Soft Comput
DOI 10.1007/s00500-014-1234-8

METHODOLOGIES AND APPLICATION

MOEA/D with uniform decomposition measurement for many-objective problems

Xiaoliang Ma · Yutao Qi · Lingling Li · Fang Liu · Licheng Jiao · Jianshe Wu

Abstract Many-objective problems (MAPs) have put forward a number of challenges to classical Pareto-dominance based multi-objective evolutionary algorithms (MOEAs) for the past few years. Recently, researchers have suggested that MOEA/D (multi-objective evolutionary algorithm based on decomposition) can work for MAPs. However, there exist two difficulties in applying MOEA/D to solve MAPs directly. One is that the number of constructed weight vectors is not arbitrary and the weight vectors are mainly distributed on the boundary of weight space for MAPs. The other is that the relationship between the optimal solution of subproblem and its weight vector is nonlinear for the Tchebycheff decomposition approach used by MOEA/D. To deal with these two difficulties, we propose an improved MOEA/D with uniform decomposition measurement and the modified Tchebycheff decomposition approach (MOEA/D-UDM) in this paper. Firstly, a novel weight vectors initialization method based on the uniform decomposition measurement is introduced to obtain uniform weight vectors in any amount, which is one of great merits to use our proposed algorithm. The modified Tchebycheff decomposition approach, instead of the Tchebycheff decomposition approach, is used in MOEA/D-UDM to alleviate the inconsistency between the weight vector of subproblem and the direction of its optimal solution in the Tchebycheff decomposition approach. The proposed MOEA/D-UDM is compared with two state-of-the-art MOEAs, namely MOEA/D and UMOEA/D on a number of MAPs. Experimental results suggest that the proposed MOEA/D-UDM outperforms or performs similarly to the other compared algorithms in terms of hypervolume and inverted generational distance metrics on different types of problems. The effects of uniform weight vector initializing method and the modified Tchebycheff decomposition are also studied separately.

Keywords Many-objective optimization · Evolutionary algorithm · Decomposition · Uniform design · Uniform decomposition measurement

Communicated by Y. Jin.

X. Ma · Y. Qi · F. Liu (✉)
School of Computer Science and Technology, Xidian University, Xi'an 710071, China
e-mail: f63liu@163.com

X. Ma
e-mail: maxiaoliang@yeah.net

Y. Qi
e-mail: qi_yutao@163.com

X. Ma · Y. Qi · L. Li · F. Liu · L. Jiao · J. Wu
Key Laboratory of Intelligent Perception and Image Understanding of Ministry of Education of China, Xidian University, Xi'an 710071, China

L. Li
e-mail: linglingxidian@gmail.com

L. Jiao
e-mail: lchjiao@mail.xidian.edu.cn

J. Wu
e-mail: jshwu@mail.xidian.edu.cn

1 Introduction

Multi-objective evolutionary algorithms (MOEAs) have attracted more and more attention for their ability of generating a group of approximation solutions in a single run for multi-objective optimization problems (MOPs). It has achieved great success in solving the increasing problems from various fields such as finance, aeronautics, industry, computer, computer science and so on (Coello Coello et al. 2007). When MOEAs have shown their advantages in solving two or three objective problems, there are growing needs for

Published online: 26 February 2014

many-objective optimizations with four or more objectives (Fonseca and Fleming 1998; Fleming et al. 2005; Hughes 2007; Deb and Jain 2012a,b).

One of the main challenges in many-objective problems is that the ability of most existing MOEAs based on Pareto-dominance is severely deteriorated when the number of objectives increases (Ishibuchi et al. 2008; Saxena et al. 2013). Almost the whole population has the same rank of non-domination with an increase in the number of objective functions. This makes the main selection based on Pareto-dominance ineffective and the effect of diversity maintaining based secondary selection gets significant. Classical MOEAs based on Pareto-dominance, such as NSGA-II (Deb et al. 2002) and SPEA2 (Zitzler et al. 2002), deteriorate the situation by preferring the distant and boundary non-dominance solutions, which suggests that the solution with worse convergence becomes the point with better diversity (Saxena et al. 2013). This could explain the performance drop reported by Khare et al. (2003), Hughes (2005), Knowles and Corne (2007).

As suggested by many researches (Deb and Jain 2012a; Ishibuchi et al. 2011a,b; Tan et al. 2012), MOEA/D (Zhang and Li 2007; Li and Zhang 2009) has been widely used to handle the MAPs. It explicitly decomposes a MOP into a set of scalar optimization subproblems and solves them by evolving a population of solutions simultaneously. The objective of these subproblems is an aggregation of all the objective functions in the MOP. Compared with the dominance based MOEAs, MOEA/D is expected to have better performances in many-objective problems. The reason is that it uses uniformly distributed subproblems to keep the diversity of the current population. Moreover, MOEA/D adopts scalarizing function-based fitness evaluation scheme to make the incomparable solutions, based on domination principle, become comparable. Due to the good performance of MOEA/D, it has attracted more and more attention (Ishibuchi and Nojima 2010; Cheng et al. 2011; Sindhya et al. 2011).

Zhang and Li (2007) suggested that, in a sense, the uniformity of the weight vectors naturally leads to the diversity of the Pareto optimal solutions over the Pareto-optimal front (PF) for MOEA/D. Various methods to obtain the evenly distributed weight vectors can be found in the specialized literature (Cornell 1990). MOEA/D used the simplex-lattice design method (Scheffe 1958; Cornell 1990) to construct the weight vectors and obtained good results in many studies (Zhang and Li 2007; Ishibuchi et al. 2011a; Li and Zhang 2009). Although the weight vectors generated by simplex-lattice design seem to be uniformly scattered on the weight space, it may not work well for MAPs. The reason can be concluded as the following three aspects: (1) the number of weight vectors cannot be set at will. (2) too many weight vectors are distributed on the boundary of the weight space (Ning et al. 2011a), (3) the low dimensional projection of weight vectors in the simplex-lattice method is much overlap as shown in Figs. 2 (middle) and 4.

In order to construct uniformly distributed weight vectors in any amount, Tan et al. (2012) and Jiang et al. (2011) used a transformation method (Fang and Wang 1994; Fang and Lin 2003) to generate the weight vectors of subproblems. It achieves a good performance in their studies. However, the transformation method is only a general method, which cannot guarantee the uniformity of the constructed weight vectors (Ning et al. 2011a). Therefore, how to design uniform weight vectors on the weight space becomes the essential issue. Several uniform decomposition measurements defined on the weight space are proposed. Maximin-discrepancy (MD) and average discrepancy (AD), suggested by (Borkowski and Piepel 2009), measure the uniformity of weight vectors in the sense of Euclidean distance. DM_2-discrepancy and CDM_2-discrepancy (Ning 2008; Ning et al. 2011a) are introduced to measure the uniformity of low dimensional projection of the weight vectors.

Multi-objective evolutionary algorithm based on decomposition using uniform decomposition measurement and suitable decomposition approach (MOEA/D-UDM) is proposed in this paper. It firstly combines simplex-lattice design with transformation method to generate the alternative weight vectors and then selects uniform weight vectors from the alternative weight vectors based on the uniform design measurement. In order to obtain uniformly distributed Pareto optimal solutions over the PF, MOEA/D-UDM uses the modified Tchebycheff decomposition approach (Deb and Jain 2012a) instead of the Tchebycheff decomposition approach. Because the relationship between the weight vector of a subproblem and the direction of its optimal solution is the same for the modified Tchebycheff decomposition but nonlinear for the Tchebycheff decomposition (Qi et al. 2013; Liu et al. 2009; Deb and Jain 2012a). MOEA/D-UDM is expected to have superiority over MOEA/D in the following three aspects: (1) generate the weight vectors in any amount. (2) Reduce the number of weight vectors on the boundary of the weight space. (3) Lessen the overlap of the weight vectors in low dimensional projection. Therefore, our motivation is not to modify MOEA/D but to extend its application for many-objective problems.

The remainder of this paper is organized as follows. Section 2 describes related backgrounds including multi-objective optimization, decomposition of multi-objective optimization, uniform design methods for constructing the weight vectors and their drawbacks. Section 3 introduces the framework of MOEA/D-UDM. Section 4 compares MOEA/D-UDM with MOEA/D and UMOEA/D (Tan et al. 2012). Then, experimental results and analysis are given in this section. Section 5 concludes this paper.

2 Related backgrounds

2.1 Definition and notation of multi-objective optimization

MOP (Deb 2001) is to optimize a vector of functions simultaneously. The problem can be described as follows:

$$\begin{cases} \min \mathbf{F}(\mathbf{x}) = (f_1(\mathbf{x}), f_2(\mathbf{x}), \ldots, f_m(\mathbf{x}))^T \\ subject\ to:\ \mathbf{x} \in \Omega \end{cases} \tag{1}$$

where $\Omega \subset R^n$ is the feasible parameter space, and $\mathbf{x} \in \Omega$ is called decision vector. $\mathbf{F}(\mathbf{x}) : \Omega \rightarrow R^m$ consists of m objective functions that are conflicting with each other.

$\mathbf{x}_A \in \Omega$ is said to dominate the vector $\mathbf{x}_B \in \Omega$ (marked as $\mathbf{x}_A \prec \mathbf{x}_B$) if and only if $f_i(\mathbf{x}_A) \leq f_i(\mathbf{x}_B)$ for each $i \in \{1, \ldots, m\}$ and $f_j(\mathbf{x}_A) < f_j(\mathbf{x}_B)$ for at least one index $j \in \{1, \ldots, m\}$. If there does not exist any vector $\mathbf{x} \in \Omega$ such that $\mathbf{x} \prec \mathbf{x}^*$, we can say that the decision vector $\mathbf{x}^* \in \Omega$ is a Pareto optimal solution. Then the Pareto optimal set is named as $PS = \{\mathbf{x}^* \mid \neg\exists \mathbf{x} \in \Omega, \mathbf{x} \prec \mathbf{x}^*\}$. The corresponding mapping of the Pareto-optimal set on the objective function space is $PF = \{\mathbf{F}(\mathbf{x}) \mid \mathbf{x} \in PS\}$ stated as the Pareto optimal front. In the absence of the decision maker's preference information, the aim of a MOEA is to find a set of Pareto optimal solutions approximating the true Pareto optimal front.

2.2 MOEA/D

Multi-objective evolutionary algorithm based on decomposition (MOEA/D) was proposed by Zhang and Li (2007). It converts the MOP (1) into a number of scalar optimization subproblems using the following Tchebycheff approach:

$$\min_{\mathbf{x}\in\Omega} g^{tch}(\mathbf{x}|\mathbf{w}, \mathbf{z}^*) = \min_{\mathbf{x}\in\Omega} \max_{1\leq i\leq m} \{w_i |f_i(\mathbf{x}) - z_i^*|\} \tag{2}$$

where $\mathbf{w} = (w_1, w_2, \ldots, w_m)(\sum_{i=1}^m w_i = 1, w_i \geq 0, i = 1, \ldots, m)$ is the weight vector of the scalar optimization problem, and $\mathbf{z}^* = (z_1^*, z_2^*, \ldots, z_m^*)$ is the reference point (i.e., $z_i^* = \min\{f_i(\mathbf{x})|\mathbf{x} \in \Omega\}, i = 1, \ldots, m$ when the target of (1) is minimization).

These subproblems are optimized by evolving a set of solutions simultaneously. The population saves the best solution found so far for each subproblem at each generation. The neighborhood relations among these subproblems are based on the Euclidean distances between their weight vectors. It solves each subproblem by only using the information of its neighboring subproblems.

2.3 Uniform design methods for constructing the weight vectors

Under weak conditions, each optimal solution x^* of (2) is a Pareto optimal solution of (1). For each Pareto optimal solution x^*, there has a weight vector w such that x^* is the optimal solution of (2). Therefore, one can obtain different Pareto optimal solutions by changing their weight vectors. Zhang and Li (2007) suggested that the uniformity of the weight vectors will naturally lead to the uniformity of the Pareto optimal solutions along the PF. It is the uniformity of weight vectors that makes MOEA/D achieve success. Therefore, it is important to construct uniform weight vectors of the subproblems on the weight space $T^m = \{(w_1, w_2, \ldots, w_m)| \sum_{i=1}^m w_i = 1, w_i \geq 0, i = 1, \ldots, m\}$.

2.3.1 Classical methods for generating the uniform weight vectors

Over the past 60 years, many work have been proposed on how to generate the uniform weight vectors. The representative approaches include the simplex-lattice design (Scheffe 1958), the simplex-centroid design (Scheffe 1963), and axial design (Cornell 1975).

1. Simplex-lattice design
   Scheffe (1958) firstly introduced the simplex-lattice design to obtain the uniformly distributed weight vectors. Assume that the weight vector has m components and let H be a positive integer. The simplex, recorded as a $\{m, H\}$ simplex-lattice, is composed of experimental design points whose components have the following proportions:

$$\begin{cases} w_i \in \left\{\frac{0}{H}, \frac{1}{H}, \ldots, \frac{H}{H}\right\}, \quad i = 1, \ldots, m \\ \sum_{i=1}^m w_i = 1 \end{cases} \tag{3}$$

   And the $\{m, H\}$ simplex-lattice is composed of all possible combinations of the components where the proportions (3) for each component are adopted. Therefore, the number of weight vectors is C_{H+m-1}^{m-1} for the $\{m, H\}$ simplex-lattice. Figure 1 (left) gives the illustration of experimental design points in $\{3, 3\}$ simplex-lattice.
2. Simplex-centroid design
   Another popular design, introduced by Scheffe (1963), is the simplex-centroid design. In a m-components simplex-centroid design, the weight vectors consist of m vertexes of T^m, C_m^2 binary mixtures of the vertexes of T^m, C_m^3

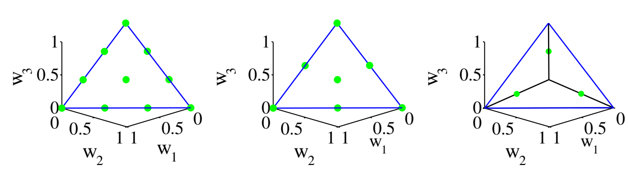

Fig. 1 Illustration of the generated weight vectors by the three classical uniform design methods, including simplex-lattice design with $H = 3$ (*left*), Simplex-centroid design (*middle*) and axial design (*right*), for tri-objective problems ($m = 3$)

ternary mixtures of the vertexes of T^m and so on. Therefore, there are $2^m - 1$ weight vectors in a m-components simplex-centroid design. Figure 1 (middle) gives the illustration of weight vectors in 3-components simplex-centroid design.

3. Axial design
Another popular design is axial design put forward by Cornell (1975). In axial design, an axis refers as the line segment connecting a vertex of the simplex T^m with its centroid. The weight vectors of the axial design are m points on m axes such that each point to the centroid is the same distance d, where $0 < d < \frac{m-1}{m}$ in general. Therefore, there are m weight vectors in a m-components axial design. Figure 1 (right) gives the illustration of weight vectors in 3-components axial design.

The number of weight vectors is restricted for the three classical methods (Tan et al. 2012). However, the needed number of weight vectors (i.e., the population size) is very flexible in the most experiments. So we need other methods to generate the weight vectors.

