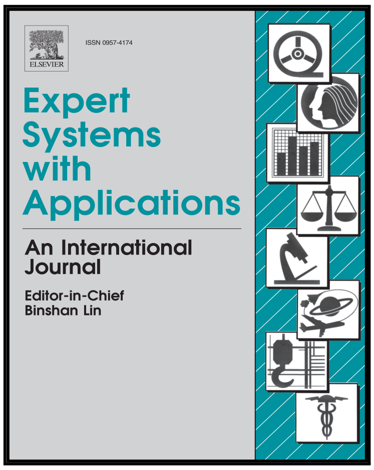


DCDG-EA: Dynamic Convergence-Diversity Guided Evolutionary Algorithm for Many-Objective Optimization

Zhiyong Li, Ke Lin, Mourad Nouioua, Shilong Jiang, Yu Gu

PII: S0957-4174(18)30602-X
DOI: https://doi.org/10.1016/j.eswa.2018.09.025
Reference: ESWA 12213

To appear in: *Expert Systems With Applications*

Received date: 27 November 2017
Revised date: 26 August 2018
Accepted date: 10 September 2018

Please cite this article as: Zhiyong Li, Ke Lin, Mourad Nouioua, Shilong Jiang, Yu Gu, DCDG-EA: Dynamic Convergence-Diversity Guided Evolutionary Algorithm for Many-Objective Optimization, *Expert Systems With Applications* (2018), doi: https://doi.org/10.1016/j.eswa.2018.09.025

Highlights

- DCDG-EA algorithm uses reference vector decomposition to solve MaOPs.
- CDOS selects an appropriate operator to generate offspring..
- CDIS strategy simultaneously considers the convergence and diversity of solutions..

DCDG-EA: Dynamic Convergence-Diversity Guided Evolutionary Algorithm for Many-Objective Optimization

Zhiyong Li[a], Ke Lin[a], Mourad Nouioua[a], Shilong Jiang[b], Yu Gu[c]

[a] College of Computer Science and Electronic Engineering of Hunan University, and Key Laboratory for Embedded and Network Computing of Hunan Province, Changsha, China
[b] PKU-HKUST Shenzhen-HongKong Institution, Shenzhen, China
[c] Beijing Advanced Innovation Center for Soft Matter Science and Engineering, Beijing University of Chemical Technology, Beijing 100029, China, and Institute for Inorganic and Analytical Chemistry, University of Frankfurt, Max-von-Laue-Str. 7, 60438 Frankfurt, Germany

Abstract

Maintaining a good balance between the convergence and the diversity is particularly crucial for the performance of the evolutionary algorithms (EAs). However, the traditional multi-objective evolutionary algorithms, which have shown their competitive performance with a variety of practical problems involving two or three objectives, face significant challenges in case of problems with more than three objectives, namely many-objective optimization problems (MaOPs). This paper proposes a dynamic convergence-diversity guided evolutionary algorithm, nammely (DCDG-EA) for MaOPs by employing the decomposition technique. Besides, the objective space of MaOPs is divided into K subspaces by a set of uniformly distributed reference vectors. Each subspace has its own subpopulation and evolves in parallel with the other subspaces. In DCDG-EA, the balance between the convergence and the diversity is achieved through the convergence-diversity based operator selection (CDOS) strategy and convergence-diversity based individual selection (CDIS) strategy. In CDOS, for each operator of the set of operators, a selection probability is assigned which is related to its convergence and diversity capabilities. Based on the attributed selection probabilities, an appropriate operator is selected to generate the offsprings. Furthermore, CDIS is used which allows to greatly overcome the inefficiency of the Pareto dominance approaches. It updates each subpopulation by using two independent distance measures that represent the convergence and the control diversity respectively. The experimental results on DTLZ and WFG benchmark problems with up to 15 objectives show that our algorithm is highly competitive comparing with the four state-of-the-art evolutionary algorithms in terms of convergence and diversity.

Keywords: Many-objective optimization, convergence, diversity, decomposition, evolutionary algorithm, Pareto optimality.

1. Introduction

Multi-objective optimization problems (MOPs) are characterized with multiple conflicting objectives. Problems with more than three objectives are defined as many-objective optimization problems (MaOP). They commonly exist in real-world applications, e.g., water allocation problems (Kasprzyk et al., 2009) and industrial scheduling problems (Dhall & Liu, 1978). Due to the conflicts among the different objectives, generally no single optimal solution exists to optimize all the objectives simulta-

Email addresses: zhiyong.li@hnu.edu.cn (Zhiyong Li), kelin_0808@hnu.edu.cn (Ke Lin), mouradnouioua@gmail.com (Mourad Nouioua), jiangshilong03@126.com (Shilong Jiang), guyu@mail.buct.edu.cn (Yu Gu)

neously. Without loss of generality, the MOP can be formulated as follows:

$$\begin{aligned} \min \quad & F(x) = (f_1(x), f_1(x), ..., f_m(x)) \\ subject \quad to \quad & x \in \prod_{i=1}^{D} [a_i, b_i], \end{aligned} \tag{1}$$

where D is the number of variable dimensions; m is the number of objectives; $\prod_{i=1}^{D}[a_i, b_i]$ is the decision space, $x = \{x | x = (x_1, x_2, ..., x_D)\}$, $a_j < x_j < b_j, j = 1, 2, ..., D$, is the decision variable vector, a_j and b_j are the constant bound constraints of x; $F(x)$ consists a set of m objective functions and is a mapping from D-dimensional decision space to m-dimensional objective space. When $M > 3$, this problem is a MaOP. Many-objective optimization has become a really hot research topic because of its wide applications and great importance.

For any two different solutions x, y of Formula (1), we say that x pareto dominates $y(x \prec y)$, if

$$\begin{aligned} (\forall i \in \{1, 2, ..., m\}: \quad & f_i(x) \leq f_i(y)) \wedge \\ (\exists i \in \{1, 2, ..., m\}: \quad & f_i(x) < f_i(y)), \end{aligned} \tag{2}$$

if there is no $x^{'} \in \prod_{i=1}^{D}[a_i, b_i]$ satisfying $x^{'} \prec x^*$, then x^* is the Pareto optimal solution. The set of all Pareto optimal solutions is called Pareto set (PS) and the set of all Pareto optimal objective vectors is called the Pareto front (PF) (Zhang & Li, 2007).

Over the last decades, multi-objective evolutionary algorithms (MOEAs) (Deb et al., 2002; Li et al., 2015a) have been successfully applied for MOPs. However, they noticeably deteriorate their search ability when solving MaOPs. The major reason is the exponentially increase of the number of non dominated solutions when the number of objectives increases. More precisely, Pareto-based MOEAs, such as NSGAII, fail to achieve the convergence (minimizing the distance of solutions toward the PF) and the diversity (maximizing the spread of solutions along the PF) requirements of MaOPs. Based on this results, Recently, a number of MOEAs have been especially designed for MaOPs. Globally speaking, the proposed MOEAs can be classified on four categories: convergence enhancement based approaches, performance indicator based approaches, objective reduction based approaches and decomposition-based approaches.

Achieving the convergence, maintaining the diversity and attaining a compromise between the convergence and the diversity are not easy tasks (Liu et al., 2017). Among the above methods, the decomposition based MOEAs are very promising techniques for the many-objective optimization problems. In fact, the decomposition based techniques can not only overcome the disadvantage of the indistinguishable solutions obtained in case of high-dimensional objective spaces. but also, it can maintain the population diversity on coarse granularity. Moreover, the decomposition based approaches have an acceptable computational effort comparing with the other methods. This observation has motivated us to propose a dynamic convergence-diversity guided evolutionary algorithm (DCDG-EA) which uses a reference vector decomposition to solve MaOPs. More precisely, a set of uniformly distributed reference vectors is used to divide the objective space into K subspaces. Each subspace has its own subpopulation and evolves simultaneously with the others. The balance between the convergence and the diversity is achieved by the convergence-diversity based operator selection (CDOS) strategy and the convergence-diversity based individual selection (CDIS) strategy. The main contributions of the presented work can be summarized as follows:

- We proposed a CDOS strategy which can achieve a good balance between the convergence and the diversity. The proposed CDOS strategy allows to overcome the inefficiency of the traditional genetic operators in the high dimensional spaces. Besides, a selection probability is assigned to each operator in a set of operators. Later, an appropriate operator is selected. The selected operator is the most conducive operator to generate offspring with good balance between the convergence and the diversity. To be more specific, the dynamically updated selection probability of an operator is related to its quality which is determined by its compromise reward (consisting of the convergence reward and the diversity reward). Moreover, the proportions of the two rewards are dynamically adjusted during the evolutionary process.

- The PS obtained by most of MOEAs is either not convergent enough or not sufficiently diversified. Our CDIS strategy simultaneously considers the convergence and diversity of the solutions to update each subpopulation. The CDIS strategy updates each subpopulation by using two independent distance measures (d_1 and d_2) that represent the convergence and the control diversity respectively. The proposed measures allows to overcome the inefficiency of the Pareto domination approaches and improve the selection pressure toward the PF. Different from other MOEAs, pre_d1, which records the smallest d_1 value of the solutions in the $(t-1)$-th generation, is the criterion used for dividing the t-th generation subpopulation into P_1 and P_2 set. The convergence of the solutions in P_1 are superior than those in P_2. The subpopulation is updated according to the convergence and the diversity of the individuals in P_1 and P_2 sets which allows to achieve a balance between the convergence and the diversity of the population.

The rest of this paper is organized as follows: Section 2 reviews the related works. Section 3 is devoted to describe the proposed DCDG-EA algorithm. Section 4 presents the experimental results with a detailed analysis. Finally, the conclusion and the future works are provided in Section 5.

2. Related Work

In this section, first, we briefly summarize the recent studies related to MaOPs. Then, we introduce the original MOEA/D (Zhang & Li, 2007) and NSGAIII (Deb & Jain, 2014a) which are the basis of our proposed algorithm. Finally, the widely used Das and Denniss systematic method to generate reference vectors (Das & Dennis, 2000) is presented.

2.1. Many-objective evolutionary optimization

The recently proposed MOEAs for solving MaOPs can be classified on four categories:

Convergence enhancement based approaches. The loss of the convergence pressure in most of traditional MOEAs occurs because of the failure of the traditional Pareto dominance approaches to distinguish the different solutions. Thus, the most intuitive idea of the convergence enhancement technique is to modify the domination relationship to increase the selection pressure on the PF. Based on this idea, a number of MOEAs such as grid-dominance (Yang et al., 2013), ε-dominance (Laumanns et al., 2002), rank-dominance (Kukkonen & Lampinen, 2007), k-optimality (Farina & Amato, 2004), fuzzy-based Pareto optimality (He et al., 2014) and L-optimality (Zou et al., 2008) have been proposed. These methods can enhance the selection pressure. However, one or more parameters have a significant effect on the performances of these methods.

Performance indicator based approaches. In these approaches, the notion of pareto dominance is not used and the selection pressure is enhanced using the well known performance indicators. The most representative performance indicator methods are: fast HV-based EA (HypE) (Bader & Zitzler, 2011), dominated hypervolume EA (Beume et al., 2007a), the ε-metric based evolutionary multi-objective algorithm (Basseur & Zitzler, 2006; LOpez et al., 2013), stochastic ranking EA (Li et al., 2016) and indicator-based EA (Zitzler & Knzli, 2004). Although the selection pressure is enhanced, this kind of methods requires high computational complexity.

Objective reduction based approaches. These algorithms are based on the application of certain dimensionality reduction techniques such as principal component analysis (PCA) (Saxena et al., 2013) and unsupervised feature selection (Pez Jaimes et al., 2008) to deal with the hardness of the MaOPs. If the number of objectives is reduced to two or three, the traditional MOEAs can be used to solve the problem. However, this approach may lead to lose objective information and it is only effective for MaOPs with redundant objectives.

The decomposition-based approaches. Generally, these approaches include two techniques, namely weight aggregation-based techniques and reference vector-based techniques. the weight aggregation-based MOEAs decompose a MOP into a set of scalar optimization subproblems by a set of uniformly distributed weight vectors and the objectives of the MOP (MaOP) are aggregated into a scalar. The most representative MOEAs based on this concept are (Zhang & Li, 2007) and MSOPS (Hughes, 2004), which are considered as important approaches for solving MaOPs. On the contrary with the aggregation based MOEAs, the reference vector-based MOEAs are independent from aggregate functions or explicit single-objective subproblems. They directly divide the objective space into subspaces by a set of evenly distributed reference vectors and search on a muliple objective subspaces. MOEA/D-M2M (Liu et al., 2014) and NSGAIII (Deb & Jain, 2014a) are decomposition-based approaches.

2.2. MOEA/D

The key idea of MOEA/D is to decompose the MOP into a number of single-objective optimization subproblems through the aggregation functions and it optimizes them simultaneously. Since the optimal solution of each subproblem is a Pareto optimal solution for the given MOP, the collection of the optimal solutions can be treated as a good PF approximation. Generally, three aggregation functions, Penalty Boundary Intersection (PBI), weight sum function and Tchebycheff function can be used to meet the purpose of MOEA/D. Moreover, the mating parents are usually selected from some neighboring weight vectors and a crossover and mutation over the selected parents are applied to generate an offspring. Offspring replaces its parent only if it has a better aggregation function value.

