Honda Research Institute EU
Honda Research Institute Europe GmbH
https://www.honda-ri.de/

Preference representation using Gaussian functions on a hyperplane in evolutionary many-objective optimization

Kaname Narukawa, Yu Setoguchi, Yuki Tanigaki, Markus Olhofer, Bernhard Sendhoff, Hisao Ishibuchi

2016

Preprint:

This is an accepted article published in Soft Computing. The final authenticated version is available online at: https://doi.org/[DOI not available]

Noname manuscript No.
(will be inserted by the editor)

Preference representation using Gaussian functions on a hyperplane in evolutionary multi-objective optimization

Kaname Narukawa · Yu Setoguchi · Yuki Tanigaki · Markus Olhofer · Bernhard Sendhoff · Hisao Ishibuchi

Received: date / Accepted: date

Abstract Many-objective optimization has attracted much attention in evolutionary multi-objective optimization (EMO). This is because EMO algorithms developed so far often degrade their search ability for optimization problems with four or more objectives, which are frequently referred to as many-objective problems. One of promising approaches to handle many objectives is to incorporate the preference of a decision maker (DM) into EMO algorithms. With the preference, EMO algorithms can focus the search on regions preferred by the DM, resulting in solutions close to the Pareto front around the preferred regions. Although a number of preference-based EMO algorithms have been proposed, it is not trivial for the DM to reflect his/her actual preference in the search. We previously proposed to represent the preference of the DM using Gaussian functions on a hyperplane. The DM specifies the center and spread vectors of the Gaussian functions so as to represent his/her preference. The preference handling is integrated into the framework of NSGA-II. This paper extends our previous work so that obtained solutions follow the distribution of Gaussian functions specified. The performance of our proposed method is demonstrated mainly for benchmark problems and real-world applications with a few objectives in this paper. We also show the applicability of our method to many-objective problems.

Keywords Many-objective optimization · Preference · Evolutionary multi-objective optimization (EMO) · Decision making · Hyperplane · Gaussian function

K. Narukawa · M. Olhofer · B. Sendhoff
Honda Research Institute Europe GmbH
Carl-Legien-Straße 30, 63073 Offenbach am Main, Germany
E-mail: kaname.narukawa@honda-ri.de

M. Olhofer
E-mail: markus.olhofer@honda-ri.de

B. Sendhoff
E-mail: bs@honda-ri.de

Y. Setoguchi · Y. Tanigaki · H. Ishibuchi
Osaka Prefecture University
1-1 Gakuen-cho, Nakaku, Sakai, Osaka 599-8531, Japan
E-mail: yu.setoguchi@ci.cs.osakafu-u.ac.jp

Y. Tanigaki
E-mail: yuki.tanigaki@ci.cs.osakafu-u.ac.jp

H. Ishibuchi
E-mail: hisaoi@cs.osakafu-u.ac.jp

1 Introduction

Evolutionary multi-objective optimization (EMO) has been one of hottest research topics over the last decade in the community of soft computing [9,10]. The task of EMO is to find a set of non-dominated solutions that approximates the Pareto front well. Thus far a number of EMO algorithms have been proposed and the high search ability has been demonstrated for benchmark problems and real-world applications [13,25,48,49].

As reviewed in [22], however, EMO algorithms often show difficulties in solving optimization problems having four or more objectives (i.e., many-objective problems[1]). For example, selection pressure will be lost in the high-dimensional objective space. In many EMO algorithms, the selection pressure is generated by a difference in ranks that are assigned to solutions based on the Pareto dominance relation. In the many-objective space, almost all of solutions become non-dominated, resulting in solutions having the same rank (i.e., the

[1] It should be noted that optimization problems with more than three objectives are called many-objective problems in this paper as there is no clear definition on the terminology in the literature.

first rank). Therefore, selection pressure will not be generated from the same rank, leading to less convergence to the Pareto front. For this issue, several remedies have been proposed in [3,31,38,46,47].

Another difficulty is that with an increase in the number of objectives, it becomes exponentially difficult for EMO algorithms to approximate the Pareto front by a finite number of solutions at the same quality. EMO algorithms normally operate with a few hundreds of solutions in a population. Therefore, practitioners who are satisfied with the quality of approximating the three-dimensional Pareto front with 100 solutions may not accept the quality of approximating the more than three-dimensional Pareto front with the 100 solutions.

The other difficulty is a visualization of solutions. We cannot recognize solutions intuitively in the more than three-dimensional space. Therefore, it is hard for us to pick up one or some solutions from those obtained by EMO algorithms as we cannot compare the solutions visually. For the visualization issue, some approaches to visualize the high-dimensional solutions have been proposed [26,27].

In order to handle many objectives, the preference of a decision maker (DM) can be exploited in EMO algorithms, that are often referred to as preference-based EMO algorithms [5,8]. Having the preference of the DM in EMO algorithms will improve the selection pressure and keep the approximation quality. While a number of preference-based EMO algorithms have been proposed [2,4,12,14,21,24,29,37,41–43,45], it is not a trivial task to reflect the DM's preference into the search of EMO algorithms.

We previously proposed to easily represent the DM's preference using Gaussian functions on a hyperplane [33,39]. Our idea of representing the preference is implemented in NSGA-II [13]. In our method, a hyperplane is constructed every generation based on solutions found thus far. On the hyperplane, the DM's preference is defined by Gaussian functions and regions preferred by the DM are emphasized on the hyperplane during optimization. The DM can reflect his/her preference dynamically into the search by specifying the center and spread vectors of the Gaussian functions during the search. This paper extends our former work by considering an allowable radius and a shortest distance in order to realize a distribution that follows Gaussian functions. The performance of our proposed method is demonstrated for benchmark problems and real-world applications.

The rest of this paper is organized as follows. Section 2 introduces problem definitions including a multi-objective optimization problem, Pareto dominance relation, and Pareto optimality. Section 3 briefly gives related works on preference-based EMO algorithms and the motivation of this paper. Our method to represent preference using Gaussian functions on a hyperplane is proposed in Section 4 and the effect of the preference handling on the performance of NSGA-II is demonstrated for many-objective DTLZ problems, a two-objective welded beam design problem, and a 7-objective hybrid car controller optimization problem in Section 5. Finally, Section 6 concludes the paper.

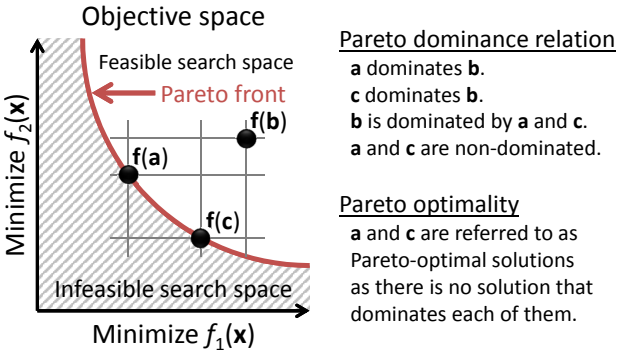


Fig. 1 Pareto dominance relation and Pareto optimality.

2 Problem definitions

An m-objective minimization problem is defined as follows:

$$\text{Minimize } \mathbf{f}(\mathbf{x}) = (f_1(\mathbf{x}), f_2(\mathbf{x}), ..., f_m(\mathbf{x})), \tag{1}$$

$$\text{subject to } \mathbf{x} \in \mathbf{X}, \tag{2}$$

where $\mathbf{f}(\mathbf{x})$ is the m-dimensional objective vector, $f_i(\mathbf{x})$ is the i-th objective function to be minimized, $\mathbf{x}$ is the decision vector consisting of decision variables, $\mathbf{X}$ is the feasible region in the decision space. When two feasible solutions $\mathbf{a}$ and $\mathbf{b}$ of Eq. (2) satisfy the following conditions[2] , it can be said that $\mathbf{a}$ dominates $\mathbf{b}$.

$$\forall i : f_i(\mathbf{a}) \leq f_i(\mathbf{b}) \text{ and } \exists j : f_j(\mathbf{a}) < f_j(\mathbf{b}). \tag{3}$$

If $\mathbf{a}$ is not dominated by any other feasible solutions, $\mathbf{a}$ is referred to as a Pareto-optimal solution. The entire set of Pareto-optimal solutions forms the Pareto front in the objective space. The Pareto dominance relation among $\mathbf{a}$, $\mathbf{b}$, and another feasible solution $\mathbf{c}$ is illustrated in Fig. 1 for a two-objective minimization problem. The task in the m-objective minimization problem is to find non-dominated solutions that approximate the Pareto front as well as possible.

[2] For maximization problems, the following conditions must be satisfied for $\mathbf{a}$ dominating $\mathbf{b}$.
$\forall i : f_i(\mathbf{a}) \geq f_i(\mathbf{b})$ and $\exists j : f_j(\mathbf{a}) > f_j(\mathbf{b})$.

3 Related works and motivation

This section provides related works on handling the preference of a DM in EMO algorithms. Depending on when and how the DM's preference is used, preference-based EMO algorithms are classified into three types: *a posteriori*, *a priori*, and interactive approaches. In *a posteriori* approaches, the DM will make a decision on which solution(s) to select based on his/her preference after EMO algorithms complete. On the other hand in *a priori* approaches, the DM represents his/her preference in some form before optimization starts and the preference information is exploited during optimization by EMO algorithms to focus the search on regions favored by the DM. Interactive approaches make it possible to adapt the preference during optimization by having an interaction between the DM and EMO algorithms.

