Das, and B. K. Panigrahi, "Multi-objective optimization with artificial weed colonies," *Information Sciences*, vol. 181, no. 12, pp. 2441–2454, 2011.
- [34] K. Miettinen, *Nonlinear Multiobjective Optimization*, Kluwer Academic Publishers, Norwell, Mass, USA, 1999.
- [35] K. Deb, *Multiobjective Optimization Using Evolutionary Algorithms*, John Wiley, 2001.
- [36] Y. T. Qi, X. L. Ma, F. Liu, L. C. Jiao, J. Y. Sun, and J. S. Wu, "MOEA/D with adaptive weight adjustment," *Evolutionary Computation*, vol. 22, no. 2, pp. 231–264, 2014.
- [37] C. A. C. Coello and S. W. Martinez, "A multi-objective particle swarm optimizer based on decomposition," in *Proceedings of the annual conference on Genetic and evolutionary computation*, pp. 69–76, 2011.
- [38] E. Zitzler, L. Thiele, M. Laumanns, C. M. Fonseca, and V. G. Da Fonseca, "Performance assessment of multiobjective optimizers: an analysis and review," *IEEE Transactions on Evolutionary Computation*, vol. 7, no. 2, pp. 117–132, 2003.
- [39] J. R. Schott, *Fault tolerant design using single and multicriteria genetic algorithm optimization [MS. Thesis]*, Massachusetts Institute of Technology, 1995.
- [40] E. Zitzler and L. Thiele, "Multiobjective evolutionary algorithms: a comparative case study and the strength pareto approach," *IEEE Transactions on Evolutionary Computation*, vol. 3, no. 4, pp. 257–271, 1999.
- [41] Y. Tian, R. Cheng, X. Zhang, and Y. Jin, "PlatEMO: a matlab platform for evolutionary multi-objective optimization," *IEEE Computational Intelligence Magazine*, vol. 12, no. 4, pp. 73–87, 2017.

