

Article

An Improved MOEA/D with Optimal DE Schemes for Many-Objective Optimization Problems

Wei Zheng ¹ , Yanyan Tan ^{1,2,*}, Xiaonan Fang ^{1,2} and Shengtao Li ^{1,2}

¹ School of Information Science and Engineering, Shandong Normal University, Jinan 250358, China; zhengshu56@126.com (W.Z.); franknan@126.com (X.F.); shengtaoli1985@163.com (S.L.)

² Institute of Data Science and Technology, Shandong Normal University, Jinan 250358, China

* Correspondence: yytan928@sdsu.edu.cn; Tel.: +86-0531-8961-0750

Received: 13 June 2017; Accepted: 24 July 2017; Published: 26 July 2017

Abstract: MOEA/D is a promising multi-objective evolutionary algorithm based on decomposition, and it has been used to solve many multi-objective optimization problems very well. However, there is a class of multi-objective problems, called many-objective optimization problems, but the original MOEA/D cannot solve them well. In this paper, an improved MOEA/D with optimal differential evolution (oDE) schemes is proposed, called MOEA/D-oDE, aiming to solve many-objective optimization problems. Compared with MOEA/D, MOEA/D-oDE has two distinguishing points. On the one hand, MOEA/D-oDE adopts a newly-introduced decomposition approach to decompose the many-objective optimization problems, which combines the advantages of the weighted sum approach and the Tchebycheff approach. On the other hand, a kind of combination mechanism for DE operators is designed for finding the best child solution so as to do the a posteriori computing. In our experimental study, six continuous test instances with 4–6 objectives comparing NSGA-II (nondominated sorting genetic algorithm II) and MOEA/D as accompanying experiments are applied. Additionally, the final results indicate that MOEA/D-oDE outperforms NSGA-II and MOEA/D in almost all cases, particularly in those problems that have complicated Pareto shapes and higher dimensional objectives, where its advantages are more obvious.

Keywords: many-objective optimization; multi-objective evolutionary optimization based on decomposition (MOEA/D); differential evolutionary schemes

1. Introduction

In daily life and work, multi-objective optimization problems (MOPs) have a wide range of applications, such as water distribution systems [1], land use management problems [2], automotive engine calibration problems [3], and so on, which are all real-world applications of MOPs. All of these problems are needed to optimize two or more than two objectives simultaneously. Without a doubt, everyone wants to make all objectives achieve optimization. However, we cannot find a solution to achieve this goal, because all of the objectives are restricting each other. If one objective achieves the optimization, the other objectives must achieve more useless solutions, which the decision makers do not want to see. Due to the importance of MOPs, scholars have been trying to solve this effectively. In order to facilitate the study of MOPs, scholars summed up the general mathematical expression of the MOPs. Therefore, a multi-objective optimization problem (MOP) can be mathematically defined as follows:

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

where $F : \Omega \rightarrow \mathbb{R}^m$ is made up of m real-valued objective functions, Ω is the decision (variable) space and \mathbb{R}^m is called the objective space.

Given two objective vectors: $u, v \in \mathfrak{R}^m$, vector u is said to dominate v , which means if and only if $u_i \leq v_i$ for every $i \in \{1, \dots, m\}$ and there must exist at least one $j \in \{1, \dots, m\}$ satisfied with $u_j < v_j$. Additionally, if $F(x)$ dominates $F(y)$, a solution $x \in \Omega$ is said to dominate another solution $y \in \Omega$. A solution $x^* \in \Omega$ is defined as a Pareto-optimal solution in the case that $F(x^*)$ is not dominated by any other solutions in the objective space. Then, $F(x^*)$ is called a Pareto-optimal objective vector. The set of all the Pareto-optimal solutions is known as the Pareto set (PS). In addition, the set of all of the Pareto-optimal objective vectors is the Pareto front (PF) [4,5]. Additionally, any improvement of one objective in the Pareto-optimal solutions is bound to the decline of at least one other objective.

In most real-world applications, because the objectives in (1) conflict with each other, there is no solution that could minimize all of the objectives simultaneously. Because of the complexity of MOPs, the traditional mathematical methods will meet different levels of difficulty while solving the MOPs. Nevertheless, the evolutionary multi-objective optimization (EMO) algorithm based on natural evolution can solve the MOPs exceedingly well to some extent. Over the past twenty years, EMO algorithms and their applications came into being and developed rapidly, which has attracted a large number of researchers to study them [6–13].

Owing to the research of EMO, the evolutionary algorithm (EA) has some certain advantages in obtaining the multiple Pareto-optimal solutions; for example, it always can obtain more than one optimal solution in a single run. Therefore, it is very common to use multi-objective evolutionary algorithms (MOEAs) to solve MOPs. Many EMO algorithms did the fitness assignment through using Pareto dominance, which are called the domination-based algorithms. In these algorithms, they tried to find a set of solutions, which should be as approximate as possible to the true PF, and this is known as convergence. The allocation of the fitness scheme to solutions is based on the Pareto-dominance principle, and the convergence of the basic algorithm plays a pivotal role in this principle. On the other hand, these algorithms need an explicit diversity preservation scheme to maintain a diverse set of solutions. In this way, two of the most prevailing MOEAs are NSGA-II [14] and SPEA2 (the improved version of strength Pareto evolutionary algorithm) [15]. They have achieved remarkable results in solving MOPs. Although the dominance-based algorithms are prevailing, they also have their own limitations to overcome. One of the main drawbacks is that they are not very suitable for solving the many-objective optimization problems. Because in the face of the many-objective optimization problems, almost all of the solutions will become non-dominated with respect to each other in the population, this will reduce the selection pressure and hinder the process of evolution [16,17].

With the continuous in-depth study of scholars around the world, the recent well-known MOEA, the multi-objective evolutionary algorithm based on decomposition, named MOEA/D [18], proposed by Zhang and Li in 2007, was dissimilar from the past algorithms. MOEA/D does not use the Pareto dominance, but uses the scalarization technique for fitness assignment to solve the multi-objective optimization problems. It uses scalarizing functions such as the weighted sum approach and the weighted Tchebycheff approach to decompose a MOP into a number of single subproblems and solves all of these subproblems in a single run (note that the description of the weighted sum approach and the weighted Tchebycheff will be shown in Section 2.2). Each subproblem is relevant to a weighted scalarizing function and neighborhood relationship, which should be considered between any two sub-problems. These algorithms based on MOEA/D in solving continuous and combinatorial MOPs have already achieved extremely promising results [19–21].

On the other hand, as an efficient and vigorous heuristic for the global optimization algorithm, differential evolution (DE) was proposed by Storn and Price [22,23]. As a prevalent EA, DE exhibits outstanding performance in solving a wide variety of problems in different fields. It employs crossover, mutation and selection operators to move its population toward the global optimum in each generation. DE as a successful EA is widely used to cope with many optimization problems, as well as it has good adaptability, so that it can be combined with other algorithms to solve MOPs at another level.

In the course of [24], a simple DE scheme is combined with MOEA/D, called MOEA/D-DE, which solves a class of new MOPs that have complicated Pareto sets. In this version of MOEA/D, called MOEA/D-DE, mating parents are selected from the neighborhood and the whole population together. It also points out that the parameter setting in the DE scheme is very important for the performance of the MOEA/D-DE algorithm. The successful implementation of MOEA/D-DE not only shows the good performance of MOEA/D, but also shows the good adaptability of the DE algorithm. Therefore, we can make some effort following this means to attempt to solve more difficult MOPs.

Moreover, it is undoubted that (1) is a multi-objective optimization problem and usually has two or three objectives. Nevertheless, in daily life, many real-world applications often involve four or more objectives for optimizing. Thus, we say that many-objective optimization problems are the class of MOPs that have four or more than four objectives. Because of its usefulness, it is not strange that the research on many-objective optimization problems has been one of the major research areas in the EMO community during recent years. To name only a few, in 2013, Tan et al. [25] proposed an improved MOEA/D, named UMOEA/D (MOEA/D with uniform design). They used the uniform design method to set the weight vectors of MOEA/D for solving the many-objective problems and got some effective results. Qi et al. [26] proposed an improved MOEA/D in the weight vector design method in 2014, called MOEA/D-AWA (MOEA/D with adaptive weight vector adjustment). They used a new weight vector initialization method and made the weight vectors adaptive. There were many experiments on MOPs and a small amount of experiments on the many-objective optimization problems, which certified how effective MOEA/D-AWA was. In 2014, Deb and Himanshu [27] suggested a reference point-based NSGA-II for many-objective problems, similar to NSGA-II, called NSGA-III. NSGA-III still employed the Pareto non-dominated sorting method and added the decomposition concept to solve the many-objective optimization problems. In 2015, Li et al. [28] proposed a new algorithm that exploited the merits from both the dominance- and decomposition-based approaches to balance the convergence and diversity of the evolutionary process to solve the many-objective optimization very well. From these points of view, while solving this kind of many-objective optimization problem, the original MOEA/D will meet some limitations, such as the distributions of weighted vectors not being very uniform, and it just uses a single approach to decompose the MOPs. All of these limitations are encouraging researchers to study it. This provides the major impetus for us to study how to improve the performance of MOEA/D on solving the many-objective optimization problems. Therefore, an improved MOEA/D with optimal DE schemes (MOEA/D-oDE) is proposed in this paper. Our purposes are to employ and use the advantages of different DE schemes. Firstly, MOEA/D-oDE uses the alterable decomposition approach. Besides, MOEA/D-oDE uses the optimal DE schemes. More precisely, the proposed algorithm MOEA/D-oDE combines the weighted sum approach and the Tchebycheff approach to produce a new scalarizing function to decompose the many-objective optimization problems. Additionally, we use the alterable DE's trial vector operators to improve the performance of MOEA/D in solving the many-objective optimization problems.

The remaining part of this paper is arranged as follows. Section 2 presents the preliminaries of our study, including the basic idea of MOEA/D, the basic decomposition approaches and a simple introduction to the DE's trail vector design methods. Section 3 presents our newly-proposed algorithm, named MOEA/D-oDE, which attaches importance to how to decompose the many-objective optimization problems and how to make the results of the DE operators perform well. Section 4 presents the experimental studies on MOEA/D-oDE, comparing MOEA/D-oDE with MOEA/D and NSGA-II in solving the many-objective problems, and the experimental results demonstrate that MOEA/D-oDE outperforms or performs similarly to MOEA/D and NSGA-II. Finally, Section 5 concludes this paper concisely.

2. Preliminaries

In this section, there is some basic knowledge that will be employed in this paper. First of all, the basic idea of MOEA/D is introduced in Section 2.1, which presents briefly how MOEA/D works,

but not providing the framework and details of MOEA/D. Thereafter, the decomposition approaches used in MOEA/D are introduced in Section 2.2, which introduces the weighted sum approach, the Tchebycheff approach and the penalty-based boundary intersection approach, respectively. Eventually, how trail vectors of differential evolution are generated is interpreted in Section 2.3 in detail.