Three positive effects are expected from considering the preference in *a priori* and interactive approaches. One is a focus by the preference. Based on the focus in EMO algorithms, solutions that are of great interest to the DM will be obtained. As a result, the DM does not suffer with a number of solutions that are mostly irrelevant. The second effect is an efficient computation. With the preference, EMO algorithms focus the search on regions preferred by the DM, and solutions the DM expects will be obtained quickly. In other words, computational efforts spent on finding solutions that are not expected by the DM will be avoided. The last effect is selection pressure. In many-objective optimization, Pareto dominance-based EMO algorithms often lose the selection pressure as almost all of solutions become non-dominated (i.e., incomparable). Considering the preference into the search provides the selection pressure.

In what follows, we show some preference representations that are used in preference-based EMO algorithms. For each representation, its advantages and disadvantages are remarked. Then, the motivation of our paper is given at the end of this section.

3.1 Weight

This is a basic approach to represent the preference of a DM by a weight vector $\mathbf{w} = (w_1, w_2, ..., w_m)$ where m is the number of objectives. With the weight vector, the minimization problem of Eq. (1) is written as follows:

$$\text{Minimize } \sum_{i=1}^{m} w_i f_i(\mathbf{x}), \tag{4}$$

where $w_i \geq 0$ for $i = 1, 2, ..., m$. The relative importance of each objective function is described by the weight vector. In practice, each objective function should be normalized in the same or similar scale so that each element of the weight vector has an equal effect on the corresponding objective function. This representation is simple and straightforward from a practical perspective. However, the weighting method has a serious flaw, meaning that solutions on concave regions of the Pareto front are not captured by any weight vectors [6]. If the Pareto front is known to be convex in advance, this representation will be useful. It should be noted that the weight vector can be changed during optimization in interactive approaches although a small change in the weight vector may result in a big difference in solutions obtained by EMO algorithms.

3.2 Bound

Another approach to represent the preference is to specify bounds for objective functions. This idea is used in the ϵ-constraint method [7,19] and then, the minimization problem of Eq. (1) can be replaced by the following:

$$\text{Minimize } f_i(\mathbf{x}), \tag{5}$$

$$\text{subject to } f_j(\mathbf{x}) \leq \epsilon_j, \text{ for all } j \in \{1, 2, ..., m\} \backslash \{i\}, \tag{6}$$

where i is selected from $\{1, 2, ..., m\}$ and ϵ_j is an upper bound for the j-th objective function. In the ϵ-constraint method, one of the objective functions is minimized in Eq. (5) and its optimal solution is proven to be weakly Pareto optimal[3]. Furthermore, a feasible solution x^* is Pareto optimal if and only if x^* solves Eq. (5) for all $i \in \{1, 2, ..., m\}$ where $f_j(\mathbf{x}^*) = \epsilon_j$ for all $j \in \{1, 2, ..., m\} \backslash \{i\}$. A unique solution of Eq. (5) is also proven to be Pareto optimal for any upper bounds. One of advantages of this method is that concave regions of the Pareto front can also be handled as well as the convex ones. However, it will be difficult to specify the upper bounds in Eq. (6) so as to obtain the solution in Eq. (5) that is Pareto optimal. Moreover, this issue becomes more difficult with an increase in the number of objectives.

3.3 Reference point

Either a reference point or a set of reference points can also be used to represent the preference of a DM. In R-NSGA-II [14], for example, solutions close to the reference point(s) in the objective space are emphasized.

[3] A solution is weakly Pareto optimal if there does not exist any other solutions for which all the objective functions are better [28].

The diversity of solutions is also considered by specifying a niching parameter ϵ in R-NSGA-II[4]. The performance of R-NSGA-II is demonstrated in computational experiments later in this chapter.

The reference point can also be used in achievement scalarizing functions (ASFs) [44]. The ASF is used as a Tchebycheff approach in MOEA/D [46] which is one of state-of-the-art EMO algorithms for many objectives. It is straightforward to represent the DM's preference by the reference points and to minimize, for example, a distance of solutions to the reference points in EMO algorithms. However, it is difficult for the DM to specify the reference point without any information about the Pareto front. Moreover, more objectives overcomplicate the difficulty.

3.4 Value function

It is great if a DM can explicitly formulate his/her preference using a mathematical expression (i.e., a value function) and if the function actually represents the DM's preference. The minimization problem in Eq. (5) can be written with the value function as follows:

$$\text{Maximize } v(\mathbf{f}(\mathbf{x})), \tag{7}$$

where $v(\mathbf{f}(\mathbf{x}))$ returns how much the DM prefers the objective vector $\mathbf{f}(\mathbf{x})$ for a solution $\mathbf{x}$. However, unfortunately, it may be nearly impossible to find the true value function. Even if the value function can be formulated, the task of optimizing the function will also be very difficult due to its possible complicated nature. One remedy to ease the difficulty is to use an approximation model of the value function. The approximation model can be built using an artificial neural network in [35] and used in EMO algorithms in an interactive manner [36].

3.5 Lexicographic order

In this approach, the preference of a DM is represented by sorting objective functions based on their importance [18]. After the sort, the most important objective is optimized with the original constraint condition(s). If there is a unique solution, the optimization ends up with the solution. Otherwise, the second most important objective is optimized with the most important objective being fixed to the optimal value. This process iterates until the unique solution is obtained. The unique solution is proven to be Pareto optimal. Although this approach is very simple, it will be difficult to rank the objective functions in order of importance.

[4] R-NSGA-II allows only one solution to exist within a distance of ϵ in the objective space.

3.6 Brief summary and motivation

From a practical perspective, bound and lexicographic order approaches are difficult to adopt in EMO algorithms for many-objective optimization as mentioned above.

If the Pareto front is known to a DM in advance, it may be easy for him/her to represent the preference using a reference point. The task here is to minimize a distance of solutions to the reference point. However, it is rare that the Pareto front is known in advance especially when real-world applications are treated. Even if we know the Pareto front, it will become much more difficult to specify the reference point with an increase in the number of objectives as we cannot visually recognize solutions in the high-dimensional objective space.

Without any information about the Pareto front, it will be promising to specify a weight vector as the preference to represent the importance of objective functions. If the Pareto front is concave, however, the optimization always ends up with extreme solutions on the Pareto front. Even if the Pareto front is convex, it is hard for a DM to expect which regions solutions are obtained based on the current weight vector as the regions depend not only on the weight vector but also on the shape of the Pareto front.

In our previous work, we proposed a new approach to represent the preference of a DM [33,39]. In the approach, a hyperplane is constructed in the objective space. On the hyperplane, the DM can specify his/her preference by center and spread vectors of Gaussian functions. Our approach is integrated into the framework of NSGA-II. In the method, the DM can visually specify his/her preference. The detail of how to visually specify the preference is explained in Subsection 4.4.

Although our previously proposed method can focus the search on regions preferred by the DM, the obtained solutions are uniformly distributed on a hyperplane around the preferred regions in the objective space (refer to Subsection 5.4 for details).

This paper extends the previous work by taking into account an allowable radius and a shortest distance so that the distribution of obtained solutions follows Gaussian functions specified by the DM. Our aim is to obtain dense solutions around center vectors and sparse solutions far away from the center vectors on a hyperplane. The reason why we use Gaussian functions can be justified by [35,36] where the actual (simple) preference of the DM is modeled by Gaussian functions.

Not only for the preference but also for generating noise [1] and data set [20], correlation functions of Krigging [30], and so on, the Gaussian functions can be used in the field of evolutionary multi-criterion optimization.

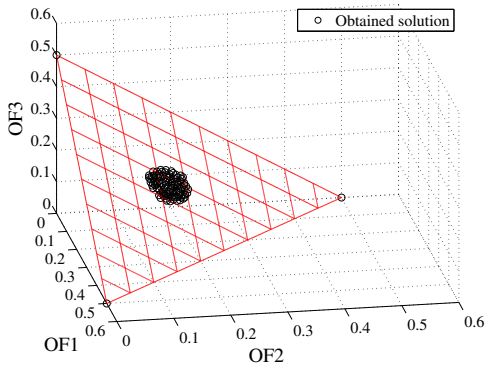


Fig. 2 Uniformly distributed solutions obtained by our previous method for DTLZ1 with three objectives.

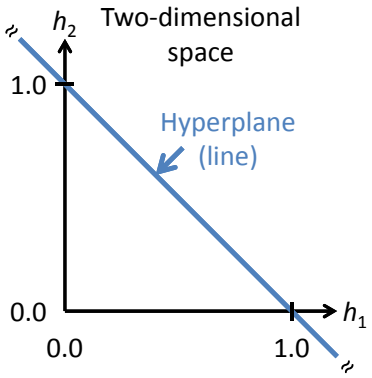


Fig. 3 Hyperplane (line) in two-objective optimization.

The detail of our proposal in this paper is described in the next section.

4 Proposed method

This section describes how to represent the preference of a DM and how the representation is integrated into NSGA-II in the following subsections.