2.3.2 *Transformation method for generating the aggregation weight vectors*

The essence of uniform design is to find a set of aggregation weight vectors, which are uniformly distributed on the weight space T^m. The transformation method (Fang and Wang 1994; Fang and Yang 2000; Prescott 2008; Borkowski and Piepel 2009) was proposed for this purpose. It can construct the weight vectors with arbitrary amount. Let $P = \{\mathbf{x}^k = (x_1^k, ..., x_{m-1}^k)|k = 1, ..., N\}$ be a uniform points on $[0, 1]^{m-1}$. Each weight vector $\mathbf{w}_k = (w_1^k, ..., w_m^k)$, $k = 1, ..., m$ can be calculated by

$$\begin{aligned} w_i^k &= (1 - (x_i^k)^{\frac{1}{m-j}}) \prod_{j=1}^{i-1} (x_j^k)^{\frac{1}{m-j}}, \quad i = 1, ..., m-1 \\ w_m^k &= \prod_{j=1}^{m-1} (x_j^k)^{\frac{1}{m-j}} \end{aligned} \tag{4}$$

Then in a sense, the set $\{\mathbf{w}_k = (w_1^k, ..., w_{m-1}^k)|k = 1, ..., N\}$ will have a good approximation to the uniform weight vectors on T^m (Fang and Wang 1994). The weight vectors generated by the transformation method for 3-objective and 4-objective problems are shown in Fig. 3 (right).

3 The new proposal algorithm: MOEA/D-UDM

In this section, the basic idea of MOEA/D-UDM will be firstly described. The novel weight vectors initialization method, the adopted decomposition approach and the main framework of MOEA/D-UDM will be analyzed afterwards.

3.1 Basic idea of MOEA/D-UDM

Zhang and Li (2007) suggested that in order to obtain uniformly distributed Pareto optimal solutions over the PF, the decomposition method and the weight vectors should be properly chosen. Following this suggestion, the aims of this work are constructing evenly scattered weight vectors and selecting corresponding decomposition method.

The simplex-lattice design method obtains a great success in solving two or three objective problems (Zhang and Li 2007; Li and Zhang 2009). However, little focus has been placed on the ability of MOEA/D for many-objective problems. To the best of our knowledge, there exist two shortages by using MOEA/D for many-objective problems. The first one is that the weight vectors in simplex-lattice design can't be specified with an arbitrary number, which restricts the use of MOEA/D to a certain extent (Tan et al. 2012). The another is that the weight vectors are mainly distributed on the boundary of T^m leaving the interior of T^m lack of weight vectors. This case will make the optimal solutions chiefly distribute on the boundary of PF and have the interior of PF lack Pareto-optimal solutions. The reason is that the optimal solution of subproblem with boundary weight vector is intended to locate on the boundary of PF. In order to deal with these two drawbacks, Jiang et al. (2011) and Tan et al. (2012) used the transformation method to generate the weight vectors. The transformation method can generate the weight vectors with any number. However, there is no weight vector assigned on the boundary of weight space. There exists a very strong complementary relationship between the simplex-lattice design method and the transformation method. Therefore, we combine simplex-lattice design with transformation method to generate the alternative weight vectors and select the needed number of weight vectors based on the uniform design measurement from the alternative weight vectors. The potential idea, leaving weight vectors mainly in the interior and partially on the boundary, is to combine the advantages of two weight vectors constructing methods.

After constructing the uniform weight vectors, a suitable decomposition approach is needed to obtain uniformly distributed Pareto-optimal solutions over the PF. We suggest to use the modified Tchebycheff decomposition approach instead of Tchebycheff decomposition approach used in MOEA/D. Since the relationship between the weight vector of subproblem and the direction of its optimal solution is the same for the former while nonlinear for the later (Qi et al. 2013; Liu et al. 2009; Deb and Jain 2012a).

3.2 The novel weight vectors initialization method

In this section, we will introduce a method to obtain the uniformly distribued weight vectors. The constructed weight vectors have the following characteristics. (1) Whose number can be specified at will. (2) They are mainly distributed on the interior of weight space and partly assigned on the boundary of weight space. (3) The diversity of their low dimensional projection is as good as the uniformity of the weight vectors.

3.2.1 Uniform design measurements on the weight vector space

The uniform experimental design seeks its design points to be uniformly scattered on the experimental domain. Many different measures of uniformity for a set of points $\mathbf{P}_N = \{\mathbf{x}^k = (x_1^k, \ldots, x_m^k) | k = 1, \ldots, N\}$ are defined in the space $[0, 1]^m$ rather than T^m (Fang and Wang 1994; Fang and Lin 2003). The most popular uniformity measurement is the L_p-discrepancy (Hickernell 1998)

$$D_p(\mathbf{P}_N) = \left\{ \int\limits_{[0,1]^m} \left| V([\mathbf{0}, \mathbf{x})) - \frac{N(\mathbf{P}_N, [\mathbf{0}, \mathbf{x}))}{N} \right|^p d\mathbf{x} \right\}^{\frac{1}{p}}$$

where $[\mathbf{0}, \mathbf{x})$ denotes the interval $[0, x_1] \times \cdots \times [0, x_m]$, $N(\mathbf{P}_N, [\mathbf{0}, \mathbf{x}))$ is the number of points of $\mathbf{P}_N$ falling in $[\mathbf{0}, \mathbf{x})$, and $V(A)$ is the volume of A.

When applying the above measurements to generate weight vectors, their optimal designs tend to distribute most of the weight vectors on or near the boundary of T^m leaving the interior essentially lack of design points (Ning et al. 2011a). Due to the constrain condition $w_1 + \cdot + w_m = 1$, to find the weight vectors is quite different from uniform sampling on $[0, 1]^m$. It is necessary to design special criteria to measure the uniformity of weight vectors. Borkowski and Piepel (2009) suggested maximin-discrepancy (MD) and average discrepancy (AD), which measure the uniformity of weight vectors in the sense of Euclidean distance. In general, the metrics of MD and AD are calculated by approximate values. In order to calculate the exact value, Ning et al. (2011a) and Ning (2008) denote L2-discrepancy for mixture experimental design (DM_2) to measure the uniformity of generated weight vectors. Let $\mathbf{W} = \{\mathbf{w}^k = (w_1^k, \ldots, w_m^k) | k = 1, \ldots, N\}$ be a set of weight vectors. From the Fig. 2 (left), we can see that if the weight vectors are uniformly distributed on T^m, the rate between the number of weight vectors locating on $R(\mathbf{x}) = [\mathbf{0}, \mathbf{x})$ and N will be similar to that between volume of $R(\mathbf{x}) = [\mathbf{0}, \mathbf{x})$ and T^m for arbitrary $\mathbf{x}$. $DM_2(\mathbf{W})$ measures the expectation difference between the both rates. The calculate formula can be find in Ning et al. (2011a). The corresponding $CDM_2(\mathbf{W})$ is also defined as follows:

$$MD(\mathbf{W}) \approx \max_{1 \le k \le N} \left\{ \min_{1 \le j \le N, j \ne k} d(\mathbf{w}^j, \mathbf{w}^k) \right\} \tag{5}$$

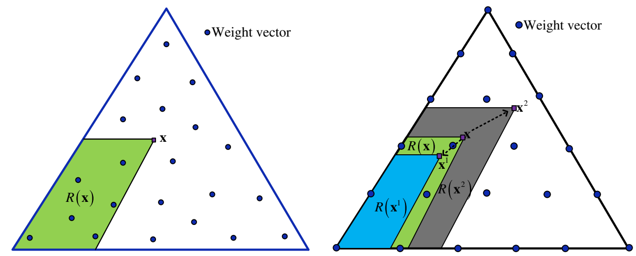

Fig. 2 Plot the dominated region $R(\mathbf{x}) = [\mathbf{0}, \mathbf{x})$ of $\mathbf{x}$ in weight space T^3 on the *left*. On the *right*, illustrate different dominated regions for simplex-lattice design

$$AD(\mathbf{W}) \approx \frac{1}{N} \sum_{k=1}^{N} \left\{ \min_{1 \le j \le N, j \ne k} d(\mathbf{w}^j, \mathbf{w}^k) \right\} \tag{6}$$

$$DM_2(\mathbf{W}) = \left[\int\limits_{T^m} \left| \frac{V(R(\mathbf{x}))}{V(T^m)} - \frac{N(\mathbf{W}, R(\mathbf{x}))}{N} \right|^2 d\mathbf{x} \right]^{\frac{1}{2}} \tag{7}$$

$$CDM_2(\mathbf{W}) = \frac{1}{m} \sum_{i=1}^{m} DM_2(\mathbf{W}\{i\}) \tag{8}$$

where $W\{i\}$, $i = 2, \ldots, m$ are new weight vectors which just exchange the first dimension with the i-th dimension of the weight vectors W. The higher values of the MD and AD metrics are preferred and the lower values of DM_2 and CDM_2 metrics are desired.

Figure 2 (right) illustrates the dominated regions for three different weight vectors $\mathbf{x}$, $\mathbf{x}_1$ and $\mathbf{x}_2$. If $\frac{V(R(\mathbf{x}))}{V(T^m)} \approx \frac{N(\mathbf{W}, R(\mathbf{x}))}{N}$, then the integral item $\left| \frac{V(R(\mathbf{x}))}{V(T^m)} - \frac{N(\mathbf{W}, R(\mathbf{x}))}{N} \right|^2$ in DM_2 can be small in the neighborhood of $\mathbf{x}$, while large for the neighborhood of $\mathbf{x}_1$ and $\mathbf{x}_2$ as shown in Fig. 2. The reason for this phenomenon is that each component of weight vector in simplex-lattice design has only $H + 1$ possible values. Form Fig. 2 (right) we can see that for simplex-lattice design, a small change of $\frac{V(R(\mathbf{x}))}{V(T^m)}$ has large effect on changing $\frac{N(\mathbf{W}, R(\mathbf{x}))}{N}$ from $\mathbf{x}$ to $\mathbf{x}_1$, while a large change of $\frac{V(R(\mathbf{x}))}{V(T^m)}$ has no effect on changing $\frac{N(\mathbf{W}, R(\mathbf{x}))}{N}$ from $\mathbf{x}$ to $\mathbf{x}_2$. Therefore, DM_2 metric value of the weight vectors generated by simplex-lattice design is large.

3.2.2 The framework of constructing weight vectors based on uniform design measurements

In this section, based on uniform design measurements, we combine the simplex-lattice design with transformation method to construct the uniformly distributed weight vectors. The detail of generating weight vectors is described in the Algorithm 1.

In step 1, the transformation method (4) (Tan et al. 2012; Fang and Wang 1994; Fang and Lin 2003; Jiang et al. 2011) is

Algorithm 1 Generating the weight vectors based on uniform decomposition measurement

Require: N : the number of the subproblems used in MOEA/D-UDM;
m : the number of objective functions in MOP (1);
r : the rate of boundary weight vectors;
Ensure: the constructed weight vectors $W_{selected}$.

1: **Step 1: Generate alternative weight vectors $W_{alternative}$, using the transformation approach**
  1.1 Use the good lattice method (Fang and Lin 2003) to generate a large number of uniformly distributed points P on $[0, 1]^{m-1}$, where $|P| >> N$.
  1.2 Apply the transformation method (4) (Fang and Wang 1994) on P to obtain nearly uniformly distributed alternative weight vectors $W_{alternative}$, where $|W_{alternative}| = |P| >> N$.
2: **Step 2: Apply uniform design measurements to select weight vectors from alternative weight vectors $W_{alternative}$**
  2.1 Initialize $W_{selected}$ as the set containing all vertex of T^m, i.e. $W_{selected} = \{(1, 0, ..., 0), (0, 1, ..., 0), ..., (0, 0, ..., 1)\}$.
  2.2 Find the element $w^{'} \in W_{alternative}$ with the optimal uniform design measurement using (5) or (6) or (7) or (8) to the selected weight vectors $W_{selected}$. Delete it from $W_{alternative}$ and add it to $W_{selected}$.
  2.3 If the size of $W_{selected}$ reaches the needed number $(1 - r) * N$, then go to Step 3. Otherwise, go to step 2.2.
3: **Step 3: Generate boundary weight vectors $W_{boundary}$, using the simplex-lattice design**
  3.1 Use the simplex-lattice design described in Sect. 2.3.1 to generate a large number of weight vectors W_{sld}.
  3.2 Select all the boundary weight vectors in W_{sld} as $W_{boundary}$.
4: **Step 4: Apply uniform design measurements to select weight vectors from the boundary weight vectors $W_{boundary}$**
  4.1 Find the element $\hat{w} \in W_{boundary}$ with the optimal uniform design measurement using (5) or (6) or (7) or (8) to the selected weight vectors $W_{selected}$. Then delete it from $W_{boundary}$. and add it to $W_{selected}$.
  4.2 If the size of $W_{selected}$ reaches the size of population N, then stop. Otherwise, go to step 4.1.

used to generate a large number of alternative weight vectors, whose number is beyond the population size N (for example 10 N), on the T^m. Then uniform design measure (Ning et al. 2011a; Borkowski and Piepel 2009) is applied to select the weight vectors from the set of alternative weight vectors until the number of selected weight vectors reaches the needed number (for example 0.9 N) in the step 2. In step 3, the simplex-lattice design is used to generate a large number of boundary weight vectors. Then uniform design measures is adopted to select the boundary weight vectors from the set of alternative boundary weight vectors until the number of selected weight vectors reaches the required number (for example $0.1N$) in step 4.

The weight vectors generated by simplex-lattice design, the transformation method and uniform decomposition based on measurements are shown in the Fig. 3 for 3–4 objective problems.

3.3 Compare UDM with simplex-lattice design and the transformation method

3.3.1 Flexibility of the number of constructed weight vectors

The numbers of weight vectors in simplex-lattice design are limited, i.e., the number of weight vectors is of the type C_{H+m-1}^{m-1} , where m is the number of objectives and H is a positive integer. However, the number of needed weight vectors (i.e., size of the evolutionary population) is reasonably flexible. With the number of objectives increasing, the limit will be worsened especially for many-objective problems.

On the contrary, the transformation method and the proposed UDM method can construct any number of weight vectors. We provide the explanation in Sect. 2.3.2 and Algorithm 1 respectively.

3.3.2 Rate of boundary weight vectors

Simplex-lattice design assigns too many weight vectors at the boundary of weight space T^m making the interior essentially lack of weight vectors (Ning et al. 2011a,b; Tan et al. 2012). On the contrary, the transformation method thought that the boundary weight vectors are insignificant leaving some weight vectors may be near but none on the boundary of weight space. The proposed UDM combines simplex-lattice design with the transformation method by letting the main weight vectors locate in the interior of weight space and a small amount of weight vectors distribute on the boundary of weight vector space.

The boundary weight vector refers to the weight vector which has zero weight value. The rate of boundary weight vectors in the $\{m, H\}$ simplex-lattice method is shown in Fig. 4 (left). Clearly, it increases with the number of objective functions m. For many-objective problems ($m \geq 4$), the rate of boundary weight vectors is excessively overabundance. There are two limitations brought by overmuch boundary weight vectors. Firstly, the optimal solution of subproblem with boundary weight vector is intended to locate the boundary of PF. It leaves the interior essentially lack of Pareto optimal solutions. However, in general, the most preferred solutions of decision maker (DM) are mainly located in the interior of PF. Secondly, It largely decreases the diversity of the optimal solutions of subproblems. It is due to that many subproblems with boundary weight vectors have the same optimal solution for the Tchebycheff decomposition (Deb and Jain 2012a). For example, the subproblems with boundary weight vectors in the type $(0, w_2, w_3, w_4)$ have the same optimal solution $(0.5, 0, 0, 0)$ in objective space for 4-objective DTLZ1 problem using the Tchebycheff decomposition approach, where $w_2 \neq 0, w_3 \neq 0, w_4 \neq 0$.