2.3. NSGAIII

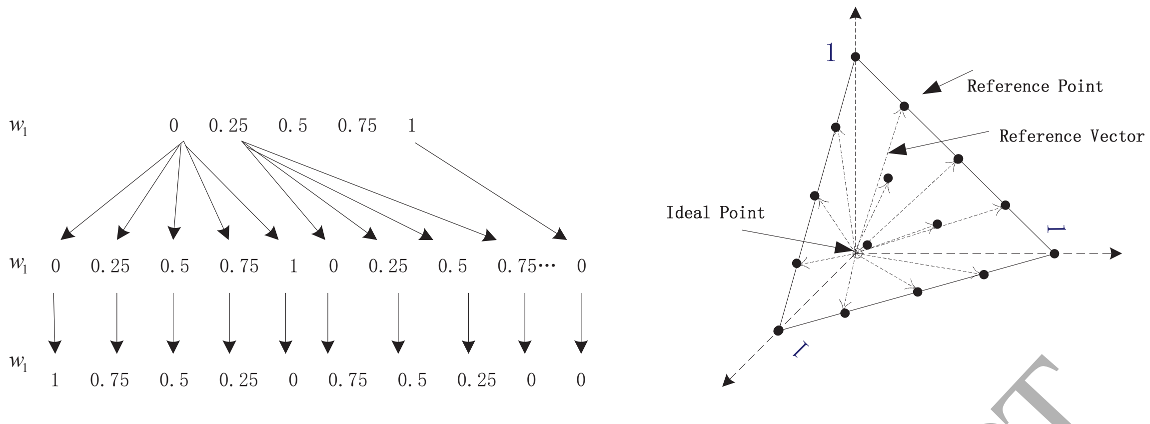
The basic framework of NSGAIII remains similar to its predecessor NSGAII. Unlike MOEA/D, the aggregation function is not used in NSGAIII. NSGAIII decomposes the objective space into set of subspaces through a set of evenly spread reference vectors. Each solution is associated with a reference vector based on its perpendicular distance to the reference line. The new generated offspring solutions are combined with their parents to form a hybrid population. Then, the fast nondominated sorting is used to divide the hybrid population into several nondomination levels. Environmental selection process is similar to NSGAII. However, the solutions in the last acceptable level are selected based on a niche-preservation operator. More precisely, the solutions associated with the less crowded reference vectors have a great chance to be selected.

2.4. Reference Vectors Generation

The Das and Denniss systematic method (Das & Dennis, 2000) to generate a set of uniformly distributed reference vectors $W = \{w^1, w^2, ..., w^K\}$, which decompose the objective space into K subspaces is also used in our algorithm. Besides, K points on the hyperplane with a uniform spacing of $\delta = 1/H$ are generated, where $H(H > 0)$ is the number of divisions in each objective coordinate. The size of the reference points (K) is given by

$$K = C_{H+m-1}^{m-1}, \tag{3}$$

For instance, in a three-objective problem $(m = 3)$, the reference points are created on a triangle with the apex at (1, 0, 0), (0, 1, 0) and (0, 0, 1). If four divisions $H = 4$ $(\delta = {}^1/_H = 0.25)$ are selected for each objective axis, $K = C_{4+3-1}^{3-1}$ or 15 reference points will be created. Fig. 1(a) provides the process of a simple example using the Das and Denniss reference vector generation method. Furthermore, Fig. 1(b) presents the distribution of the reference points.



(a) Reference points are generated recursively

(b) Set of reference points in a normalized hyper-plan where $m = 3$ and $H = 4$

Figure 1: The Das and Denniss reference vector generation method

3. Proposed Algorithm

3.1. Basic Idea and Framework of Proposed Algorithm

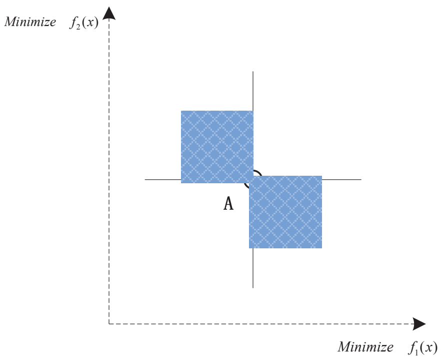
Normally, for a solution A, there are about $(1-(1/2^{m-1}))$ percent of non-dominated solutions with A for $m-$objective problem space (XIAO et al. (2014)). Fig. 2 gives simple examples for two-objective and three-objective problems respectively. In other words, as the number of objectives increases, almost all solutions in the current population become non-dominated with each others in the early phases of the evolution. Thus, the Pareto dominance relation fails to distinguish the different solutions and it greatly weaken the selection pressure toward PF. Moreover, it is worth noticing that the maintenance of the population diversity is particularly important and nontrivial on solving the high dimension optimization problems.

The reference vector-based decomposition techniques can robustly maintain the population diversity and reduce the complexity of the problem simultaneously. Based on this motivation, we propose a convergence-diversity guided evolutionary algorithm, namely DCDG-EA, that uses a reference vector-based decomposition technique to solve MaOPs. Besides, the population of MaOPs is divided into several subpopulations by a set of uniformly distributed reference vectors and each subpopulation evolves independently in parallel with the others.

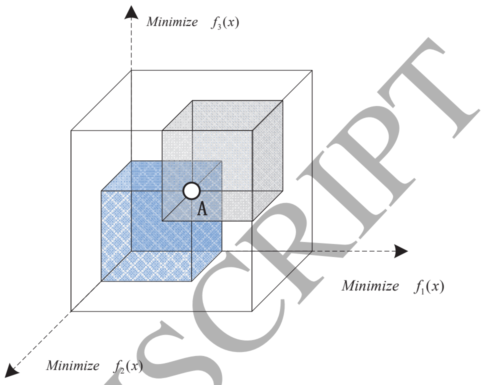
Fig. 3 shows that the Pareto front of MaOPs is also divided into K small segments, each solution is assigned to a reference vector. Our goal is to find the intersection points between the PF and the set of the reference vectors, these points provide a good approximation to the entire PF. When three solutions are selected from these six solutions, B, C, and E can be selected according to the Pareto dominance before decomposition. However, A, B, and E can be selected after the decomposition. simply, if a solution located in an isolated subspace but is dominated by solutions in other subpopulations, this solution must be retained to maintain the diversity of the population. This approach improves the diversity of population in coarse grain size.

In order to achieve a good balance between the convergence and the diversity, we propose an CDIS strategy. Besides, two distance metrics, the projected distance d_1 of a solution along the reference vector is used to evaluate its convergence toward PF and the perpendicular distance d_2 from a solution to a reference vector is used to control diversity respectively (Fig. 3(b)). As illustrated in Fig. 4, pre_d1 records the smallest d_1 value of solutions in the $(t-1)$-th generation subpopulation and it divides the t-th generation subpopulation into P_1 and P_2 set. Obviously, the convergence of solutions in P_1 are superior than the others of P_2. Thus, the solutions in P_1 are preferred to be selected according to their convergence and diversity performance. However, only convergence of solutions is considered in P_2. Thus, the balance between the convergence and the diversity will be achieved.

The traditional genetic operator may become extremely disruptive on solving the high-dimensional objective space and it may have inefficient performance. Simulated Binary Crossover (SBX) (Deb & Agrawal, 1994), Crossover and Mutation of Differential Evolution (DE) (Storn & Price, 1997), are often suitable to problems with different PF shapes, different stages of the optimization process, even obtain a population with different degree of convergence and diversity. Thus, we propose a CDOS strategy that selects an appropriate operator from a set of operators based on their selection probability to produce offspring with good balance between the convergence and the diversity. CDOS will be described in detail later.

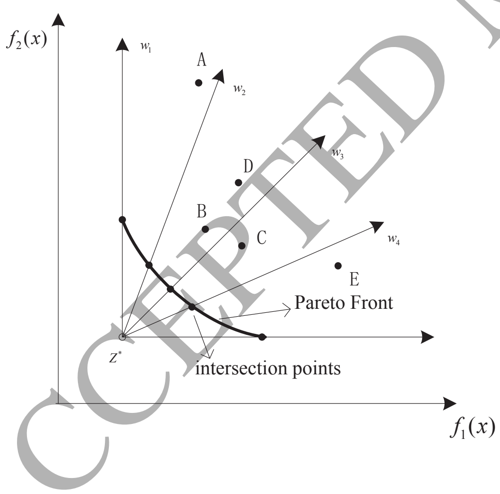


(a) Solutions in the shadow region, which account for about 1/2 of the population, are non-dominated by A in the two-objective problem.

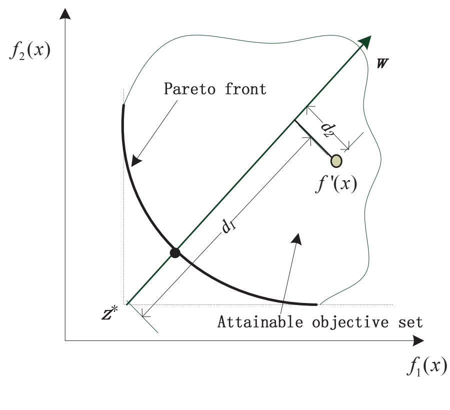


(b) Solutions in the blank region, which account for about 3/4 of the population, are non-dominated by A in the three-objective problem.

Figure 2: Ratio of non-dominated solutions



(a) The population decomposition strengthens the diversity of population in coarse grain size.



(b) Illustration of the two distances d_1 and d_2

Figure 3:

The framework of our proposed DCDG-EA algorithm is shown in Algorithm 1 and it consists of five components: 1) population initialization and distributed reference vectors generation, 2) objective space or population decomposition, 3) calculate distances along and perpendicular to each reference direction, 4) use CDOS strategy to select an operator for offspring generation and 5) update every subpopulation by CDIS strategy.

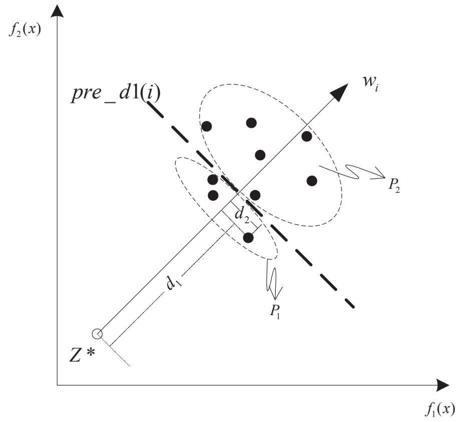


Figure 4: The subpopulation $SubPop\{i\}$ is divided into P_1 and P_2 by $pre\_d1(i)$

Algorithm 1 Framework of DCDG-EA

Require:

1) maximum number of generation $t_{\max}$;
2) the set of reference vectors $W = \{w^1, w^2, ..., w^K\}$;
3) the number of objectives m;
4) the population size N, $n = N/K$ is the size of each subpopulation;
5) Parameters in CDOS: the qualities of operators $Q = \{q_{1,t}, q_{2,t}, ..., q_{l,t}\}$, the adaptation rate ω, learning factor δ, minimum selection possibility of every operator $P_{\min}$.

Ensure:

population $Pop \leftarrow \bigcup_{i=1}^{K} SubPop(i)$

```
Initialize the population Pop
SubPop ← decomposition(Pop, W)
while termination condition is not fulfilled do
    newPop ← ∅
    for i ← 1 to K do
        compute the distances d_1 and d_2 of each solution with respect to its reference direction in SubPop{i} according to Formulas (7) and (8)
    end for
    if t == 1 then
        select operator o_j from the set of operators according to Formula (15) by roulette selection
    else
        select operator o_j from the set of operators by CDOS(Pop, newPop, t, O, Q, ω, δ)
    end if
    for i ← 1 to K do
        newSubpop{i} ← o_j.operator(SubPop{i})
        newPop ← newPop ∪ newSubpop{i}
    end for
    mergePop ← Pop ∪ newPop
    SubPop ← decomposition(mergePop, W)
    SubPop ← use our proposed CDIS to update every subpopulation
end while
return Pop ← ⋃_{i=1}^{K} SubPop(i)
```

3.2. Objective Space Decomposition

Reference vectors Generation is presented in Section 2. These K reference vectors are evenly distributed in the objective space and can explicitly divide the objective space into K subspaces $\{\Omega^1, \Omega^2, ..., \Omega^K\}$. $w^i = (w_1^i, w_2^i, ..., w_m^i)^T, i = \{1, 2, ..., K\}$ is a unit vector along any given reference direction, each w^i specifies a unique subregion in the objective space, denoted as Ω^i and is defined by Formula (4):

$$\Omega^i = \{f'(x) \in R_+^m| < f'(x), w^i > \leq < f'(x), w^j >\} \tag{4}$$

where

$$f'(x) = f(x) - Z^*, \tag{5}$$

$j = \{1, ..., K\}$, $Z^* = (f_1^{\min}, ..., f_m^{\min})$, $< F(x), w^i >$ is the angle between $F(x)$ and w^i, Formula (5) denotes the objective functions translation. In other words, $F(x)$ is in Ω^i if and only if w^i has the smallest angle to $F(x)$ among all the K reference vectors. Based on this strategy, the population is divided into K subpopulations $(P_1, P_2, ..., P_K)$. The cosine value of the acute angel θ between the solution x and reference vector w can be calculated through Formula (6):

$$\cos\theta = \frac{w \cdot f'(x)}{||f'(x)||}, \tag{6}$$

where $|| \cdot ||$ calculates the norm, i.e., the length of the $f'(x)$. The pseudocode of objective space decomposition is given in Algorithm 2. It differs from (Liu et al., 2014) in the fact that $nicheCount$ (the number of solutions per subpopulation) is used to measure the discrepancy of the entire population in CDIS strategy which ensures further the diversity of the population.

Algorithm 2 Objective space decomposition: $decomposition(Pop, W)$

Require:
  1) parent population Pop;
  2) the set of reference vectors $W = \{w^1, w^2, ..., w^K\}$.
Ensure:
  $SubPop$

```
1: objective functions translation of all the solution objectives
2: for x ∈ Pop do
3:     for w ∈ W do
4:         calculate the cosine value: cos Value ← cos(x, w) according to (6)
5:     end for
6:     index(x) ← w : arg max_{w∈W} {cos value}
7: end for
8: According to index divide Pop to K SubPops and compute nicheCount (the number of solutions of
   each subpopulation)
9: return SubPop
```

3.3. Distances Calculation

As shown in Fig. 3(b), the convergence and the diversity of a solution with respect to a given reference vector can be judged using two measures d_1 and d_2 calculated by Formulas (7) and (8) respectively:

$$d_1(x) = w^T f'(x) \tag{7}$$

and

$$d_2(x) = \sqrt{||f'(x)||^2 - (w^T f'(x))^2}, \tag{8}$$

where w is a unit vector along any given reference direction. A smaller value of projected distance d_1 along the reference vector indicates superior convergence. Geometrically, DCDG-EA aims to find the intersection points between the PF and the reference vectors. Since the set of reference vectors are evenly distributed in the objective space, one can expect that the resultant intersection points provide a good approximation to the whole PF. It is clear that $d_2 = 0$ means the solution is perfectly aligned along the reference direction associated with it thereby ensuring a perfect diversity. Thus, we use these two measures in CDIS strategy to manage the convergence and the diversity of the population.