4.1 Preference representation

In our proposal, the preference of a DM is represented by Gaussian functions on a hyperplane, in which the DM specifies the center and spread values. First of all, we consider the hyperplane passing through m vectors $(1, 0, ..., 0)$, $(0, 1, ..., 0)$, ..., $(0, 0, ..., 1)$ for m-objective minimization. For two-objective optimization, the hyperplane concerned is shown in Fig. 3. Next, q sets of m Gaussian functions are used to represent the preference of the DM for a vector $\mathbf{v} = (v_1, v_2, ..., v_m)$ on the hyperplane (i.e., $v_1, v_2, ..., v_m = 1$) as follows:

$$p(\mathbf{v}) = \max_{i=1}^{q} p_i(\mathbf{v}), \tag{8}$$

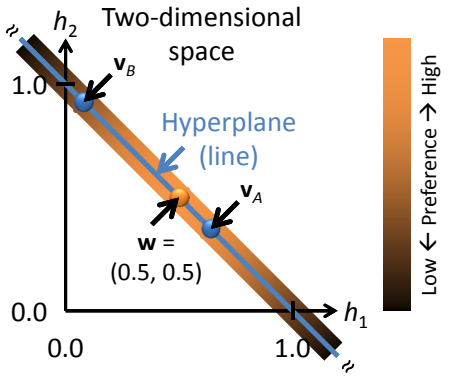


Fig. 4 Preference on a hyperplane (line) in two-objective optimization.

$$p_i(\mathbf{v}) = \exp\{-\sum_{l=1}^{m} \frac{(v_l - w_l^i)^2}{(s_l^i)^2}\},\ i = 1, 2, ..., q, \tag{9}$$

where $w_l^i \geq 0$ is the center (or the mean) and $s_l^i > 0$ is the spread (or the standard deviation) of a Gaussian function. The larger $p(\mathbf{v})$ is, the more $\mathbf{v}$ is preferred. The center vector $\mathbf{w}^i = (w_1^i, w_2^i, ..., w_m^i)$ and the spread vector $\mathbf{s}^i = (s_1^i, s_2^i, ..., s_m^i)$ are specified for the i-th set of m Gaussian functions by the DM under the following condition:

$$w_1^i + w_2^i + ... + w_m^i = 1. \tag{10}$$

We visualize the preference of the DM for two-objective optimization in Fig. 4 where the center vector and the spread vector of the Gaussian functions are specified as $\mathbf{w} = (w_1, w_2) = (0.5, 0.5)$ and $\mathbf{s} = (s_1, s_2) = (1.0, 1.0)$, respectively. We can say that $\mathbf{v}_A$ is preferred more than $\mathbf{v}_B$ as $p(\mathbf{v}_A) > p(\mathbf{v}_B)$ in Fig. 4.

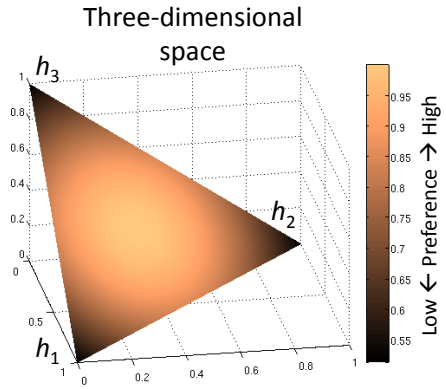
The preference of the DM is also visualized for three-objective optimization in Fig. 5. The center vector $\mathbf{w} = (w_1, w_2, w_3) = (1/3, 1/3, 1/3)$ and the spread vector $\mathbf{s} = (s_1, s_2, s_3) = (1.0, 1.0, 1.0)$ are used in Fig. 5(a). In Fig. 5(b), the center and spread vectors are specified as follows:

$$\mathbf{w}^1 = (w_1^1, w_2^1, w_3^1) = (0.8, 0.1, 0.1),$$
$$\mathbf{w}^2 = (w_1^2, w_2^2, w_3^2) = (0.12, 0.76, 0.12),$$
$$\mathbf{w}^3 = (w_1^3, w_2^3, w_3^3) = (0.14, 0.14, 0.72),$$
$$\mathbf{s}^1 = (s_1^1, s_2^1, s_3^1) = (1.0, 1.0, 1.0),$$
$$\mathbf{s}^2 = (s_1^2, s_2^2, s_3^2) = (0.1, 1.0, 1.0),$$
$$\mathbf{s}^3 = (s_1^3, s_2^3, s_3^2) = (1.0, 1.0, 0.1).$$

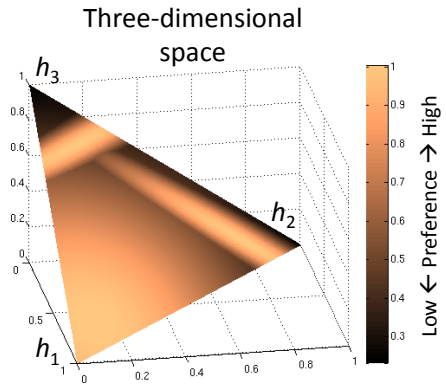
4.2 Preference calculation for solutions

In order to calculate how much the DM prefers a solution giving an objective vector $\mathbf{f} = (f_1, f_2, ..., f_m)$, $\mathbf{f}$ is first shifted to $\mathbf{f}' = (f_1', f_2', ..., f_m')$, where

$$f_k' = f_k - f_k^{\min},\ k = 1, 2, ..., m, \tag{11}$$



(a) A center region is preferred.



(b) Three regions are preferred.

Fig. 5 Preference on a hyperplane (plane) in three-objective optimization.

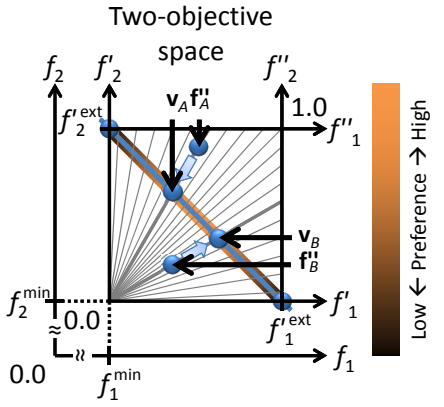


Fig. 6 Coordinate transformations of solutions to a hyperplane.

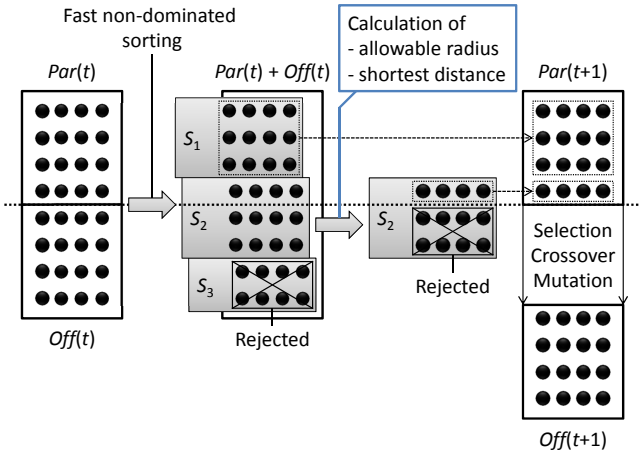


Fig. 7 Parent population update.

and $f_k^{\min}$ is the minimum value found so far for the k-th objective. Next, $\mathbf{f}'$ is normalized to $\mathbf{f}'' = (f''_1, f''_2, ..., f''_m)$ as follows:

$$f''_k = \frac{f'_k}{f'^{\text{ext}}_k}, \ k = 1, 2, ..., m, \tag{12}$$

where f'^{ext}_k is the k-th objective value of an extreme vector $\mathbf{f}'^{\text{ext}}_k$ for the k-th objective. The extreme vector $\mathbf{f}'^{\text{ext}}_k$ is identified by finding a vector that minimizes the following achievement scalarizing function ASF with a weight vector $\mathbf{d}^k = (d^k_1, d^k_2, ..., d^k_m)$ indicating the k-th objective direction (e.g., $\mathbf{d}^1 = (1, 0, ..., 0)$);

$$ASF(\mathbf{f}', \mathbf{d}_k) = \max_{i=1}^{m} f'_i / d^k_i, \tag{13}$$

where $d^k_i = 0$ is replaced with $d^k_i = 10^{-6}$ for avoiding a division by zero. Then, a hyperplane passing through m extreme vectors is generated and $\mathbf{f}''$ is mapped to $\mathbf{v} = (v_1, v_2, ..., v_m)$ on the hyperplane as follows:

$$v_i = \frac{f''_i}{f''_1 + f''_2 + ... + f''_m}, \ i = 1, 2, ..., m. \tag{14}$$

It should be noted that $v_1 + v_2 + ... + v_m = 1$ holds as $\mathbf{v}$ is on the hyperplane. These coordinate transformations are depicted in Fig. 6. Now we obtain the preference of the DM on the solution with the objective vector $\mathbf{f}$ using $p(\mathbf{v})$ in Eq. (8).

4.3 Integration into NSGA-II

The preference of a DM is considered in a parent population update of NSGA-II. The update procedure of the proposed method is illustrated in Fig. 7 where $Par(t)$ and $Off(t)$ mean the parent population and the offspring population at the t-th generation, respectively.

In the same manner as in NSGA-II, $Par(t)$ and $Off(t)$ are first merged into one population in Fig. 7. Then, non-dominated solutions in the merged population are identified as a set S_1. Next, non-dominated solutions in the merged population where solutions in S_1 are removed are identified as a set S_2. This procedure iterates while non-dominated solutions are identified. It should be noted that solutions in S_i have a rank of i.