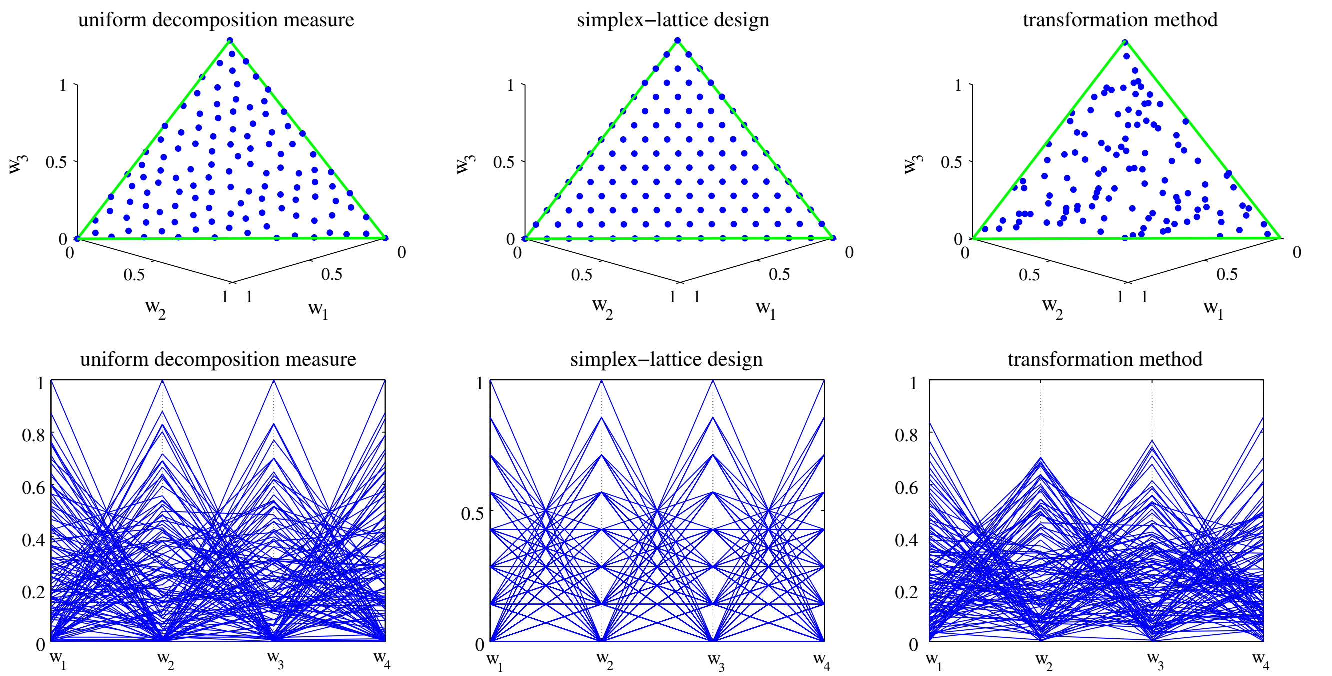

Fig. 3 Illustration the three methods, including uniform design based on measures (*left*), simplex-lattice design (*middle*) and the transformation method (*right*), scattering weight vectors on the weight space for 3-objective and 4-objective problems. Where the number of weight vectors is set as N = 120. In this case, the lower bound and upper bound of the generated weight vectors are $[0.0021, 0.0041, 0.0021]$ and $[0.9355, 0.9481, 0.9896]$ respectively for transformation method on three objectives, while [0, 0, 0] and [1, 1, 1] for simplex-lattice design and uniform design based on measures. The lower bound and upper bound of the generated weight vectors are $[0.0014, 0.0018, 0.0019, 0.0027]$ and $[0.8391, 0.7399, 0.7871, 0.8512]$ respectively for transformation method on four objectives, while [0, 0, 0, 0] and [1, 1, 1, 1] for simplex-lattice design and uniform design based on measures on four objectives

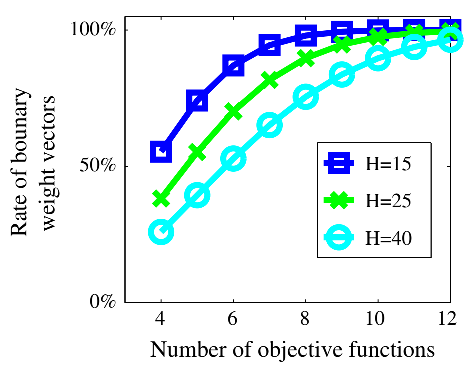

Fig. 4 Illustration the rate of boundary weight vectors for simplex-lattice design method on the left and the problem of “Curse of dimensionality” for constructing weight vector on the *right*

Table 1 The measurement values of the weight vectors in Fig. 3

Uniform design method	m	MD	AD	DM2	CDM2
Simplex-lattice design	3	**0.1010**	**0.1010**	0.0285	0.0285
	4	**0.0217**	**0.0217**	0.2020	24.244
The transformation method	3	0.0170	0.0495	0.0160	0.0159
	4	0.0075	0.0081	0.2068	18.655
Uniform design based on measure	3	0.0736	0.0816	**0.0152**	**0.0152**
	4	0.0158	0.0148	**0.2702**	**11.293**

The value of metric with bold is the best among the three compared algorithms

The transformation method thought that the boundary weight vectors are insignificant leaving no weight vector on the boundary of T^m. There are two limitations as follows. Firstly, the boundary weight vectors are important to extend the boundary of obtaining non-dominated solutions in the process of evolution, since the optimal solutions of the subproblems with boundary weight vectors tend to locate in the boundary of PF. Secondly, when a small population is used to evolve, it is risk for UMOEA/D missing some parts of boundary of PF especially for many-objective problems. In general, the PF of MOP is $m - 1$ dimension, the number of solutions required to approximate the whole PF increases exponentially with m (Saxena et al. 2013). Figure 3 (bottom right) shows the case that the weight vectors generated by transformation method misses the region with $w_i > 0.86, i = 1, ..., 4$.

3.3.3 *Uniformity and diversity of constructed weight vectors*

From the Fig. 3 and Table 1, we can see that

1. Simplex-lattice design

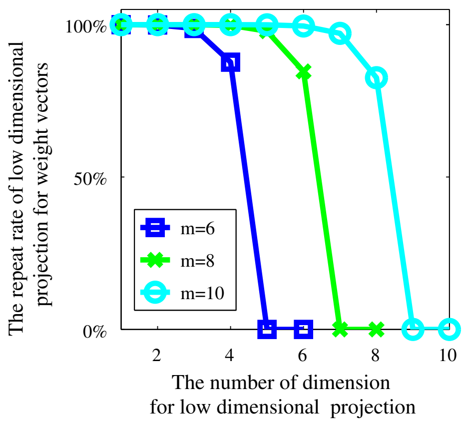

Fig. 5 Illustration the repeat rate of weight vectors constructed by simplex-lattice design in low dimensional projection

In the sense of Euclidean distance, the uniformity of the weight vectors generated by simplex-lattice design is the best on the weight space for the three methods as shown in the Fig. 3. MD and AD measurements, which are based on the Euclidean distance between weight vectors, also show the same result in the Table 1. However, the diversity of low dimensional projection of weight vectors in the simplex-lattice method is poor. Figure 3 (bottom middle) shows that the diversity of wight vectors, which is generated by simplex-lattice design, from w_1 to w_2, w_2 to w_3 and w_3 to w_4 is less than that of the other methods. Figure 5 shows the repeat rate of weight vectors in low dimensional projection for the $\{m, 10\}$ simplex-lattice. From the Fig. 5, we can clearly see that the repeat rate of weight vectors in low dimensional projection increases with the dimension of low dimensional projection decreasing. DM_2 and CDM_2 measurements, which pay more attention to the diversity of low dimensional projection of weight vector than the whole weight vector, show that it is the worst in three compared methods from the Table 1.

2. The transformation method
   The weight vectors generated by the transformation method (4) are uniform in a sense (Fang and Wang 1994). DM_2 and CDM_2 measurements show that the diversity of low dimensional projection of weight vectors using the transformation method is better than simplex-lattice design. Figure 3 can give a clear illustration.
3. Uniform decomposition measurement
   Uniform decomposition measurement (UDM) combines the both advantages of simplex-lattice design with the transformation method. In the sense of Euclidean distance, the uniformity of the weight vectors generated by uniform decomposition measure is quite close to the simplex-lattice design as shown in Fig. 3. DM_2 and CDM_2 measurements show that the diversity of low dimensional projection of weight vectors performs the best in the Table 1.

3.4 The adopted decomposition approach: modified Tchebycheff decomposition approach

After constructing the uniform weight vectors, a suitable decomposition approach is needed to obtain uniformly distributed optimal solutions over the PF. There are many approaches for converting a MOP into a number of scalar optimization problems. Tchebycheff decomposition approach is popular for its ability of handling the MOPs with non-convex PFs. However, for Tchebycheff decomposition approach, the uniformity of weight vectors still cannot guarantee that these optimal solutions uniformly distribute over the PF. The reason is that the relationship between weight vector and the direction of its optimal solution is nonlinear (Qi et al. 2013; Liu et al. 2009; Deb and Jain 2012a).

In order to ease the inconsistency between the weight vector of subproblem and the direction of its optimal solution, Deb and Jain (2012a) proposed an improvement to Tchebycheff decomposition approach, called modified Tchebycheff decomposition approach. They introduced a remedy to Tchebycheff decomposition approach by dividing the term with w_i, instead of multiplying w_i, in Eq. (2):

$$\min_{\mathbf{x}\in\Omega} g^{mtch}(\mathbf{x}|\mathbf{w}, \mathbf{z}^*) = \min_{\mathbf{x}\in\Omega} \max_{1\leq i\leq m} \left\{ \frac{|f_i(x) - z_i^*|}{w_i} \right\} \tag{9}$$

where $\mathbf{w}$ is a direction vector as well as a weight vector. For the difficulty $w_i = 0, i = 1, ..., m$, it can be handled by using a small number (for example, 10^{-6}) to replace zero.

According to the analysis in Qi et al. (2013) and Liu et al. (2009), if the straight line $\frac{|f_1(\mathbf{x})-z_1^*|}{w_1} = \frac{|f_2(\mathbf{x})-z_2^*|}{w_2} = \cdots = \frac{|f_m(\mathbf{x})-z_m^*|}{w_m}$ has an intersection with PF, then the intersection is the optimal solution to the subproblem with weight vector $\mathbf{w} = (w_1, w_2, ..., w_m)$ in the modified Tchebycheff decomposition approach. Therefore, the direction of the optimal solution is consisted with the weight vector, since the straight line $\frac{|f_1(\mathbf{x})-z_1^*|}{w_1} = \frac{|f_2(\mathbf{x})-z_2^*|}{w_2} = \cdots = \frac{|f_m(\mathbf{x})-z_m^*|}{w_m}$ across the reference point $\mathbf{z}^* = (z_1^*, z_2^*, ..., z_m^*)$ with the direction vector $\mathbf{w} = (w_1, w_2, ..., w_m)$. With the uniformly distributed weight vectors, the modified Tchebycheff decomposition approach is expected to obtain more uniform optimal solutions over the PF than the Tchebycheff decomposition approach. The reason is that the relationship between the weight vector of a subproblem and the direction of its optimal solution is consistent in the former while nonlinear in the later.

3.5 The framework of MOEA/D-UDM

MOEA/D-UDM has the same framework as MOEA/D. MOEA/D-UDM minimizes each subproblem using the information of its neighboring subproblems. For each subproblem, its neighborhood subproblems are defined as a set of the closest subproblems which have closest weight vectors to it in Euclidean distance. During the search procedure, MOEA/D-UDM maintains the following items:

- An evolutionary population on the decision space $\mathbf{x}^1, \ldots, \mathbf{x}^N \in \Omega$, where $\mathbf{x}^i$ is the best solution obtained so far to the i-th subproblem.
- An evolutionary population on the objective space $\mathbf{F}^1, \ldots, \mathbf{F}^N$, where $\mathbf{F}^i = \mathbf{F}(\mathbf{x}^i)$.
- Weight vectors $\{\mathbf{w}^1, \ldots, \mathbf{w}^N\}$, where $\mathbf{w}^i$ is the weight vector to the i-th scalar optimization subproblem.
- A reference point $\mathbf{z} = (z_1, \ldots, z_m)$ where z_i is the current optimal value obtained so far for the i-th objective.

There are only two strategies, which are different from MOEA/D (Zhang and Li 2007), in the framework of MOEA/D-UDM. One is the novel weight vectors which are constructed based on the uniform decomposition measurement. The other is the modified Tchebycheff decomposition approach instead of Tchebycheff decomposition approach. Here, our motivation is not to improve the framework of MOEA/D but to expand its application for many-objective optimization problems. The description of MOEA/D-UDM is given in Algorithm 2.

Algorithm 2 MOEA/D-UDM

Require: N : the size of population;
T : the neighborhood size of each subproblem;
Ensure: $\{\mathbf{x}^1, \cdots, \mathbf{x}^N\}$: the evolutionary population on the decision space;
$\{\mathbf{F}^1, \cdots, \mathbf{F}^N\}$: the evolutionary population on the objective space;

1: **Step 1: Initialization**

1.1 Construct the uniformly distributed weight vectors $\{\mathbf{w}^1, \cdots, \mathbf{w}^N\}$ by applying Algorithm 1. Calculate the Euclidean distances between any two weight vectors and then compute the T nearest weight vectors to each weight vector. For each $i \in 1, 2, \ldots, N$, Suppose $\mathbf{N}(i) = \{i_1, i_2, \ldots, i_T\}$, where $\{\mathbf{w}^{i_1}, \cdots, \mathbf{w}^{i_T}\}$ are the T nearest weight vectors to $\mathbf{w}^i$;

1.2 Produce the initial population $\mathbf{x}^1, \cdots, \mathbf{x}^N$ randomly and let $\mathbf{F}^i = \mathbf{F}(\mathbf{x}^i), i = 1, \ldots, m$;

1.3 Initialize the reference $\mathbf{z} = (z_1, \ldots, z_m)$ by setting $z_i = \min\{f_i(\mathbf{x}^1), \ldots, f_i(\mathbf{x}^N)\}$.

2: **Step 2: Evolution**

for each $i = 1, \ldots, N$, do

2.1 Reproduce: Let $r_1 = i$ and choose two indexes r_2 and r_3 from $N(i)$ randomly, produce a solution $\bar{x}$ from $\mathbf{x}_1^r$, $\mathbf{x}_2^r$ and $\mathbf{x}_3^r$ by applying the genetic operator to construct a new solution $\mathbf{x}'$.

2.2 Repair and evaluate: When an element of $\mathbf{x}'$ is out of Ω, its value will be considered as a chose value within Ω randomly. Let $\widehat{\mathbf{x}}$ be the repaired solution, and then evaluate the new solution $\mathbf{F}(\widehat{\mathbf{x}}) = (f_1(\widehat{\mathbf{x}}), \ldots, f_m(\widehat{\mathbf{x}}))$.

2.3 Update the reference: For each $j = 1, \ldots, m$, if $f_j(\widehat{\mathbf{x}}) < z_j$, then let $z_j = f_j(\widehat{\mathbf{x}})$.

2.4 Update the neighboring solutions based on the modified Tchebycheff approach: For each index $k \in N(i)$, if $g^{mtch}(\widehat{\mathbf{x}}|\mathbf{w}^k, \mathbf{z}) \leq g^{mtch}(\mathbf{x}^k|\mathbf{w}^k, \mathbf{z})$, then set $\mathbf{x}^k = \widehat{\mathbf{x}}$ and $\mathbf{F}^k = \mathbf{F}(\widehat{\mathbf{x}})$.