3.4. Convergence-diversity Based Operator Selection Strategy

In our proposed CDOS strategy, population at generation t is generated by an appropriate operator $o_{j,t}$, which is selected from a set of operators $O = \{o_1, o_2, ..., o_l\}$ based on their selection probabilities $P = \{p_{1,t}, p_{2,t}, ..., p_{l,t}\}$, where l is the number of operators. The selection probabilities are related to the operators' quality $Q = \{q_{1,t}, q_{2,t}, ..., q_{l,t}\}$, which are decided by the compromise reward $ComR$ of the operators. In this paper, the roulette selection method is adopted for calculating the selection probability of the operators.

For each population generated by $o_{j,t}$ at iteration t, the compromise reward $ComR$ of the operator at iteration t consists of two components, namely, diversity reward and convergence reward. Specifically, the convergence reward R_{con} of the operator is expressed by the ability of the entire population to approach the PF and is provided in Formula (9). The diversity reward R_{div} of the operator is measured by the distribution density of each subpopulation in the entire population and is provided in Formula (10). Moreover, the compromise reward $ComR$ of the operator calculated by Formula (11) is employed to update the probability $P = \{p_{1,t}, p_{2,t}, ..., p_{l,t}\}$ of the group of operators, the appropriate operator o_j is then selected for the next generation. This operator is conducive to the balance between the convergence and diversity of the population.

$$R_{con}(o_j, t) = \frac{(\frac{1}{n}\sum_{s=1}^{n} d_1(x_{s,t-1}^{P})) - (\frac{1}{n}\sum_{s=1}^{n} d_1(x_{s,t}^{o_j}))}{\frac{1}{n}\sum_{s=1}^{n} d_1(x_{s,t-1}^{P})}, \tag{9}$$

$$R_{div}(o_j, t) = \frac{\sigma_{nicheCount,t} - \sigma_{nicheCount,t-1}^{o_j}}{\sigma_{nicheCount,t}}, \tag{10}$$

and

$$ComR(o_j, t) = (1-\alpha) \times R_{con}(o_j, t) + \alpha \times R_{div}(o_j, t), \tag{11}$$

where

$$\sigma_{nicheCount,t} = \frac{\sum_{i=1}^{K} (nicheCount_{i,t} - \frac{1}{K}\sum_{i=1}^{K} nicheCount_{i,t})^2}{K} \tag{12}$$

$$\alpha = \frac{1}{1 + e^{-(10\times t/t_{\max}-5)}}, \tag{13}$$

$x_{s,t}^{o_j}$ denotes the solution s generated by operator $o_{j,t}$ at iteration t, and $x_{s,t-1}^{P}$ denotes its parent. As previously mentioned, d_1 indicates the convergence of solution toward the PF. When the d_1 values of all the solutions at iteration t decrease, the convergence of the entire population enhances. The convergence reward R_{con} of the operator o_j also increases. Moreover, $nicheCount_{i,t}$ denotes the number of solutions in subpopulation i at iteration t and we use $\sigma_{nicheCount,t}$ which can measure the degree of dispersion of the solutions in the entire population, to denote the standard deviation of the number of solutions in the K subpopulations. The smaller the $\sigma_{nicheCount,t}$, the more dispersed the entire population in each subpopulation, thereby indicating that the diversity is improved. On the contrary, the larger the $\sigma_{nicheCount,t}$, the greater the indication that the solution may be concentrated in one or several subpopulations. Thus, when the population distribution becomes increasingly uniform, the diversity reward R_{div} of the operator o_j increase.

The compromise reward $ComR$ considers the diversity and the convergence of the population. It dynamically adjusts the proportion of their importances by α according to the evolutionary stages of the population. In reality, for MaOPs, the solutions are dispersed in a high dimensional objective space at the early stages of the evolution. The selection pressure for the convergence plays a major role in population evolution. However, in the later stages of population evolution, the solutions are extremely close to the Pareto front. In this case, the population diversity should be emphasized to produce a well-distributed set of solutions. Furthermore, the parameter α should be small at the early stages of evolution and it gradually increases with the evolution of the population. (Formula (13) is translated from the Sigmoid function to the right by 5 units. A sigmoid function is constrained by a pair of horizontal asymptotes as $x \to \pm\infty$ and its range is (0,1). When the decision variable is close to 5, the function value will be close to 1).

Certain probability allocation methods show a good performance for the single objective optimization problems. Fortunately, these methods can easily extended to the multi-objective optimization. Thus, in this paper, we directly use Adaptive Pursuit (AP) proposed in (Thierens, 2005) with the same parameters selected in the original work. The quality update rule for an operator is calculated by Formula (14) and the selection probability $P_{j,t+1}$ is updated according to Formula (15),

$$q_{j,t+1} = (1-\omega) \times q_{j,t} + \omega \times ComR(o_j, t) \tag{14}$$

and

$$p_{j,t+1} = \begin{cases} p_{j,t} + \delta \times (p_{\max} - p_{j,t}) & if o_j = \arg\max_{o_j \in O} q_{j,t} \\ p_{j,t} + \delta \times (p_{\min} - p_{j,t}) & otherwise, \end{cases} \tag{15}$$

where adaptation rate $\omega \in [0, 1]$ controls the memory of the operator's quality, $\delta \in (0, 1)$ is the learning factor. $P_{\min} \in (0, 1)$ for each selection probability that ensures that no operator is lost. Thus the maximum probability is $P_{\max} = 1 - (l-1) \times P_{\min}$. The pseudocode of CDOS is provided in Algorithm 3.

Algorithm 3 $CDOS(Pop, newPop, t, O, Q, \omega, \delta)$

Require:

1) parent population Pop;
2) offspring $newPop$;
3) current generation t;
4) the set of operators O;
5) the qualities of operators $Q = \{q_{1,t}, q_{2,t}, ..., q_{l,t}\}$
6) adaptation rate ω;
7) learning factor δ;

Ensure:

the selected operator o_j

```
for each operator o_j ∈ O do
    R_con ← calculate the convergence reward of o_j according to Formula (9)
    R_div ← calculate the diversity reward of o_j according to Formula (10)
    calculate the combined reward of o_j according to Formula (11)
    update the qualities of operators according to Formula (14)
    update the selection possibility of o_j by Formula (15)
end for
select operator o_j from the set of operators according to Formula (15) by roulette selection
return o_j
```

CDOS differs from (Hadka & Reed, 2013) firstly on how the probabilities are updated. Our feedback loop updates the probabilities by the convergence and diversity rewards of operators. whereas, the probabilities of operators in (Hadka & Reed, 2013) updated by counting the number of solutions produced by each operator in the $\varepsilon-$ box dominance archive. In particular, CDOS uses both the speed of the population toward the PF and the degree of dispersion of the population in each subpopulation to update the selection probabilities which favors the operator producing offspring with better convergence and diversity.

3.5. Convergence-diversity Based Individual Selection Strategy (CDIS)

The major purpose of CDIS is the selection of a fixed number of solutions for each subpopulation for the next generation. The PF of MaOPs is divided into K small segments by a set of reference vectors. The intersection points between the PF and the set of reference vectors can provide a good approximation to the entire PF. Thus, our goal is to push the objective vectors of the solutions as far as possible such that the algorithm can reach the boundaries of the feasible objective space.

With the evolution of the population, the solutions are dispersed in the objective space and they quickly approache the PF. d_1 value of solutions at the t-th generation should be smaller than the $(t-1)$-th generation. We use $pre\_d1(i), (i = 1, ...K)$ to record the smallest d_1 value of the solutions in the $(t-1)$-th generation subpopulation $SubPop\{i\}$. $pre\_d1(i)$ is initialized by the average d_1 value of $SubPop\{i\}$. As illustrated in Fig. 4, solutions in the t-th generation subpopulation $SubPop\{i\}$ are divided into P_1 and P_2 set according to $pre\_d1(i)$. If the projected distance $d_1 < pre\_d1(i)$, the solution is placed on the set P_1. Otherwise, it is placed on P_2 set. In this case, the convergence of the solutions in P_1 is more superior than solutions in P_2. Thus, the solutions in P_1 are preferred to be selected. In order to achieve the balance between the convergence and the diversity, the combined distance $comd = d_1 + \theta \times d_2$, is used to select solutions from P_1 (θ is the penalty parameter). Firstly, the solutions with small combined distances are preferred. By contrast, in the later stages of evolution, the solutions are very close to PF. Thus, the difference pre_d1 values between the t-th generation and the $(t+1)$-th generation population is particularly small and the solution will most likely be divided into P_2 set. Solutions in P_2 have poor convergence and select a solution with small d_1 value instead of the combination distances will speed up convergence speed and will achieve the balance between the convergence and the diversity. Therefore, CDIS weighs well the convergence and the diversity of the solutions. The pseudo code of CDIS is illustrated in Algorithm 4.

1) If $|P_1| > N/K$ (N/K is the size of the subpopulation), we sort the solutions of P_1 by the combined distance $comd$ in ascending order, and select the top N/K solutions as the next generation subpopulation;

2) If $|P_1| = N/K$, we output P_1 as the next subpopulation;

3) If $|P_1| < N/K$, we sort solutions of P_2 by d_1 in ascending order, top $(N/K - |P_1|)$ of P_2 and solutions in P_1 are selected to the next subpopulation.

3.6. Algorithm Complexity Analysis

Assuming that the size of population is N, the number of reference vectors is K (hence the size of each subpopulation is $n = N/K$), the number of decision variables is D and the number of operators is l. The time complexity of one generation of DCDG-EA algorithm is analyzed as follows:

Step 1: The time complexity of population initialization is $O(ND)$.

Step 2: The decomposition of the initialized population requires $O(K^2n)$ computation (line 2 of Algorithm 1).

Step 3: Judging the stopping criteria, the time complexity is $O(1)$.

Step 4: The distances d_1 and d_2 of all solutions with respect to their reference vectors in each subpopulation require $O(nK)$ computation (lines 5 to 7 of Algorithm 1).

Step 5: In CDOS, we select an appropriate operator from l operators based on their selection possibilities, update the compromise reward $ComR$, the probability $P = \{p_{1,t}, p_{2,t}, ...p_{l,t}\}$ and quality $Q = \{q_{1,t}, q_{2,t}, ..., q_{l,t}\}$ require $O(l)$ computation (Algorithm 3).

Step 6: The operator is performed on each decision variable of the parent solutions to generate N offspring, which need $O(ND)$ (lines 13 to 16 of Algorithm 1).

Step 7: Decomposition of merged population requires $O(K^2n)$ computation (line 18 of Algorithm 1).

Algorithm 4 $CDIS(SubPop, pre\_d_1, n)$

Require:
1) sub populations $SubPop$;
2) pre_d1;
3) the size of each subpopulation $n = N/K$.

Ensure:
$SubPop$ and pre_d1

$newSubpop\{i\} \leftarrow \emptyset, i = \{1, 2, ..., K\}$
for $i = 1; i < K; i++$ **do**
  if $|SubPop\{i\}| < n$ **then**
    randomly select $n - |SubPop\{i\}|$ solutions from the entire population add to $SubPop\{i\}$
  else
    sort the d_1 distance of all the solutions in $SubPop\{i\}$ in ascending order
    for each $x \in SubPop\{i\}$ **do**
      if $d_1(x) < pre\_d1$ **then**
        put x into P_1
      else
        put x into P_2
      end if
      if $\min(d_1(x)_{x \in SubPop\{i\}}) < pre\_d1(i)$ **then**
        $pre\_d1(i) \leftarrow \min(d_1(x)_{x \in SubPop\{i\}})$
      end if
    end for
    if $|P_1| == n$ **then**
      $newSubpop\{i\} \leftarrow P_1$
    else if $|P_1| < n$ **then**
      sort the d_1 distance of all the solutions in P_2 by an ascending sort order and $topP_2$ is the top of $n - |P_1|$ solutions in P_2
      $newSubpop\{i\} \leftarrow |P_1| \cup topP_2$
    else
      calculate the combined distance $comd \leftarrow d_1 + \theta \times d_2$ of every solution in P_1 and $topP_1$ is the top n solutions in P_1
      $newSubpop\{i\} \leftarrow topP_1$
    end if
  end if
end for
$Subpop \leftarrow newSubpop$
return $SubPop$ and pre_d1

Step 8: CDIS sorts the d_1 distance of all the solutions in each subpopulation and divides them into P_1 and P_2. If all solutions are in the same subpopulation, $O(NlogN)$ is needed in the best case; otherwise $O(Kn^2)$ in the worst case (Algorithm 4).

Thus, for K subpopulations, the complexity of one generation of DCDG-EA is $O(K^2n + Kn^2)$ in the worst case and $O(K^2n + NlogN)$ in the best case. In particular, when the number of reference vectors is equal to population size, the complexity of one generation of DCDG-EA is $O(N^2)$. Additionally, the convergence analysis of the algorithm can refer to some other studies (Nguyen et al., 2014; Li et al., 2015b).

4. Experimental Setup

This section presents the experiment results of the proposed DCDG-EA algorithm. First, we describe the benchmark problems and the performance metrics used in our experiments. Then, we briefly introduce several state-of-the-art EMO algorithms (NSGAIII Deb & Jain (2014a), GREA (Yang et al., 2013), MOEA/D (Zhang & Li, 2007), and MOEA/D-M2M (Liu et al., 2014)), which are used in comparison in order to verify the performance of the proposed DCDG-EA algorithm. The general parameter settings for the comparative studies of these algorithms are also presented. Finally, the experimental results with the analysis are provided.

4.1. Benchmark Problems and Performance Metrics

The well-known DTLZ and WFG test suite for many-objective optimization are considered in our experimental studies. For each test instance, the number of objectives varies from 3 to 15, i.e., $m \in \{3, 5, 8, 10, 15\}$. All these problems can be scaled to any number of objectives and decision variables. As suggested in (Deb et al., 2005), the number of decision variables is set as $D{=}m+k-1$, and $k = 5$ for DTLZ1, $k = 10$ for DTLZ2, DTLZ3 and DTLZ4. As suggested in Huband et al. (2006a), the number of decision variables is set as $D = k + r$, where the position-related variable $k = m - 1$ and the distance-related variable $r = 20$ for WFG test instances. The characteristics of all test instances are summarized in Table 1.