2.1. Basic Idea of MOEA/D

MOEA/D is a representative algorithm based on decomposition, which was proposed by Zhang and Li. In addition, Zhang and Li had proven that it is superior to the other state-of-the-art MOEAs in 2007 [18]. Moreover, a variant of MOEA/D, which is named MOEA/D-DRA (MOEA/D with dynamical resource allocation), had achieved first place at CEC2009 (2009 IEEE Congress on Evolutionary Computation) [29]. The basic idea of MOEA/D is using some kind of aggregation function to decompose an MOP into some single objective optimization sub-problems and to use some other evolutionary algorithm to optimize them simultaneously.

In the general MOEA/D framework, let N be the population size, m be the number of the objectives, for all $i = 1, \dots, N$, and $\lambda_1, \lambda_2, \dots, \lambda_N$ be a set of aggregation coefficient weight vectors, where $\lambda_i = (\lambda_i^1, \lambda_i^2, \dots, \lambda_i^m)^T$, $j = 1, \dots, m$, and subject to $\lambda_i^j \geq 0$, $\sum_{j=1}^m \lambda_i^j = 1$. With these weight vectors, MOEA/D employs a decomposition method (discussed in Section 2.2) to decompose an MOP (1) into single objective optimization subproblems and minimizes all of these N subproblems in a single run. Each subproblem i corresponding to vector λ_i is optimized by using the information just from its neighborhood. The neighborhood of weight vector λ_i is defined as a group of its T closest weight vectors in the entire set $\{\lambda_1, \lambda_2, \dots, \lambda_N\}$. Therefore, the neighborhood of the i -th subproblem is composed of all of the subproblems whose weight vectors are from the neighborhood of λ_i . The constantly-updated population consists of the best solution found up to now for each subproblem. Only the present solutions to these neighboring subproblems are utilized for optimizing a subproblem in the algorithm. For more details about MOEA/D, readers may refer to [18].

2.2. Decomposition Approaches Used in MOEA/D

In the study of the original MOEA/D, a total of three kinds of decomposition approaches are introduced simply here. They are the weighted sum approach (WS), the Tchebycheff approach (TCH) and the penalty-based boundary intersection approach (PBI). A brief introduction of them is shown in the following:

- (1) Weighted sum (WS) approach: The optimization problem of the WS approach is defined as:

$$\begin{aligned} & \text{minimize} && g^{ws}(x|\lambda) = \sum_{i=1}^m \lambda_i f_i(x) \\ & \text{subject to} && x \in \Omega \end{aligned} \quad (2)$$

where $\lambda = \{\lambda_1, \lambda_2, \dots, \lambda_N\}$ is a set of weight vectors, for all $i = 1, \dots, N$, $\lambda_i = (\lambda_i^1, \lambda_i^2, \dots, \lambda_i^m)^T$, and subject to $\lambda_i^j \geq 0$, $\sum_{j=1}^m \lambda_i^j = 1$ for all $j = 1, \dots, m$. This approach could work well if the PF is convex (for minimization problems). Nevertheless, not every Pareto-optimal vector can be obtained through this approach if the PF is non-convex.

- (2) The Tchebycheff (TCH) approach: The optimization problem of the TCH approach is defined as:

$$\begin{aligned} & \text{minimize} && g^{tch}(x|\lambda, z^*) = \max_{1 \leq i \leq m} \{\lambda_i |f_i(x) - z_i^*|\} \\ & \text{subject to} && x \in \Omega \end{aligned} \quad (3)$$

where λ is the same as above in (2), and $z^* = (z_1^*, \dots, z_m^*)^T$ is the ideal reference point, i.e., for each $i = 1, \dots, m$, $z_i^* < \min\{f_i(x) | x \in \Omega\}$.

As we know, there must exist a weight vector λ for every Pareto-optimal point x^* , such that x^* is the optimal solution of (3). In addition, every optimal solution of (3) should be a Pareto-optimal solution of (1). Consequently, we can obtain different Pareto-optimal solutions by altering the weight vectors.

- (3) Penalty-based boundary intersection (PBI) approach: The optimization problem of the PBI approach is defined as:

$$\begin{aligned}
 &\text{minimize} && g^{pbi}(x|\lambda, z^*) = d_1 + \theta d_2 \\
 &\text{where} && d_1 = \frac{\|(F(x)-z^*)^T \lambda\|}{\|\lambda\|} \\
 &&& d_2 = \|F(x) - (z^* - d_1 \lambda)\| \\
 &\text{subject to} && x \in \Omega
 \end{aligned} \tag{4}$$

In (4), λ is the weight vector as defined in (2), z^* is the ideal reference point as defined in (3) and $\theta > 0$ is a penalty parameter. From the previous study, using the same distributed weighted vectors, the PBI approach has some advantages over the TCH approach when solving an MOP whose number of objectives is more than two. However, the benefits must come with a price, which means the θ as a penalty parameter requires proper adjusting.

Some of these approaches have been successfully used in solving real-world problems. For example, in [8], Tan et al. used the weighted sum approach to design a pattern reconfigurable array antenna and got some certain results. In [20], Ke et al. used the Tchebycheff approach to solve the multi-objective traveling salesman problem (MTSP) and multiobjective 0-1 knapsack problem (MOKP) well.

2.3. Differential Evolution's Trial Vectors

DE is a kind of algorithm that deals with continuous optimization problems. To illustrate the basic concepts of DE, suppose that the objective function for this part is minimizing the $f(\vec{x})$, $\vec{x} = (x_1, x_2, \dots, x_D) \in \mathbb{R}^D$, and the feasible solution region is $\Omega = \prod_{i=1}^D [L_i, U_i]$. At generation $G = 0$, a random initial population $\{\vec{x}_{i,0} = (x_{i,1,0}, x_{i,2,0}, \dots, x_{i,D,0}), i = 1, \dots, N, j = 1, \dots, D\}$ is generated from the feasible solution region, where N is the population size. In the evolution of the DE algorithm, for each generation G of the current population, DE will create a mutation vector $\vec{v}_{i,G} = (v_{i,1,G}, v_{i,2,G}, \dots, v_{i,D,G})$ for each individual $\vec{x}_{i,G}$, which is called a target vector. Five kinds of widely-used DE mutation operators are shown as follows.

- (1) "DE/rand/1"

$$\vec{v}_{i,G} = \vec{x}_{r1,G} + F \cdot (\vec{x}_{r2,G} - \vec{x}_{r3,G}) \tag{5}$$

- (2) "DE/best/1"

$$\vec{v}_{i,G} = \vec{x}_{best,G} + F \cdot (\vec{x}_{r1,G} - \vec{x}_{r2,G}) \tag{6}$$

- (3) "DE/current-to-best/1"

$$\vec{v}_{i,G} = \vec{x}_{i,G} + F_1 \cdot (\vec{x}_{best,G} - \vec{x}_{i,G}) + F_2 \cdot (\vec{x}_{r1,G} - \vec{x}_{r2,G}) \tag{7}$$

- (4) "DE/best/2"

$$\vec{v}_{i,G} = \vec{x}_{best,G} + F_1 \cdot (\vec{x}_{r1,G} - \vec{x}_{r2,G}) + F_2 \cdot (\vec{x}_{r3,G} - \vec{x}_{r4,G}) \tag{8}$$

(5) “DE/rand/2”

$$\vec{v}_{i,G} = \vec{x}_{r1,G} + F_1 \cdot (\vec{x}_{r2,G} - \vec{x}_{r3,G}) + F_2 \cdot (\vec{x}_{r4,G} - \vec{x}_{r5,G}) \tag{9}$$

In the above five equations, $r1, r2, r3, r4$ and $r5$ are the independent individuals randomly selected from the population size and are different from i . $\vec{x}_{best,G}$ is the first-rate individual in the current population. Additionally, the parameters F, F_1 and F_2 are called the scaling factors, and F and F_2 are usually equal; all of them control the degree of mutation.

After mutation, DE employs a crossover operator on $\vec{v}_{i,G}$ and $\vec{x}_{i,G}$ to generate a trial vector $\vec{u}_{i,G}$ and $\vec{u}_{i,G} = (u_{i,1,G}, u_{i,2,G}, \dots, u_{i,D,G})$, described as follows:

$$u_{i,j,G} = \begin{cases} v_{i,j,G}, & \text{if } rand_j(0,1) \leq Cr \text{ or } j = j_{rand} \\ x_{i,j,G}, & \text{otherwise} \end{cases} \tag{10}$$

where $i = 1, 2, \dots, N, j = 1, 2, \dots, D, rand_j(0,1)$ is a random number uniformly distributed from zero to one, requiring to be generated for each j . Additionally, $Cr \in [0, 1]$ is called the control parameter for crossover. The number j_{rand} is an integer in $[1, D]$, which is also chosen randomly. Using j_{rand} , the trial vector $\vec{u}_{i,G}$ could always be different from the target vector $\vec{x}_{i,G}$. More details about DE and the DE’s operator may be found in [22,23,30].

3. MOEA/D with Optimal DE Schemes

In order to solve the many-objective problems well, this paper proposes an improved MOEA/D with optimal DE schemes, named MOEA/D-oDE. Our main purposes are to explore and employ the advantages of different DE schemes. Apart from this, a combined decomposition approach is introduced and adopted in the proposed MOEA/D-oDE. Therefore, the combined decomposition method is given in Section 3.1, and the method of optimal DE schemes is given in Section 3.2 to introduce them, respectively. The final subsection will give the framework of the proposed algorithm, MOEA/D-oDE, and interpret it briefly.

3.1. The Combined Decomposition Method

As shown in Section 2.2, three decomposition approaches are introduced, on account of the friendly expansibility of the weighted sum approach and the Tchebycheff approach. We combine them in advance and propose a combined decomposition method, named the weighted sum δ Tchebycheff (WST) approach. Here, we refer to a concept from [31]: for subproblem i , let x^i be the current solution in the decision space; the improvement region of x^i can be defined as: $\{F(x)|x \text{ is better than } x^i \text{ for subproblem } i\}$. As illustrated in Figure 1, for two objective optimization as an example, $\lambda_i = \{0.5, 0.5\}, i = 1, \dots, N$, the improvement regions of two commonly-used decomposition approaches, (a) the weighted sum approach and (b) weighted Tchebycheff approach, are shown. In each sub-figure, the grey area is the improvement region. The square point is the current solution of the subproblem i with the direction vector and the triangle point being its optimal solution. Thereafter, the optimal target of the square point is to find its corresponding triangle point. As described in Equations (2) and (3) and from Figure 1, we inform the reader that the weighted Tchebycheff approach barely gives priority to the salient objective, while on the contrary, the weighted sum approach attaches importance to all of the objectives. Therefore, we would like to combine these two approaches together by considering them on both a salient field and an entire field. A combined decomposition method, named the WST method, is introduced. The optimization problem of the WST approach is defined as:

$$\begin{aligned}
 \text{minimize } g^{wst}(x|\lambda, z^*) &= \frac{(m-1)^2}{m^2} \max_{1 \leq i \leq m} \{\lambda_i |f_i(x) - z_i^*|\} + \\
 & \left[1 - \frac{(m-1)^2}{m^2}\right] \sum_{i=1}^m \lambda_i f_i(x) \tag{11} \\
 \text{subject to } & x \in \Omega
 \end{aligned}$$

where m is the number of objectives, λ is the weight vector as defined in (2) and $z^* = (z_1^*, \dots, z_m^*)$ is the reference points as defined in (3). For each $i = 1, \dots, m$, $z_i^* = \min\{f_i(x) | x \in \Omega\}$.