As solutions having a smaller rank are preferred, solutions in S_1 are copied to $Par(t+1)$ whereas those in S_3 are rejected in Fig. 7. For each solution in S_2, an

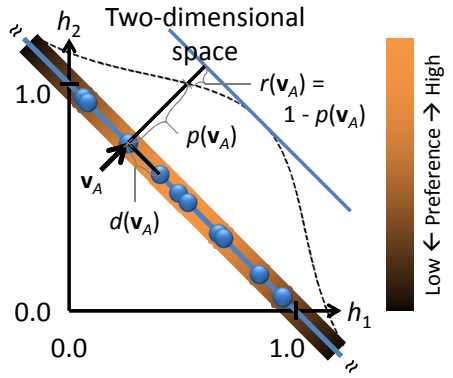


Fig. 8 Allowable radius and shortest distance for a solution that is mapped onto the hyperplane as $\mathbf{v}_A$.

allowable radius and a shortest distance are considered in this paper instead of a crowding distance in NSGA-II.

The allowable radius is defined for a solution having an objective vector $\mathbf{f} = (f_1, f_2, ..., f_m)$ as follows:

$$r(\mathbf{v}) = 1 - p(\mathbf{v}), \tag{15}$$

where $\mathbf{v}$ is obtained from $\mathbf{f}$ by Eqs. (11)-(12), (14), and $p(\mathbf{v})$ is calculated by Eqs. (8)-(9). The shortest distance is defined as the Euclidean distance of a solution to its nearest neighbor on the hyperplane. The allowable radius and the shortest distance are illustrated for a solution that is mapped onto the hyperplane as $\mathbf{v}_A$ in Fig. 8 where a dashed line represents the preference function $p(\mathbf{v})$ in Eq. (8).

In our proposal, the allowable radius and the shortest distance are considered for a set S_L of solutions having the L-th rank. The value of L is the minimum integer that makes $|S_1 \cup S_2 \cup ... \cup S_L| \geq \mu$ hold, where μ is the parent population size ($L = 2$ in Fig. 7). First of all, a solution is randomly selected from S_L. Then the allowable radius r and the shortest distance d is calculated for the solution. If $r > d$ holds, the solution is removed from S_L. Otherwise, another solution is randomly selected from S_L prohibiting the same solution(s) to be chosen, and the allowable radius and the shortest distance are compared. If any solutions in S_L are not removed from the set, the solution that has the minimum shortest distance is removed from S_L. We iterate this procedure until $|S_1 \cup S_2 \cup ... \cup S_L| = \mu$ holds. By removing solutions whose shortest distance is smaller than the allowable radius, we expect that solutions in S_L follows the distribution of $p(\mathbf{v})$ in Eq. (8).

4.4 Preference update

As mentioned in Subsection 3.6, a DM can visually specify his/her preference in our proposed method. In the method, current solutions and center vectors of Gaussian functions on the hyperplane can be depicted on a parallel coordinate plot. If the current solutions distribute around the center vectors on a hyperplane, we can say that the current solutions are preferred by the DM. If the DM does not like the distribution of current solutions in the original objective space that can also be drawn in a parallel coordinate plot, the DM will update his/her preference by changing the center and spread vectors of Gaussian functions so that current solutions follow the distribution of the Gaussian functions.

5 Computational experiments

This section demonstrates the effect of integrating the preference representation into NSGA-II. The NSGA-II algorithm with the preference representation is denoted as P-NSGA-II in our paper. P-NSGA-II is applied to DTLZ problems, a two-objective welded beam design problem (WBDP), and a 7-objective hybrid car controller optimization problem (HCCOP). In the following subsections, the optimization problems, performance indicators to evaluate obtained solutions, and the results of computational experiments are provided.

5.1 Optimization problems

We apply P-NSGA-II to benchmark problems and real-world applications. As the benchmark problems, DTLZ problems[5] [15–17] are selected, that are scalable with respect to the number of decision variables and the number of objectives.

We employ a two-objective welded beam design problem (WBDP) [14] and a 7-objective hybrid car controller optimization problem (HCCOP) [32] as real-world applications. In WBDP, four decision variables are optimized so that an end deflection of a welded beam and the fabrication cost are minimized. It is apparent that WBDP is no longer a many-objective optimization problem. However, it is worth testing that our method developed for many objectives can solve less objectives (i.e., two objectives in WBDP).

HCCOP is given as a black box that has 11 decision variables and 7 objectives. It is often the case that

[5] The total number of DTLZ problems depends on which paper to be referred to. For example, [15] has nine DTLZ problems whereas [16] has seven DTLZ problems in which DTLZ5 and DTLZ9 in [15] are removed, and DTLZ6, DTLZ7, DTLZ8 in [15] are described as DTLZ5, DTLZ6, DTLZ7, respectively. This paper uses the first seven DTLZ problems in [15]

the contents of optimization problems are not available in companies due to, for example, their policy. Therefore, it makes sense to validate the performance of our method on such a black box problem.

5.2 Performance indicators

For optimization problems with less than four objectives, the obtained solutions are visualized using a scatter plot so as to evaluate them. With more than three objectives, some performance indicators developed so far can be used to evaluate the obtained solutions [34]. Among them, we use inverse generational distance, convergence, and hypervolume measures as they are frequently used to evaluate solutions from the following perspectives: convergence, diversity of solutions in the objective space, and satisfaction of a DM. Each performance indicator is defined as follows.

5.2.1 Inverse generational distance

In order to assess how well P-NSGA-II searches for preferred regions, an inverse generational distance is used in the same manner as in [11]. This metric is calculated for a solutions set $S = \{s_1, s_2, ..., s_\mu\}$ as follows:

$$IGD(S, R) = \frac{1}{|R|} \sum_{i=1}^{|R|} \min_{j=1}^{\mu} d(r_i, s_j), \tag{16}$$

where $d(r_i, s_j)$ is the Euclidean distance between r_i and s_j in the objective space, $R = \{r_1, r_2, ..., r_{|R|}\}$ is a set of targeted points on the Pareto front. The targeted point is an intersection of the Pareto front and a line through the origin and the center vector specified by a DM on a hyperplane in P-NSGA-II. The hyperplane is constructed by using extreme points of the Pareto front. Needless to say, smaller values of $IGD(S, R)$ indicate better convergence of S to regions preferred by the DM.

5.2.2 Convergence

A convergence measure $CONV(S, P)$ evaluates the convergence of a solutions set $S = \{s_1, s_2, ..., s_\mu\}$ to the Pareto front P as follows:

$$CONV(S, P) = \frac{1}{\mu} \sum_{i=1}^{\mu} d(s_i, P), \tag{17}$$

where $d(s_i, P)$ is the minimal Euclidean distance from s_i to the Pareto front in the objective space. Smaller values of $CONV(S, P)$ indicate better convergence of S to the Pareto front.

Table 1 Parameters specification.

Parameter	Value
Population size	100
Generation number	1001
Crossover probability	1.0
Mutation probability	$1/n$
Crossover distribution index η_c	15
Mutation distribution index η_m	20

Table 2 The number of decision variables n in m-objective DTLZ problems.

Problem	n
DTLZ1	$m + 4$
DTLZ2, 3, 4, 5	$m + 9$
DTLZ7	$m + 19$

5.2.3 Hypervolume

A hypervolume $HV(S, \mathbf{r}^*)$ is calculated in the objective space for a set S of obtained solutions as a space covered by a reference vector $\mathbf{r}^*$ and solutions in S that dominate $\mathbf{r}^*$. The nadir vector[6] or a vector that is dominated by the nadir vector will be used as $\mathbf{r}^*$. It is reasonable to monitor $HV(S, \mathbf{r}^*)$ as $HV(S, \mathbf{r}^*)$ is maximized by the Pareto front P (i.e., $0 \leq HV(S, \mathbf{r}^*) \leq HV(P, \mathbf{r}^*)$) [50, 47].

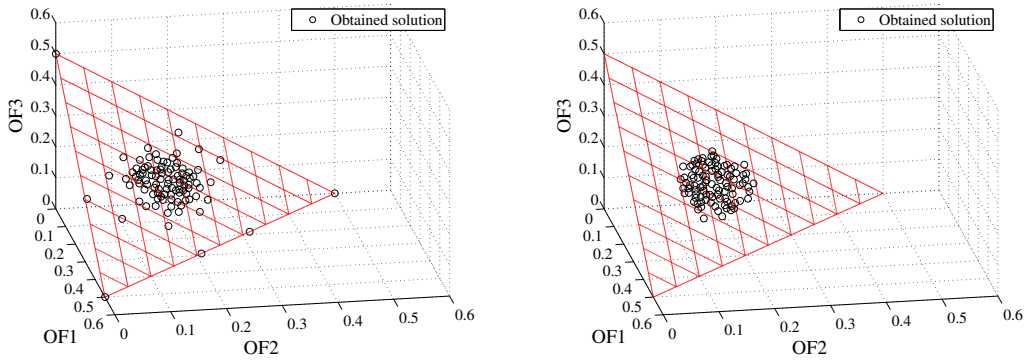
If the Pareto front P is known and given, P is used to calculate the relative hypervolume $RHV(S, P, \mathbf{r}^*) = HV(S, \mathbf{r}^*)/HV(P, \mathbf{r}^*)$ as suggested in [40, 23]. $RHV(S, P, \mathbf{r}^*)$ takes values from 0 to 1. If $RHV(S, P, \mathbf{r}^*) = 0$, any solutions that dominate $\mathbf{r}^*$ are not included in S. On the other hand, $RHV(S, P, \mathbf{r}^*)$ being close to one implies the sufficient approximation of P by S. It should be noted that $RHV(S, P, \mathbf{r}^*) = 1$ can not be achieved by a finite number of solutions for continuous problems, in general.