Table 1: Characteristics of test instances

Test instance	Characteristics
DTLZ1	Linear, multimodal
DTLZ2	Concave
DTLZ3	Concave, multimodal
DTLZ4	Concave, biased
WFG1	Mixed, biased
WFG2	Convex, disconnected, multi-modal, non-separable
WFG3	linear, degenerate, non-separable
WFG4	Concave, multi-modal
WFG5	Concave, deceptive
WFG6	Concave, non-separable
WFG7	Concave, biased
WFG8	Concave, biased, non-separable
WFG9	Concave, biased, multi-modal, deceptive, non-separable

Table 2: setting of reference points for HV computation

Test instance	Reference point
DTLZ1	$(1.0, ..., 1.0)^T$
DTLZ2 to DTLZ4	$(2.0, ..., 2.0)^T$
WFG1-WFG9	$(3.0, ..., 2.0 \times m + 1.0)^T$

Inverted generational distance (IGD) (Bosman & Thierens, 2003) and hypervolume (HV) (Zitzler & Thiele, 1999) are used in our paper to compare the performance of the different compared algorithms.

IGD metric (Bosman & Thierens, 2003): IGD measures the average distance from a set of reference points P^* in the PF to the approximation set P. In this paper, P^* is composed by uniformly sampling 10000 points which takes into account the reference vectors over the true PF. The IGD value of S is computed as

$$IGD(S, P^*) = \frac{\sum_{x^* \in P^*} dist(x^*, S)}{|P^*|}, \tag{16}$$

where $dist(x^*, S)$ denotes the Euclidean distance between the solution $x^* \in P^*$ and its nearest neighbor in S, and $|P^*|$ is the cardinality of P^*. $IGD(S, P^*)$ can measure both the convergence and the diversity of S. The lower the IGD value is, the better the quality of S is for the approximation of the entire PF.

HV metric (Zitzler & Thiele, 1999) can simultaneously measure the convergence and diversity of the obtained solutions. We let $Z^r = (z_1^r, z_2^r, ..., z_m^r)^T$ be a reference point in the objective space that is dominated by all Pareto-optimal objective vectors. HV metric measures the size of the objective space dominated by the solutions in S and bounded by Z^r

$$HV(S) = VOL(\bigcup_{x \in S} [f_1(x), z_1^r] \times \cdots [f_m(x), z_m^r]), \tag{17}$$

where $VOL(\cdot)$ indicates the Lebesgue measure. The larger the HV value is, the better the quality of S is for the approximation of the entire PF.

For problems with less than eight objectives, the exact hypervolume was computed by the recently proposed WFG algorithm (While et al., 2012). Additionally, when the number of objectives exceeds 8, we use the Monte Carlo approach (Beume et al., 2007b) to estimate the hypervolume through the calculation of the percentage of 1,000,000 random points in the objective space to be dominated by the PF. Table 2 shows the setting of reference points in our experiment.

4.2. *EMO Algorithms for Comparisons*

We consider four state-of-the-art EMO algorithms: NSGAIII, GrEA, MOEA/D and MOEA/D-M2M for comparisons [1]. Since the NSGAIII and MOEA/D algorithms have already been introduced in Section 2, only the GrEA and MOEA/D M2M algorithms are briefly introduced here.

1) GrEA combines two new concepts (grid dominance and grid difference), three types of indicators (grid ranking, grid crowding distance and grid coordinate point distance) and a fitness adjustment strategy.

2) MOEA/D-M2M decomposes a MOP into a set of simple multi-objective optimization subproblems via a set of evenly distributed reference vectors that each of them can specify a unique subspace. Each objective subspace allocates a subpopulation that contains a certain number of solutions to solve the corresponding MOP. solution competition in the population is achieved by the Pareto non-dominated sorting method.

4.3. *General Parameter Settings*

The five EMO algorithms considered in this paper have several parameters that are summarized as follows:

1) Crossover mutation parameters of the SBX operator are shown in Table 3. Sato et al. (2011) suggests that only a small number of genes can be exchanged between two parent solutions during offspring generation. Therefore, the crossover probability CR of DE are in the range of $(0, 0.1]$ and 0.1 in our paper.

2) The population size N and the number of reference vectors K for different number of objectives are given in Table 4. For instances with more than 8 objectives, the reference points are generated via a two-layer sampling scheme with two values of H i.e., one for each layer as outlined in (Deb & Jain, 2014b) (Table 4).

3) Each algorithm runs independently 30 times on each test instance.

4) Neighborhood size T in MOEA/D is 20, the probability of selecting in the neighborhood is 0.9.

5) The settings of div in GrEA are summarized in Table 5.

6) In the CDOS of our DCDG-EA algorithm, we exploit the SBX and the DE operators. The adaptation rate $\omega = 0.8$, learning factor $\delta = 0.8$, the smallest selection possibility of every operator $P_{\min} = 0.1$, the initial selection possibility and quality of every operator are $p_{i,t} = 1/l$, $i = \{1, 2, ..., l\}$ and $q_{i,t} = 1, i = \{1, 2, ..., l\}$ respectively. The parameter θ in CDIS is 5.

[1]The source code of NSGAIII, GrEA and MOEA/D comes from PlatEMO Tian et al. (2017), which can be downloaded from http://bimk.ahu.edu.cn/index.php?s=/Index/Software/index.html

The algorithms used on the comparison have their own unique parameters which are set here the same as that used in their original papers.

Table 3: parameter values of SBX used in DCDG-EA and MOEA/D

Parameter	DCDG-EA	MOEA/D
crossover probability p_c	1	1
crossover distribution index η_c	30	20
mutation probability p_m	1	1
mutation distribution index η_m	20	20

Table 4: Number of sampling size / reference points / population size

m	H	No.of reference vectors	Population size
3	$H = 12$	91	91
5	$H = 6$	210	210
8	$H_1 = 3, H_2 = 2$	156	156
10	$H_1 = 3, H_2 = 2$	275	276
15	$H_1 = 2, H_2 = 1$	135	135

Table 5: setting of grid division div in GrEA

Test instance	m	div
DTLZ1	3 5 8 10 15	10 10 10 12 12
DTLZ2	3 5 8 10 15	10 9 8 8 10
DTLZ3	3 5 8 10 15	11 11 10 10 12
DTLZ4	3 5 8 10 15	10 9 8 9 10
WFG1	3 5 8 10 15	7 14 14 13 14
WFG2	3 5 8 10 15	15 16 14 18 20
WFG3	3 5 8 10 15	14 14 13 14 15
WFG4	3 5 8 10 15	10 11 10 11 12
WFG5	3 5 8 10 15	10 11 12 14 14
WFG6	3 5 8 10 15	11 12 13 14 15
WFG7	3 5 8 10 15	10 11 12 14 16
WFG8	3 5 8 10 15	10 12 11 13 14
WFG9	3 5 8 10 15	9 12 13 15 15

5. Experiment Result and Analysis

5.1. Performance Comparisons on DTLZ

The statistical results of IGD and HV values obtained by DCDG-EA, NSGAIII, GrEA, MOEA/D and MOEA/D-M2M on DTLZ instances with different number of objectives are shown in Table 6 and Table 7 respectively, where the best values are highlighted in bold. A smaller IGD value means that the obtained Pareto set gets closer to the true front. Additionally, the larger HV value means that the quality of the obtained Pareto set is good. Fig. 5 shows the variation of IGD metric obtained by these five algorithms on 15-objective DTLZ test suit in a single run. The experimental results show the proposed MOEA-APS algorithm is highly competitive comparing with the other four state-of-the-art MOEAs. The results are represented by instance.

The optimal front of DTLZ1 is a linear hyper-plane that includes $(11^k - 1)$ local optimal fronts. Thus, the convergence to the global optimal front is not an easy task. The statistical results in Table 6 indicates that DCDG-EA has achieved the best performance among the five algorithms on all

Table 6: Statistical results (mean and deviation) of the IGD values obtained by the five algorithms on DTLZ instances with different number of objectives, the best results are in bold type

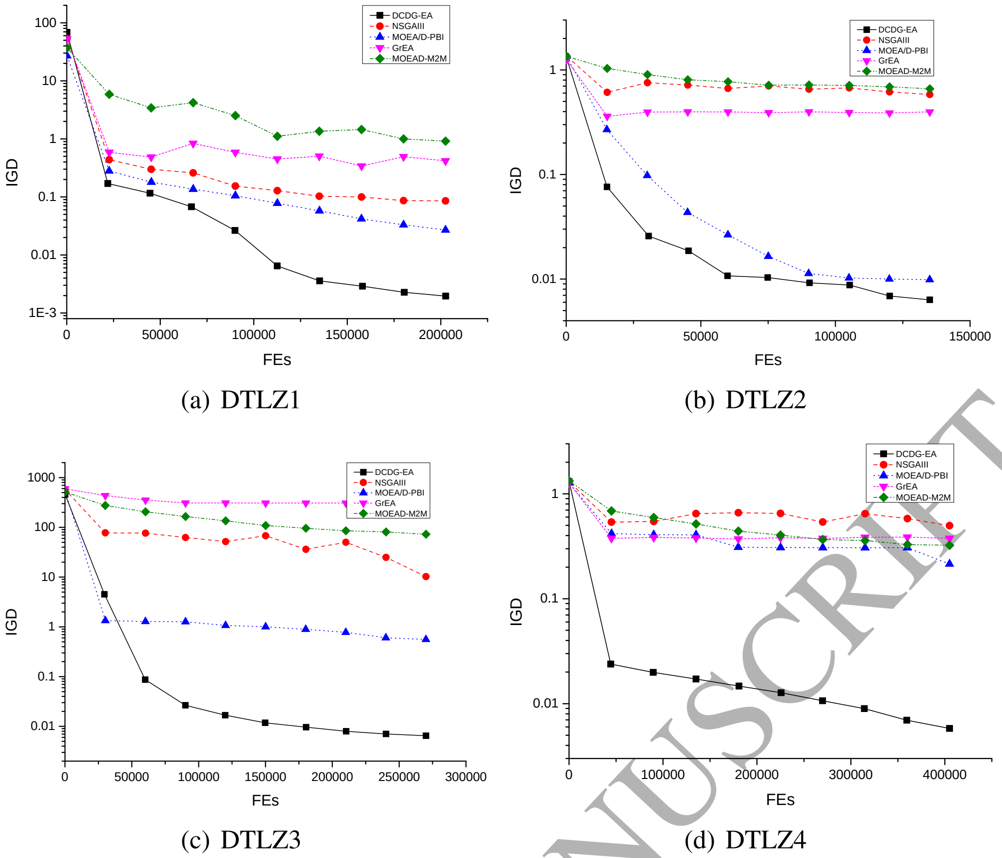
Problem	m	FEs	DCDG-EA	NSGAIII	MOEA/D	GrEA	MOEAD-M2M
DTLZ1	3	36400	1.9553e-3 (1.14e-3)	**7.1231e-4 (2.81e-4)**	1.6092e-3 (1.14e-3)	1.1253e-1 (9.13e-2)	2.0353e-2 (1.01e-3)
	5	126000	**3.4146e-4 (2.66e-5)**	4.5773e-4 (1.41e-4)	5.8192e-4 (2.64e-4)	2.0706e-1 (6.25e-2)	4.5676e-2 (1.75e-3)
	8	117000	**2.3453e-3 (4.83e-4)**	4.8700e-2 (5.90e-2)	6.3036e-3 (1.24e-3)	3.3277e-1 (8.88e-2)	1.8982e-1 (1.81e-1)
	10	275000	**1.9593e-3 (3.75e-4)**	4.0404e-3 (2.45e-3)	2.4974e-3 (3.10e-4)	3.8506e-1 (1.15e-1)	1.1858e-1 (5.07e-2)
	15	202500	**2.1990e-3 (3.83e-4)**	8.5352e-2 (1.43e-1)	2.2776e-2 (1.80e-3)	4.4263e-1 (5.40e-2)	1.5314e+0 (1.11e+0)
DTLZ2	3	22750	9.6795e-4 (1.05e-4)	1.1563e-3 (1.21e-4)	**5.5364e-4 (7.68e-5)**	7.3694e-2 (6.51e-4)	4.9756e-2 (2.11e-3)
	5	73500	1.6322e-3 (9.78e-5)	2.4882e-3 (4.31e-4)	**8.5719e-4 (8.77e-5)**	1.4524e-1 (1.52e-3)	1.4764e-1 (8.91e-4)
	8	78000	3.1652e-3 (3.31e-4)	1.2331e-1 (2.55e-1)	**2.0801e-3 (3.32e-4)**	3.0384e-1 (2.06e-3)	2.6511e-1 (6.42e-3)
	10	206250	**2.0256e-3 (1.37e-4)**	1.9812e-1 (2.60e-1)	2.0317e-3 (8.96e-5)	3.4491e-1 (1.16e-3)	3.0361e-1 (1.29e-2)
	15	135000	**6.5004e-3 (1.75e-3)**	6.7881e-1 (5.28e-2)	1.0307e-2 (4.26e-3)	4.7855e-1 (3.80e-3)	6.9949e-1 (3.45e-2)
DTLZ3	3	91000	**2.5388e-3 (2.90e-3)**	4.4992e-3 (1.85e-3)	3.7167e-3 (2.36e-3)	1.2352e-1 (1.12e-1)	5.1982e-2 (2.00e-3)
	5	210000	**1.6498e-3 (8.85e-4)**	3.4424e-3 (8.24e-4)	4.8851e-3 (1.70e-3)	7.2446e-1 (3.55e-1)	1.3774e-1 (2.34e-3)
	8	156000	**8.8657e-3 (1.31e-3)**	7.7826e-1 (8.96e-1)	1.2584e-2 (5.50e-3)	1.4899e+0 (4.26e-1)	5.4573e+0 (3.27e+0)
	10	412500	**5.5387e-3 (7.01e-4)**	6.7360e-1 (6.77e-1)	2.4551e-1 (4.52e-1)	1.2230e+0 (9.36e-2)	2.2910e+0 (1.02e+0)
	15	270000	**7.1965e-3 (8.63e-4)**	1.5498e+1 (1.20e+1)	5.1251e-1 (6.69e-1)	2.4408e+2 (3.99e+1)	7.1998e+1 (1.12e+1)
DTLZ4	3	54600	**2.1381e-4 (2.12e-5)**	1.0680e-1 (2.37e-1)	3.1839e-1 (2.91e-1)	7.2293e-2 (1.00e-3)	5.2575e-2 (1.48e-3)
	5	210000	**2.9681e-4 (4.38e-5)**	1.1396e-3 (3.97e-4)	1.3743e-1 (1.88e-1)	1.4653e-1 (5.35e-3)	1.3861e-1 (3.93e-3)
	8	195000	**2.9801e-3 (3.16e-4)**	1.0761e-1 (2.32e-1)	3.4930e-1 (2.13e-1)	3.0433e-1 (3.56e-3)	2.5767e-1 (5.97e-3)
	10	550000	**3.0785e-3 (2.78e-4)**	4.2996e-3 (6.53e-4)	2.4151e-1 (8.45e-2)	3.4220e-1 (2.92e-3)	2.6241e-1 (3.67e-3)
	15	405000	**6.6477e-3 (1.95e-3)**	4.9734e-1 (2.78e-1)	2.6555e-1 (1.49e-1)	4.7161e-1 (1.58e-2)	3.3453e-1 (1.36e-2)