3: **Step 3: Stopping Criteria**

If the stopping standard is satisfied, then stop and output $\mathbf{x}^1, \cdots, \mathbf{x}^N$ and $\mathbf{F}^1, \cdots, \mathbf{F}^N$. Otherwise, go to Step 2.

4 Experimental study

In this section, we firstly compare MOEA/D-UDM with UMOEA/D (Tan et al. 2012) and MOEA/D (Zhang and Li 2007) for different types of many-objective problems. UMOEA/D is a new MOEA/D algorithm using transformation method to generate uniform weight vectors. Secondly, we study the effectiveness of two strategies including uniform weight vector design method based on measurements and the modified Tchebycheff decomposition approach. Thirdly, certain special types of problems are tested. The simulation codes of the compared approaches are developed by visual C++ 6.0 and run on a workstation with Inter Core6 2.8 GHz CPU and 32 GB RAM.

4.1 Multi-objective problems

In the experimental study, four widely used many-objective DTLZ problems (Deb et al. 2005) and six WFG problems (Huband et al. 2006) with 3–6 objectives are selected as the test problems. Detailed description of these problems can be found in Deb et al. (2005) and Huband et al. (2006), where the number of decision vectors is set as 12 for 3-objective DTLZ1-DTLZ4 and convex DTLZ2, 13 for 4-objective DTLZ1-DTLZ4 and convex DTLZ2, 14 for 5-objective DTLZ1-DTLZ4 and convex DTLZ2, 15 for 6-objective DTLZ1-DTLZ4 and convex DTLZ2, and 15 for 3–6 objective WFG4-WFG9 problems.

4.2 Performance metric

In the experimental studies of this work, the hypervolume (HV) and inverted generational distance (IGD) metrics are selected to compare the performance of the different algorithms quantitatively.

Hypervolume metric measures the volume of the region which is dominated by the obtained non-dominated solutions. Let $\{\mathbf{p}^1, \ldots, \mathbf{p}^N\}$ be the approximate non-dominated solutions of the PF and $\mathbf{r}$ be the reference point such that $\forall 1 \leq i \leq m, \mathbf{p}^i \prec \mathbf{r}$. Then the hypervolume of $\{\mathbf{p}^1, \ldots, \mathbf{p}^N\}$ can be defined as follows:

$$HV = volume\left(\bigcup_{i=1}^{N} v_i\right)$$

where v_i indicates a hyper-rectangle formed by the reference point $\mathbf{r}$ and the i-th non-dominated solution $\mathbf{p}^i$.

In this work, the reference point $\mathbf{r}$ is set as the maximum values of non-dominated solutions obtained by the compared algorithms.

The inverted generational distance (IGD) metric is used, which is a comprehensive index of convergence and diversity (Zitzler et al. 2003). Suppose $\bar{P}$ is a set of uniformly distributed solutions along the PF (on the objective space). Let $P^{'}$ be the approximate solutions of the PF. The average distance from $\bar{P}$ to $P^{'}$ is defined as:

$$IGD(\bar{P}, P^{'}) = \frac{\sum_{v\in\bar{P}} d(v, P^{'})}{|\bar{P}|} \quad (10)$$

where $d(v, P^{'})$ is the minimum Euclidean distance between v and the points in $P^{'}$. In a sense, $IGD(\bar{P}, P^{'})$ can measure both the uniformity and convergence of $P^{'}$ when $\bar{P}$ is large enough to represent the PF very well. To have a low value of $IGD(\bar{P}, P^{'})$, the set $P^{'}$ should be very near to the PF and cannot lose any part of the whole PF.

The number of the solutions in set $\bar{P}$ used to calculate the value of IGD is 2,000 for 4-objective problems, 5,000 for 5-objective problems and 10,000 for 6-objective problems. Each compared algorithm is independently run 30 times to calculate the statistical values.

4.3 Experimental setting

In our experimental study, the source code of MOEA/D (Zhang and Li 2007) can be found at the web site http://cswww.essex.ac.uk/staff/zhang/. UMOEA/D follows the implementation of Tan et al. (2012). The simulated binary crossover (SBX) operator and polynomial mutation (Deb and Beyer 2001) are used in MOEA/D-UDM, UMOEA/D and MOEA/D for solving the selected test problems. The parameter values are listed in Table 2, where n is the number of variables and $rand$ is a uniform random number in [0, 1].

Table 2 Parameter settings for SBX and PM

Parameter	MOEA/D-UDM	MOEA/D	UMOEA/D
Crossover probability p_c	1	1	1
Distribution index for crossover	20	20	20
Mutation probability p_m	1/n	1/n	1/n
Distribution index for mutation	20	20	20

Population size $N = 120$ for the 3-objective DTLZ, convex DTLZ2 and WFG problems. $N = 220$ for the 4-objective DTLZ, convex DTLZ2 and WFG problems. $N = 495$ for the 5-objective DTLZ, convex DTLZ2 and WFG problems. $N = 792$ for the 6-objective DTLZ, convex DTLZ2 and WFG problems.

All the compared algorithms quit when the number of function evaluations reaches the maximum value. The maximum number is set as 800,000 for 3-objective DTLZ1-DTLZ4, convex DTLZ2 and WFG4-WFG9 problems, 1,200,000 for 4-objective DTLZ1-DTLZ4, convex DTLZ2 and WFG4-WFG9 problems, 1,500,000 for 5-objective DTLZ1-DTLZ4, convex DTLZ2 and WFG4-WFG9 problems, 2,100,000 for 6-objective DTLZ1- DTLZ4, convex DTLZ2 and WFG4-WFG9 problems.

4.4 Experimental results of MOEA/D-UDM and comparisons

This part of experiments is designed to study the effectiveness of MOEA/D-UDM on different types of many-objective problems. The classical DTLZ problems are investigated firstly. And then performances of MOEA/D-UDM on WFG problems are studied. Thirdly, many-objective problem with convex PF is tested.

4.4.1 Comparison of performances on normalized test problems: DTLZ1-DTLZ4 problems

At first, we use 3–6 objective DTLZ1-DTLZ4 problems (Deb et al. 2005) as the test problems. In this paper, these problems are called 'normalized test problems' (Deb and Jain 2012a) since they have the same range of values for each objective (the PF of DTLZ1 is in the domain $[0, 0.5]^m$ and the PFs of DTLZ2-DTLZ4 lie in the domain $[0, 1]^m$).

Tables 3 and 4 respectively present the mean and standard deviation of the hypervolume and IGD values of the final solutions obtained by each algorithm for 3–6 objective DTLZ1-DTLZ4 problems. It is evident from Tables 3 and 4 that, the final solutions obtained by MOEA/D-UDM are better than UMOEA/D and MOEA/D for 3–6 objective normalized test problems as far as the hypervolume and IGD metrics are concerned, as far as the hypervolume and IGD metrics are concerned. It means that MOEA/D-UDM needs fewer function evaluations than MOEA/D to maximize the hypervolume value and minimize the IGD-metric value for the selected test problems. Table 4 also shows that UMOEA/D performs slightly better than MOEA/D in 3–6 objective DTLZ1-DTLZ4 problems in terms of IGD metric. This result was also reported in UMOEA/D (Tan et al. 2012). However, according to the hypervolume metric, Table 3 presents a bit different answer. The performance of MOEA/D is better than UMOEA/D for 3-objective and 5-

Table 3 The value of the hypervolume between the algorithms MOEA/D-UDM, UMOEA/D and MOEA/D for DTLZ1-DTLZ4 problems

HV	m	MOEA/D-UDM		MOEA/D		UMOEA/D	
		Mean	std	Mean	std	Mean	std
DTLZ1	3	**1.2767**	1.1517e−3	1.2718	5.8402e−4	1.26210	1.4880e−3
	4	**1.2829e−1**	1.0632e−4	1.2325e−1	1.2328e−4	1.2604e−1	1.4525e−4
	5	**1.6836**	3.7257e−4	1.6819	2.3674e−4	1.6803	7.0581e−4
	6	**5.3847e−1**	2.1147e−5	5.3803e−1	3.1672e−5	5.3819e−1	5.1240e−5
DTLZ2	3	**4.5142e−1**	1.7172e−3	4.1829e−1	1.2517e−3	4.2484e−1	1.2545e−3
	4	**6.2574e−1**	2.2596e−3	5.0137e−1	1.9916e−3	5.5939e−1	3.4627e−3
	5	**9.5963e−1**	1.9180e−3	7.9077e−1	1.6479e−3	8.9073e−1	2.8019e−3
	6	**9.4513e−1**	9.1571e−4	7.9911e−1	1.4170e−3	8.8099e−1	1.3938e−3
DTLZ3	3	**4.4674e+1**	3.2140e−2	4.4649e+1	1.6749e−2	4.4430e+1	3.6043e−2
	4	**3.0052e+1**	3.4925e−2	2.9940e+1	1.7827e−2	2.9911e+1	2.9924e−2
	5	**2.3995**	3.1730e−3	2.2310	3.2708e−3	2.3290	4.1047e−3
	6	**3.2539e+1**	6.4458e−3	3.2394e+1	6.6606e−3	3.2476e+1	8.4992e−3
DTLZ4	3	**4.2521e−1**	1.5729e−3	3.9235e−1	1.0432e−3	3.9952e−1	1.4794e−3
	4	**7.2818e−1**	2.3800e−3	6.0350e−1	2.0992e−3	6.6319e−1	3.5465e−3
	5	**7.2399e−1**	1.7411e−3	5.5265e−1	1.3567e−3	6.6609e−1	1.8653e−3
	6	**8.6161e−1**	9.3143e−4	7.1128e−1	1.3539e−3	8.1244e−1	1.8057e−3

The value of metric with bold is the best among the three compared algorithms

Table 4 The value of the IGD between the algorithms MOEA/D-UDM, UMOEA/D and MOEA/D for DTLZ1-DTLZ4 problems

IGD	m	MOEA/D-UDM		MOEA/D		UMOEA/D	
		Mean	std	Mean	std	Mean	std
DTLZ1	3	**1.7597e−2**	1.1689e−5	2.6680e−2	1.5298e−5	2.1399e−2	4.2563e−5
	4	**3.1093e−2**	8.4999e−6	7.3962e−2	2.4286e−5	3.8003e−2	8.6977e−5
	5	**3.8500e−2**	1.3207e−5	1.0423e−1	3.4305e−5	4.5539e−2	5.9535e−5
	6	**4.6554e−2**	1.9995e−5	1.1223e−1	1.2662e−5	5.2926e−2	3.7438e−5
DTLZ2	3	**5.0030e−2**	2.8377e−5	6.3302e−2	2.4241e−5	5.8073e−2	5.1984e−5
	4	**9.1252e−2**	6.7913e−5	1.9217e−1	9.9002e−5	1.1190e−1	1.9990e−4
	5	**1.2220e−1**	6.4824e−5	2.9578e−1	1.3786e−4	1.4456e−1	1.0993e−4
	6	**1.5719e−1**	8.7903e−5	3.3510e−1	2.1463e−5	1.7945e−1	1.6940e−4
DTLZ3	3	**5.0048e−2**	4.0864e−5	6.3292e−2	2.5071e−5	5.8066e−2	5.8840e−5
	4	**9.1363e−2**	8.4966e−5	1.9188e−1	2.0503e−4	1.1177e−1	2.9555e−4
	5	**1.2236e−1**	8.8210e−5	2.9532e−1	2.3616e−4	1.4431e−1	2.0970e−4
	6	**1.5776e−1**	1.5494e−4	3.3512e−1	6.5616e−5	1.7918e−1	2.4032e−4
DTLZ4	3	**5.0031e−2**	3.2331e−5	6.3293e−2	3.1993e−5	5.8080e−2	6.4039e−5
	4	**9.1391e−2**	7.1282e−5	1.9210e−1	1.3697e−4	1.1187e−1	1.6535e−4
	5	**1.2266e−1**	7.7961e−5	2.9546e−1	2.6265e−4	1.4430e−1	2.0003e−4
	6	**1.5830e−1**	1.5926e−4	3.3520e−1	1.4606e−4	1.7950e−1	2.4950e−4

The value of metric with bold is the best among the three compared algorithms

objective DTLZ1 problems and 3–4 objective DTLZ3 problems as far as the hypervolume metric is concerned. The reason may be that a boundary solution of PF is intended to contribute larger exclusive hypervolume value than a interior solution of PF. MOEA/D can obtain more boundary solutions of PF than UMOEA/D since simplex-lattice design used by MOEA/D distributes more weight vectors on the boundary of weight space than the transformation method applied by UMOEA/D, especially for many-objective problems as analyzed in Sect. 3.3.

In order to help the reader understand the effectiveness of the compared methods, we give the graphical representation of Pareto front for tri-objective problems. Figure 6 shows the obtained solutions with biggest hypervolume values on

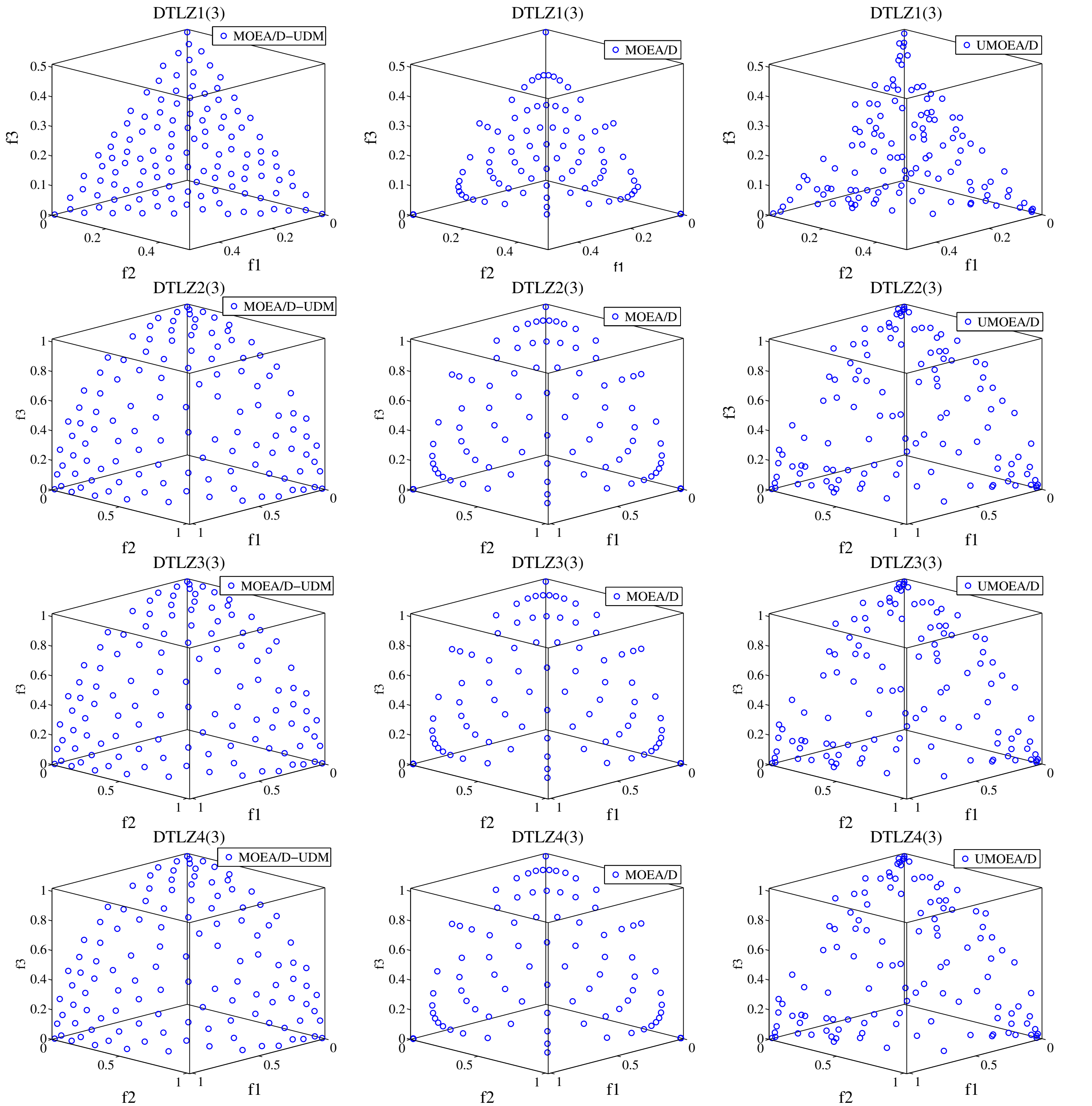

Fig. 6 The distribution of the obtained solutions with biggest hypervolume values found by MOEA/D-UDM, MOEA/D and UMOEA/D in solving the tri-objective DTLZ1-DTLZ4 test problems

tri-objective DTLZ1-DTLZ4 in objective space. The PF of DTLZ1 problem is a hyper-plane in the first quadrant. The PFs of DTLZ2-DTLZ4 are unit spheres in the first quadrant. The weight vectors used by each algorithm are shown in Fig. 3 (upper). It is clear that MOEA/D-UDM is significantly better than the original MOEA/D and UMOEA/D as far as the uniformity of final solutions is concerned. The solutions obtained by MOEA/D distribute with some regularity. This observation is similar to the result in the original MOEA/D (Zhang and Li 2007). The solutions found by UMOEA/D have a good diversity. But its uniformity is not very good.