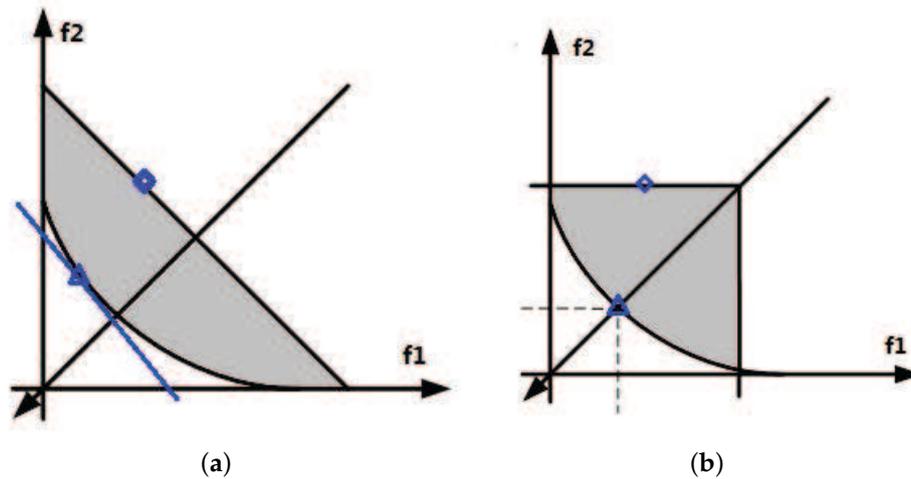


Figure 1. Illustration of the improvement regions for the (a) weighted sum approach and the (b) weighted Tchebycheff approach.

Additionally, we would like to point out that the WST method combines the advantages of the weighted sum and the weighted Tchebycheff approach. It not only emphasizes one single aspect, but also pays attention to the entire direction of the objects. Our experimental results have proven its validity in the end.

3.2. The Optimal DE Schemes

As presented in Section 2.3, there are five commonly-used DE mutation operators. In our algorithm, in order to make full use of DE operators and improve the search ability, due to the previous literature [22,32], and the fact that the value of F could always be different for different problems, we decide to choose the set of Equations (6) and (8) and the scaling factors set of $\{0.3, 0.5\}$ to generate trial vectors used in the later evolutionary computation to do our experiments. In this case, we could get four trial vectors (Operator (6) with $F = 0.3$, Operator (6) with $F = 0.5$, Operator (8) with $F = 0.3$ and Operator (8) with $F = 0.5$) in a single run, and then, we would like to find and select the best child solution for the a posteriori evolution. Besides, for Formula (8), the value of F_1 is suggested to be a random number, which is uniformly distributed from zero to 0.5, which is beneficial for the convergence. Figure 2 shows the flow chart of our method roughly. In the end, the experimental results have proven its effectiveness.

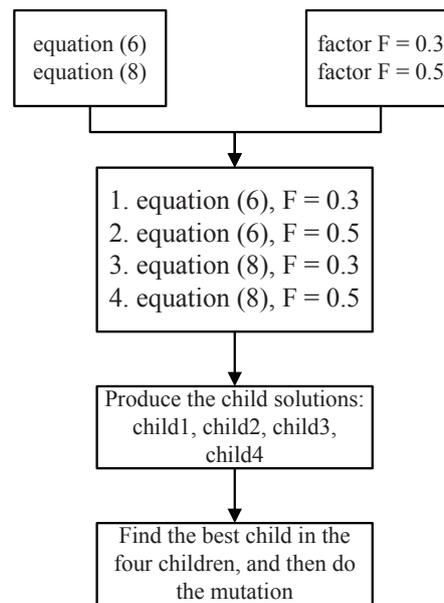


Figure 2. Flowchart of generating differential evolution (DE) operators.

3.3. Framework of MOEA/D-oDE

MOEA/D-oDE decomposes a many-objective problem into a set of single objective problems and solves them in parallel. Each decomposed single objective problem is handled with optimal DE schemes. The framework of MOEA/D-oDE is based on MOEA/D. We would like to point out that there are two distinguishing points from MOEA/D: (1) we select the WST decomposition approach to decompose the many-objective optimization problems; (2) optimal DE schemes instead of single DE operator schemes are employed in our experiments.

Following the recounted methods, at each generation, MOEA/D-oDE with the WST approach maintains the following: (1) a population of N points x_1, x_2, \dots, x_N , where x_i is the present solution to the i -th subproblem; (2) reference ideal point $z = (z_1, z_2, \dots, z_m)^T$, where z_j is the best value for objective f_j that is found up to now, and $\forall j = 1, 2, \dots, m$; (3) the external population (EP), which is used to store the non-dominated solutions that have been found in the search process.

The pseudocodes of our proposed algorithm MOEA/D-oDE are shown in Algorithm 1. As shown in Algorithm 1, in the initialization procedure of MOEA/D-oDE (Lines 8–17), we dispose of the neighborhood structure $B(i)$ by computing the Euclidean distances between any two weight vectors and then arrange the T closest weight vectors with respect to each weight vector firstly. Furthermore, we generate the initial population, and for $\forall i = 1, \dots, N$, evaluate $F(x_i)$ respectively, and mainly initialize z by the condition: $z_j = \min_{1 \leq i \leq N} f_j(x_i), \forall j = 1, \dots, m$ in the end. In the update procedure of MOEA/D-oDE (Lines 19–27), if the stopping criteria are not satisfied, then go to Step 2.1 firstly, which does the reproduction by using optimal DE schemes, and then, the values that overstep the boundary should be repaired in Step 2.2. Secondly, we evaluate the function and update of z by the new minimum evaluation. Finally, through the $g^{wst}(x'|\lambda_j, z) \leq g^{wst}(x_j|\lambda_j, z)$, we renew the best solutions and obtain the final results.

Algorithm 1: The pseudocode of MOEA/D-oDE.

```

1 Input:
2   MOP(1);
3    $N$ : the number of the population size;
4    $\{\lambda_1, \lambda_2, \dots, \lambda_N\}$ : a set of  $N$  improved uniformly-distributed weight vectors;
5    $T$ : the neighborhood size;
6 Output:
7   EP;
8 Step 1: Initialization
9   Step 1.1) Set  $EP = \emptyset$ ;
10  Step 1.2) Compute the Euclidean distances between any two weight vectors, and then
11           dispose of the  $T$  closest weight vectors with respect to each weight vector. For
12           each  $i = 1, \dots, N$ , set  $B(i) = \{i_1, i_2, \dots, i_T\}$ , where  $\forall j \in B(i)$ ,  $\lambda_j$  are the  $T$ 
13           closest vectors to  $\lambda_i$ ;
14  Step 1.3) Generate the initial population randomly;
15  Step 1.4)  $\forall i = 1, \dots, N$ , evaluate  $F(x_i)$ ;
16  Step 1.5) Initialize  $z$ : the original reference point  $(z_1, z_2, \dots, z_m)$  generated by the
17           condition:  $z_j = \min_{1 \leq i \leq N} f_j(x_i), \forall j = 1, \dots, m$ ;
18 Step 2: Update
19  Step 2.1) Reproduction: Randomly select two indexes  $k$  and  $l$  from  $B(i)$ , and generate a
20           child solution  $x'$  from the parents  $x_k$  and  $x_l$  by using the optimal DE schemes;
21  Step 2.2) Repair: Repair  $x'$  by applying a problem-specific repair/improvement
           heuristic;
22  Step 2.3) Function evaluation: Evaluate  $F(x')$ ;
23  Step 2.4) Update of  $z$ : For each  $j = 1, \dots, m$ , if  $z_j > f_j(x')$ , then set  $z_j = f_j(x')$ ;
24  Step 2.5) Replacement/update of solutions: For each index  $j \in B(i)$ , if
25            $g^{wst}(x'|\lambda_j, z) \leq g^{wst}(x_j|\lambda_j, z)$  then set  $x_j = x'$  and  $F(x_j) = F(x')$ ;
26  Step 2.6) Remove from EP all of the vectors dominated by  $F(x')$ . Add  $F(x')$  to EP if no
27           vector in EP dominates  $F(x')$ .
28 Step 3: Stopping criteria
29   If the stopping criteria is satisfied, then stop, and output EP. Otherwise, go to Step 2.

```

4. Experimental Study

In this section, the experimental studies are shown to validate the effectiveness of MOEA/D-oDE. Test instances are used for testing the performance of the algorithms, and performance metrics are used for reflecting the performance of the algorithms, which are exhibited firstly. Additionally, there are three parts of the experiments in the following subsection. One compares MOEA/D with the WST approach to the other two approaches (the WS and the Tchebycheff approach). The other compares MOEA/D-oDE with MOEA/D [18] (MOEA/D with the Tchebycheff approach) and NSGA-II [14]. In the end, MOEA/D-oDE with MOEA/DD (a recent algorithm proposed by Li et al. which solves many-objective optimization problems well) [28] and MOEA/D-AWA [26] are compared as further experiments to reflect the performance of MOEA/D-oDE.

4.1. Test Instance

In our experiments, we choose the family of the four DTLZ functions that are of a scalable objective [33] for benchmark test instances. Additionally, the other two scalable continuous test instances (F1 and F2) which have complicated PS shapes are also employed in our experiments. Taking into account the limited space, we put the mathematic representations of all test functions in

the Appendix, where we can find that the DTLZ functions' family is different from the F1 and F2. Additionally, we can learn from the previous literature [28,33] that DTLZ1 is linear and multi-modal, DTLZ2 is concave, DTLZ3 is concave and multimodal and DTLZ4 is concave and biased. On the other hand, F1 and F2 have variable complicated PS shapes, which have certificated that these problems could cause some trouble for the algorithms. More details of the six continuous test instances can be seen in [25]. Here, all of these test instances are for minimization.

4.2. Comparison Algorithms and Performance Metrics

In this paper, we compare MOEA/D with the WST approach to MOEA/D with WS and the Tchebycheff approach, named MOEA/D-WST, MOEA/D-WS and MOEA/D-TCH, respectively, to test the performance of the WST approach firstly. For this part, we test the DTLZ1–DTLZ4 with 4–6 objectives representatively. After that, we compare MOEA/D-oDE with two other well-known algorithms, MOEA/D [18] and NSGA-II [14], on all six test instances. Lastly, MOEA/DD [28] with DE operators and MOEA/D-AWA [26] have been tested to compare MOEA/D-oDE on DTLZ1–DTLZ4 with four objectives representatively.