Table 7: Statistical results (mean and deviation) of the HV values obtained by the five algorithms on DTLZ instances with different number of objectives, the best results are in bold type

Problem	M	FEs	DCDG-EA	NSGAIII	MOEA/D	GrEA	MOEAD-M2M
DTLZ1	3	36400	1.3966e-1 (2.64e-4)	**1.3994e-1 (7.72e-5)**	1.3970e-1 (2.48e-4)	1.0969e-1 (2.55e-2)	1.3899e-1 (1.45e-4)
	5	126000	**4.9320e-2 (8.86e-6)**	4.9316e-2 (6.50e-6)	4.9299e-2 (9.02e-6)	3.3067e-2 (6.02e-3)	4.9227e-2 (2.16e-5)
	8	117000	**8.3532e-3 (3.15e-7)**	8.3308e-3 (3.13e-5)	8.3498e-3 (9.51e-7)	4.7494e-3 (2.48e-3)	6.7183e-3 (3.40e-3)
	10	275000	2.5321e-3 (3.39e-8)	**2.5322e-3 (4.52e-8)**	2.5321e-3 (2.97e-8)	1.3011e-3 (8.71e-4)	2.5137e-3 (3.91e-5)
	15	202500	**1.2747e-4 (1.03e-9)**	1.2590e-4 (1.79e-6)	1.2742e-4 (3.98e-8)	5.8033e-5 (1.83e-5)	1.8717e-6 (3.12e-6)
DTLZ2	3	22750	7.4398e-1 (1.74e-4)	**7.4426e-1 (1.04e-4)**	7.4423e-1 (1.59e-4)	7.2376e-1 (1.30e-3)	7.2867e-1 (1.64e-3)
	5	73500	1.3074e+0 (7.86e-4)	1.3079e+0 (9.55e-4)	**1.3080e+0 (7.60e-4)**	1.3075e+0 (1.95e-3)	1.2742e+0 (9.28e-4)
	8	78000	1.9808e+0 (3.79e-4)	1.9031e+0 (1.69e-1)	1.9800e+0 (7.37e-4)	**1.9874e+0 (9.91e-4)**	1.9161e+0 (1.23e-2)
	10	206250	**2.5153e+0 (2.96e-4)**	2.4318e+0 (1.13e-1)	2.5151e+0 (2.15e-4)	2.4998e+0 (1.92e-3)	2.4546e+0 (1.23e-2)
	15	135000	**4.1378e+0 (6.33e-4)**	3.4145e+0 (2.40e-1)	4.1374e+0 (7.17e-4)	4.0530e+0 (3.93e-3)	2.1288e+0 (1.42e-1)
DTLZ3	3	91000	**7.4120e-1 (5.36e-3)**	7.3717e-1 (3.41e-3)	7.3869e-1 (4.21e-3)	6.6234e-1 (1.21e-1)	7.2353e-1 (6.13e-3)
	5	210000	**1.3072e+0 (1.73e-3)**	1.3048e+0 (1.55e-3)	1.3014e+0 (2.71e-3)	5.2974e-1 (4.08e-1)	1.2796e+0 (4.70e-3)
	8	156000	**1.9790e+0 (9.20e-4)**	1.1232e+0 (1.03e+0)	1.9653e+0 (8.05e-3)	1.5336e-1 (2.51e-1)	0.0000e+0 (0.00e+0)
	10	412500	**2.5155e+0 (4.56e-4)**	1.4039e+0 (1.30e+0)	2.1030e+0 (8.76e-1)	2.6973e-1 (7.83e-2)	7.3161e-2 (1.64e-1)
	15	270000	**4.1375e+0 (4.24e-4)**	0.0000e+0 (0.00e+0)	2.6457e+0 (2.04e+0)	0.0000e+0 (0.00e+0)	0.0000e+0 (0.00e+0)
DTLZ4	3	54600	**7.4485e-1 (3.12e-6)**	6.8671e-1 (1.30e-1)	5.7059e-1 (1.59e-1)	7.2482e-1 (1.37e-3)	7.3127e-1 (2.23e-3)
	5	210000	**1.3085e+0 (7.06e-4)**	1.3084e+0 (5.94e-4)	1.2451e+0 (8.70e-2)	1.3073e+0 (1.93e-3)	1.2888e+0 (2.61e-3)
	8	195000	1.9811e+0 (3.88e-4)	1.9232e+0 (1.28e-1)	1.8220e+0 (1.21e-1)	**1.9877e+0 (8.66e-4)**	1.9679e+0 (5.82e-3)
	10	550000	**2.5154e+0 (1.03e-4)**	2.5149e+0 (7.37e-4)	2.4685e+0 (3.35e-2)	2.5029e+0 (2.43e-3)	2.5107e+0 (2.41e-3)
	15	405000	**4.1381e+0 (4.71e-4)**	3.7711e+0 (2.35e-1)	4.0597e+0 (5.92e-2)	4.0614e+0 (1.53e-2)	3.8863e+0 (9.15e-2)

DTLZ1 instances, except for the 3 objectives instance where NSGAIII performs better. The parallel coordinates presented in Fig. 6 clearly shows that DCDG-EA, NSGAIII and MOEA/D can find an effectively converged and widely distributed set of points for DTLZ1 instance with 15 objectives. Whereas, the IDG and HV values obtained by DCDG-EA as shown in Table 6 and Table 7 are the best. By contrast, the non-dominating fronts obtained by GrEA and MOEA/D-M2M are far from the PF.

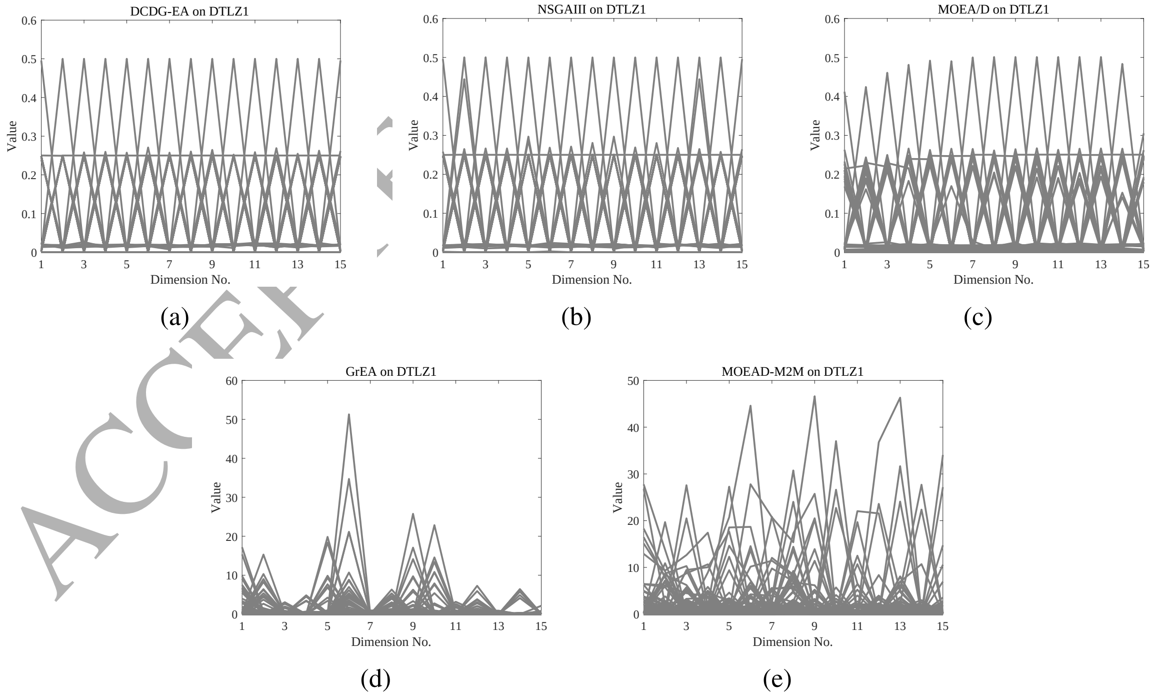
DTLZ2 is a simple test problem that can also be used to investigate the ability of MOEAs (Deb et al., 2005). Unlike MOEA/D which demonstrates its superiority only for the low-dimensional instances, DCDG-EA becomes more prominent when the number of objectives increases. More precisely, DCDG-EA shows its high performance on 10 and 15-objective DTLZ2 instances. Fig. 7 shows the nondominated fronts obtained by these five MOEAs for 15-objective DTLZ2 instance. It is clear that both DCDG-EA and MOEA/D can achieve a well converged and widely distributed set



(a) DTLZ1 (b) DTLZ2

(c) DTLZ3 (d) DTLZ4

Figure 5: Variation of IGD metric obtained by DCDG-EA, NSGAIII, MOEA/D, GrEA and MOEA/D-M2M algorithms on 15-objective DTLZ test suit



(a) (b) (c)

(d) (e)

Figure 6: Parallel coordinates of nondominated fronts obtained by DCDG-EA, NSGAIII, MOEA/D, GrEA and MOEA/D-M2M reference solutions sets sampled from true Pareto front set on the 15-objective DTLZ1 problem

of nondominated points comparing with the other algorithms. However, NSGAIII can only get some part of the true PF. It is worth mentioning that although the solutions distribution obtained by GrEA are rather messy, GrEA is still able to find a widely distributed set of nondominated points for 15-objective DTLZ2 instance. Moreover, the set of nondominated points obtained by MOEAD-M2M are not completely uniform.

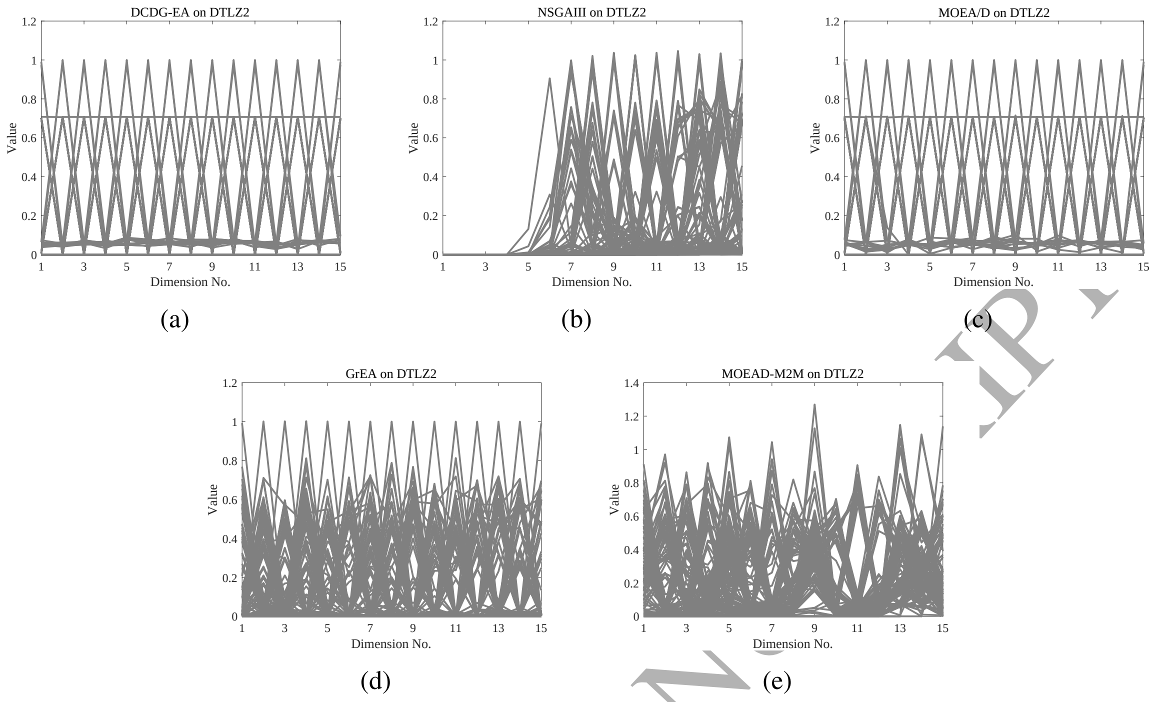


Figure 7: Parallel coordinates of nondominated fronts obtained by DCDG-EA, NSGAIII, MOEA/D, GrEA and MOEA/D-M2M reference solutions sets sampled from true Pareto front set on the 15-objective DTLZ2 problem

DTLZ3 is a highly multimodal problem that has $(3^k - 1)$ local Pareto-optimal fronts and may make an MOEA become stuck at any of these local Pareto-optimal fronts before converging to the global Pareto-optimal front (Deb et al., 2005). From the IGD results shown in Table 6 and Table 7, DCDG-EA significantly outperforms the other in all instances. From Fig. 8, we can clearly see that NSGAIII performs poorly on 15-objectives DTLZ3 since the non-dominated sorting employed in NSGA-III has a low efficiency in selecting solutions with a good convergence performance when the number of objective is very large. Moreover, GrEA and MOEA/D-M2M encounter considerable difficulties on the convergence to the PF. However, MOEA/D has slightly better performance than GrEA and MOEA/D-M2M.

DTLZ4 is designed to investigate the ability of a MOEA to maintain a good distribution of solutions (Deb et al., 2005). DCDG-EA achieves the best performance in all DTLZ4 instances. From the parallel coordinate slots illustrated in Fig. 9, MOEA/D performs better than NSGAIII, but it does not performs well as DCDG-EA. Moreover, GrEA and MOEA/D-M2M can obtain a solution set with good convergence and wide distribution. However, the obtained solutions are messy.

Globally speaking, from the empirical studies on DTLZ test suite, we conclude that the promising results obtained by DCDG-EA should be attributed to its advanced techniques used for balancing the convergence and the diversity.