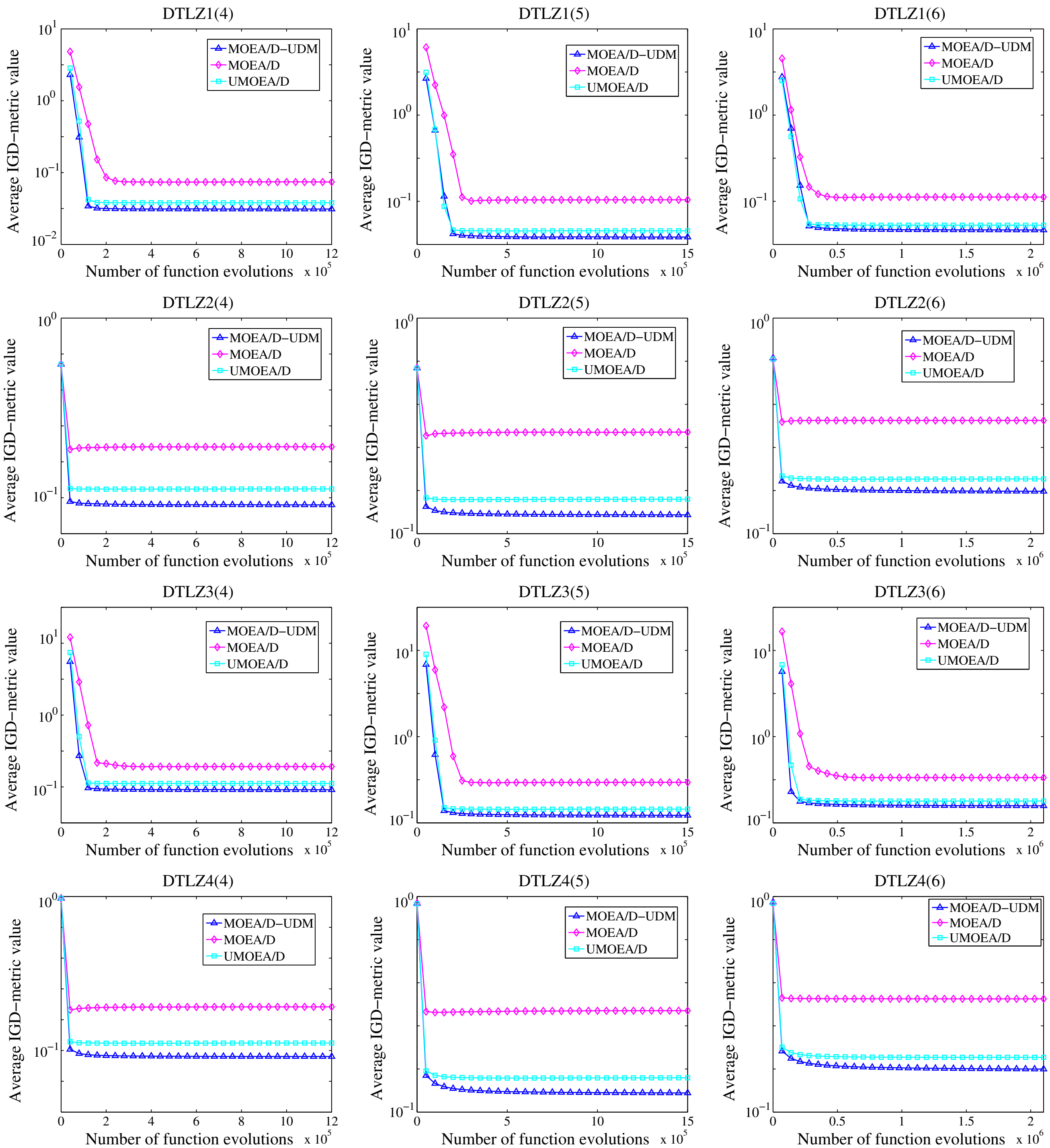

Fig. 7 Evolution of the mean of IGD-metric values for 4–6 objective DTLZ1-DTLZ4 test problems

Figure 7 plots the evolution of the average IGD-metric values of the current population for 4–6 objective DTLZ1-DTLZ4 problems respectively. These results show that MOEA/D-UDM converges much faster than MOEA/D and UMOEA/D in minimizing the IGD-metric value for the normalized test problems, which suggests that MOEA/D-UDM is more efficient and effective than UMOEA/D and MOEA/D for these 4–6 objective DTLZ1-DTLZ4 problems. It is due to that MOEA/D-UDM adopts the uniformly distributed weight vectors and suitable decomposition approach. Figure 7 also shows that UMOEA/D performs better than MOEA/D for 4–6 objective DTLZ1-DTLZ4 problems in terms of IGD met-

Table 5 The value of the Hypervolume between the algorithms MOEA/D-UDM, UMOEA/D and MOEA/D for WFG4-WFG9 problems

HV	m	MOEA/D-UDM		MOEA/D		UMOEA/D	
		Mean	std	Mean	std	Mean	std
WFG4	3	**2.0628e+1**	7.8003e−2	1.9048e+1	5.5268e−2	1.9398e+1	6.4214e−2
	4	**2.2650e+2**	8.9119e−1	1.7891e+2	8.3683e−1	2.0136e+2	2.0965
	5	**2.8409e+3**	7.2619	2.1952e+3	5.1260	2.6095e+3	8.0002
	6	**3.8976e+4**	6.0513e+1	3.2298e+4	5.9572e+1	3.6495e+4	1.0462e+2
WFG5	3	**2.1918e+1**	1.8436e−1	2.0566e+1	2.2169e−1	2.0713e+1	1.2315e−1
	4	**2.1892e+2**	1.7453	1.7129e+2	5.1674e	1.9633e+2	1.6526
	5	**2.7608e+3**	2.1588e+1	2.0016e+3	1.1951e+2	2.4981e+3	1.0239e+1
	6	**3.7258e+4**	5.7016e+2	2.7136e+4	2.0626e+3	3.4305e+4	1.2296e+2
WFG6	3	**2.3586e+1**	6.9424e−1	2.2265e+1	7.7789e−1	2.2598e+1	7.1919e−1
	4	**2.4218e+2**	6.3865	1.9650e+2	6.0767	2.1373e+2	6.1894
	5	**2.9340e+3**	5.1017e+1	2.2989e+3	7.2910e+1	2.6724e+3	4.0319e+1
	6	**3.9518e+4**	1.6587e+2	3.3077e+4	6.5028e+2	3.6649e+4	3.5338e+2
WFG7	3	**2.0682e+1**	7.9743e−2	1.9080e+1	5.7567e−2	1.9469e+1	8.8913e−2
	4	**2.2558e+2**	9.1079e−1	1.7753e+2	8.3906e−1	2.0183e+2	1.1817
	5	**2.8119e+3**	6.7492	2.1585e+3	4.9457	2.5927e+3	9.2266
	6	**3.7878e+4**	4.2984e+1	3.1115e+4	5.6432e+1	3.5617e+4	6.9392e+1
WFG8	3	**2.1737e+1**	1.8547e−1	2.0423e+1	1.8373e−1	2.0952e+1	2.4595e−1
	4	**2.2576e+2**	1.0422	1.6650e+2	2.5277	2.0710e+2	1.3684
	5	**2.8745e+3**	7.6075	1.6326e+3	6.1114e+1	2.6633e+3	1.2988e+1
	6	**3.9392e+4**	7.9001e+1	2.3496e+4	1.2765e+3	3.7115e+4	1.0285e+2
WFG9	3	**2.5454e+1**	1.7261	2.4120e+1	1.6855	2.4800e+1	1.2183
	4	**2.7691e+2**	1.8559e+1	2.2414e+2	2.9669	2.6377e+2	1.3389e+1
	5	**2.6693e+3**	2.3281e+1	1.5826e+3	8.6413e+1	2.5463e+3	1.7772e+1
	6	**3.5863e+4**	2.0787e+2	1.7269e+4	1.6462e+3	3.4678e+4	2.1642e+2

The value of metric with bold is the best among the three compared algorithms

ric. UMOEA/D (Tan et al. 2012) also gave a similar result. Although it is difficult to visualize, a small IGD value in each case refers to a well-distributed set of solutions.

4.4.2 *Comparison of performances on scaled test problems: WFG4-WFG9 problems*

To observe the algorithm's performance on problems having differently scaled objective values, WFG4-WFG9 problems are selected as the test problems. The PFs of WFG4-WFG9 are spheroids with $\mathbf{v} = (2, 4, \ldots, 2m)$ as the semi-major axis vector in the first quadrant. Each objective $f_i, i = 1, \ldots, m$ of PF lies in $[0, 2i]$ for WFG4-WFG9 problems.

To handle different scaling of objectives, the objective normalization technique as suggested in MOEA/D (Zhang and Li 2007) is used.

Tables 5 and 6 respectively show the mean and standard deviation of the hypervolume and IGD values of the final solutions obtained by each algorithm for 3–6 objective WFG4-WFG9 problems. It can be seen from Tables 5 and 6 that MOEA/D-UDM performs the best in three compared algorithms. It obtains bigger hypervolume and smaller IGD values for 3–6 objective scaled test problems. In terms of IGD metric, the performance of MOEA/D-UDM is much better than that of MOEA/D especially for 4–6 objective WFG4-WFG9 test problems. Tables 5 and 6 also suggest that UMOEA/D performs better than MOEA/D in 3–6 objective WFG4-WFG9 problems in terms of hypervolumer and IGD metrics.

Figure 8 shows in objective space, the obtained solutions with biggest hypervolume values found by each algorithm in solving the tri-objective WFG4-WFG9 test problems. The weight vectors used by each algorithm are shown in Fig. 3 (upper). It is clear that MOEA/D-UDM is better than the original MOEA/D and UMOEA/D as far as the uniformity of final solutions is concerned. The solutions obtained by MOEA/D distribute with some regularity. This observation is similar to the result that showed in the original MOEA/D (Zhang and Li 2007). The uniformity of solutions found by UMOEA/D is not very good.

Table 6 The value of the IGD between the algorithms MOEA/D-UDM, UMOEA/D and MOEA/D for WFG4-WFG9 problems

IGD	m	MOEA/D-UDM		MOEA/D		UMOEA/D	
		Mean	std	Mean	std	Mean	std
WFG4	3	**1.9869e−1**	2.5701e−4	2.4836e−1	1.8625e−4	2.2913e−1	4.3191e−4
	4	**4.4644e−1**	6.3335e−4	9.1009e−1	2.9544e−3	5.3822e−1	2.6528e−3
	5	**7.0978e−1**	1.5377e−3	1.7230	1.7699e−3	8.0218e−1	2.7822e−3
	6	**1.0493**	2.4190e−3	2.4302	3.5099e−3	1.1673	3.2371e−3
WFG5	3	**2.1432e−1**	1.9258e−3	2.5805e−1	2.9421e−3	2.4907e−1	1.2026e−3
	4	**4.6330e−1**	1.9221e−3	8.7639e−1	2.9506e−2	5.4893e−1	2.4582e−3
	5	**7.2521e−1**	2.8216e−3	1.6879	2.8870e−2	8.2340e−1	1.7263e−3
	6	**1.0670**	5.7322e−3	2.3695	4.7496e−2	1.1865	2.3710e−3
WFG6	3	**2.0941e−1**	6.8731e−3	2.5740e−1	7.0841e−3	2.3916e−1	7.5142e−3
	4	**4.5124e−1**	3.7243e−3	9.0460e−1	1.6024e−2	5.4914e−1	5.7662e−3
	5	**7.1620e−1**	5.7546e−3	1.7157	2.0221e−2	8.1276e−1	3.5591e−3
	6	**1.0555**	6.2477e−3	2.4079	3.6575e−2	1.1773	3.7382e−3
WFG7	3	**1.9917e−1**	2.5254e−4	2.4829e−1	1.4603e−4	2.2924e−1	2.5390e−4
	4	**4.4811e−1**	5.8268e−4	9.1005e−1	2.0634e−3	5.3873e−1	1.7869e−3
	5	**7.1012e−1**	1.0072e−3	1.7206	2.9405e−3	7.9911e−1	1.9503e−3
	6	**1.0525**	1.7846e−3	2.4317	4.8342e−3	1.1595	2.2550e−3
WFG8	3	**2.2980e−1**	5.4934e−3	2.6784e−1	4.6078e−3	2.4901e−1	5.7707e−3
	4	**5.0691e−1**	6.0238e−3	8.7683e−1	1.3476e−2	5.4982e−1	4.0241e−3
	5	**7.7061e−1**	3.9662e−3	1.7816	3.0155e−2	8.1575e−1	3.6613e−3
	6	**1.1389**	8.4181e−3	2.7538	8.3527e−2	1.1877	3.7205e−3
WFG9	3	**2.1273e−1**	3.1899e−2	2.4973e−1	3.0612e−2	2.3785e−1	2.2871e−2
	4	**4.9754e−1**	3.2155e−2	8.1323e−1	1.1244e−2	5.5196e−1	2.3190e−2
	5	**8.0401e−1**	1.4053e−2	1.6603	2.5566e−2	8.2759e−1	3.3368e−3
	6	**1.2001**	1.6095e−2	2.8528	5.5394e−2	1.2050	3.1023e−3

The value of metric with bold is the best among the three compared algorithms

Figure 9 plots the evolution of the average IGD-metric values of the current evolutionary population for continuous WFG4-WFG9 problems with 4–6 objectives respectively. These results show that MOEA/D-UDM converges much faster than MOEA/D and UMOEA/D in minimizing the IGD-metric value for the selected test instances, which suggests that MOEA/D-UDM is more efficient and effective than UMOEA/D and MOEA/D for these 4–6 objective scaled test problems. Figure 9 also shows that UMOEA/D performs better than MOEA/D for 4–6 objective WFG4-WFG9 problems in terms of IGD metric.