As shown above, MOEA/D is a famous MOEA based on decomposition, and MOEA/D-oDE is developed from the MOEA/D, so we choose the MOEA/D to test all six test instances at the same time, and we should point out that the Tchebycheff approach is used in MOEA/D for comparing. In addition, NSGA-II is the most widely-cited algorithm in the MOEA field, which is based on the Pareto-dominance principle. Although NSGA-II is relatively long, its influence is still profound. On the other hand, MOEA/DD and MOEA/D-AWA are all of the recently-proposed promising algorithms. In other words, all four algorithms are influential, so it is worth comparing them. In addition, we should point out that the results of MOEA/DD are run by “PlatEMO” [34], and we acknowledge the team that made this. The source codes of MOEA/D-AWA can be found on the personal web sites of the author Qi.

During the experiments, the inverted generational distance (IGD) [35–37] metric and the generational distance (GD) [38] are used to determine the quality of an algorithm. The IGD metric value can reflect the convergence and diversity of the results obtained by algorithms. Let P^* be a group of uniformly-distributed Pareto-optimal points in PF, as well as P be a group of points for PF approximation; then, the IGD metric can be defined as follows [10,37]:

$$\text{IGD}(P^*, P) = \frac{\sum_{v \in P^*} d(v, P)}{|P^*|} \quad (12)$$

where $d(v, P)$ is the minimum Euclidean distance between v and any point in P and $|P^*|$ is the cardinal number of P^* . If P^* is a set of large points that could represent the PF well, in a way, it means that the $\text{IGD}(P^*, P)$ could measure the diversity and convergence of P . Therefore, P must be very close to the true PF and does not miss any part of the PF in order to have a smaller $\text{IGD}(P^*, P)$ value. Thus, we can conclude that the smaller the IGD metric value is, the better the algorithm performs.

For generational distance (GD) [38]: Let P^* be a set of uniformly-distributed points in the objective space along the PF and P be an approximation set to the PF; the generational distance from P to P^* is defined as:

$$\text{GD}(P, P^*) = \frac{\sqrt{\sum_{v \in P} (d(v, P^*))^2}}{|P|} \quad (13)$$

$$d(v, P^*) = \min_{x^* \in P^*} \left\{ \sqrt{\sum_{i=1}^m (f_i(x) - f_i(x^*))^2} \right\}$$

where m is the number of objectives and $d(v, P^*)$ is the minimum Euclidean distance between v and the points in P^* . The generational distance could indicate the distance between the approximate

Pareto-optimal front obtained by the algorithms and the ideal Pareto-optimal front distances. Consequently, the value of $GD(P, P^*)$ is smaller, which means the convergence of the approximate Pareto-optimal front obtained by the algorithms is better; meanwhile, it is closer to the ideal Pareto-optimal front distances. The best situation is that while $GD(P, P^*) = 0$, it means that all solutions in P are the Pareto-optimal solutions.

In our experiments, the number of the solutions in setting P^* selected to calculate the value of the IGD metric is 1000 uniformly-distributed points for four-objective problems, 2000 uniformly-distributed points for five-objective problems and 3003 uniformly-distributed points for six-objective problems.

4.3. Parameter Setting

Almost all of these algorithms are implemented by using C++ programming, and the implementation environment is Visual C++ 6.0 on a personal computer (Intel Core 3.8 GHz CPU and 8.0 GB RAM). In the implementation of MOEA/D-oDE, we carry out the trial DE vector operators and polynomial mutation on all of the test instances. Meanwhile, for the algorithms compared to MOEA/D [18,24], the simulated binary crossover (SBX) [4] operator is employed for DTLZ problems, and the DE crossover operator is employed for F1 and F2. Polynomial mutation is used in MOEA/D for all of the test problems. Evolution operators used in NSGA-II are the same as in MOEA/D. The other parameters used in MOEA/D and NSGA-II are shown in Table 1, where n is the number of variables. Additionally, the parameters of MOEA/DD and MOEA/D-AWA are the same as the parameters used in MOEA/D, and the probability of choosing parents locally of MOEA/DD is set to 0.9.

Table 1. Parameter setting in MOEA/D-optimal DE (oDE), MOEA/D and NSGA-II.

Parameter	MOEA/D-oDE	MOEA/D	NSGA-II
Crossover probability p_c	1	1	1
Distribution index η_c for crossover	20	20	20
Mutation probability p_m	$1/n$	$1/n$	$1/n$
Distribution index η_m for mutation	20	20	20
Crossover rate CR in DE	1	1	
Scaling factor F in DE	0.3; 0.5	0.5	

Moreover, in MOEA/D, both the number of objectives m and the parameter H control the number of set $\{\lambda^1, \dots, \lambda^N\}$, where the population size N is the amount of sub-problems. Each individual weight vector ought to take a value from $\{0/H, 1/H, \dots, H/H\}$. Therefore, the population size, which means the number of weight vectors, is $N = C_{H+m-1}^{m-1}$, where m is the number of objectives. Additionally, for NSGA-II, the number of the non-dominated solutions would increase as the number of objectives increases [39]. For fair comparisons, we set the values of H as 17, 13 and 10 for all of the test instances with 4, 5 and 6 objectives, respectively, in MOEA/D. Hence, we can obtain that the values of the population size N are 1140, 2380 and 3003 for 4-objective, 5-objective and 6-objective continuous test problems respectively in all three algorithms. The number of the neighborhood size is set to $T = 20$ for all of the test instances in MOEA/D-oDE and MOEA/D. Each compared algorithm runs 20 times independently for each instance. When the number of function evaluations reaches the given upper limitation, three algorithms run out. The max number of the function evaluations is set to $250 \times N$ for four- and five- objective problems and $500 \times N$ for six-objective problems.

4.4. Experimental Results and Analysis

In this subsection, three parts of experimental results and analyses are shown respectively in the following.

4.4.1. MOEA/D-WST Compared with MOEA/D-WS and MOEA/D-TCH

Since the proposed WST approach is a combination of the weighted sum and the Tchebycheff approach, we have tested the MOEA/D-WST, MOEA/D-WS and MOEA/D-TCH on DTLZ1–DTLZ4 with 4–6 objectives. These three algorithms function with the same condition without the decomposition approach. Table 2 shows the IGD metric values obtained by three algorithms, including the mean, the best and the standard values of the 20 IGD metric results. Table 3 shows the GD metric values obtained by the three algorithms, including the mean, the best and the standard values of the 20 IGD metric results, as well. We can find from these two tables that the MOEA/D-WST has better performance on both the IGD metric and the GD metric values for most of the instances. This is because the WST approach has the features of both the weighted sum approach and the Tchebycheff approach. Therefore, MOEA/D with the WST approach has some certain superiorities over MOEA/D with the weighted sum approach and MOEA/D with the Tchebycheff approach.

Table 2. Statistics of the IGD metric obtained by MOEA/D-WST, MOEA/D-WS and MOEA/D-TCH on DTLZ1–DTLZ4.

Objectives	Problem	MOEA/D-WST			MOEA/D-WS			MOEA/D-TCH		
		Mean	Best	Std	Mean	Best	Std	Mean	Best	Std
4	DTLZ1	0.2590	0.2580	0.0110	0.2640	0.2630	0.1620	0.2630	0.2630	0.0003
	DTLZ2	0.2040	0.2030	0.0007	0.2740	0.2720	0.0010	0.2710	0.2680	0.0010
	DTLZ3	0.2050	0.2040	0.0004	0.2740	0.2480	0.0067	0.2720	0.2710	0.0007
	DTLZ4	0.2840	0.2030	0.1380	0.5710	0.2730	0.1900	0.4540	0.2710	0.1780
5	DTLZ1	0.2430	0.2420	0.0008	0.2830	0.2740	0.0183	0.2760	0.2740	0.0025
	DTLZ2	0.3140	0.3120	0.0013	0.5500	0.5440	0.0047	0.4700	0.4680	0.0014
	DTLZ3	0.3110	0.3100	0.0007	0.4710	0.4690	0.0008	0.5500	0.5180	0.0120
	DTLZ4	0.4060	0.3100	0.1140	0.7300	0.5610	0.1280	0.5980	0.4690	0.1400
6	DTLZ1	0.2680	0.2680	0	0.2420	0.2060	0.0173	0.2680	0.2680	0
	DTLZ2	0.3470	0.3420	0.0026	0.7350	0.7230	0.0099	0.4810	0.4750	0.0121
	DTLZ3	0.3440	0.3410	0.0021	0.7340	0.7090	0.0109	0.4890	0.4750	0.0146
	DTLZ4	0.5880	0.3410	0.1590	0.8100	0.7250	0.0826	0.6910	0.4740	0.1340

Table 3. Statistics of the GD metric obtained by MOEA/D-WST, MOEA/D-WS and MOEA/D-TCH on DTLZ1–DTLZ4.

Objectives	Problem	MOEA/D-WST			MOEA/D-WS			MOEA/D-TCH		
		Mean	Best	Std	Mean	Best	Std	Mean	Best	Std
4	DTLZ1	0.0220	0.0219	0.0224	10.300	2.7300	17.300	0.0307	0.0306	0.0308
	DTLZ2	0.0061	0.0061	0.0062	0.0064	0.0063	0.0066	0.0072	0.0072	0.0073
	DTLZ3	0.0061	0.0061	0.0062	1.8300	0.0063	9.8600	0.0072	0.0072	0.0072
	DTLZ4	0.0062	0.0061	0.0062	0.0070	0.0062	0.0079	0.0072	0.0069	0.0075
5	DTLZ1	0.0228	0.0227	0.0230	10.300	4.8800	14.600	0.0302	0.0302	0.0303
	DTLZ2	0.0099	0.0098	0.0101	0.0100	0.0096	0.0107	0.0146	0.0145	0.0146
	DTLZ3	0.0098	0.0096	0.0098	2.4400	0.0098	5.1600	0.0146	0.0145	0.0146
	DTLZ4	0.0103	0.0096	0.0098	0.0121	0.0102	0.0197	0.0140	0.0129	0.0147
6	DTLZ1	0.0131	0.0131	0.0131	22.000	8.1700	47.200	0.0131	0.0131	0.0131
	DTLZ2	0.0117	0.0116	0.0119	0.0099	0.0095	0.0104	0.0169	0.0160	0.0172
	DTLZ3	0.0117	0.0114	0.0119	1.7600	0.0397	3.3000	0.0168	0.0159	0.0171
	DTLZ4	0.0131	0.0114	0.0119	0.0209	0.0134	0.0349	0.0171	0.0154	0.0182

4.4.2. MOEA/D-oDE Compared with MOEA/D and NSGA-II

All compared algorithms run 20-times independently for each test problem in the process of our experiments. Table 4 shows the IGD metric values obtained by three algorithms, including the mean, the best and the standard values of the 20 IGD metric results. From the table, we can see that the performance of MOEA/D-oDE is better than the other two algorithms except for the DTLZ4 problem with six objectives. For this problem, NSGA-II performs the best. Because of this, we cannot say that

the decomposition-based methods must be better than dominance-based methods. Apart from this instance, for the other test instances, MOEA/D-oDE performs better or much better than MOEA/D and NSGA-II. Meanwhile, NSGA-II performs better than MOEA/D on the DTLZ function family with 4–5 objectives. However, for F1 and F2, MOEA/D always performs better than NSGA-II, no matter with many objectives. With the number of objectives increasing, MOEA/D compared to NSGA-II becomes better on DTLZ1 and DTLZ3 with six objectives. For this, we can see that MOEA/D with DE operators in dealing with problems with complicated PS shapes has its unique advantages, but for problems such as the DTLZ functions' family, its superiority is not very obvious. Nevertheless, we have used the DE operators for crossover in MOEA/D-oDE for all of the test problems. In terms of the IGD metric values, the results obtained by MOEA/D-oDE are much better than those obtained by MOEA/D and NSGA-II for almost all of these test problems. Its advantages are very clear, especially for F1 and F2. Then, we can conclude that MOEA/D-oDE can handle the many-objective optimization problems well, whether the problems have complicated PS shapes or not. This also indicates that MOEA/D-oDE is robust for solving these problems.