5.2. Performance Comparisons on WFG

The WFG test suite provides a truer means for assessing the performance of the optimization algorithms using a wide range of different problems Huband et al. (2006b). Table 8 presents the statistical comparison results of DCDG-EA with the other MOEAs in terms of HV values. The best results are in bold type. Based on the above results, we can obtain some observations for each algorithm on WFG test suite. It is clear that DCDG-EA has the best performance in most cases.

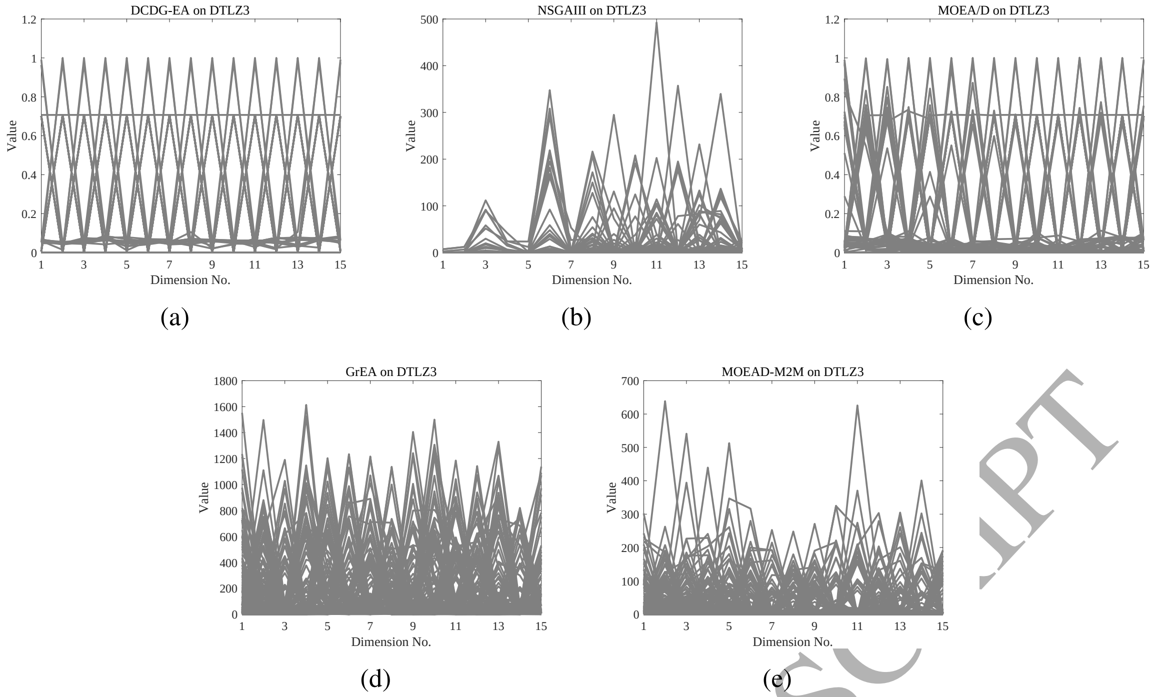


Figure 8: Parallel coordinates of nondominated fronts obtained by DCDG-EA, NSGAIII, MOEA/D, GrEA and MOEA/D-M2M reference solutions sets sampled from true Pareto front set on the 15-objective DTLZ3 problem

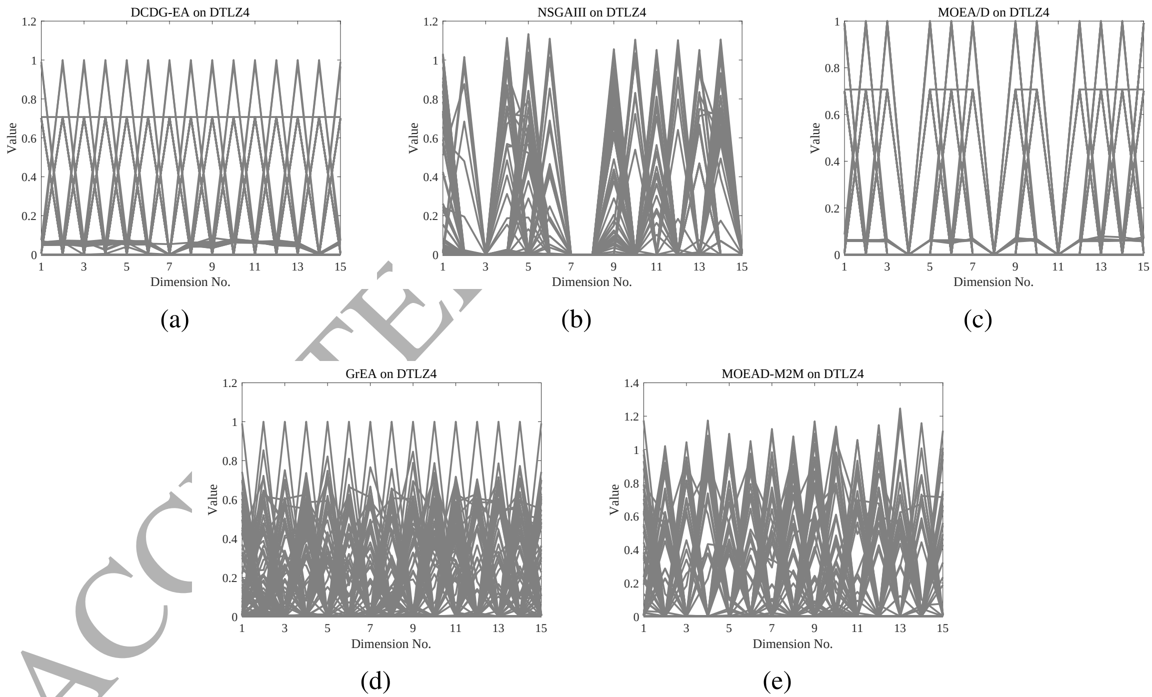


Figure 9: Parallel coordinates of nondominated fronts obtained by DCDG-EA, NSGAIII, MOEA/D, GrEA and MOEA/D-M2M reference solutions sets sampled from true Pareto front set on the 15-objective DTLZ4 problem

WFG1 investigates MOEAs' ability for solving instances with mixed and biased PF. From Table 8, we can see DCDG-EA shows the best performance on WFG1 and WFG2 with 3-, 5- and 8- objective. Relatively speaking, DCDG-EA may have difficulty on handling problems with high-dimensional convex disconnected PFs, because the DCDG-EA does not show such outstanding performance as NSGAIII on WFG2 with 10- and 15-objective. Moreover, NSGAIII may struggle on relatively low-dimensional biased problems having mixed PFs (e.g., 3-, 5- and 8-objective WFG1). WFG3 has the linear and degenerate PF and DCDG-EA performs best with 8-, 10- and 15-objective cases, while GrEA wins in the 3- and 5-objective cases. NSGAIII does not behave quite well on this problem.

WFG4 is featured by multimodality and has larger hill sizes which can cause MOEAs to be struggled in local optima. DCDG-EA shows a good performance on solving 3-, 5- and 15-objective, except the 8 and 10 objectives instances where DCDG-EA performs slightly worse than NSGAIII and GrEA. The differences in the HV values of DCDG-EA, NSGAIII and GrEA on WFG5-6 are not significant. However, MOEA/D and MOEA/D M2M are not good as others. the compared MOEAs do not differ greatly on WFG7-9 with 3 and 5 objective. However, the superiority becomes more evident when the number of objectives increases, especially on WFG8-9 instances. It is worth noting that the overall performance of the MOEA/D and MOEA/D M2M algorithms on the WFG test suite is not particularly promising comparing with the other algorithms.

Table 8: Statistical results (mean and deviation) of the HV values obtained by the five algorithms on WFG instances with different number of objectives, the best results are in bold type

Problem	M	FEs	DCDG-EA	NSGAIII	MOEA/D	GrEA	MOEAD-M2M
WFG1	3	36400	**5.9462e+1 (2.20e-1)**	4.6956e+1 (5.18e-1)	4.6445e+1 (1.14e+0)	5.8502e+1 (1.13e-1)	2.9025e+1 (7.99e+0)
	5	157500	**6.0249e+3 (4.58e+0)**	5.9748e+3 (8.80e+1)	5.8517e+3 (5.22e+1)	5.9377e+3 (1.26e+1)	2.1761e+3 (2.70e+2)
	8	234000	**2.0728e+7 (4.97e+2)**	2.0651e+7 (3.45e+3)	1.5096e+7 (3.40e+6)	2.0402e+7 (9.03e+4)	6.8262e+6 (7.60e+5)
	10	550000	8.6406e+9 (5.17e+6)	**8.6663e+9 (7.20e+4)**	7.1927e+9 (4.00e+7)	8.5098e+9 (2.29e+7)	2.6022e+9 (8.54e+7)
	15	405000	1.3843e+17 (7.73e+13)	**1.3914e+17 (1.68e+12)**	6.9228e+16 (1.74e+16)	1.3672e+17 (6.62e+14)	3.9456e+16 (6.16e+14)
WFG2	3	36400	**5.9317e+1 (4.43e-2)**	5.9037e+1 (4.49e-2)	5.1531e+1 (3.74e+0)	5.4449e+1 (6.47e+0)	5.8159e+1 (3.75e-1)
	5	157500	**6.1419e+3 (7.20e-1)**	6.1322e+3 (5.72e+0)	4.9291e+3 (2.65e+0)	6.0004e+3 (6.17e+0)	5.9832e+3 (4.53e+0)
	8	234000	**2.2101e+7 (4.19e+3)**	2.2003e+7 (9.49e+4)	2.0891e+7 (2.77e+5)	2.1694e+7 (6.63e+4)	2.1817e+7 (8.09e+4)
	10	550000	9.6036e+9 (1.59e+7)	**9.6351e+9 (1.13e+6)**	9.1529e+9 (3.10e+7)	9.4315e+9 (6.42e+6)	9.5478e+9 (1.56e+7)
	15	405000	1.5138e+17 (8.46e+15)	**1.7846e+17 (5.35e+14)**	1.6828e+17 (1.14e+15)	1.7368e+17 (6.01e+14)	1.7288e+17 (2.38e+15)
WFG3	3	36400	6.1591e+0 (7.13e-2)	6.1675e+0 (1.68e-2)	5.3875e+0 (3.64e-1)	**6.2556e+0 (7.03e-2)**	4.8623e+0 (2.96e-1)
	5	157500	2.5396e+0 (9.59e-4)	1.6589e+0 (2.67e-2)	3.8624e-2 (5.46e-2)	**2.6832e+0 (1.03e-2)**	0.0000e+0 (0.00e+0)
	8	234000	**1.4089e-2 (1.65e-4)**	1.3214e-2 (8.94e-4)	0.0000e+0 (0.00e+0)	0.0000e+0 (0.00e+0)	0.0000e+0 (0.00e+0)
	10	550000	**4.1853e-5 (4.18e-6)**	1.8051e-5 (1.07e-5)	0.0000e+0 (0.00e+0)	0.0000e+0 (0.00e+0)	0.0000e+0 (0.00e+0)
	15	405000	**0.0000e+0 (0.00e+0)**	**0.0000e+0 (0.00e+0)**	**0.0000e+0 (0.00e+0)**	**0.0000e+0 (0.00e+0)**	**0.0000e+0 (0.00e+0)**
WFG4	3	36400	**3.5107e+1 (3.38e-1)**	3.4904e+1 (4.39e-2)	3.3355e+1 (4.46e-1)	3.5033e+1 (1.23e-1)	3.3068e+1 (3.28e-1)
	5	157500	**4.9139e+3 (1.19e+1)**	4.8907e+3 (1.84e+1)	4.3725e+3 (2.58e+1)	4.8998e+3 (1.02e+1)	4.1780e+3 (1.27e+2)
	8	234000	2.0417e+7 (9.27e+3)	1.9514e+7 (1.24e+6)	1.1617e+7 (2.67e+5)	**2.0516e+7 (3.44e+3)**	1.7253e+7 (5.59e+4)
	10	550000	9.3269e+9 (7.09e+6)	**9.3276e+9 (4.07e+6)**	5.8620e+9 (3.15e+8)	9.3233e+9 (8.02e+6)	8.2001e+9 (2.10e+8)
	15	405000	**1.7731e+17 (1.42e+13)**	1.6144e+17 (1.31e+15)	5.9014e+16 (7.14e+14)	1.6880e+17 (1.54e+14)	1.3086e+17 (6.05e+15)
WFG5	3	36400	3.2419e+1 (3.26e-1)	**3.2624e+1 (3.23e-1)**	3.1724e+1 (1.22e-1)	3.2242e+1 (6.50e-2)	3.0709e+1 (2.20e-1)
	5	157500	**4.6449e+3 (1.54e+1)**	4.6268e+3 (2.57e+1)	4.2204e+3 (9.08e+1)	4.5937e+3 (3.05e+0)	4.1249e+3 (6.12e+1)
	8	234000	1.9106e+7 (3.00e+3)	1.9102e+7 (2.15e+3)	1.2191e+7 (7.15e+4)	**1.9237e+7 (4.35e+2)**	1.6709e+7 (2.97e+5)
	10	550000	8.7211e+9 (2.19e+6)	8.7197e+9 (1.72e+6)	5.8331e+9 (1.86e+8)	**8.7223e+9 (4.97e+6)**	7.8561e+9 (3.35e+7)
	15	405000	**1.6422e+17 (1.87e+11)**	1.5681e+17 (1.05e+16)	7.9483e+16 (1.21e+15)	1.5238e+17 (3.49e+15)	1.0642e+17 (3.20e+15)
WFG6	3	36400	3.2046e+1 (1.61e+0)	**3.2861e+1 (1.82e+0)**	3.1565e+1 (6.40e-1)	3.2047e+1 (9.81e-1)	2.9923e+1 (6.14e-1)
	5	157500	**4.6659e+3 (6.16e+1)**	4.4445e+3 (1.55e+2)	2.4163e+3 (3.11e+2)	4.5212e+3 (1.25e+1)	4.0038e+3 (9.48e+0)
	8	234000	1.8164e+7 (2.63e+5)	1.8034e+7 (3.63e+4)	5.4380e+6 (2.03e+5)	**1.8977e+7 (9.00e+5)**	1.5937e+7 (1.69e+6)
	10	550000	8.3980e+9 (1.58e+8)	**8.6810e+9 (1.02e+8)**	2.8478e+9 (2.52e+8)	8.6737e+9 (1.94e+8)	7.8486e+9 (2.56e+8)
	15	405000	**1.6089e+17 (1.84e+14)**	1.2743e+17 (5.11e+15)	2.0182e+16 (1.70e+15)	1.5199e+17 (9.13e+15)	7.0918e+16 (9.40e+15)
WFG7	3	36400	**3.5498e+1 (3.84e-2)**	3.5493e+1 (1.77e-2)	3.1113e+1 (1.45e+0)	3.5219e+1 (6.39e-2)	3.3106e+1 (2.99e-1)
	5	157500	4.9949e+3 (3.59e+0)	4.9823e+3 (8.01e+0)	4.1472e+3 (6.37e+1)	**5.0029e+3 (4.02e+0)**	4.1686e+3 (4.71e+1)
	8	234000	2.0397e+7 (2.59e+2)	2.0376e+7 (2.63e+3)	7.9760e+6 (1.66e+5)	**2.0573e+7 (3.41e+4)**	1.7164e+7 (6.98e+4)
	10	550000	**9.3550e+9 (6.24e+6)**	9.3297e+9 (3.22e+6)	4.2622e+9 (7.63e+8)	9.3345e+9 (1.08e+5)	8.4085e+9 (2.87e+7)
	15	405000	**1.7730e+17 (1.37e+13)**	1.7276e+17 (6.40e+15)	4.0078e+16 (1.51e+16)	1.7050e+17 (1.43e+15)	6.1823e+16 (3.06e+16)
WFG8	3	36400	2.6891e+1 (1.32e-1)	2.7294e+1 (8.52e-2)	2.5829e+1 (3.19e-1)	**2.7718e+1 (1.41e-1)**	2.2974e+1 (5.90e-1)
	5	157500	**4.7175e+3 (4.52e+0)**	4.5416e+3 (9.15e+1)	3.3601e+3 (3.35e+2)	4.5064e+3 (3.20e+2)	3.8746e+3 (2.83e+2)
	8	234000	**1.8122e+7 (4.00e+4)**	1.7043e+7 (8.30e+5)	2.8876e+6 (1.98e+6)	1.7500e+7 (3.09e+4)	1.2942e+7 (2.27e+6)
	10	550000	**8.5572e+9 (2.65e+8)**	8.1102e+9 (3.62e+8)	9.4290e+8 (1.23e+8)	8.4329e+9 (9.37e+7)	7.4096e+9 (9.62e+8)
	15	405000	**1.6918e+17 (4.55e+15)**	1.0159e+17 (1.44e+16)	8.0044e+15 (4.14e+13)	1.6151e+17 (9.90e+14)	3.5743e+16 (9.18e+14)
WFG9	3	36400	3.2951e+1 (4.35e-2)	3.3000e+1 (8.47e-2)	3.0647e+1 (1.78e+0)	**3.3483e+1 (1.22e-2)**	2.7321e+1 (5.11e+0)
	5	157500	**4.5344e+3 (5.72e+0)**	4.2323e+3 (2.31e+2)	3.8730e+3 (2.12e+1)	4.5090e+3 (2.07e+1)	3.9352e+3 (9.79e+1)
	8	234000	1.9282e+7 (6.43e+4)	1.9111e+7 (1.48e+5)	5.5875e+6 (1.36e+6)	**1.9530e+7 (1.08e+4)**	1.5587e+7 (4.52e+5)
	10	550000	**8.9416e+9 (1.78e+7)**	8.8845e+9 (1.72e+7)	3.7180e+9 (1.09e+9)	8.8122e+9 (1.81e+7)	6.8060e+9 (3.40e+7)
	15	405000	**1.6972e+17 (1.27e+15)**	1.4814e+17 (3.40e+15)	2.1441e+16 (2.67e+16)	1.5908e+17 (1.54e+15)	8.2894e+16 (5.42e+15)