4.4.3 Comparison of performances on convex DTLZ2 problems

Having studied the problems, such as DTLZ1-DTLZ4 and WFG4-WFG9, with concave PFs, one may ponder whether the proposed MOEA/D-UDM can handle the many-objective problems with convex PFs. To investigate the performance of MOEA/D-UDM on such a case, we use convex DTLZ2 problem suggested by Deb and Jain (Deb and Jain 2012a). The description of convex DTLZ2 problem can be found in (Deb and Jain 2012a). Figure 10 (left) shows the PF of 3-objective convex DTLZ2 problem, which is almost flat at the edges but changes sharply in the middle.

Table 7 shows the mean and standard deviation of the hypervolume of the obtained solutions for all three methods. Clearly, MOEA/D-UDM performs well by finding a set of solutions with bigger hypervolume values for 3–6 objective convex DTLZ2 problems.

Figure 10 (right) plots 120 obtained solutions on a 3-objective convex PF problem. Although the weight vectors are uniformly scattered on the weight space, the distribution of obtained solutions on the objective space is not uniform. MOEA/D-UDM is able to find a well-distributed set of solutions on the edges as well as in the middle of the surface. As shown in Fig. 10, the obtained solutions by MOEA/D distribute with some rule and mainly assembled on the middle of PF. UMOEA/D obtains a diverse solutions but its uniformity is not very well.

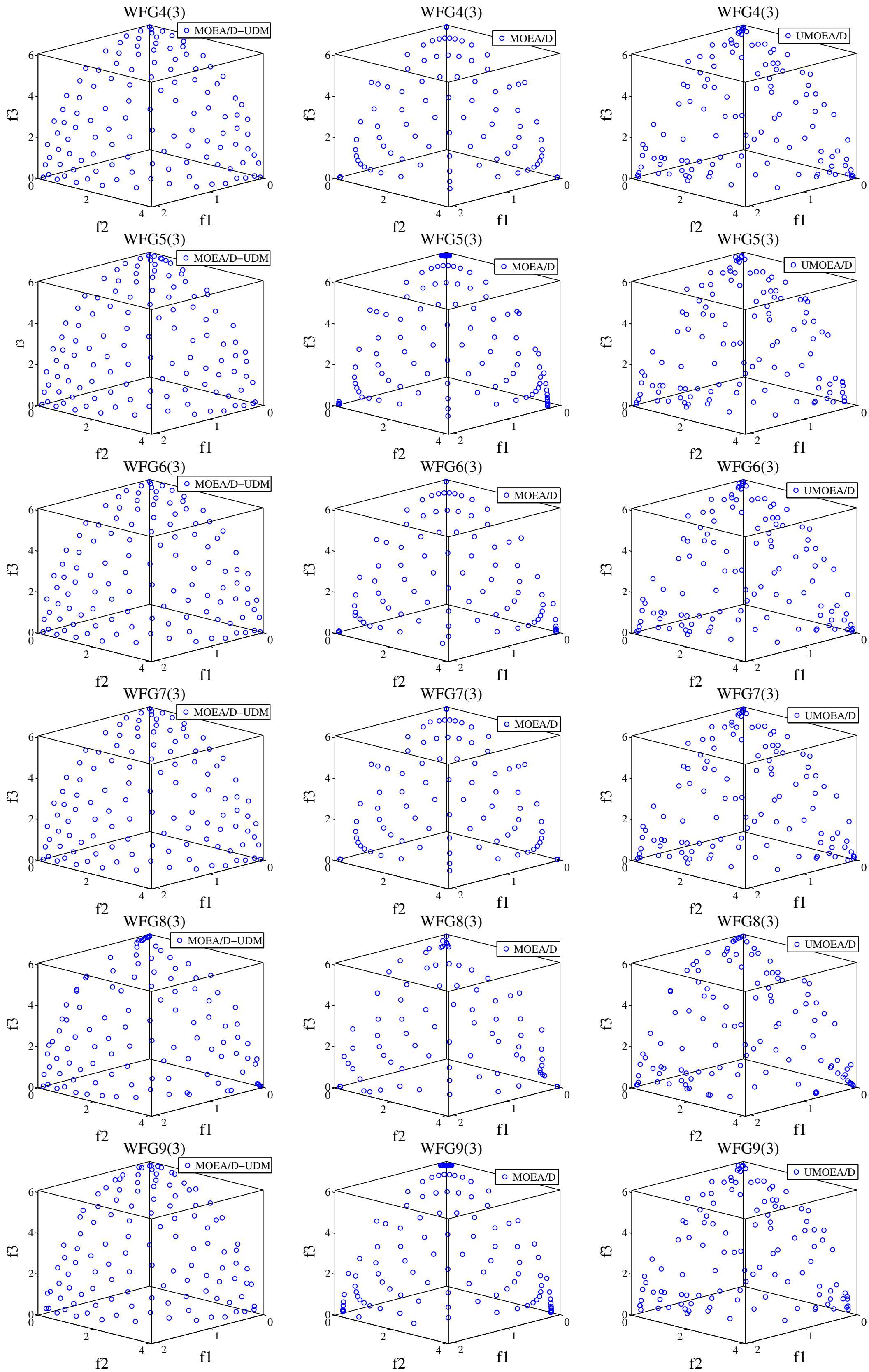

Fig. 8 The distribution of the obtained solutions with biggest hypervolume values found by MOEA/D-UDM, MOEA/D and UMOEA/D in solving the tri-objective WFG4-WFG9 test problems

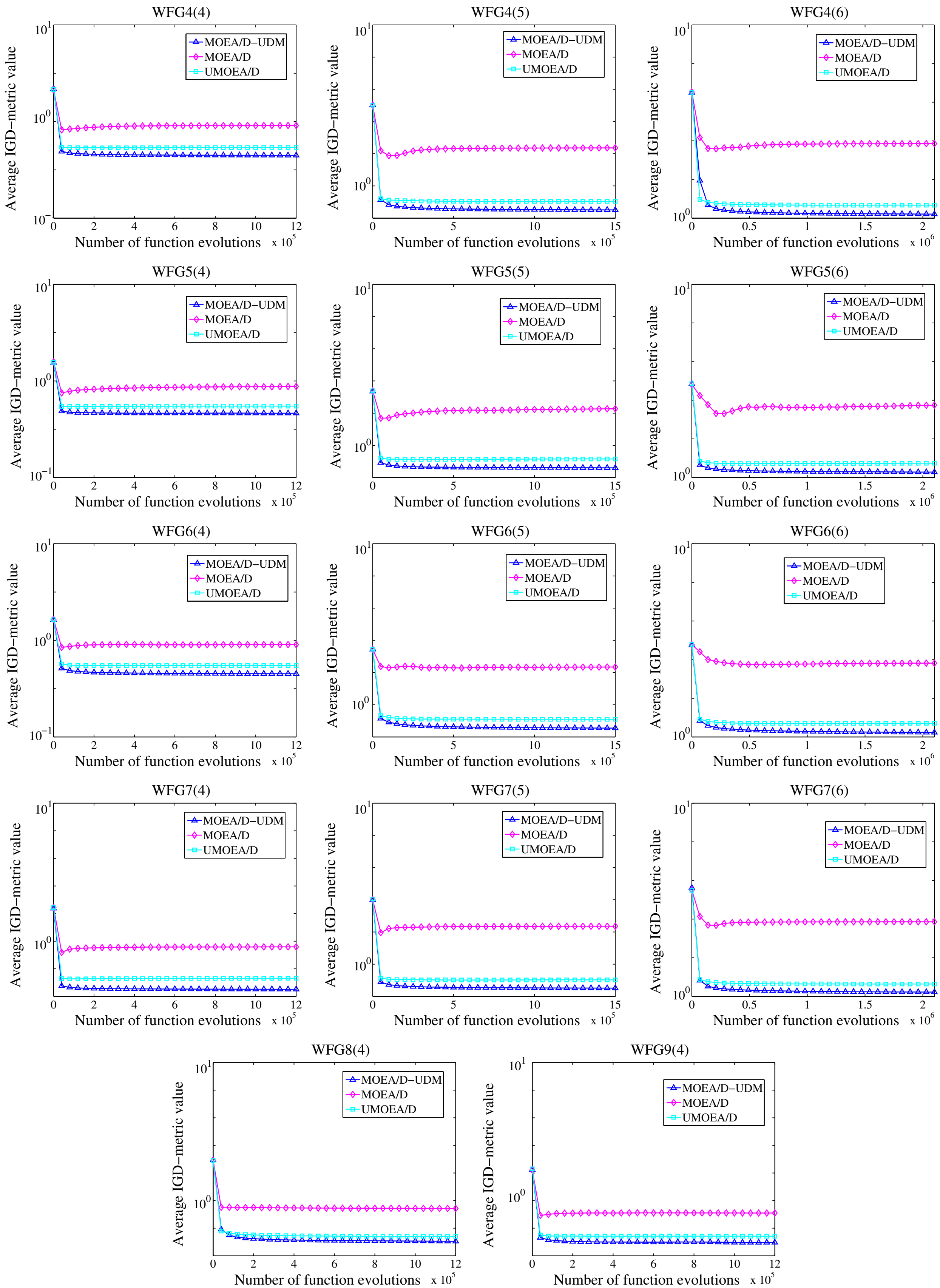

Fig. 9 Evolution of the mean of IGD-metric values for 4–6 objective WFG4-WFG9 test problems

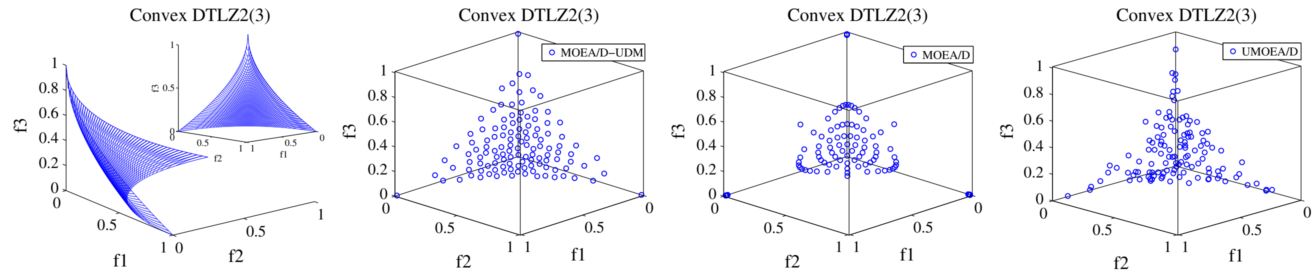

Fig. 10 *Left* plot of the PF of convex DTLZ2 problem. The others: the distribution of the obtained solutions with biggest hypervolume values found by MOEA/D-UDM, MOEA/D and UMOEA/D in solving the tri-objective convex DTLZ2 problem

Table 7 The value of the hypervolume between the algorithms MOEA/D-UDM, MOEA/D and UMOEA/D for convex DTLZ2 problems

HV	m	MOEA/D-UDM		MOEA/D		UMOEA/D	
		Mean	std	Mean	std	Mean	std
Convex DTLZ2	3	**9.3253e−1**	7.1416e−4	9.0613e−1	6.7073e−4	9.2008e−1	5.5299e−4
	4	**1.1128**	2.8083e−4	1.0983	3.9276e−4	1.1088	4.7672e−4
	5	**9.7251e−1**	9.1759e−5	9.6972e−1	1.5153e−4	9.7084e−1	1.7538e−4
	6	**2.4777**	4.9403e−5	2.4770	7.9878e−5	2.4773	1.4383e−4

The value of metric with bold is the best among the three compared algorithms

4.5 Effectiveness of the two proposed strategies

In this section, we empirically study the effectiveness of the two proposed modifies. Two variants of MOEA/D-UDM, namely UDM-MOEA/D-TCH and MOEA/D-MTCH were developed to carry out the study. MOEA/D-MTCH is the same as MOEA/D-UDM and the new weight vector initialization is not applied, In UDM-MOEA/D-TCH, the modified Tchebycheff decomposition approach is not used.

Table 8 presents the mean and standard deviation of the IGD-metric values of the final solutions obtained by each algorithm for 4–6 objectives DTLZ1-DTLZ4 and WFG4-WFG9 problems.

Table 8 shows that, in terms of IGD metric, using the same decomposition approach, UDM-MOEA/D-TCH and MOEA/D-UDM are respectively better than MOEA/D and MOEA/D-MTCH for all the selected test problems, which suggests that the strategy of generating weight vectors based on uniform design measurements is more efficient and effective than simplex-lattice design for 4–6 objective DTLZ1-DTLZ4 and WFG4-WFG9 problems. Simplex-lattice design used by MOEA/D assigns too many weight vectors on the boundary, which lets the interior of weight space lack of weight vectors especially for many-objective problems. So the diversity of optimal solutions using simplex-lattice design will be reduced.

It is clear that, from the Table 8, using the same weight vectors, MOEA/D-MTCH and MOEA/D-UDM are respectively better than MOEA/D and UDM-MOEA/D-TCH for the selected problems based on the values of the IGD metric. The reason is analyzed in the Sect. 3.3 that the relationship between weight vector and the direction of its optimal solution is nonlinear for Tchebycheff decomposition approach while the same for the modified Tchebycheff decomposition approach.

Overall, the proposed strategies of constructing weight vectors based on the uniform design measurement and the modified Tchebycheff decomposition approach are necessary and effective for our proposed MOEA/D-UDM according to the experiment studies.

4.6 Further investigations of MOEA/D-UDM

In this section, we want to investigate MOEA/D-UDM's performance in certain special types of many-objective problems.

4.6.1 Finding a preference part of PF

In many-objective optimization problems, there still exist quite a few difficulties, such as visualization of higher-dimensional PF, the number of points needed to approximate the whole PF increasing exponentially with m, and so on (Saxena et al. 2013). Therefore, the decision-maker(DM) may not always be interested in searching the whole PF. Naturally, some desired parts of PF may be more interested to the DM. Specially, he/she may offer his/her preference by using a few representative reference directions (weight vectors).