On the other hand, Table 5 shows the GD metric values obtained by three algorithms, including the mean, the best and the standard values of the 20 IGD metric results, as well. From the table, we can see that the mean GD metrics of MOEA/D-oDE are superior to MOEA/D and NSGA-II on most test instances, as well as almost all of the best values of the GD metric belong to MOEA/D-oDE. However, the stability of MOEA/D-oDE is not very good. Through this table, we can obtain the same conclusions as Table 4.

The evolution of the mean IGD metric values based on 20 independent runs for three algorithms is plotted in Figures 3, 4 and 5, respectively. As shown in these figures, we can find that the convergence speed of the MOEA/D-oDE is faster than the other two algorithms. However, there are many final IGD metric values obtained by NSGA-II that are lower than those obtained by MOEA/D. The convergence speed of NSGA-II is not faster than the other two algorithms in most instances. Meanwhile, the runtime of NSGA-II is the longest of the three algorithms. For NSGA-II, on F1 and F2 with 5–6 objectives, in the later stage of evolution, the obtained IGD metric values do not decrease, but increase. Hence, this indicates that NSGA-II is not good at solving the many objective problems with complicated PS shapes in another way.

As described in Figures 6, 7 and 8, box plots of the IGD metric values among these three algorithms for the continuous problems with 4–6 objectives based on 20 independent runs are visualized. The box diagram can reflect the stability of a set of data. From Table 4 and those three box diagrams, we can conclude that MOEA/D-oDE performs more stably, especially for the F1 and F2 test instances. On the contrary, though NSGA-II has superior performance with respect to MOEA/D on some instances, its stability is usually not very good. The reason may be that the original NSGA-II could not solve the problems with multi-modality and bias very well. Additionally, the stability of MOEA/D does well in some test instances. In some of these box figures, such as Figures 6 and 8, there are some '+' on some test instances, which means the outliers. We can see that MOEA/D and NSGA-II have some larger outliers, which means that the two algorithms may perform worse in one of 20 runs. In other words, to some extent, MOEA/D and NSGA-II do not behave stably on some instances, such as DTLZ4 with four objectives. However, compared to MOEA/D and NSGA-II, the performance of MOEA/D-oDE is more stable.

Table 4. Statistics of the IGD metric obtained by MOEA/D-oDE, MOEA/D and NSGA-II on the test problems.

Objectives	Problem	MOEA/D-oDE			MOEA/D			NSGA-II		
		Mean	Best	Std	Mean	Best	Std	Mean	Best	Std
4	DTLZ1	0.0422	0.0361	0.0172	0.0862	0.0853	0.0004	0.0478	0.0444	0.0018
	DTLZ2	0.0538	0.0530	0.0003	0.0879	0.0868	0.0005	0.0621	0.0602	0.0015
	DTLZ3	0.0586	0.0540	0.0094	0.0864	0.0844	0.0007	0.0631	0.0606	0.0015
	DTLZ4	0.0019	0.0017	0.0038	0.0166	0.0129	0.0032	0.0113	0.0075	0.0031
	F1	0.0408	0.0403	0.0003	0.0827	0.0802	0.0011	0.0974	0.0901	0.0041
	F2	0.0634	0.0611	0.0014	0.0871	0.0821	0.0019	0.1086	0.0979	0.0071
5	DTLZ1	0.0459	0.0448	0.0008	0.1644	0.1618	0.0009	0.1008	0.0854	0.0154
	DTLZ2	0.0772	0.0765	0.0003	0.1750	0.1711	0.0016	0.0988	0.0937	0.0025
	DTLZ3	0.0801	0.0790	0.0008	0.1710	0.1642	0.0026	0.1231	0.1085	0.0200
	DTLZ4	0.0024	0.0022	0.0001	0.0256	0.0255	2.27e-5	0.0235	0.0158	0.0060
	F1	0.0466	0.0461	0.0008	0.1598	0.1575	0.0010	0.2736	0.2305	0.0238
	F2	0.0815	0.0803	0.0007	0.1651	0.1537	0.0067	0.5061	0.3217	0.0826
6	DTLZ1	0.2653	0.2653	1.5e-6	0.2694	0.2694	5.6e-17	3.1426	0.2711	1.6169
	DTLZ2	0.3043	0.2456	0.0299	0.4724	0.4714	0.0005	0.4324	0.4155	0.0101
	DTLZ3	0.3875	0.2768	0.3123	0.4723	0.4706	0.0010	296.57	175.67	67.982
	DTLZ4	0.4131	0.3786	0.0194	0.4870	0.4715	0.0405	0.3598	0.3475	0.0062
	F1	0.4833	0.4750	0.0046	0.5598	0.5434	0.0074	1.5215	1.2525	1.8710
	F2	0.3102	0.2907	0.0074	0.4582	0.4499	0.0037	6.1305	4.4551	4.4551

Table 5. Statistics of the GD metric obtained by MOEA/D-oDE, MOEA/D and NSGA-II on the test problems.

Objectives	Problem	MOEA/D-oDE			MOEA/D			NSGA-II		
		Mean	Best	Std	Mean	Best	Std	Mean	Best	Std
4	DTLZ1	0.0017	0.0016	0.0032	0.0021	0.0021	0	0.1028	0.0107	0.1072
	DTLZ2	0.0025	0.0022	0.0003	0.0028	0.0028	0	0.0022	0.0021	0
	DTLZ3	0.0028	0.0023	0.0049	0.0029	0.0028	0	0.0365	0.0062	0.0481
	DTLZ4	0.0110	0.0107	0.0001	0.0106	0.0085	0.0009	0.0140	0.0137	0.0001
	F1	0.0018	0.0016	0.0001	0.0025	0.0021	0.0003	0.0125	0.0097	0.0022
	F2	0.0024	0.0022	0.0003	0.0044	0.0028	0.0038	0.0396	0.0218	0.0172
5	DTLZ1	0.0039	0.0021	0.3068	0.0032	0.0032	0	0.2345	0.0933	0.0761
	DTLZ2	0.0029	0.0028	0.0001	0.0035	0.0035	0	0.0040	0.0035	0.0002
	DTLZ3	0.0031	0.0028	0.0010	0.0035	0.0035	0	0.2111	0.0910	0.0534
	DTLZ4	0.0081	0.0079	0.0004	0.0090	0.0090	0	0.0101	0.0099	0
	F1	0.0022	0.0021	0	0.0035	0.0032	0.0003	0.0454	0.0312	0.0076
	F2	0.0029	0.0028	0	0.0036	0.0036	0	0.0985	0.0916	0.0037
6	DTLZ1	0.0123	0.0037	0.0036	0.0038	0.0037	0	4.9076	2.2325	1.4610
	DTLZ2	0.0025	0.0021	0.0005	0.0028	0.0027	0	0.0235	0.0216	0.0009
	DTLZ3	0.0027	0.0021	0.0009	0.0028	0.0027	0	25.5919	24.1220	0.7563
	DTLZ4	0.0028	0.0025	0.0004	0.0027	0.0026	0	0.0123	0.0100	0.0012
	F1	0.0068	0.0067	0	0.0064	0.0055	0.0024	0.1042	0.1016	0.0011
	F2	0.0023	0.0021	0.0002	0.0029	0.0026	0	0.4386	0.4321	0.0037

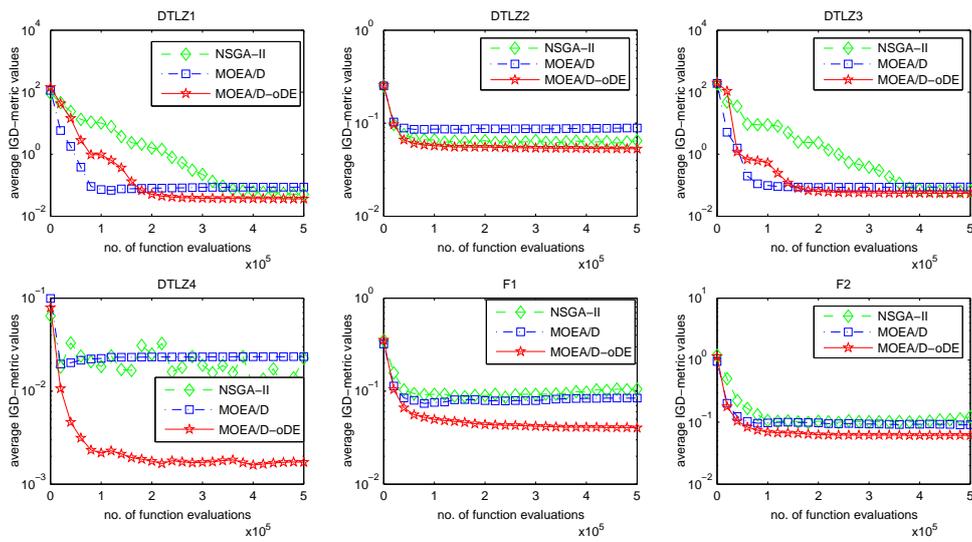


Figure 3. Evolution of the mean of IGD metric values for four-objective test problems.