5.3. Non-parametric Statistical Test

In addition to the above analysis, the non-parametric statistical test is used to prove the efficiency of the proposed DCDG-EA algorithm. Friedman test is a multiple comparison test that aims to detect whether a significant differences exist between the results of two or more algorithms. Friedman test ranks the algorithms for each problem separately, the best performing algorithm ranks first, the second best has the second rank and so on. In our test, we use the average ranks of the IGD metric and HV metric on all the instances to evaluate the performance of an algorithm. Besides, The null hypothesis states that all the algorithms behave similarly. The Friedman statistic F_f can be computed as follows:

$$F_f = \frac{12n}{k\,(k+1)}\left[\sum\nolimits_j R_j^2 - \frac{k\,(k+1)}{4}\right], \tag{18}$$

which is compared with a χ^2 distribution with $(k-1)$ degrees of freedom, critical values have been evaluated in Sheskin (2004). The final ranks achieved by the Friedman test, the statistic F_f value and the P_Value of IGD metric and HV metric are presented in Table 9 and Table 10 respectively. As we can see, DCDG-EA achieves the best ranks (1.3000 and 1.6385) on this test.

Table 9: Ranks achieved by the Friedman test on IGD values of all DTLZ instances

Algorithms	Friedmanranks
DCDG-EA	1.3000
NSGAIII	2.8500
MOEA/D	2.3000
GrEA	4.5000
MOEAD-M2M	4.0500
Statistic	39.5516
P_Value	<0.00001

Table 10: Ranks achieved by the Friedman test on HV values of all DTLZ and WFG instances

Algorithms	Friedmanranks
DCDG-EA	1.6385
NSGAIII	2.5077
MOEA/D	4.0077
GrEA	2.7077
MOEAD-M2M	4.1385
Statistic	56.0554
P_Value	<0.00001

5.4. Comparisons Between CDOS, CDIS and PBI

CDOS and CDIS strategies are the key components in DCDG-EA. most scalarization approaches used in the decomposition-based MOEAs are also applicable to the proposed DCDG-EA. In this section, we will carry out some comparative studies on these two strategies with PBI approach. Table 11 represents DCDG-EA with different approaches. DCDG-EA is our proposed original algorithm with CDOS and CDIS strategies, 'Y' indicates that the method is used in DCDG-EA. In DCDG-EA_PBI, all the individuals are selected using $d_1 + \theta \times d_2$ only.

Table 11: Different methods are integrated into DCDG-EA

	CDOS	CDIS	PBI
DCDG-EA	Y	Y	N
DCDG-EA_CDOS	Y	N	N
DCDG-EA_CDIS	N	Y	N
DCDG-EA_PBI	Y	N	Y

In order to study the influence of CDOS and CDIS strategies in the DCDG-EA, Table 12 shows the statistical results of the IDG and HV metrics values on DTLZ test suit when DCDG-EA contains only one of the two strategies. It is clear that the overall performance of the DCDG-EA algorithm is the best when combining these two strategies, especially the HV metric value. The high performance of DCDG-EA should be attributed to the combined effect of these two strategies for balancing the convergence and the diversity. On the one hand, during the evolution of the population, CDOS is committed to producing well-converged and well-distributed offspring. On the other hand, CDIS selects solutions with good convergence and diversity to the next generation at different stages of population evolution.

Table 12: Comparison of CDOS and CDIS strategies

Problem	m	FEs	DCDG-EA	DCDG-EA_CDOS	DCDG-EA_CDIS	DCDG-EA	DCDG-EA_CDOS	DCDG-EA_CDIS
			IGD			HV		
DTLZ1	3	36400	1.9553e-3 (1.14e-3)	2.1273e-3 (2.91e-3)	**9.7373e-4 (4.29e-4)**	1.3966e-1 (2.64e-4)	1.3966e-1 (6.98e-4)	**1.3989e-1 (9.90e-5)**
	5	126000	**3.4146e-4 (2.66e-5)**	4.0901e-4 (1.27e-4)	4.2768e-4 (1.30e-4)	**4.9320e-2 (8.86e-6)**	4.9312e-2 (6.71e-6)	4.9313e-2 (5.89e-6)
	8	117000	2.3453e-3 (4.83e-4)	**2.0786e-3 (3.41e-4)**	2.0919e-3 (2.74e-4)	8.3532e-3 (3.15e-7)	8.3531e-3 (4.46e-7)	**8.3534e-3 (3.35e-7)**
	10	275000	1.9593e-3 (3.75e-4)	1.8373e-3 (1.34e-4)	**1.7809e-3 (2.05e-4)**	**2.5321e-3 (3.39e-8)**	2.5321e-3 (5.43e-8)	2.5320e-3 (2.10e-8)
	15	202500	**2.1990e-3 (3.83e-4)**	2.4483e-3 (3.98e-4)	2.5735e-3 (6.61e-4)	**1.2747e-4 (1.03e-9)**	1.2747e-4 (6.60e-10)	1.2747e-4 (1.41e-9)
DTLZ2	3	22750	9.6795e-4 (1.05e-4)	**9.6570e-4 (1.01e-4)**	9.8083e-4 (6.45e-5)	7.4398e-1 (1.74e-4)	**7.4414e-1 (2.24e-4)**	7.4402e-1 (8.30e-5)
	5	73500	**1.6322e-3 (9.78e-5)**	1.6872e-3 (5.50e-5)	1.6475e-3 (9.21e-5)	1.3074e+0 (7.86e-4)	**1.3082e+0 (4.58e-4)**	1.3076e+0 (4.37e-4)
	8	78000	5.1652e-3 (3.31e-4)	**4.9578e-3 (2.59e-4)**	5.1019e-3 (2.84e-4)	**1.9808e+0 (3.79e-4)**	1.9797e+0 (3.94e-4)	1.9805e+0 (6.69e-4)
	10	206250	**5.0656e-3 (1.37e-4)**	5.1042e-3 (3.69e-4)	5.1010e-3 (3.42e-4)	**2.5153e+0 (2.96e-4)**	2.5151e+0 (7.82e-4)	2.5151e+0 (6.22e-4)
	15	135000	6.5004e-3 (1.75e-3)	6.1568e-3 (5.58e-4)	**6.1104e-3 (1.06e-3)**	4.1378e+0 (6.33e-4)	4.1380e+0 (4.74e-4)	**4.1382e+0 (6.61e-5)**
DTLZ3	3	91000	**2.5388e-3 (2.90e-3)**	2.7860e-3 (1.51e-3)	5.0634e-3 (5.28e-3)	**7.4120e-1 (5.36e-3)**	7.4062e-1 (2.96e-3)	7.3738e-1 (7.12e-3)
	5	210000	1.6498e-3 (8.85e-4)	**1.3294e-3 (4.06e-4)**	2.1079e-3 (1.48e-3)	1.3072e+0 (1.73e-3)	**1.3073e+0 (6.96e-4)**	1.3067e+0 (2.25e-3)
	8	156000	**8.8657e-3 (1.31e-3)**	1.0188e-2 (1.34e-3)	1.0847e-2 (3.92e-3)	**1.9790e+0 (9.20e-4)**	1.9773e+0 (1.23e-3)	1.9755e+0 (4.92e-3)
	10	412500	5.5387e-3 (7.01e-4)	5.6686e-3 (3.38e-4)	**5.0298e-3 (3.71e-4)**	**2.5155e+0 (4.56e-4)**	2.5153e+0 (6.86e-4)	2.5150e+0 (4.37e-4)
	15	270000	**7.1965e-3 (8.63e-4)**	8.6569e-3 (7.61e-4)	4.7750e-2 (8.71e-2)	**4.1375e+0 (4.24e-4)**	4.1290e+0 (2.06e-2)	4.1375e+0 (6.13e-4)
DTLZ4	3	54600	2.1381e-4 (2.12e-5)	2.1799e-4 (2.65e-5)	**2.0951e-4 (3.65e-5)**	**7.4485e-1 (3.12e-6)**	7.4484e-1 (1.64e-5)	7.4484e-1 (1.08e-5)
	5	210000	2.9681e-4 (4.38e-5)	**2.5829e-4 (2.54e-5)**	2.5849e-4 (3.02e-5)	1.3085e+0 (7.06e-4)	1.3084e+0 (9.34e-4)	**1.3091e+0 (4.12e-4)**
	8	195000	**2.9801e-3 (3.16e-4)**	3.2167e-3 (2.88e-4)	3.0389e-3 (4.69e-4)	**1.9811e+0 (3.88e-4)**	1.9803e+0 (6.20e-4)	1.9808e+0 (6.28e-4)
	10	550000	**3.0785e-3 (2.78e-4)**	3.2927e-3 (2.36e-4)	3.3312e-3 (1.15e-4)	**2.5154e+0 (1.03e-4)**	2.5152e+0 (6.65e-4)	2.5151e+0 (5.56e-4)
	15	405000	**6.6477e-3 (1.95e-3)**	2.6114e-2 (4.47e-2)	2.6298e-2 (4.47e-2)	**4.1381e+0 (4.71e-4)**	4.1346e+0 (7.99e-3)	4.1344e+0 (8.22e-3)

Table 13 shows the selection probabilities of the operators in the process of population evolution on 15-objective DTLZ3 and DTLZ4. In other words, the number of times each operator is selected divided by the sum of the number of times all the operators are selected 15-objective DTLZ3 and 15-objective DTLZ4 are reported. It is clear that SBX is more likely to be selected than the DE operators for the highly multimodal 15-objective DTLZ3 problem because SBX has a strong ability to escape from the local optima. Moreover, DTLZ4 is designed to investigate the ability of a MOEA to maintain a good distribution of solutions. Since the unique memory capability of DE makes it possible to dynamically track the current search situation to adjust its search strategy, the selection probability of DE is greater than SBX. Therefore, we can conclude that CDOS has a strong ability to maintain the search diversity and enhance the search on solving wide assortment of problems.

Table 13: Selection probabilities of evolutionary operators DE and SBX

Test instances	DE	SBX
15-objective DTLZ3	0.2074	0.7926
15-objective DTLZ4	0.8208	0.1792

Similarly, Table 14 gives the comparison results of CDIS strategy with PBI approach on solving DTLZ test suit. It is clear that CDIS strategy is much better than PBI approach. In the PBI approach, the convergence and diversity of the solutions are always controlled by the penalty factor θ. Unlike PBI, we consider the evolutionary state of solutions in the population. In CDIS strategy, the solutions in each subpopulation are divided into P_1 and P_2 sets according to pre_d1 which records the best d_1 value in the previous generation subpopulation. The convergence of solutions in P_1 is obviously better than solutions in P_2. In the early stages of evolution, the solutions are dispersed in the objective space and quickly approaches the PF. d_1 value rapidly decreases. If the P_1 set is not empty, selecting a solution with a small combined distance $comd = d_1 + \theta \times d_2$ in P_1 to the next generation will maintain the balance between the convergence and diversity of population. In the later stages of the evolution, the solutions are very close to the PF, the difference pre_d1 values between the t-th generation and the $(t+1)$-th generation population is particularly small and the solution will most likely be divided into P_2 set. If the P_2 set is not empty, selecting a solution with a small d_1 value instead of the combination distance will speed up convergence speed and achieve the balance between the convergence and diversity.