Table 8 The value of the IGD between the algorithm MOEA/D-UDM, UDM-MOEA/D-TCH, MOEA/D-MTCH and MOEA/D

IGD	m	MOEA/D-UDM		UDM-MOEA/D-TCH		MOEA/D-MTCH		MOEA/D	
		Mean	std	Mean	std	Mean	std	Mean	std
DTLZ1	3	**1.7597e−2**	1.1689e−5	2.1710e−2	2.0088e−5	1.8509e−2	2.1605e−5	2.6680e−2	1.5298e−5
	4	**3.1093e−2**	8.4999e−6	4.0582e−2	4.8426e−5	3.6223e−2	4.1389e−3	7.3962e−2	2.4286e−5
	5	**3.8500e−2**	1.3207e−5	4.9322e−2	5.7816e−5	5.1629e−2	5.3196e−3	1.0423e−1	3.4305e−5
	6	**4.6554e−2**	1.9995e−5	5.8567e−2	7.5443e−5	6.7401e−2	6.8614e−3	1.1223e−1	1.2662e−5
DTLZ2	3	**5.0030e−2**	2.8377e−5	6.0431e−2	4.8851e−5	5.3405e−2	6.8623e−6	6.3302e−2	2.4241e−5
	4	**9.1252e−2**	6.7913e−5	1.1934e−1	1.7485e−4	9.5047e−2	5.3320e−3	1.9217e−1	9.9002e−5
	5	**1.2220e−1**	6.4824e−5	1.5870e−1	2.1621e−4	1.5018e−1	1.1455e−2	2.9578e−1	1.3786e−4
	6	**1.5719e−1**	8.7903e−5	2.0059e−1	3.5796e−4	2.0040e−1	1.1201e−2	3.3510e−1	2.1463e−5
DTLZ3	3	**5.0048e−2**	4.0864e−5	6.0444e−2	7.9997e−5	5.3416e−2	8.3829e−6	6.3292e−2	2.5071e−5
	4	**9.1363e−2**	8.4966e−5	1.1929e−1	2.2906e−4	1.1106e−1	1.1589e−2	1.9188e−1	2.0503e−4
	5	**1.2236e−1**	8.8210e−5	1.5879e−1	3.6764e−4	1.7194e−1	1.7573e−2	2.9532e−1	2.3616e−4
	6	**1.5776e−1**	1.5494e−4	2.0115e−1	3.3529e−4	2.3230e−1	2.3823e−2	3.3512e−1	6.5616e−5
DTLZ4	3	**5.0031e−2**	3.2331e−5	6.0435e−2	3.5248e−5	5.3408e−2	8.4033e−6	6.3293e−2	3.1993e−5
	4	**9.1391e−2**	7.1282e−5	1.1930e−1	1.4982e−4	9.2612e−2	2.5889e−3	1.9210e−1	1.3697e−4
	5	**1.2266e−1**	7.7961e−5	1.5900e−1	2.3522e−4	1.3745e−1	5.8484e−3	2.9546e−1	2.6265e−4
	6	**1.5830e−1**	1.5926e−4	2.0289e−1	3.2091e−4	1.7956e−1	8.4977e−3	3.3520e−1	1.4606e−4
WFG4	3	**1.9869e−1**	2.5701e−4	2.4223e−1	6.0435e−4	2.4569e−1	5.9484e−4	2.4836e−1	1.8625e−4
	4	**4.4644e−1**	6.3335e−4	5.6435e−1	1.9141e−3	6.9935e−1	3.6233e−3	9.1009e−1	2.9544e−3
	5	**7.0978e−1**	1.5377e−3	8.8225e−1	2.3447e−3	9.9762e−1	5.7607e−3	1.7230	1.7699e−3
	6	**1.0493**	2.4190e−3	1.2960e	4.2618e−3	1.3456	9.2707e−3	2.4302	3.5099e−3
WFG5	3	**2.1432e−1**	1.9258e−3	2.5464e−1	2.3971e−3	2.3735e−1	3.2326e−3	2.5805e−1	2.9421e−3
	4	**4.6330e−1**	1.9221e−3	5.7964e−1	4.4857e−3	7.3553e−1	4.6862e−3	8.7639e−1	2.9506e−2
	5	**7.2521e−1**	2.8216e−3	8.8588e−1	8.3039e−3	1.0263	5.5210e−3	1.6879	2.8870e−2
	6	**1.0670**	5.7322e−3	1.2839e	1.1215e−2	1.3673	5.8758e−3	2.3695	4.7496e−2
WFG6	3	**2.0941e−1**	6.8731e−3	2.5135e−1	5.1084e−3	2.5584e−1	8.5190e−4	2.5740e−1	7.0841e−3
	4	**4.5124e−1**	3.7243e−3	5.7267e−1	7.2806e−3	7.0849e−1	5.5246e−3	9.0460e−1	1.6024e−2
	5	**7.1620e−1**	5.7546e−3	8.9068e−1	4.0056e−3	9.9803e−1	6.1261e−3	1.7157	2.0221e−2
	6	**1.0555**	6.2477e−3	1.2992	5.4060e−3	1.3263	7.3154e−3	2.4079	3.6575e−2
WFG7	3	**1.9917e−1**	2.5254e−4	2.4221e−1	3.3549e−4	2.4458e−1	6.0167e−4	2.4829e−1	1.4603e−4
	4	**4.4811e−1**	5.8268e−4	5.6432e−1	9.9417e−4	7.0893e−1	3.8010e−3	9.1005e−1	2.0634e−3
	5	**7.1012e−1**	1.0072e−3	8.8138e−1	1.8580e−3	1.0095	4.3953e−3	1.7206	2.9405e−3
	6	**1.0525**	1.7846e−3	1.3018	3.2884e−3	1.3518	6.4112e−3	2.4317	4.8342e−3
WFG8	3	**2.2980e−1**	5.4934e−3	2.5840e−1	1.1033e−2	2.6055e−1	3.5261e−4	2.6784e−1	4.6078e−3
	4	**5.0691e−1**	6.0238e−3	5.9774e−1	5.4665e−3	7.0715e−1	3.8844e−3	8.7683e−1	1.3476e−2
	5	**7.7061e−1**	3.9662e−3	9.3829e−1	4.8050e−3	9.9078e−1	5.0878e−3	1.7816	3.0155e−2
	6	**1.1389**	8.4181e−3	1.4002	7.6076e−3	1.3187	5.7687e−3	2.7538	8.3527e−2
WFG9	3	**2.1273e−1**	3.1899e−2	2.4438e−1	3.0951e−2	2.4841e−1	2.6508e−3	2.4973e−1	3.0612e−2
	4	**4.9754e−1**	3.2155e−2	5.8761e−1	3.3025e−3	7.5456e−1	7.7947e−3	8.1323e−1	1.1244e−2
	5	**8.0401e−1**	1.4053e−2	9.1352e−1	7.0490e−3	1.0696	1.4088e−2	1.6603	2.5566e−2
	6	**1.2001**	1.6095e−2	1.3436	1.3477e−2	1.4423	2.2007e−2	2.8528	5.5394e−2

The value of metric with bold is the best among the three compared algorithms

We show the performance of MOEA/D-UDM for 3-objective DTLZ1 and DTLZ2 problems with some user-offered preferred directions. For both DTLZ1 and DTLZ2 problems, 859 reference directions with a shape of smile are constructed as shown in Fig. 11 (right). MOEA/D and MOEA/D-UDM use the above same weight vectors. The obtained solutions by MOEA/D-UDM (MOEA/D-MTCH) is plotted on the left and the obtained solutions by MOEA/D

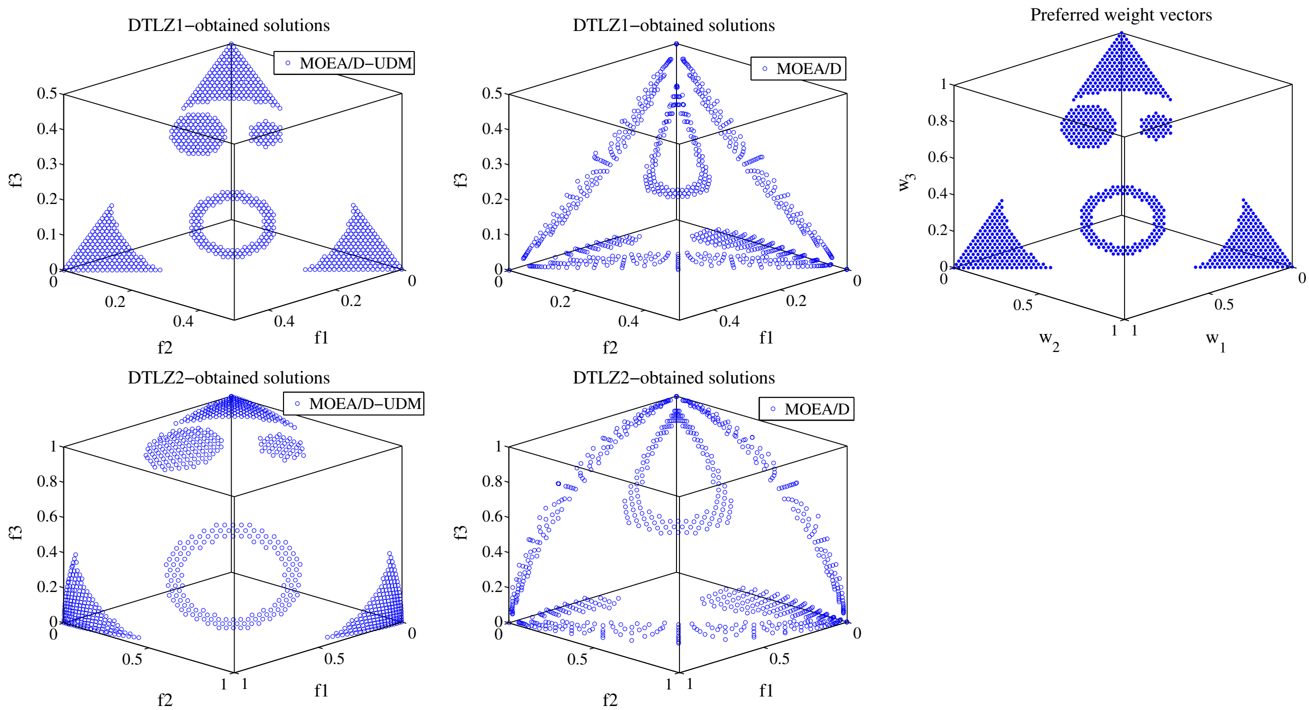

Fig. 11 *Left* found solutions by MOEA/D-UDM for 3-objective DTLZ1 and DTLZ2 problems. *Middle* obtained solutions by MOEA/D for DTLZ1-DTLZ2 problems. *Right* preferred weight vectors

are plotted on the middle. Clearly, the distributions of obtained solutions found by MOEA/D-UDM are similar to the given preferred directions for 3-objective DTLZ1 and DTLZ2 problems. While the distributions of preferred solutions found by MOEA/D are immensely different from the given preferred directions. The reason is that MOEA/D-UDM uses the modified Tchebycheff decomposition to convert a MOP into a number of subproblems while MOEA/D use the Tchebycheff decomposition. As analyzed in 3.4, the geometry relationship between the weight vector of subproblem and the direction of its optimal solution is same for the modified Tchebycheff decomposition while nonlinear for Tchebycheff decomposition.

If the DM can offer his/her preference in the form of a few representative reference directions, the modified Tchebycheff decomposition method used by MOEA/D-UDM can find the optimal solutions with the similar shape.

4.6.2 Small population size

In MOEA/D (Zhang and Li 2007), Zhang and Li showed the ability of MOEA/D with a small population in solving bi-objective problem. One may ask: "Can MOEA/D-UDM work well with a small population for many-objective problems?". In this section, we study this aspect. 3–6 objective DTLZ2 problems are selected to study. we are interested in finding 10 well-distributed solutions for the 3-objective problem, 20 solutions for 4-objective problem, 35 solutions for 5-objective problem and 56 solutions for 6-objective problem.

Figure 12 shows the small size of weight vectors used by MOEA/D-UDM, MOEA/D and UMOEA/D for 3–6 objective problem. The weight vectors used by UMOEA/D are distributed in the interior of weight space. While the weight vectors generated by MOEA/D are mainly distributed on the boundary of weight space. The weight vectors constructed by MOEA/D-UDM are between the both. Most weight vectors are assigned in the interior of weight space and few weight vectors are distributed on the boundary of weight space.

Table 9 tabulates the number of population size (N), the maximum number of function evaluation (FE) and the mean and standard deviation of the hypervolume of the obtained solutions for all three methods in solving DTLZ2 problems. Clearly, MOEA/D-UDM performs the best in finding a set of solutions with bigger hypervolume values for 3–6 objective DTLZ2 problems. The performance of UMOEA/D is better than MOEA/D.

In order to help the reader understand the performance of the compared methods with a small population, we give the graphical representation of Pareto front for these problems. Figure 13 plots the obtained solutions for 3–6 objective DTLZ2 problem with small population size. Clearly, the three compared methods can work for 3-objective DTLZ2 problem. MOEA/D uses 10 weight vectors, but it only finds 6 dif-

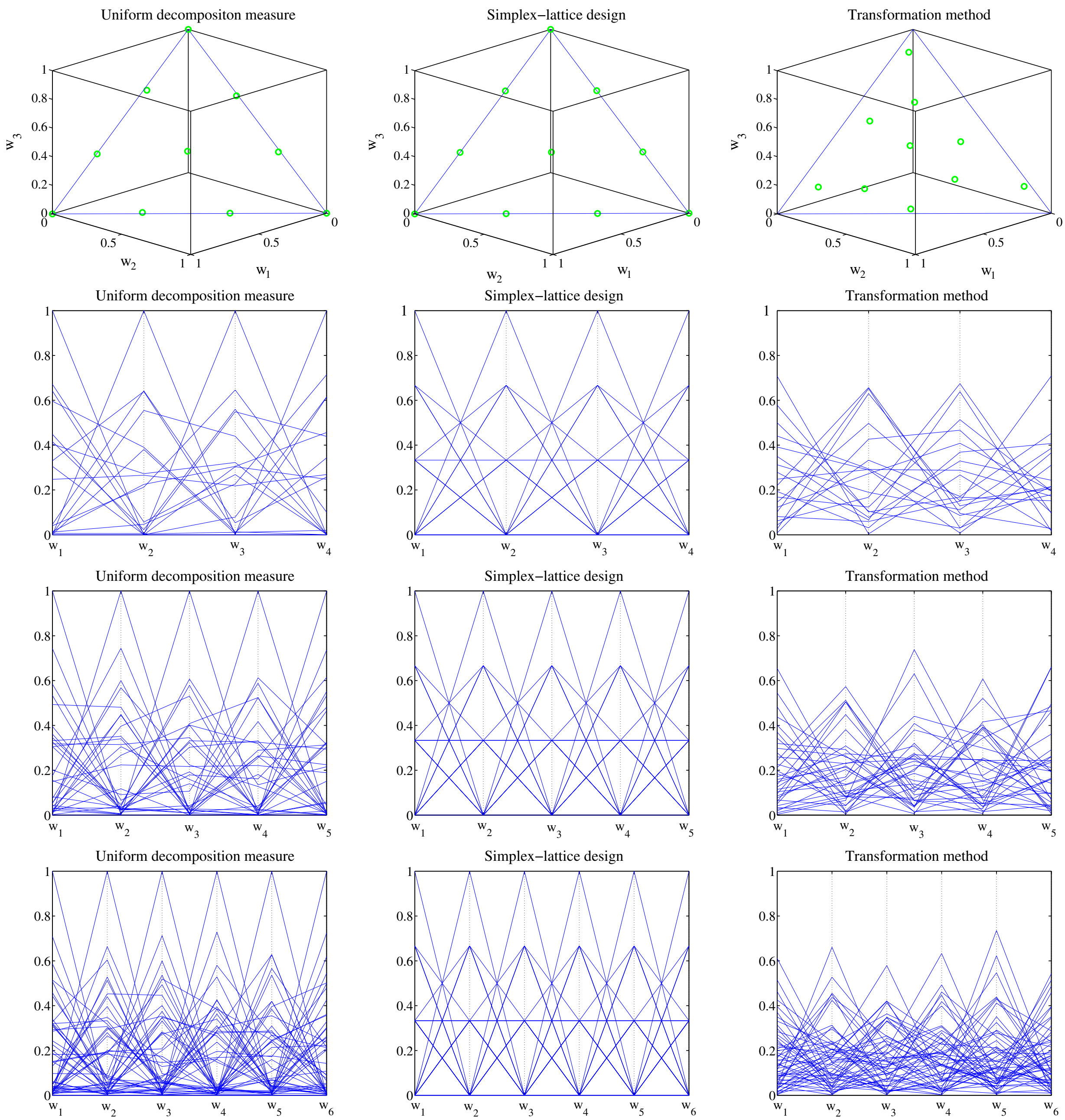

Fig. 12 Plot the small size of weight vectors used by MOEA/D-UDM, MOEA/D and UMOEA/D for 3–6 objective DTLZ2 problem

Table 9 The value of the hypervolume between the algorithms MOEA/D-UDM, MOEA/D and UMOEA/D for DTLZ2 problems with small popultion