5.3 Results of computational experiments

We applied P-NSGA-II to DTLZ problems using the parameters specification in Table 1, where n is the number of decision variables. The value of n is specified as in Table 2 for m-objective DTLZ problems. In order to show the performance of P-NSGA-II, we also applied R-NSGA-II [14] (refer to Subsection 3.3) to the same problems. R-NSGA-II prefers solutions close to the reference point(s) $\mathbf{r}$ and allows only one solution to exist within a distance of ϵ in the objective space.

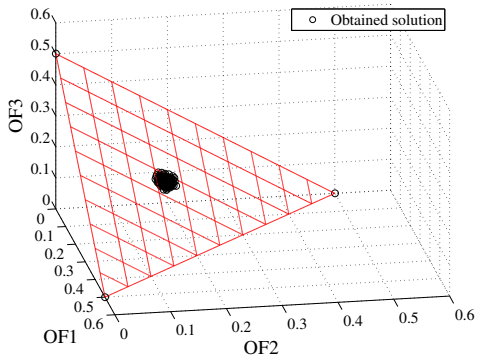
First, different spread vectors in Eq. (9) are tried for three-objective DTLZ1. Solutions obtained by P-NSGA-II and R-NSGA-II are depicted in Fig. 9. The

[6] The nadir vector is defined as a vector consisting of the worst value of each objective on the Pareto front.

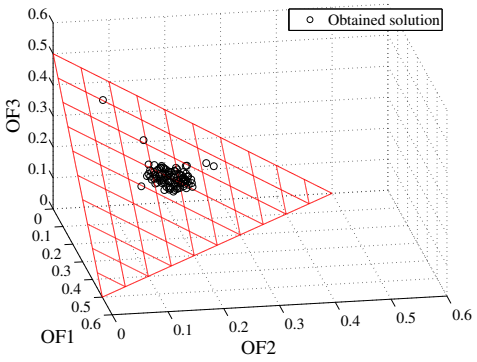


(a) P-NSGA-II

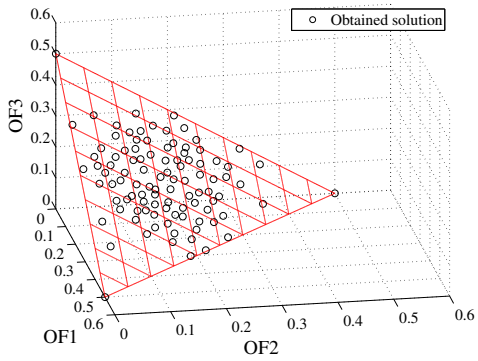
(b) R-NSGA-II



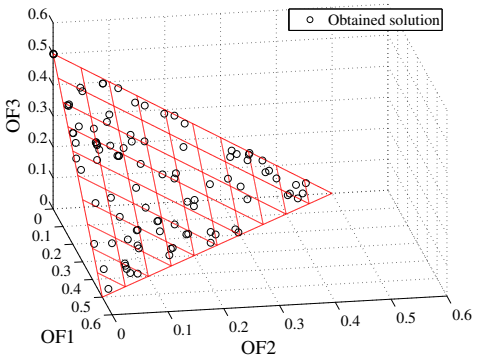
(c) P-NSGA-II



(d) R-NSGA-II



(e) P-NSGA-II



(f) R-NSGA-II

Fig. 9 Solutions obtained by P-NSGA-II and R-NSGA-II for three-objective DTLZ1 with preference of a center region. Preference is specified in Table 3.

Table 3 Preference specifications in Fig. 9.

	P-NSGA-II			R-NSGA-II	
	Center vector	Spread vector		Reference vector	Niching parameter
(a)	$\mathbf{w} = (1/3, 1/3, 1/3)$	$\mathbf{s} = (1.0, 1.0, 1.0)$	(b)	$\mathbf{r} = (1/6, 1/6, 1/6)$	$\varepsilon = 0.0001$
(c)	$\mathbf{w} = (1/3, 1/3, 1/3)$	$\mathbf{s} = (0.5, 0.5, 0.5)$	(d)	$\mathbf{r} = (1/6, 1/6, 1/6)$	$\varepsilon = 0.00005$
(e)	$\mathbf{w} = (1/3, 1/3, 1/3)$	$\mathbf{s} = (1.5, 1.5, 1.5)$	(f)	$\mathbf{r} = (1/6, 1/6, 1/6)$	$\varepsilon = 0.0005$

mean and spread vectors used in P-NSGA-II are shown in Table 3 as well as the reference vectors (points) and values of ϵ used in R-NSGA-II.

From Fig. 9(a), well-distributed solutions in the objective space are obtained by P-NSGA-II. Especially, the distribution in a center region of the Pareto front is dense while that away from the center region is sparse

whereas the previous version only obtains uniformly distributed solutions (see Fig. 16(b) for the comparison).

R-NSGA-II also obtains solutions close to the reference point in Fig. 9(b). However, the distribution is dense in some regions and sparse in the other regions, that cannot be controlled by its parameters (i.e., $\mathbf{r}$ and ϵ).

The effect of changing diversity-related parameters (i.e., $\mathbf{s}$ in P-NSGA-II and ϵ in R-NSGA-II) is examined in Figs. 9(c) and 9(e) for P-NSGA-II, and in Figs. 9(d) and 9(f) for R-NSGA-II. As can be seen from these figures, the parameters $\mathbf{s}$ and ϵ work well for controlling the diversity of solutions in the objective space in P-NSGA-II and R-NSGA-II, respectively. It should be noted that extreme solutions are always captured in P-NSGA-II to generate a hyperplane. We also note that R-NSGA-II obtains some solutions far away from the center region of the Pareto front in Fig. 9(d) and overlapped solutions in Fig. 9(f).

Next, different mean vectors in Eq. (9) are tried for P-NSGA-II on three-objective DTLZ1 in Fig. 10. Solutions obtained by R-NSGA-II are also depicted. The used parameters relating to the preference handling in P-NSGA-II and R-NSGA-II are shown in Table 4. From Fig. 10, P-NSGA-II and R-NSGA-II can focus their search on multiple regions specified in Table 4. It should be noted that the distribution of solutions obtained by P-NSGA-II looks more reasonable that those obtained by R-NSGA-II from these results.

The same consideration can be obtained for P-NSGA-II on three-objective DTLZ2 in Figs. 11 and 12. For R-NSGA-II, however, the niching parameter ϵ does not affect the distribution of obtained solutions in Figs. 11(b), 11(d), and 11(f). Moreover, the solutions obtained by R-NSGA-II do not reach the Pareto front compared to P-NSGA-II in Fig. 12.

P-NSGA-II and R-NSGA-II are applied to three objective DTLZ5 and DTLZ7 in Figs. 13 and 14, respectively. From these results, it can be said that both the methods focus the search on preferred regions except for R-NSGA-II applied to DTLZ7 in Fig. 14(b).

The performance of P-NSGA-II is also examined for two-objective WBDP with different preference specifications in Table 9. The obtained solutions by P-NSGA-II are illustrated in Fig. 15. P-NSGA-II focuses the search on regions defined in Table 9.

In order to assess how well P-NSGA-II searches for preferred regions, the inverse generational distance (IGD) metric in Eq. 16 is used in the same manner as in [11]. P-NSGA-II is applied to DTLZ1, DTLZ2, DTLZ3, and DTLZ4, each with 3, 6, 9, and 12 objectives using different values of σ in the spread vector $\mathbf{s} = (\sigma, \sigma, ..., \sigma)$: $\sigma = 0.5$, $\sigma = 1.0$, and $\sigma = 1.5$. The mean of IGD, its standard deviation (S. D.), the maximum and minimum over 20 runs are summarized in Table 10. It should be noted that the center vector is specified as $(1/m, 1/m, ..., 1/m)$ for each problem given m being the number of objectives.

From Table 10, it can be said that P-NSGA-II shows high performance except for many-objective DTLZ3 problems (i.e., DTLZ3 with 6, 9, and 12 objectives). The DTLZ3 problems include local Pareto fronts. Thereby, P-NSGA-II might get stuck in the local Pareto fronts and could not escape from there.

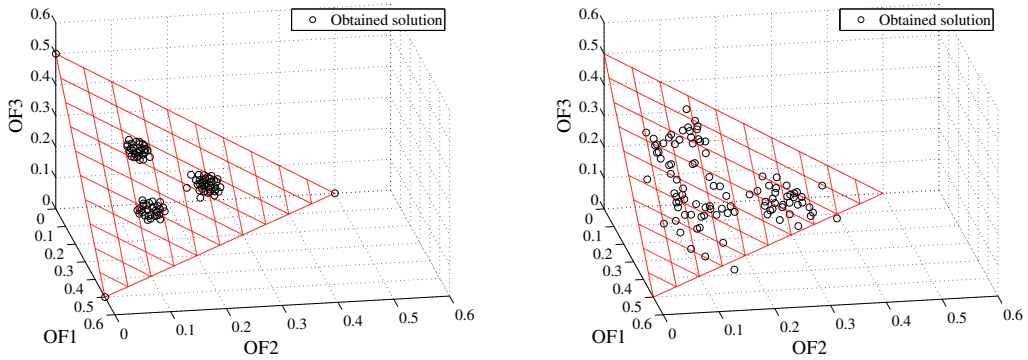
For DTLZ2 and DTLZ4, smaller means of IGD are obtained by smaller values of σ. For DTLZ1, the smallest minimum of IGD is achieved by $\sigma = 0.5$ except for DTLZ1 with six objectives. The standard deviation of IGD for DTLZ1 with 6, 9, 12 objectives is large when $\sigma = 0.5$ is used. From these facts, smaller values of σ accelerate the convergence of solutions to regions preferred by a DM whereas the robustness may be lost.