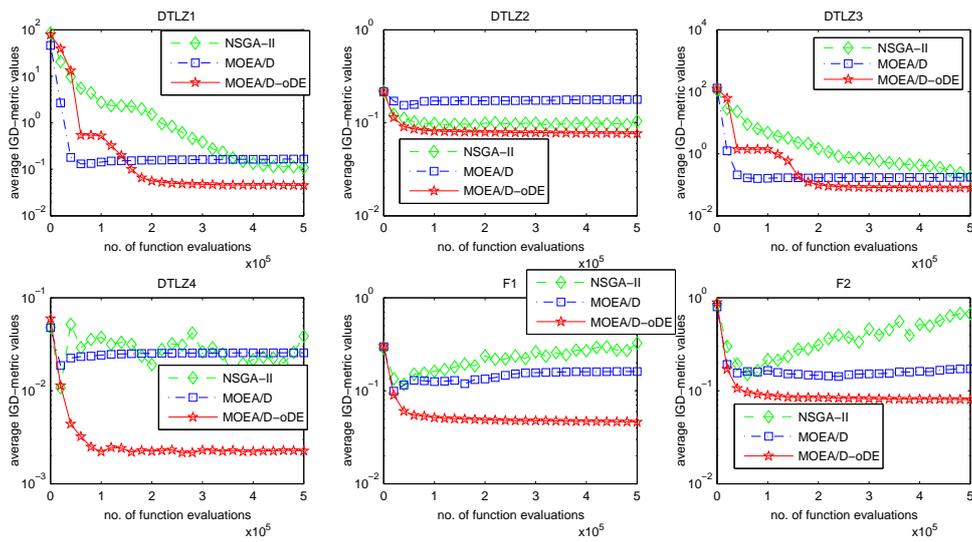


Figure 4. Evolution of the mean of IGD metric values for five-objective test problems.

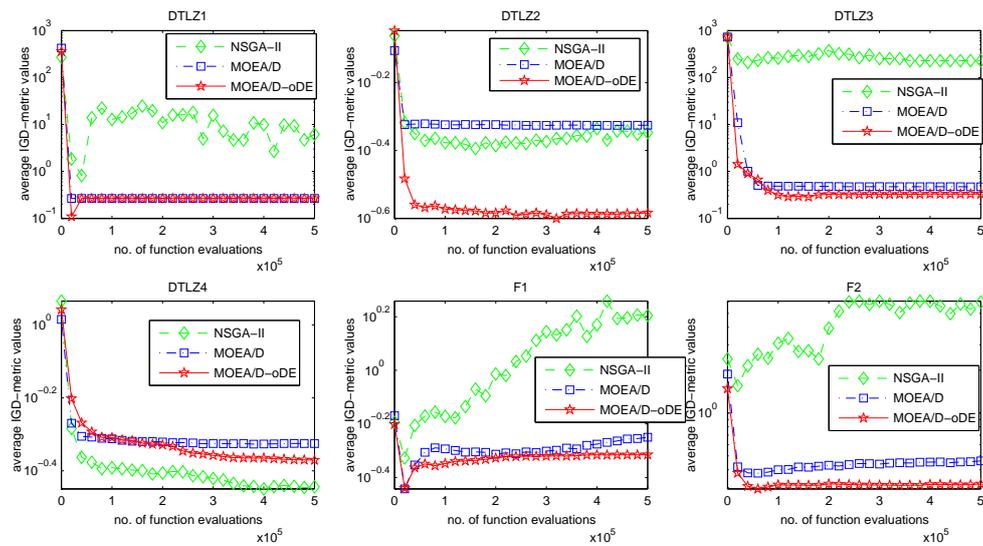


Figure 5. Evolution of the mean of IGD metric values for six-objective test problems.

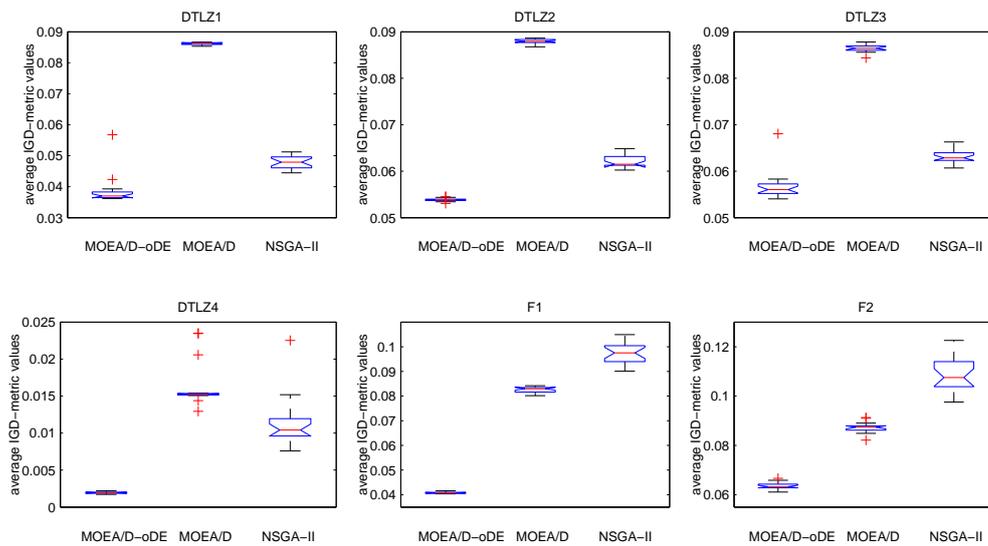


Figure 6. Box plots of the IGD metric values based on 20 independent runs for four-objective test problems.

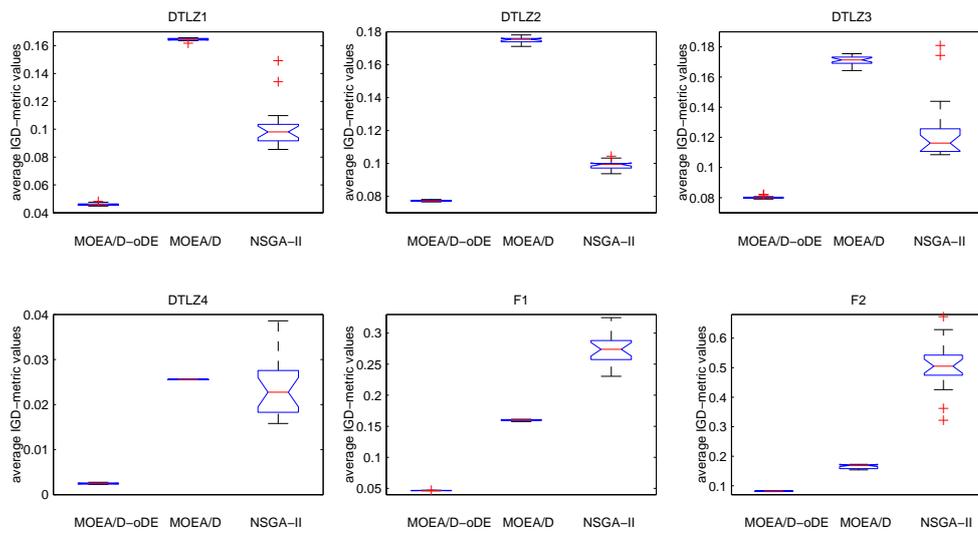


Figure 7. Box plots of the IGD metric values based on 20 independent runs for five-objective test problems.

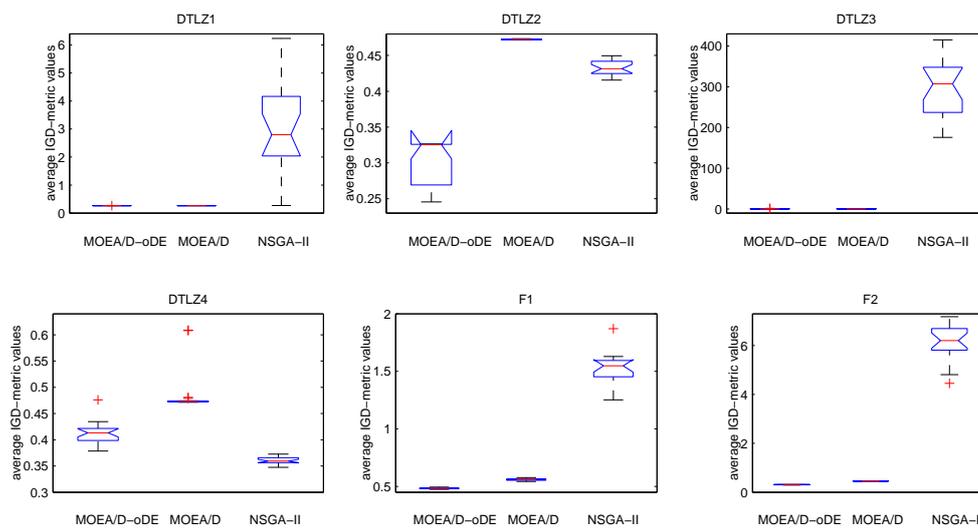


Figure 8. Box plots of the IGD metric values based on 20 independent runs for six-objective test problems.

4.4.3. MOEA/D-oDE Compared with MOEA/DD and MOEA/D-AWA

To further test the performance of MOEA/D-oDE, the two recent algorithms MOEA/DD with DE operators and MOEA/D-AWA are tested on DTLZ1–DTLZ4 with four objectives representatively. Tables 6 and 7 have shown the IGD metric and GD metric values obtained by the three algorithms, including the mean, the best and the standard values of the 20 results. From Table 6, the IGD metrics of MOEA/D-oDE have some superiorities to MOEA/DD and MOEA/D-AWA, but in Table 7, only the GD metrics of DTLZ1 and DTLZ3 of MOEA/D-oDE have the best performance. We can conclude that MOEA/D-oDE has some certain advantages in solving the many-objective optimization problems, as well as MOEA/DD is good at solving these kinds of problems. Nevertheless, MOEA/D-AWA could not obtain very good solutions.

Table 6. Statistics of the IGD metric obtained by MOEA/D-oDE, MOEA/DD and MOEA/D-AWA on DTLZ1–DTLZ4 with 4 objectives.

Objectives	Problem	MOEA/D-oDE			MOEA/DD			MOEA/D-AWA		
		Mean	Best	Std	Mean	Best	Std	Mean	Best	Std
4	DTLZ1	0.0422	0.0361	0.0172	1.2946	0.4973	0.4400	2.1900	2.1200	0.0363
	DTLZ2	0.0538	0.0530	0.0003	0.0515	0.0510	0.0003	6.0200	5.8600	0.0764
	DTLZ3	0.0586	0.0540	0.0094	4.7469	2.0666	1.7146	6.1600	5.9200	0.1360
	DTLZ4	0.0019	0.0017	0.0038	0.0549	0.0142	0.0236	5.9900	5.8900	0.0536

Table 7. Statistics of the GD metric obtained by MOEA/D-oDE, MOEA/DD and MOEA/D-AWA on DTLZ1–DTLZ4 with 4 objectives.