HV	m	N	FE	MOEA/D-UDM		MOEA/D		UMOEA/D	
				Mean	std	Mean	std	Mean	std
DTLZ2	3	10	800,000	**2.5455e−1**	1.2373e−3	2.2441e−1	1.4972e−3	2.3603e−1	1.3913e−3
	4	20	1,200,000	**4.0428e−1**	1.2788e−3	3.7741e−1	1.5607e−3	2.9080e−1	1.9060e−3
	5	35	1,500,000	5.1543e−1	1.8727e−3	**5.2296e−1**	1.3428e−2	4.1403e−1	1.8322e−3
	6	56	2,100,000	**6.6198e−1**	1.5463e−3	5.5002e−1	1.7021e−2	5.3391e−1	1.9569e−3

The value of metric with bold is the best among the three compared algorithms

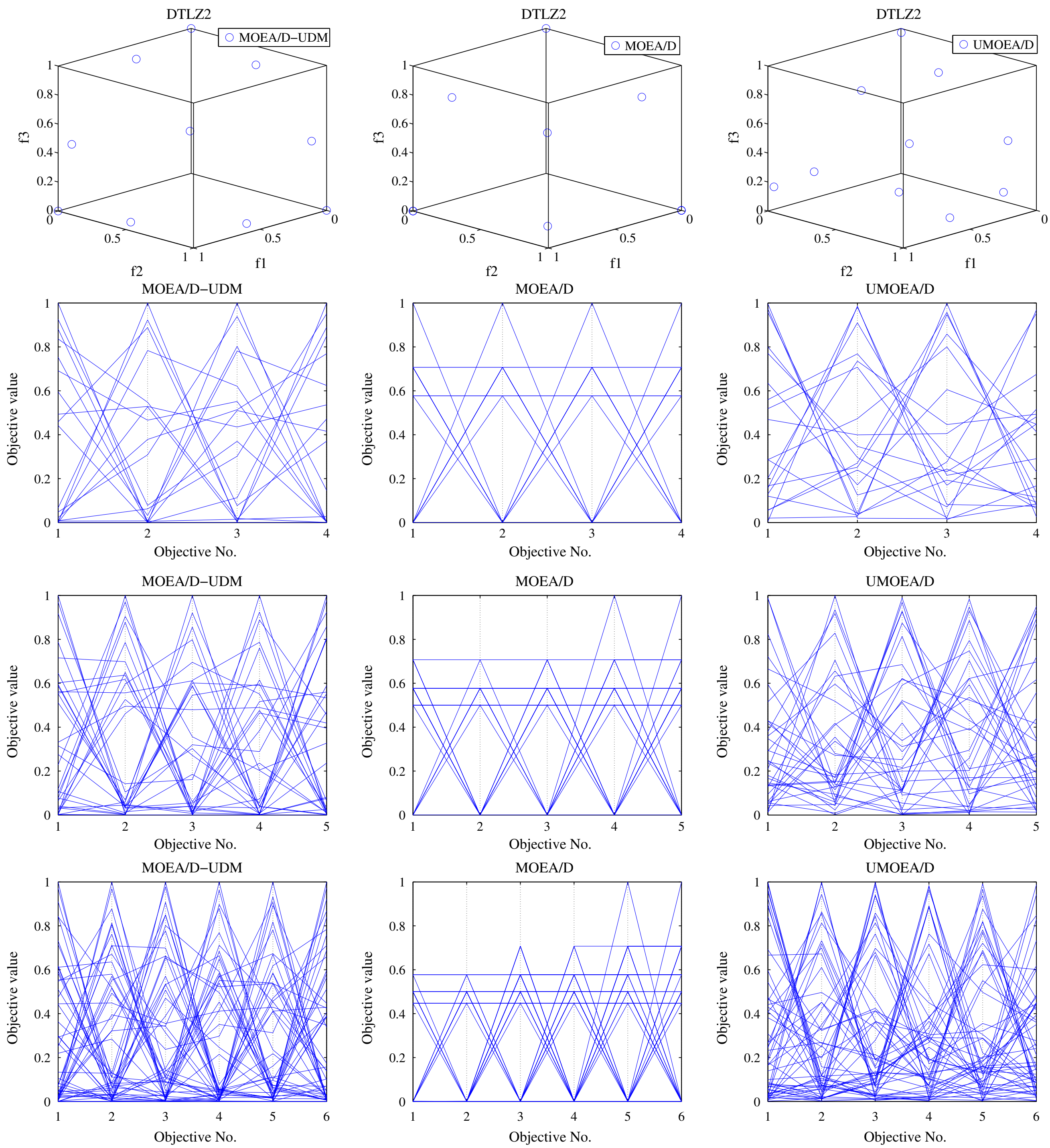

Fig. 13 Scatter plot of the obtained solutions with biggest hypervolume values for 3 objective DTLZ2 problem and parallel coordinate plot of the obtained solutions for 4–6 objective DTLZ2 problem with small population size

ferent solutions. The reason is that simplex-lattice design distributes too many weight vectors on the boundary of weight space and many subproblems with boundary weight vectors have the same optimal solution. MOEA/D-UDM and UMOEA/D can find 10 different solutions and the uniformity of obtained solutions by MOEA/D-UDM is slightly better than UMOEA/D. To show the distribution of obtained solutions on a higher-objective problem, parallel coordinate plot is used to demonstrate the obtained solutions for 4–6 DTLZ2 problems. As shown in Fig. 13, it is clear that MOEA/D-UDM and UMOEA/D are able to find a well distributed solutions in the entire range of PF with a small population, but

MOEA/D is unable to find solutions having larger objective values for 5-6 objective DTLZ2 problems with a small population.

5 Conclusions

In this paper, we have suggested a uniform design measurement based weight vectors initialization method. Combining it with the modified Tchebycheff decomposition approach, MOEA/D-UDM is proposed for handling many-objective optimization problems. The proposed MOEA/D-UDM has been used to solve 3–6 objective test problems. The PFs of these problems include convex, concave, scaled objective values, biased density of solutions over the front. In all such problems, the proposed MOEA/D-UDM approach has been able to successfully find a set of well-converged and well distributed solutions. The performance of MOEA/D-UDM has also been compared with two versions of a recently proposed MOEA/D. The experimental results show that MOEA/D-UDM is better than the both compared algorithms. Furthermore, the results of this work strongly suggest that the choice of weighting vectors and suitable decomposition method can affect not only the distribution of the obtained solutions on the PF but also the convergence of the algorithm. This issue is more evident for many-objective problems.

Moreover, its ability of MOEA/D-UDM has been demonstrated for solving different types of many-objective optimization tasks. It has been tested that MOEA/D-UDM is able to work with a small population for many-objective problems. In finding a part of PF, the ability of MOEA/D-UDM has also been shown.

In the future, firstly, in order to obtain uniformly distributed solutions on PF for different type problems, we want to dynamically adjust the distribution of weight vectors for many-objective problems in the process of evolution. The related work can be found in Li and Landa-Silva (2011), Gu and Liu (2010) and Jiang et al. (2011). Secondly, we plan to incorporate the user-preference into the framework of MOEA/D for many-objective problems. The related work can be found in Mohammadi et al. (2012). Thirdly, we intend to introduce some advanced single objective optimization algorithm, such as simulated annealing (Li and Landa-Silva 2011), ant colony optimization (Ke et al. 2010), particle swarm optimization (Moubayed et al. 2010; Martinez and Coello 2011) and estimation of distribution algorithm (Shim et al. 2012), into the framework of MOEA/D to handle hard multi-objective problems.

Acknowledgments This work was supported by the Program for Cheung Kong Scholars and Innovative Research Team in University Nos. IRT1170 and IRT1170,the National Research Foundation for the China Postdoctoral Science Foundation Special funded project under Grant Nos. 201104658, 20090203120016, 20100203120008 and 200807010003, the Doctoral Program of Higher Education of China under Grant Nos. 200807010003, 20090203120016 and 20100203120008, the National Natural Science Foundation of China under Grant Nos. 61072106, 61072139, 61001202 and 61003199, the Fundamental Research Funds for the Central Universities under Grant Nos. JY10000903007, K5051203007 and K5051203002, the Fund for Foreign Scholars in University Research and Teaching Programs the 111 Project No. B07048, and the Provincial Natural Science Foundation of Shaanxi of China under Grant Nos. 2009JQ8015 and 2011JQ8010.

References

Borkowski J, Piepel G (2009) Uniform designs for highly constrained mixture experiments. J Quality Technol 41(1):35–47

Cheng J, Zhang G, Li Z, Li Y (2011) Multi-objective ant colony optimization based on decomposition for bi-objective traveling salesman problems. Soft Comput 16:597–614

Coello Coello C, Lamont G, Van Veldhuizen D (2007) Evolutionary algorithms for solving multi-objective problems. Springer, London

Cornell J (1975) Some comments on designs for cox's mixture polynomial. Technometrics 17:25–35

Cornell J (1990) Experiments with mixtures, designs, models, and the analysis of mixture data. Wiley, New York

Deb K (2001) Multi-objective optimization using evolutionary algorithms. Wiley, London

Deb K, Beyer H (2001) Self-adaptive genetic algorithms with simulated binary crossover. Evol Comput 9(2):197–221

Deb K, Jain H (2012a), An improved nsga-ii procedure for many-objective optimization, part i: solving problems with box constraints. Tech. Rep. 2012009, KanGAL

Deb K, Jain H (2012b) An improved nsga-ii procedure for many-objective optimization, part ii: handling constraints and extending to an adaptive approach. Tech Rep 2012010, KanGAL

Deb K, Agrawal S, Pratap A, Meyarivan T (2002) A fast and elitist multi-objective genetic algorithm: Nsga-ii. IEEE Trans Evol Comput 6(2):182–197

Deb K, Thiele L, Laumanns M, Zitzler E (2005) Scalable test problems for evolutionary multi-objective optimization. Evol Multiobjective Optim :105–145

Fang K, Lin D (2003) Uniform designs and their application in industry. Handb Stat 22:131–170

Fang K, Wang Y (1994) Number-theoretic methods in statistics. Chapman and Hall, London

Fang K, Yang Z (2000) On uniform design of experiments with restricted mixture and generation of uniform distribution on some domains. Statist Probab Lett 46:113–120

Fleming P, Purshouse R, Lygoe R (2005) Many-objective optimization: an engineering design perspective. Evol Multi-Criterion Optim :14–32

Fonseca C, Fleming P (1998) Multiobjective optimization and multiple constraint handling with evolutionary algorithms-part ii: a application example. IEEE Trans Syst Man Cybernet Part A: Syst Hum 28(1):38–47

Gu F, Liu H (2010) A novel weight design in multi-objective evolutionary algorithm. CIS 2010:137–141

Hickernell F (1998) Lattice rules: how well do they measure up? In: Random and quasi-random point sets. Springer, London, pp 106–166

Huband S, Hingston P, Barone L, While L (2006) A review of multiobjective test problems and a scalable test problem toolkit. IEEE Trans Evol Comput 10(5):477–506

Hughes E (2005) Evolutionary many-objective optimisation: many once or one many. In: CEC2005, pp 222–227

Hughes E (2007) Radar waveform optimisation as a many-objective application benchmark. EMO 4403:700–714

Ishibuchi H, Nojima Y (2010) Performance evaluation of evolutionary multiobjective optimization algorithms for multiobjective fuzzy genetics-based machine learning. Soft Comput 15:2415–2434

Ishibuchi H, Tsukamoto N, Nojima Y (2008) Evolutionary many-objective optimization: a short review. CEC 2008:2424–2431

Ishibuchi H, Akedo N, Nojima Y (2011a) A many-objective test problem for visually examining diversity maintenance behavior in a decision space. GECCO 2011:649–656

Ishibuchi H, Hitotsuyanagi Y, Ohyanagi H, Nojima Y (2011b) Effects of the existence of highly correlated objectives on the behavior of moea/d. EMO 6576:166–181

Jiang S, Cai Z, Zhang J, Ong Y (2011) Multiobjective optimization by decomposition with pareto-adaptive weight vectors. In: ICNC 2011

Ke L, Zhang Q, Battiti R (2010) Multiobjective combinatorial optimization by using decomposition and ant colony. University of Essex, Tech Rep

Khare V, Yao X, Deb K (2003) Performance scaling of multi-objective evolutionary algorithms. EMO 2003:376–390

Knowles J, Corne D (2007) Quantifying the effects of objective space dimension in evolutionary multiobjective optimization. EMO 2007:757–771

Li H, Landa-Silva D (2011) An adaptive evolutionary multi-objective approach based on simulated annealing. Evol Comput 19(4):561–595

Li H, Zhang Q (2009) Multiobjective optimization problems with complicated pareto sets, moea/d and nsga-ii. IEEE Trans Evol Comput 13(2):284–302

Liu H, Wang Y, Cheung Y (2009) A multiobjective evolutionary algorithm using min-max strategy and sphere coordinate transformation. Intell Autom Soft Comput 15:361–384

Martinez Z, Coello C (2011) A multi-objective particle swarm optimizer based on decomposition. In: Genetic and Evolutionary Computation Conference

Mohammadi A, Omidvar M, Li X (2012) Reference point based multi-objective optimization through decomposition. In: IEEE world congress on computational intelligence, pp 10–15

Moubayed N, Petrovski A, McCall J (2010) A novel multi-objective particle swarm optimisation based on decomposition. In: International Conference on Parallel Problem Solving from Nature, pp 1–10

Ning J (2008) Uniform designs for mixture experiments. PhD thesis, Central China Normal University

Ning J, Zhou Y, Fang K (2011a) Discrepancy for uniform design of experiments with mixtures. J Stat Plan Inference 141:1487–1496

Ning J, Zhou Y, Fang K (2011b) Uniform design for experiments with mixtures. Commun Stat Theory Methods 40(10):1734–1742

Prescott P (2008) Nearly uniform designs for mixture experiments. Commun Statist Theory Methods 37:2095–2115

Qi Y, Ma X, Liu F, Jiao L, Sun J, Wu J (2013) Moea/d with adaptive weight adjustment. Evol Comput (in press)

Saxena D, Duro J, Tiwari A, Deb K, Zhang Q (2013) Objective reduction in many-objective optimization: linear and nonlinear algorithms. IEEE Trans Evol Comput 17(1):77–99

Scheffe H (1958) Experiments with mixtures. J Royal Stat Soc Ser B 20:344–360

Scheffe H (1963) Simplex-centroid designs for experiments with mixtures. J Royal Stat Soc Ser B 25:235–263

Shim V, Tan K, Tan K (2012) A hybrid estimation of distribution algorithm for solving the multi-objective multiple traveling salesman problem. In: WCCI 2012

Sindhya K, Ruuska S, Haanpaa T, Miettinen K (2011) A new hybrid mutation operator for multiobjective optimization with differential evolution. Soft Comput 15(10):2041–2055

Tan Y, Jiao Y, Li H, Wang X (2012) Moea/d + uniform design: a new version of moea/d for optimization problems with many objectives. Computer Oper Res

Zhang Q, Li H (2007) Moea/d: a multi-objective evolutionary algorithm based on decomposition. IEEE Trans Evol Comput 11(6):712–731

Zitzler E, Laumanns M, Thiele L (2002) Spea 2: improving the strength pareto evolutionary algorithm for multiobjective optimization. In: Evolutionary Methods for Design, Optimization, and Control, pp 19–26

Zitzler E, Thiele L, Laumanns M, Fonseca C, Fonseca V (2003) Performance assessment of multiobjective optimizers: an analysis and review. IEEE Trans Evol Comput 7(2):117–132