5.4 Comparison to the previous work

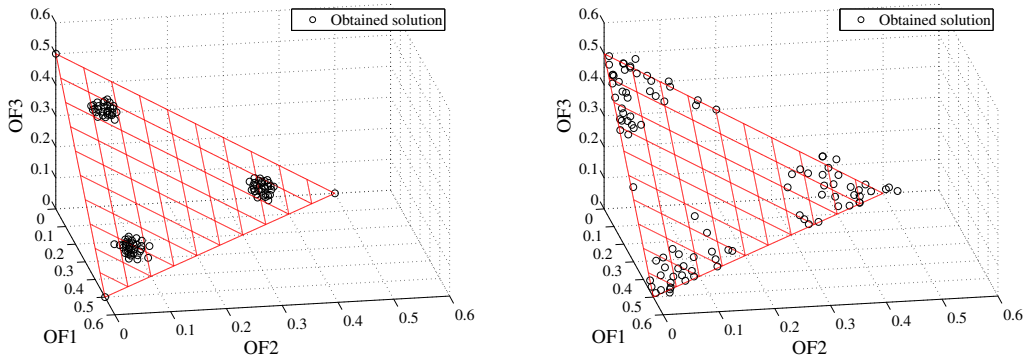
We also compare the distribution of solutions obtained by P-NSGA-II and its former version (denoted as P-NSGA-II-β in this paper) for three-objective DTLZ1 and DTLZ2 problems in order to show the difference. The used parameters specification is the same as in Table 1 and the weight vector used is $\mathbf{w} = (1/3, 1/3, 1/3)$ for P-NSGA-II and P-NSGA-II-β.

In P-NSGA-II-β, a preferred level is calculated for each solution with the same rank using Eq. (8). Among solutions with the same rank, those having the larger preferred level are preferred more. It should be noted that for each solution having the Euclidean distance of 0.01 or less to its nearest neighbor(s) in the objective space, the preferred level is specified as 0 so as to keep the diversity of solutions in the space.

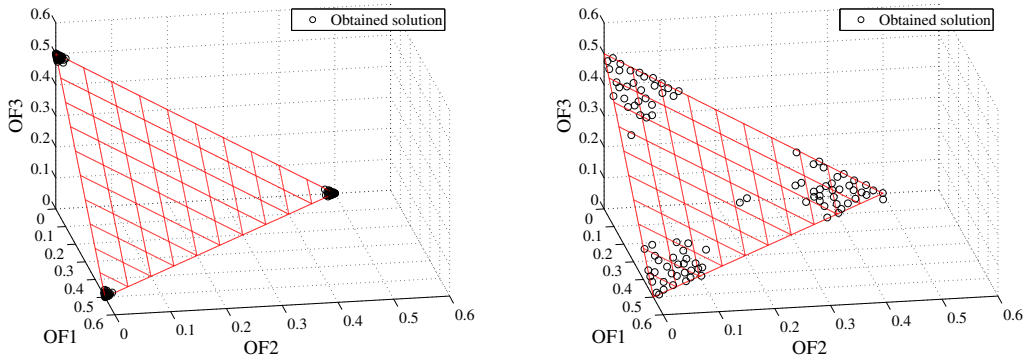
Fig. 16 plots solutions obtained by P-NSGA-II and P-NSGA-II-β for three-objective DTLZ1 and DTLZ2 problems. From the comparison in Fig. 16, we can see that for each problem P-NSGA-II-β obtains uniformly distributed solutions around a center region of the Pareto front whereas P-NSGA-II obtains dense solutions at the center and sparse ones away from there. That is, P-NSGA-II gives a decision maker more candidates to select at the center region of the Pareto front. This is the advantage of using P-NSGA-II over P-NSGA-II-β.



(a) P-NSGA-II (b) R-NSGA-II



(c) P-NSGA-II (d) R-NSGA-II

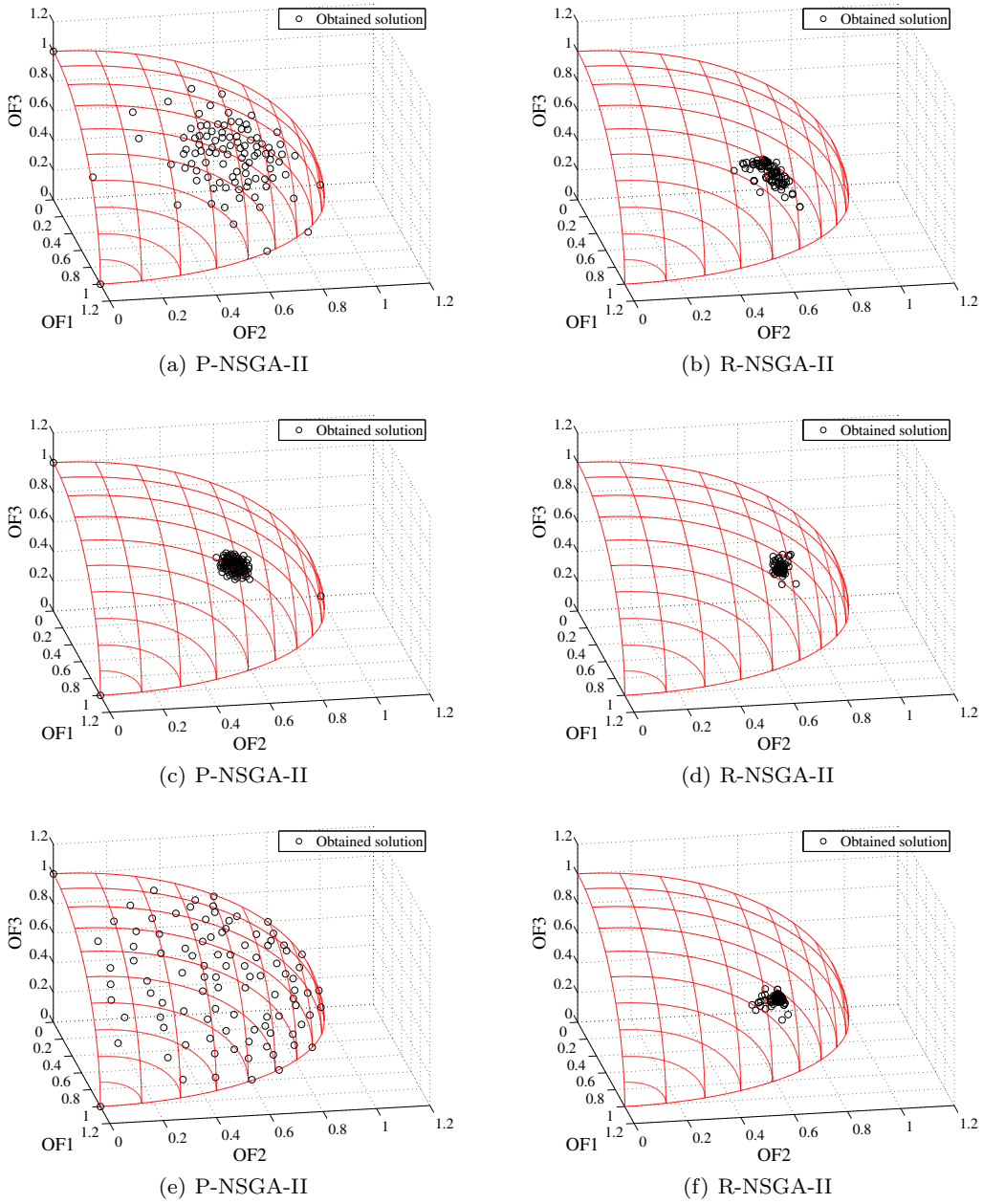


(e) P-NSGA-II (f) R-NSGA-II

Fig. 10 Solutions obtained by P-NSGA-II and R-NSGA-II for three-objective DTLZ1 with preference of three regions. Preference is specified in Table 4.

Table 4 Preference specifications in Fig. 10.