Objectives	Problem	MOEA/D-oDE			MOEA/DD			MOEA/D-AWA		
		Mean	Best	Std	Mean	Best	Std	Mean	Best	Std
4	DTLZ1	0.0017	0.0016	0.0032	0.4917	0.3544	0.1311	0.0150	0.0027	0.0276
	DTLZ2	0.0025	0.0022	0.0003	0.0007	0.0007	0	0.0074	0.0071	0.0002
	DTLZ3	0.0028	0.0023	0.0049	2.2195	1.3006	0.5813	0.0272	0.0073	0.0497
	DTLZ4	0.0110	0.0107	0.0001	0.0007	0.0006	0	0.0073	0.0068	0.0002

4.5. Parameters’ Sensitivity Analysis

Since the scaling factor F in DE is an important element, many previous attempts have been made to improve the convergence speed of DE by tuning the F [22,32]. Due to us having selected the parameter set {0.3, 0.5} for the experiments, we would like to test other parameter combinations for comparison. Here, we select the F1 with four objectives to do the comparative experiments.

We choose six parameter combinations for the comparison study: {0.3, 0.4}, {0.4, 0.5}, {0.4, 0.6}, {0.5, 0.6}, {0.6, 0.9} and {0.9, 1}. Table 8 shows the IGD metric results of all combinations of F1 with four objectives. We can find that the best IGD metric value of these seven groups is the {0.3, 0.4}; otherwise, the worst is the {0.9, 1}. Meanwhile, all of these IGD metric values are stable. From these experiments, we can find that, when the value of the factor F transforms evenly and slowly, it has no great influence on the consequences of the algorithm for F1 with four objectives. More attention to this part needs to be given in the future.

Table 8. IGD metric value statistics for F1 with 4 objectives on different parameters.

Objectives	Parameters Combination	MOEA/D-oDE		
		Mean	Best	Std
4	{0.3, 0.5}	0.0408	0.0403	0.0003
	{0.3, 0.4}	0.0406	0.0384	0.0007
	{0.4, 0.5}	0.0406	0.0388	0.0006
	{0.4, 0.6}	0.0416	0.0405	0.0006
	{0.5, 0.6}	0.0416	0.0408	0.0005
	{0.6, 0.9}	0.0415	0.0407	0.0004
	{0.9, 1}	0.0445	0.0428	0.0007

5. Conclusions

In order to solve the many-objective optimization problems well, an improved MOEA/D with optimal DE schemes, called MOEA/D-oDE, is proposed in this paper. MOEA/D-oDE under the framework of MOEA/D decomposes a many-objective problem into a set of single objective subproblems, and each subproblem is handled with evolutionary operators. All of these subproblems can be solved in parallel.

MOEA/D-oDE has two distinguishing points from MOEA/D. First, a new combined decomposition method named the WST approach is introduced and employed in MOEA/D-oDE.

The WST approach combines the advantages of the weighted sum and the weighted Tchebycheff approach. It focuses not only on one single aspect, but also the entire direction of objectives. Second, we design an optimal DE scheme, which combines two DE crossover operators and two scaling factors F to generate and select the best child solution to do the a posteriori computing in each subproblem. In the experimental studies, MOEA/D with the WST method is compared with MOEA/D with the weighted sum approach and the Tchebycheff approach in DTLZ1–DTLZ4 with 4–6 objectives firstly, which reflects the effectiveness of the new decomposition method. Then, MOEA/D-oDE is tested on three sets of six continuous test instances with 4–6 objectives, and the results obtained by MOEA/D-oDE are compared with those obtained by MOEA/D and NSGA-II. Experimental results indicate that MOEA/D-oDE in solving the problems with a higher dimension of objectives outperforms MOEA/D and NSGA-II in most situations. In the end, the comparison with MOEA/DD and MOEA/D-AWA on DTLZ1–DTLZ4 with four objectives is made to reflect the performance of MOEA/D-oDE from another point of view. Additionally, several sets of parameter sensitivity experiments were carried out. In a word, to some extent, MOEA/D-oDE can solve the many-objective optimization problems well.

In the future, we plan to research how to make the DE’s trial vectors obtain the best child solutions dynamically. Additionally, the multi-objective discrete problems, such as the multi-objective TSP (travelling salesman problem) problems, are also our research targets. It is also worthwhile to research new methods to solve the many-objective optimization problems.

Acknowledgments: The work is partially supported by the National Natural Science Foundation of China (Nos. 61401260, 61373081, 61402268, 61572298, 61602283), the Technology and Development Project of Shandong (No. 2013GGX10125), the Natural Science Foundation of Shandong, China (Nos. BS2014DX006, BS2015DX016, ZR2014FM012, ZR2016FB10), and the Taishan Scholar Project of Shandong, China.

Author Contributions: The four authors worked on the paper in a similar way and on all of its aspects. However, Wei Zheng is the main writer of this paper. Yanyan Tan is the director of this paper. Xiaonan Fang and Shengtao Li offered some useful advice to us.

Conflicts of Interest: The authors declare no conflict of interest.

Appendix. Test Instances Used in This Paper

DTLZ1

$$\left\{ \begin{array}{l} f_1(x) = 0.5x_1x_2 \cdots x_{m-1}(1 + g(x_m)) \\ f_2(x) = 0.5x_1x_2 \cdots (1 - x_{m-1})(1 + g(x_m)) \\ \vdots \\ f_{m-1}(x) = 0.5x_1(1 - x_2)(1 + g(x_m)) \\ f_m(x) = 0.5(1 - x_1)(1 + g(x_m)) \\ \text{where } g(x_m) = 100[|x_m| + \sum_{x_i \in x_m} ((x_i - 0.5)^2 - \cos(20\pi(x_i - 0.5)))] \\ \text{Domain } [0, 1]^n \end{array} \right. \quad (\text{A1})$$

DTLZ2

$$\left\{ \begin{array}{l} f_1(x) = \cos(x_1\pi/2) \cos(x_2\pi/2) \cdots \cos(x_{m-2}\pi/2) \cos(x_{m-1}\pi/2)(1 + g(x_m)) \\ f_2(x) = \cos(x_1\pi/2) \cos(x_2\pi/2) \cdots \cos(x_{m-2}\pi/2) \sin(x_{m-1}\pi/2)(1 + g(x_m)) \\ \vdots \\ f_{m-1}(x) = \cos(x_1\pi/2) \sin(x_2\pi/2)(1 + g(x_m)) \\ f_m(x) = \sin(x_1\pi/2)(1 + g(x_m)) \\ \text{where } g(x_m) = \sum_{x_i \in x_m} (x_i - 0.5)^2 \\ \text{Domain } [0, 1]^n \end{array} \right. \quad (\text{A2})$$

DTLZ3

$$\left\{ \begin{array}{l} f_1(x) = \cos(x_1\pi/2) \cos(x_2\pi/2) \cdots \cos(x_{m-2}\pi/2) \cos(x_{m-1}\pi/2)(1 + g(x_m)) \\ f_2(x) = \cos(x_1\pi/2) \cos(x_2\pi/2) \cdots \cos(x_{m-2}\pi/2) \sin(x_{m-1}\pi/2)(1 + g(x_m)) \\ \vdots \\ f_{m-1}(x) = \cos(x_1\pi/2) \sin(x_2\pi/2)(1 + g(x_m)) \\ f_m(x) = \sin(x_1\pi/2)(1 + g(x_m)) \\ \text{where } g(x_m) = 100[|x_m| + \sum_{x_i \in x_m} ((x_i - 0.5)^2 - \cos(20\pi(x_i - 0.5)))] \\ \text{Domain } [0, 1]^n \end{array} \right. \tag{A3}$$

DTLZ4

$$\left\{ \begin{array}{l} f_1(x) = \cos(x_1^\alpha\pi/2) \cos(x_2^\alpha\pi/2) \cdots \cos(x_{m-2}^\alpha\pi/2) \cos(x_{m-1}^\alpha\pi/2)(1 + g(x_m)) \\ f_2(x) = \cos(x_1^\alpha\pi/2) \cos(x_2^\alpha\pi/2) \cdots \cos(x_{m-2}^\alpha\pi/2) \sin(x_{m-1}^\alpha\pi/2)(1 + g(x_m)) \\ \vdots \\ f_{m-1}(x) = \cos(x_1^\alpha\pi/2) \sin(x_2^\alpha\pi/2)(1 + g(x_m)) \\ f_m(x) = \sin(x_1^\alpha\pi/2)(1 + g(x_m)) \\ \text{where } g(x_m) = \sum_{x_i \in x_m} (x_i - 0.5)^{2, \alpha} = 100 \\ \text{Domain } [0, 1]^n \end{array} \right. \tag{A4}$$

F1

$$\left\{ \begin{array}{l} f_1(x) = x_1x_2 \cdots x_{m-1} + \frac{2}{|J_1|} \sum_{j \in J_1} (x_j - x_2 \cdots x_{m-1} \sin(6\pi x_1 + \frac{j\pi}{n}))^2 \\ f_2(x) = x_1x_2 \cdots (1 - x_{m-1}) + \frac{2}{|J_2|} \sum_{j \in J_2} (x_j - x_2 \cdots x_{m-1} \sin(6\pi x_1 + \frac{j\pi}{n}))^2 \\ \vdots \\ f_{m-1}(x) = x_1(1 - x_2) + \frac{2}{|J_{m-1}|} \sum_{j \in J_{m-1}} (x_j - x_2 \cdots x_{m-1} \sin(6\pi x_1 + \frac{j\pi}{n}))^2 \\ f_m(x) = (1 - x_1) + \frac{2}{|J_m|} \sum_{j \in J_m} (x_j - x_2 \cdots x_{m-1} \sin(6\pi x_1 + \frac{j\pi}{n}))^2 \\ \text{where } J_1 = \{j|m \leq j \leq n, \text{ and } j \text{ is a multiplication of } m\}, \\ J_2 = \{j|m \leq j \leq n, \text{ and } j - 1 \text{ is a multiplication of } m\}, \\ \vdots \\ J_m = \{j|m \leq j \leq n, \text{ and } j - m + 1 \text{ is a multiplication of } m\}. \\ \text{Its PS is } x_j = x_2 \cdots x_{m-1} \sin(6\pi x_1 + \frac{j\pi}{n}), j = m, \cdots, n. \\ \text{Domain } [0, 1]^{m-1} \times [-1, 1]^{n-m+1} \end{array} \right. \tag{A5}$$

F2

$$\left\{ \begin{array}{l}
 f_1(x) = \cos\left(\frac{x_1\pi}{2}\right) \cdots \cos\left(\frac{x_{m-2}\pi}{2}\right) \cos\left(\frac{x_{m-1}\pi}{2}\right) + \frac{2}{|J_1|} \sum_{j \in J_1} (x_j - 2x_2 \cdots x_{m-1} \sin(2\pi x_1 + \frac{j\pi}{n}))^2 \\
 f_2(x) = \cos\left(\frac{x_1\pi}{2}\right) \cdots \cos\left(\frac{x_{m-2}\pi}{2}\right) \sin\left(\frac{x_{m-1}\pi}{2}\right) + \frac{2}{|J_2|} \sum_{j \in J_2} (x_j - 2x_2 \cdots x_{m-1} \sin(2\pi x_1 + \frac{j\pi}{n}))^2 \\
 \vdots \\
 f_{m-1}(x) = \cos(x_1\pi/2) \sin(x_2\pi/2) + \frac{2}{|J_{m-1}|} \sum_{j \in J_{m-1}} (x_j - 2x_2 \cdots x_{m-1} \sin(2\pi x_1 + \frac{j\pi}{n}))^2 \\
 f_m(x) = \sin(x_1\pi/2) + \frac{2}{|J_m|} \sum_{j \in J_m} (x_j - 2x_2 \cdots x_{m-1} \sin(2\pi x_1 + \frac{j\pi}{n}))^2 \\
 \text{where } J_1, J_2, \dots, J_m \text{ are the same as those of } F1. \\
 \text{Its PS is } x_j = 2x_2 \cdots x_{m-1} \sin(2\pi x_1 + \frac{j\pi}{n}), j = m, \dots, n. \\
 \text{Domain } [0, 1]^{m-1} \times [-2, 2]^{n-m+1}
 \end{array} \right. \quad (A6)$$