Table 14: Comparison of CDIS strategy and PBI approach

Problem	m	FEs	DCDG-EA	DCDG-EA_PBI	DCDG-EA	DCDG-EA_PBI
			IGD		HV	
	3	36400	1.9553e-3 (1.14e-3)	2.2273e-3 (2.81e-3)	**1.3966e-1 (2.64e-4)**	1.3966e-1 (6.78e-4)
	5	126000	**3.4146e-4 (2.66e-5)**	4.2901e-4 (1.46e-4)	**4.9320e-2 (8.86e-6)**	4.9112e-2 (6.91e-6)
DTLZ1	8	117000	2.3453e-3 (4.83e-4)	**2.0796e-3 (3.21e-4)**	**8.3532e-3 (3.15e-7)**	8.3431e-3 (3.46e-7)
	10	275000	1.9593e-3 (3.75e-4)	2.0373e-3 (1.31e-4)	**2.5321e-3 (3.39e-8)**	2.5321e-3 (6.43e-8)
	15	202500	**2.1990e-3 (3.83e-4)**	2.4483e-3 (3.98e-4)	**1.2747e-4 (1.03e-9)**	1.2747e-4 (6.60e-10)
	3	22750	9.6795e-4 (1.05e-4)	**9.6770e-4 (1.11e-4)**	7.4398e-1 (1.74e-4)	**7.4514e-1 (2.44e-4)**
	5	73500	**1.6322e-3 (9.78e-5)**	1.6872e-3 (5.50e-5)	1.3074e+0 (7.86e-4)	**1.3085e+0 (5.67e-4)**
DTLZ2	8	78000	**5.1652e-3 (3.31e-4)**	5.9538e-3 (2.54e-4)	**1.9808e+0 (3.79e-4)**	1.9697e+0 (3.04e-4)
	10	206250	**5.0656e-3 (1.37e-4)**	5.1042e-3 (3.69e-4)	**2.5153e+0 (2.96e-4)**	2.1243e+0 (2.34e-4)
	15	135000	6.5004e-3 (1.75e-3)	**6.1668e-3 (5.48e-4)**	**4.1378e+0 (6.33e-4)**	4.1332e+0 (4.44e-4)
	3	91000	**2.5388e-3 (2.90e-3)**	2.7960e-3 (1.81e-3)	**7.4120e-1 (5.36e-3)**	7.4098e-1 (2.76e-3)
	5	210000	1.6498e-3 (8.85e-4)	**1.4293e-3 (4.56e-4)**	1.3072e+0 (1.73e-3)	**1.3074e+0 (6.86e-4)**
DTLZ3	8	156000	**8.8657e-3 (1.31e-3)**	2.2298e-2 (1.23e-3)	**1.9790e+0 (9.20e-4)**	1.9783e+0 (1.13e-3)
	10	412500	**5.5387e-3 (7.01e-4)**	5.8686e-3 (3.37e-4)	**2.5155e+0 (4.56e-4)**	2.5153e+0 (7.86e-4)
	15	270000	**7.1965e-3 (8.63e-4)**	8.2569e-3 (6.71e-4)	**4.1375e+0 (4.24e-4)**	4.1380e+0 (2.57e-2)
	3	54600	**2.1381e-4 (2.12e-5)**	2.1643e-4 (2.33e-5)	**7.4485e-1 (3.12e-6)**	7.4483e-1 (1.25e-5)
	5	210000	2.9681e-4 (4.38e-5)	**2.5939e-4 (2.65e-5)**	**1.3085e+0 (7.06e-4)**	1.3082e+0 (9.34e-4)
DTLZ4	8	195000	**2.9801e-3 (3.16e-4)**	3.3452e-3 (2.76e-4)	**1.9811e+0 (3.88e-4)**	1.9808e+0 (6.78e-4)
	10	550000	**3.0785e-3 (2.78e-4)**	3.4526e-3 (2.25e-4)	**2.5154e+0 (1.03e-4)**	2.5152e+0 (6.89e-4)
	15	405000	**6.6477e-3 (1.95e-3)**	2.674e-2 (4.32e-2)	**4.1381e+0 (4.71e-4)**	4.1375e+0 (7.56e-3)

6. Conclusion

In this paper, a dynamic convergence-diversity guided MOEA (DCDG-EA) for MaOPs is proposed. In DCDG-EA, we use a set of reference vectors that directly divide the objective space into several subspaces where each subspace has its unique subpopulation.

During the evolutionary process, DCDG-EA dynamically adjusts the balance between the convergence and the diversity. The proposed CDIS strategy uses two distance measures (the projected distance along the reference vector that represents convergence and the perpendicular distance from solution to the reference vector that controls the diversity) to update each subpopulation, which effectively overcomes the inefficiency of Pareto domination and promotes the population evolution. Moreover, the CDOS strategy selects an appropriate operator from a set of operators for offspring generation. It is conducive to the strengthening of the balance between the convergence and diversity of population according to the intensity of the population and degree of solutions toward PF. The experimental results on DTLZ and WFG test suits with different number of objectives show that our algorithm is more superior than other compared algorithms. This result simply confirms our initial proposition. At present, some of the parameters involved in the CDOS strategy used empirical values. Hence in our future research, we intend to study the influence of these parameters in the high-dimensional objective space. Additionally, we intend to use DCDG-EA to solve some other real-world applications.

Acknowledgment

This work was partially supported by the National Natural Science Foundation of China (No.61672215, U1613209).

References

Bader, J., & Zitzler, E. (2011). Hype: An algorithm for fast hypervolume-based many-objective optimization. *Evolutionary Computation*, *19*, 45–76.

Basseur, M., & Zitzler, E. (2006). A preliminary study on handling uncertainty in indicator-based multiobjective optimization. *Lecture Notes in Computer Science*, *2*, 727–739.

Beume, N., Naujoks, B., & Emmerich, M. (2007a). Sms-emoa: Multiobjective selection based on dominated hypervolume. *European Journal of Operational Research*, *181*, 1653–1669.

Beume, N., Naujoks, B., & Emmerich, M. (2007b). Sms-emoa: Multiobjective selection based on dominated hypervolume. *European Journal of Operational Research*, *181*, 1653–1669.

Bosman, P. A. N., & Thierens, D. (2003). The balance between proximity and diversity in multiobjective evolutionary algorithms. *IEEE Transactions on Evolutionary Computation*, *7*, 174–188.

Das, I., & Dennis, J. E. (2000). Normal-boundary intersection: A new method for generating the pareto surface in nonlinear multicriteria optimization problems. *Siam Journal on Optimization*, *8*, 631–657.

Deb, K., & Agrawal, R. B. (1994). Simulated binary crossover for continuous search space. *Complex systems*, *9*, 115–148.

Deb, K., & Jain, H. (2014a). An evolutionary many-objective optimization algorithm using reference-point-based nondominated sorting approach, part i: Solving problems with box constraints. *IEEE Transactions on Evolutionary Computation*, *18*, 1001–0920.

Deb, K., & Jain, H. (2014b). An evolutionary many-objective optimization algorithm using reference-point-based nondominated sorting approach, part i: Solving problems with box constraints. *IEEE Transactions on Evolutionary Computation*, *18*, 577–601.

Deb, K., Pratap, A., Agarwal, S., & Meyarivan, T. (2002). A fast and elitist multiobjective genetic algorithm: Nsga-ii. *IEEE Transactions on Evolutionary Computation*, *6*, 182–197.

Deb, K., Thiele, L., Laumanns, M., & Zitzler, E. (2005). *Scalable Test Problems for Evolutionary Multiobjective Optimization*.

Dhall, S. K., & Liu, C. L. (1978). On a real-time scheduling problem. *Operations Research*, *26*, 127–140.

Farina, M., & Amato, P. (2004). A fuzzy definition of optimality for many-criteria optimization problems. *IEEE Transactions on Systems Man and Cybernetics - Part A Systems and Humans*, *34*, 315–326.

Hadka, D., & Reed, P. (2013). Borg: An auto-adaptive many-objective evolutionary computing framework. *Evolutionary Computation*, *21*, 231–259.

He, Z., Yen, G. G., & Zhang, J. (2014). Fuzzy-based pareto optimality for many-objective evolutionary algorithms. *IEEE Transactions on Evolutionary Computation*, *18*, 269–285.

Huband, S., Hingston, P., Barone, L., & While, L. (2006a). A review of multiobjective test problems and a scalable test problem toolkit. *IEEE Transactions on Evolutionary Computation*, *10*, 477–506.

Huband, S., Hingston, P., Barone, L., & While, L. (2006b). A review of multiobjective test problems and a scalable test problem toolkit. *IEEE Transactions on Evolutionary Computation*, *10*, 477–506.

Hughes, E. J. (2004). Multiple single objective pareto sampling. In *Evolutionary Computation, 2003. CEC '03. The 2003 Congress on* (pp. 2678–2684 Vol.4).

Kasprzyk, J. R., Reed, P. M., Kirsch, B. R., & Characklis, G. W. (2009). Managing population and drought risks using many-objective water portfolio planning under uncertainty. *Water Resources Research*, *45*, 170–170.

Kukkonen, S., & Lampinen, J. (2007). Ranking-dominance and many-objective optimization. In *Evolutionary Computation, 2007. CEC 2007. IEEE Congress on* (pp. 3983–3990).

Laumanns, M., Thiele, L., Deb, K., & Zitzler, E. (2002). Combining convergence and diversity in evolutionary multiobjective optimization. *Evolutionary Computation*, *10*, 263.

Li, B., Tang, K., Li, J., & Yao, X. (2016). Stochastic ranking algorithm for many-objective optimization based on multiple indicators. *IEEE Transactions on Evolutionary Computation*, (pp. 1–1).

Li, Z., Nguyen, T. T., Chen, S. M., & Truong, T. K. (2015a). A hybrid algorithm based on particle swarm and chemical reaction optimization for multi-object problems. *Applied Soft Computing*, *35*, 525–540.

Li, Z., Wang, W., Yan, Y., & Li, Z. (2015b). Psabc: A hybrid algorithm based on particle swarm and artificial bee colony for high-dimensional optimization problems. *Expert Systems with Applications*, *42*, 8881–8895.

Liu, H. L., Gu, F., & Zhang, Q. (2014). Decomposition of a multiobjective optimization problem into a number of simple multiobjective subproblems. *IEEE Transactions on Evolutionary Computation*, *18*, 450–455.

Liu, Y., Gong, D., Jing, S., & Jin, Y. (2017). A many-objective evolutionary algorithm using a one-by-one selection strategy. *IEEE Transactions on Cybernetics*, *47*, 2689–2702.

LOpez, A., Coello, C. A. C., Oyama, A., & Fujii, K. (2013). *An Alternative Preference Relation to Deal with Many-Objective Optimization Problems*. Springer Berlin Heidelberg.

Nguyen, T. T., Li, Z. Y., Zhang, S. W., & Truong, T. K. (2014). A hybrid algorithm based on particle swarm and chemical reaction optimization. *Expert Systems with Applications*, *41*, 2134–2143.

Pez Jaimes, A., Coello Coello, C. A., & Chakraborty, D. (2008). Objective reduction using a feature selection technique. In *Genetic and Evolutionary Computation Conference, GECCO 2008, Proceedings, Atlanta, Ga, Usa, July* (pp. 673–680).

Sato, H., Aguirre, H., n E, & Tanaka, K. (2011). Genetic diversity and effective crossover in evolutionary many-objective optimization. In *International Conference on Learning and Intelligent Optimization* (pp. 91–105).

Saxena, D. K., Duro, J. A., Tiwari, A., & Deb, K. (2013). Objective reduction in many-objective optimization: Linear and nonlinear algorithms. *IEEE Transactions on Evolutionary Computation*, *17*, 77–99.

Sheskin, D. J. (2004). *Handbook of parametric and nonparametric statistical procedures /*. Chapman & Hall/CRC.

Storn, R., & Price, K. (1997). De-a simple and efficient heuristic for global optimization over continuous space. *Journal of Global Optimization*, *114*, 341–359.

Thierens, D. (2005). An adaptive pursuit strategy for allocating operator probabilities. In *Genetic and Evolutionary Computation Conference, GECCO 2005, Proceedings, Washington Dc, Usa, June* (pp. 1539–1546).

Tian, Y., Cheng, R., Zhang, X., & Jin, Y. (2017). Platemo: A matlab platform for evolutionary multi-objective optimization [educational forum]. *IEEE Computational Intelligence Magazine*, *12*, 73–87.

While, L., Bradstreet, L., & Barone, L. (2012). A fast way of calculating exact hypervolumes. *Evolutionary Computation IEEE Transactions on*, *16*, 86–95.

XIAO, J., WANG, K., & BI, X. (2014). Multi-objective evolutionary algorithm based on improved k-dominated sorting. *Control and Decision*, *29*, 182–197.

Yang, S., Li, M., Liu, X., & Zheng, J. (2013). A grid-based evolutionary algorithm for many-objective optimization. *IEEE Transactions on Evolutionary Computation*, *17*, 721–736.

Zhang, Q., & Li, H. (2007). Moea/d: A multiobjective evolutionary algorithm based on decomposition. *IEEE Transactions on Evolutionary Computation*, *11*, 712–731.

Zitzler, E., & Knzli, S. (2004). *Indicator-Based Selection in Multiobjective Search*.

Zitzler, E., & Thiele, L. (1999). Multiobjective evolutionary algorithms: a comparative case study and the strength pareto approach. *IEEE Transactions on Evolutionary Computation*, *3*, 257–271.

Zou, X., Chen, Y., Liu, M., & Kang, L. (2008). A new evolutionary algorithm for solving many-objective optimization problems. *Systems Man & Cybernetics Part B Cybernetics IEEE Transactions on*, *38*, 1402–1412.