	P-NSGA-II			R-NSGA-II	
	Center vector	Spread vector		Reference vector	Niching parameter
(a)	$\mathbf{w}$ = (1/2, 1/4, 1/4)	$\mathbf{s}$ = (0.5, 0.5, 0.5)	(b)	$\mathbf{r}$ = (1/4, 1/8, 1/8)	ε = 0.0001
	$\mathbf{w}$ = (1/4, 1/2, 1/4)	$\mathbf{s}$ = (0.5, 0.5, 0.5)		$\mathbf{r}$ = (1/8, 1/4, 1/8)	
	$\mathbf{w}$ = (1/4, 1/4, 1/2)	$\mathbf{s}$ = (0.5, 0.5, 0.5)		$\mathbf{r}$ = (1/8, 1/8, 1/4)	
(c)	$\mathbf{w}$ = (5/7, 1/7, 1/7)	$\mathbf{s}$ = (0.5, 0.5, 0.5)	(d)	$\mathbf{r}$ = (3/10, 1/10, 1/10)	ε = 0.0001
	$\mathbf{w}$ = (1/7, 5/7, 1/7)	$\mathbf{s}$ = (0.5, 0.5, 0.5)		$\mathbf{r}$ = (1/10, 3/10, 1/10)	
	$\mathbf{w}$ = (1/7, 1/7, 5/7)	$\mathbf{s}$ = (0.5, 0.5, 0.5)		$\mathbf{r}$ = (1/10, 1/10, 3/10)	
(e)	$\mathbf{w}$ = (1.0, 0.0, 0.0)	$\mathbf{s}$ = (0.5, 0.5, 0.5)	(f)	$\mathbf{r}$ = (0.5, 0.0, 0.0)	ε = 0.0001
	$\mathbf{w}$ = (0.0, 1.0, 0.0)	$\mathbf{s}$ = (0.5, 0.5, 0.5)		$\mathbf{r}$ = (0.0, 0.5, 0.0)	
	$\mathbf{w}$ = (0.0, 0.0, 1.0)	$\mathbf{s}$ = (0.5, 0.5, 0.5)		$\mathbf{r}$ = (0.0, 0.0, 0.5)	



(a) P-NSGA-II (b) R-NSGA-II

(c) P-NSGA-II (d) R-NSGA-II

(e) P-NSGA-II (f) R-NSGA-II

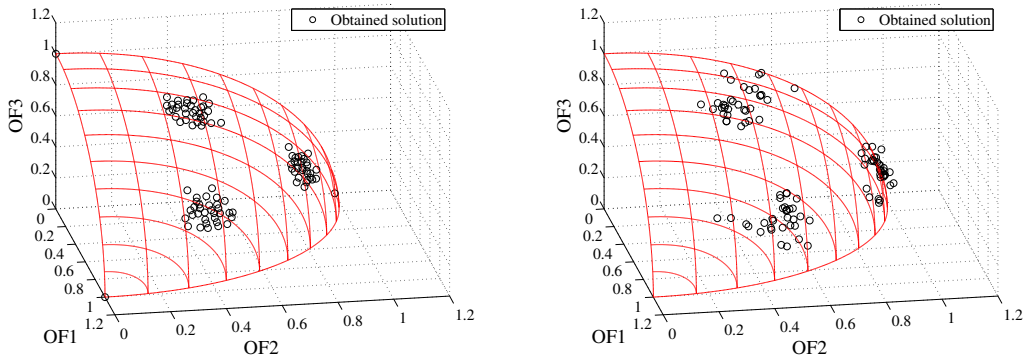
Fig. 11 Solutions obtained by P-NSGA-II and R-NSGA-II for three-objective DTLZ2 with preference of a center region. Preference is specified in Table 5.

Table 5 Preference specifications in Fig. 11.

	P-NSGA-II			R-NSGA-II	
	Center vector	Spread vector		Reference vector	Niching parameter
(a)	$\mathbf{w} = (1/3, 1/3, 1/3)$	$\mathbf{s} = (1.0, 1.0, 1.0)$	(b)	$\mathbf{r} = (1/3, 1/3, 1/3)$	$\varepsilon = 0.0001$
(c)	$\mathbf{w} = (1/3, 1/3, 1/3)$	$\mathbf{s} = (0.5, 0.5, 0.5)$	(d)	$\mathbf{r} = (1/3, 1/3, 1/3)$	$\varepsilon = 0.00005$
(e)	$\mathbf{w} = (1/3, 1/3, 1/3)$	$\mathbf{s} = (1.5, 1.5, 1.5)$	(f)	$\mathbf{r} = (1/3, 1/3, 1/3)$	$\varepsilon = 0.0005$

5.5 Change of preference during optimization

In the framework of P-NSGA-II, it is possible to change the center and spread vectors on a hyperplane during optimization by a DM. In order to examine how well P-NSGA-II handles the preferred region(s) that will be altered by a DM, P-NSGA-II is applied to three-objective



(a) P-NSGA-II (b) R-NSGA-II

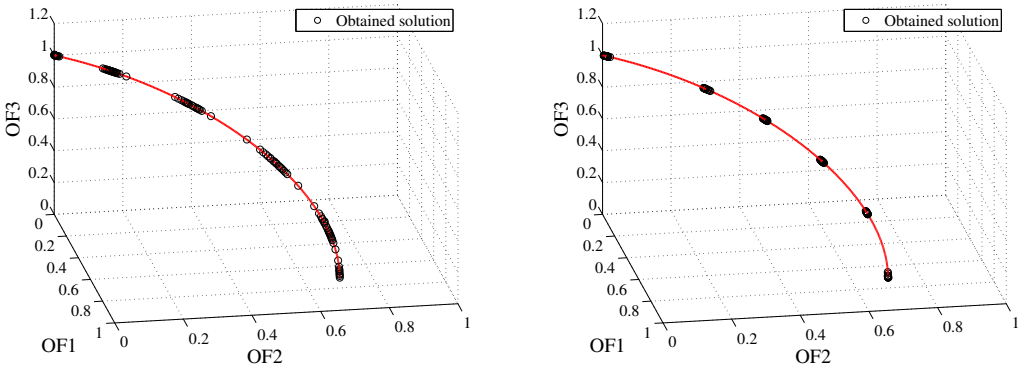
(c) P-NSGA-II (d) R-NSGA-II

(e) P-NSGA-II (f) R-NSGA-II

Fig. 12 Solutions obtained by P-NSGA-II and R-NSGA-II for three-objective DTLZ2 with preference of three regions. Preference is specified in Table 6.

Table 6 Preference specifications in Fig. 12.

	P-NSGA-II			R-NSGA-II	
	Center vector	Spread vector		Reference vector	Niching parameter
(a)	$\mathbf{w}$ = (1/2, 1/4, 1/4)	$\mathbf{s}$ = (0.5, 0.5, 0.5)	(b)	$\mathbf{r}$ = (1/2, 1/4, 1/4)	ε = 0.0001
	$\mathbf{w}$ = (1/4, 1/2, 1/4)	$\mathbf{s}$ = (0.5, 0.5, 0.5)		$\mathbf{r}$ = (1/4, 1/2, 1/4)	
	$\mathbf{w}$ = (1/4, 1/4, 1/2)	$\mathbf{s}$ = (0.5, 0.5, 0.5)		$\mathbf{r}$ = (1/4, 1/4, 1/2)	
(c)	$\mathbf{w}$ = (5/7, 1/7, 1/7)	$\mathbf{s}$ = (0.5, 0.5, 0.5)	(d)	$\mathbf{r}$ = (3/5, 1/5, 1/5)	ε = 0.0001
	$\mathbf{w}$ = (1/7, 5/7, 1/7)	$\mathbf{s}$ = (0.5, 0.5, 0.5)		$\mathbf{r}$ = (1/5, 3/5, 1/5)	
	$\mathbf{w}$ = (1/7, 1/7, 5/7)	$\mathbf{s}$ = (0.5, 0.5, 0.5)		$\mathbf{r}$ = (1/5, 1/5, 3/5)	
(e)	$\mathbf{w}$ = (1.0, 0.0, 0.0)	$\mathbf{s}$ = (0.5, 0.5, 0.5)	(f)	$\mathbf{r}$ = (1.0, 0.0, 0.0)	ε = 0.0001
	$\mathbf{w}$ = (0.0, 1.0, 0.0)	$\mathbf{s}$ = (0.5, 0.5, 0.5)		$\mathbf{r}$ = (0.0, 1.0, 0.0)	
	$\mathbf{w}$ = (0.0, 0.0, 1.0)	$\mathbf{s}$ = (0.5, 0.5, 0.5)		$\mathbf{r}$ = (0.0, 0.0, 1.0)	

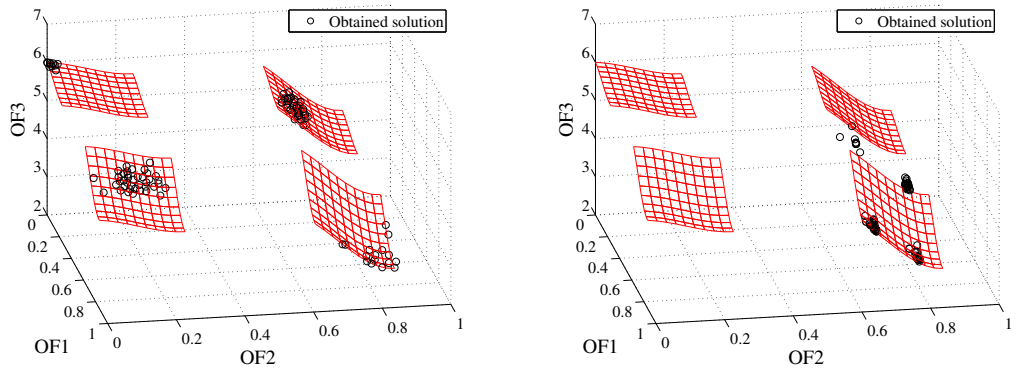


(a) P-NSGA-II (b) R-NSGA-II

Fig. 13 Solutions obtained by P-NSGA-II and R-NSGA-II for three-objective DTLZ5 with preference of six regions. Preference is specified in Table 7.