References

1. Fu, G.; Kapelan, Z.; Kasprzyk, J.R.; Reed, P. Optimal Design of Water Distribution Systems Using Many-Objective Visual Analytics. *J. Water Res. Plan. Manag.* **2012**, *139*, 624–633, doi:10.1061/(ASCE)WR.1943-5452.0000311.
2. Chikumbo, O.; Goodman, E.; Deb, K. Approximating a multi-dimensional pareto front for a land use management problem: A modified moea with an epigenetic silencing metaphor. In Proceedings of the 2012 IEEE Congress on Evolutionary Computation, Brisbane, Australia, 10–15 June 2012; pp. 1–9, doi:10.1109/CEC.2012.6256170.
3. Lygoe, R.J.; Cary, M.; Fleming, P.J. A Real-World Application of a Many-Objective Optimisation Complexity Reduction Process. In *Evolutionary Multi-Criterion Optimization*; Springer: Berlin/Heidelberg, Germany, 2013; pp. 641–655, doi:10.1007/978-3-642-37140-0_48.
4. Deb, K. *Multi-Objective Optimization Using Evolutionary Algorithms*; John Wiley & Sons: Hoboken, NJ, USA, 2001.
5. Miettinen, K. *Nonlinear Multiobjective Optimization*; Kluwer Academic Publishers: Boston, MA, USA, 1999; ISBN 978-1-4615-5563-6.
6. Ajith, A.; Lakhmi, J.; Robert, G. *Evolutionary Multiobjective Optimization*; Springer: London, UK, 2005.
7. Bai, J.; Liu, H. Multi-objective artificial bee algorithm based on decomposition by PBI method. *Appl. Intell.* **2016**, 1–16, doi:10.1007/s10489-016-0787-x.
8. Tan, Y.Y. Multi-Objective Evolutionary Algorithm Based on Decomposition and Its Applications. Ph.D. Thesis, Xidian University, Xi'an, China, 2013.
9. Zheng, X.W.; Lu, D.J.; Wang, X.G.; Liu, H. A cooperative coevolutionary biogeography-based optimizer. *Appl. Intell.* **2015**, *43*, 95–111, doi:10.1007/s10489-014-0627-9.
10. Tan, Y.Y.; Li, S.; Fang, X. MOEA/D with Adaptive IWO for Synthesizing Phase-Only Reconfigurable Linear Arrays. *Open Chem. Eng. J.* **2015**, *9*, 125–133, doi:10.2174/1874123101509010125.
11. Mei, Q.; Zhang, H.; Liang, C. A Discriminative Feature Extraction Approach for Tumor Classification Using Gene Expression Data. *Curr. Bioinform.* **2016**, *11*, 561–570, doi: 10.2174/1574893611666160728114747.
12. Liu, H.; Zhang, P.; Hu, B.; Moore, P. A novel approach to task assignment in a cooperative multi-agent design system. *Appl. Intell.* **2015**, *43*, 162–175, doi:10.1007/s10489-014-0640-z.
13. Tan, Y.Y.; Jiao, Y.C. MOEA/D with Uniform Design for Solving Multiobjective Knapsack Problems. *JCP* **2013**, *8*, 302–307, doi:10.4304/jcp.8.2.302-307.
14. Deb, K.; Pratap, A.; Agarwal, S.; Meyarivan, T.A.M.T. A fast and elitist multiobjective genetic algorithm: NSGA-II. *IEEE. Trans. Evol. Comput.* **2002**, *6*, 182–197, doi:10.1109/4235.996017.
15. Zitzler, E.; Laumanns, M.; Thiele, L. SPEA2: Improving the strength Pareto evolutionary algorithm. *Eurogen* **2001**, *3242*, 95–100.
16. Ishibuchi, H.; Tsukamoto, N.; Nojima, Y. Evolutionary many-objective optimization: A short review. *IEEE Congr. Evolut. Comput.* **2008**, 2419–2426, doi:10.1109/CEC.2008.4631121.
17. Giagkiozis, I.; Fleming, P.J. Methods for multi-objective optimization: An analysis. *Inf. Sci.* **2015**, *293*, 338–350, doi:10.1016/j.ins.2014.08.071.

18. Zhang, Q.; Li, H. MOEA/D: A multiobjective evolutionary algorithm based on decomposition. *IEEE. Trans. Evol. Comput.* **2007**, *11*, 712–731, doi:10.1109/TEVC.2007.892759.
19. Li, H.; Landa, S.D. Evolutionary Multi-objective Simulated Annealing with adaptive and competitive search direction. *IEEE. Trans. Evol. Comput.* **2008**, 3311–3318, doi:10.1109/CEC.2008.4631246.
20. Ke, L.; Zhang, Q.; Battiti, R. MOEA/D-ACO: A multiobjective evolutionary algorithm using decomposition and AntColony. *IEEE. Trans. Cybern.* **2013**, *43*, 1845–1859, doi:10.1109/TSMCB.2012.2231860.
21. Li, W.; Wang, L.; Jiang, Q.; Hei, X.; Wang, B. Multiobjective Cloud Particle Optimization Algorithm Based on Decomposition. *Algorithms* **2015**, *8*, 157–176, doi:10.3390/a8020157.
22. Storn, R.; Price, K. Differential Evolution—A Simple and Efficient Heuristic for global Optimization over Continuous Spaces. *J. Glob. Optim.* **1997**, *11*, 341–359, doi:10.1023/A:1008202821328.
23. Storn, R.; Price, K. Differential evolution: A simple and efficient adaptive scheme for global optimization over continuous spaces. 1995. Available online: <http://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.70.6764&rep=rep1&type=pdf> (accessed on 26 July 2017).
24. Li, H.; Zhang, Q. Multiobjective optimization problems with complicated Pareto sets, MOEA/D and NSGA-II. *IEEE. Trans. Evol. Comput.* **2009**, *13*, 284–302, doi:10.1109/TEVC.2008.925798.
25. Tan, Y.Y.; Jiao, Y.C.; Li, H.; Wang, X.K. MOEA/D+ uniform design: A new version of MOEA/D for optimization problems with many objectives. *Comput. Oper. Res.* **2013**, *40*, 1648–1660, doi:10.1016/j.cor.2012.01.001.
26. Qi, Y.; Ma, X.; Liu, F.; Jiao, L.; Sun, J.; Wu, J. MOEA/D with adaptive weight adjustment. *Evolut. Comput.* **2014**, *22*, 231–264, doi:10.1162/EVCO_a_00109.
27. Deb, K.; Jain, H. An evolutionary many-objective optimization algorithm using reference-point-based nondominated sorting approach, part I: Solving problems with box constraints. *IEEE. Trans. Evol. Comput.* **2014**, *18*, 577–601, doi:10.1109/TEVC.2013.2281535.
28. Li, K.; Deb, K.; Zhang, Q.; Kwong, S. An Evolutionary Many-Objective Optimization Algorithm Based on Dominance and Decomposition. *IEEE. Trans. Evol. Comput.* **2015**, *19*, 694–716, doi:10.1109/TEVC.2014.2373386.
29. Zhang, Q.; Liu, W.; Li, H. The performance of a new version of MOEA/D on CEC09 unconstrained MOP test instances. *IEEE Congr. Evolut. Comput.* **2009**, *1*, 203–208, doi:10.1109/CEC.2009.4982949.
30. Wang, Y.; Cai, Z.; Zhang, Q. Differential Evolution With Composite Trial Vector Generation Strategies and Control Parameters. *IEEE. Trans. Evol. Comput.* **2011**, *15*, 55–66, doi:10.1109/TEVC.2010.2087271.
31. Wang, L.; Zhang, Q.; Zhou, A.; Gong, M.; Jiao, L. Constrained Subproblems in a Decomposition-Based Multiobjective Evolutionary Algorithm. *IEEE. Trans. Evol. Comput.* **2016**, *20*, 475–480, doi:10.1109/TEVC.2015.2457616.
32. Gampferle, R.; Muller, S.D.; Koumoutsakos, P. A parameter study for differential evolution. *Adv. Intell. Syst. Fuzzy Syst. Evolut. Comput.* **2002**, *10*, 293–298.
33. Deb, K.; Thiele, L.; Laumanns, M.; Zitzler, E. *Scalable Test Problems for Evolutionary Multiobjective Optimization*; Springer Science: London, UK, 2005; pp. 105–145.
34. Tian, Y.; Cheng, R.; Zhang, X.; Jin, Y. PlatEMO: A MATLAB Platform for Evolutionary Multi-Objective Optimization. *arXiv* **2017**, arXiv:1701.00879.
35. Zitzler, E.; Thiele, L.; Laumanns, M.; Fonseca, C.M.; Da Fonseca, V.G. Performance assessment of multiobjective optimizers: An analysis and review. *IEEE. Trans. Evol. Comput.* **2003**, *7*, 117–132, doi:10.1109/TEVC.2003.810758.
36. Tan, Y.Y.; Jiao, Y.C.; Li, H.; Wang, X.K. A modification to MOEA/D-DE for multiobjective optimization problems with complicated Pareto sets. *Inf. Sci.* **2012**, *213*, 14–38, doi:10.1016/j.ins.2012.06.007.
37. Tan, Y.Y.; Jiao, Y.C.; Li, H.; Wang, X.K. MOEA/D-SQA: A multi-objective memetic algorithm based on decomposition. *Eng. Optim.* **2012**, *44*, 1095–1115, doi:10.1080/0305215X.2011.632008.

38. Veldhuizen, D.A.; Lamont van, G.B. Multiobjective evolutionary algorithm test suites. *ACM Symp. Appl. Comput.* **1999**, 351–357, doi:10.1145/298151.298382.
39. Sato, H.; Aguirre, H.E.; Tanaka, K. Controlling dominance area of solutions and its impact on the performance of MOEAs. In *International Conference on Evolutionary Multi-Criterion Optimization*; Springer: Berlin/Heidelberg, Germany, 2007; pp. 5–20, doi:10.1007/978-3-540-70928-2_5.



© 2017 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (<http://creativecommons.org/licenses/by/4.0/>).