Table 7 Preference specifications in Fig. 13.

	P-NSGA-II			R-NSGA-II	
	Center vector	Spread vector		Reference vector	Niching parameter
(a)	$\mathbf{w}$ = (0.0, 0.0, 1.0)	$\mathbf{s}$ = (0.5, 0.5, 0.5)	(b)	$\mathbf{r}$ = (0.0, 0.0, 0.7)	ε = 0.0001
	$\mathbf{w}$ = (1/6, 1/6, 2/3)	$\mathbf{s}$ = (0.5, 0.5, 0.5)		$\mathbf{r}$ = (0.2, 0.2, 0.5)	
	$\mathbf{w}$ = (2/7, 2/7, 3/7)	$\mathbf{s}$ = (0.5, 0.5, 0.5)		$\mathbf{r}$ = (0.3, 0.3, 0.4)	
	$\mathbf{w}$ = (3/8, 3/8, 1/4)	$\mathbf{s}$ = (0.5, 0.5, 0.5)		$\mathbf{r}$ = (0.4, 0.4, 0.3)	
	$\mathbf{w}$ = (4/9, 4/9, 1/9)	$\mathbf{s}$ = (0.5, 0.5, 0.5)		$\mathbf{r}$ = (0.5, 0.5, 0.2)	
	$\mathbf{w}$ = (0.5, 0.5, 0.0)	$\mathbf{s}$ = (0.5, 0.5, 0.5)		$\mathbf{r}$ = (0.7, 0.7, 0.0)	



(a) P-NSGA-II (b) R-NSGA-II

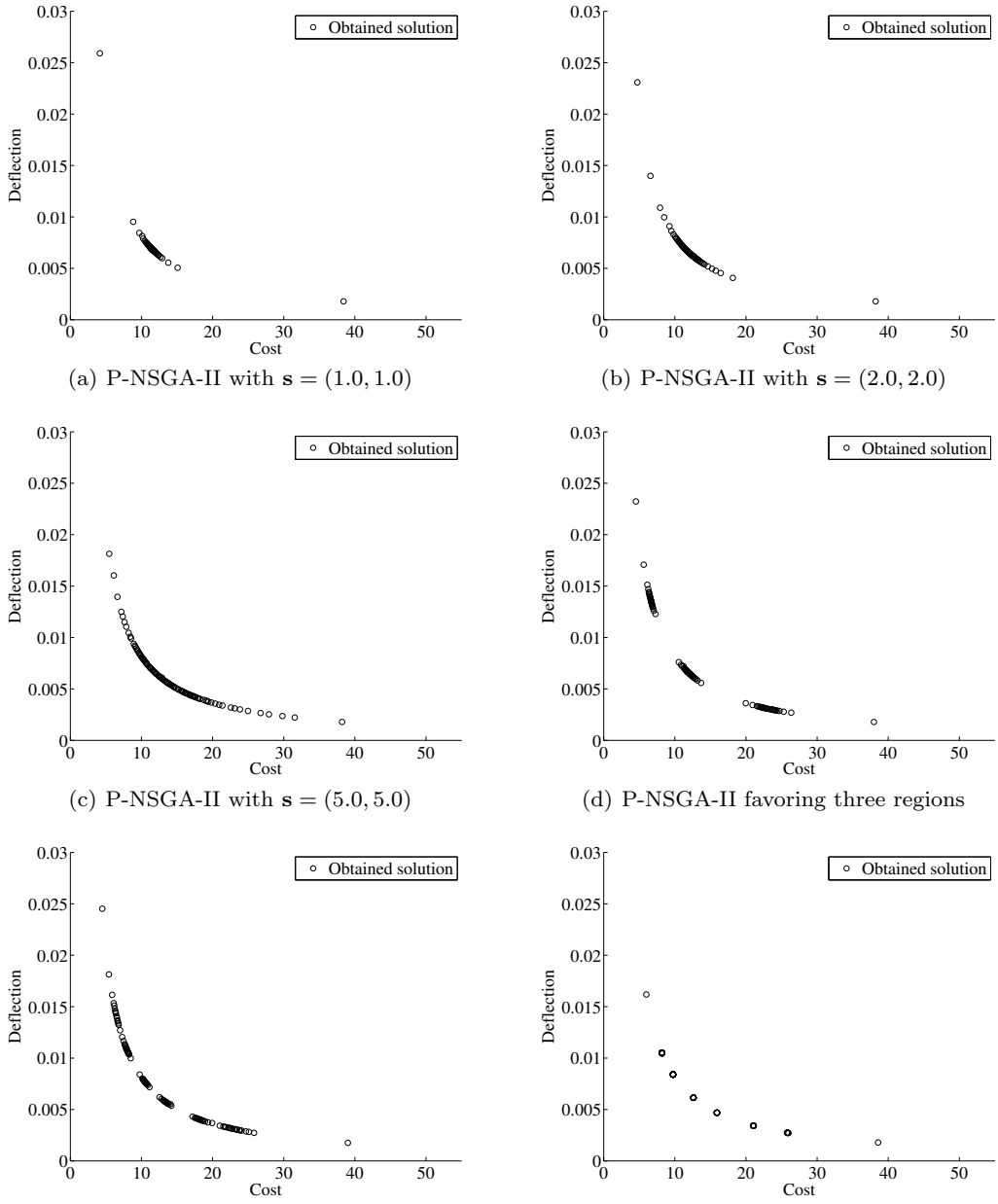
Fig. 14 Solutions obtained by P-NSGA-II and R-NSGA-II for three-objective DTLZ7 with preference of four regions. Preference is specified in Table 8.

Table 8 Preference specifications in Fig. 14.

	P-NSGA-II			R-NSGA-II	
	Center vector	Spread vector		Reference vector	Niching parameter
(a)	$\mathbf{w}$ = (0.5, 0.5, 0.0)	$\mathbf{s}$ = (0.5, 0.5, 0.5)	(b)	$\mathbf{r}$ = (0.1, 0.1, 0.55)	ε = 0.0001
	$\mathbf{w}$ = (0.0, 0.0, 1.0)	$\mathbf{s}$ = (0.5, 0.5, 0.5)		$\mathbf{r}$ = (0.7, 0.7, 0.4)	
	$\mathbf{w}$ = (0.5, 0.1, 0.4)	$\mathbf{s}$ = (0.5, 0.5, 0.5)		$\mathbf{r}$ = (0.7, 0.1, 0.5)	
	$\mathbf{w}$ = (0.1, 0.5, 0.4)	$\mathbf{s}$ = (0.5, 0.5, 0.5)		$\mathbf{r}$ = (0.1, 0.7, 0.5)	

DTLZ1 in Fig. 17 where the preference of the DM is assumed to change as in Table 11.

P-NSGA-II is also applied to WBDP in Fig. 18 with preference specifications in Table 12. From these results in Figs. 17 and 18, it can be said that P-NSGA-II easily adapts the search to the changed region(s) preferred by a DM. It should be noted that P-NSGA-II also provides the DM with discriminable options by specifying



(a) P-NSGA-II with $\mathbf{s} = (1.0, 1.0)$

(b) P-NSGA-II with $\mathbf{s} = (2.0, 2.0)$

(c) P-NSGA-II with $\mathbf{s} = (5.0, 5.0)$

(d) P-NSGA-II favoring three regions

(e) P-NSGA-II favoring six regions

(f) P-NSGA-II with smaller values in $\mathbf{s}$

Fig. 15 Solutions obtained by P-NSGA-II for two-objective WBDP. Preference is specified in Table 9.

Table 9 Preference specifications in Fig. 15.

P-NSGA-II					
	Center vector	Spread vector		Center vector	Spread vector
(a)	**w** = (0.5, 0.5)	**s** = (1.0, 1.0)	(b)	**w** = (0.5, 0.5)	**s** = (2.0, 2.0)
(c)	**w** = (0.5, 0.5)	**s** = (5.0, 5.0)	(d)	**w** = (0.9, 0.1) **w** = (0.5, 0.5) **w** = (0.1, 0.9)	**s** = (0.5, 0.5) **s** = (1.0, 1.0) **s** = (0.5, 0.5)
(e)	**w** = (0.9, 0.1) **w** = (0.8, 0.2) **w** = (0.6, 0.4) **w** = (0.4, 0.6) **w** = (0.2, 0.8) **w** = (0.1, 0.9)	**s** = (0.5, 0.5, 0.5) **s** = (0.5, 0.5, 0.5) **s** = (0.5, 0.5, 0.5) **s** = (0.5, 0.5, 0.5) **s** = (0.5, 0.5, 0.5) **s** = (0.5, 0.5, 0.5)	(f)	**w** = (0.9, 0.1) **w** = (0.8, 0.2) **w** = (0.6, 0.4) **w** = (0.4, 0.6) **w** = (0.2, 0.8) **w** = (0.1, 0.9)	**s** = (0.1, 0.1, 0.1) **s** = (0.1, 0.1, 0.1) **s** = (0.1, 0.1, 0.1) **s** = (0.1, 0.1, 0.1) **s** = (0.1, 0.1, 0.1) **s** = (0.1, 0.1, 0.1)

Table 10 *IGD* for solutions obtained by P-NSGA-II on DTLZ problems using different values of σ.

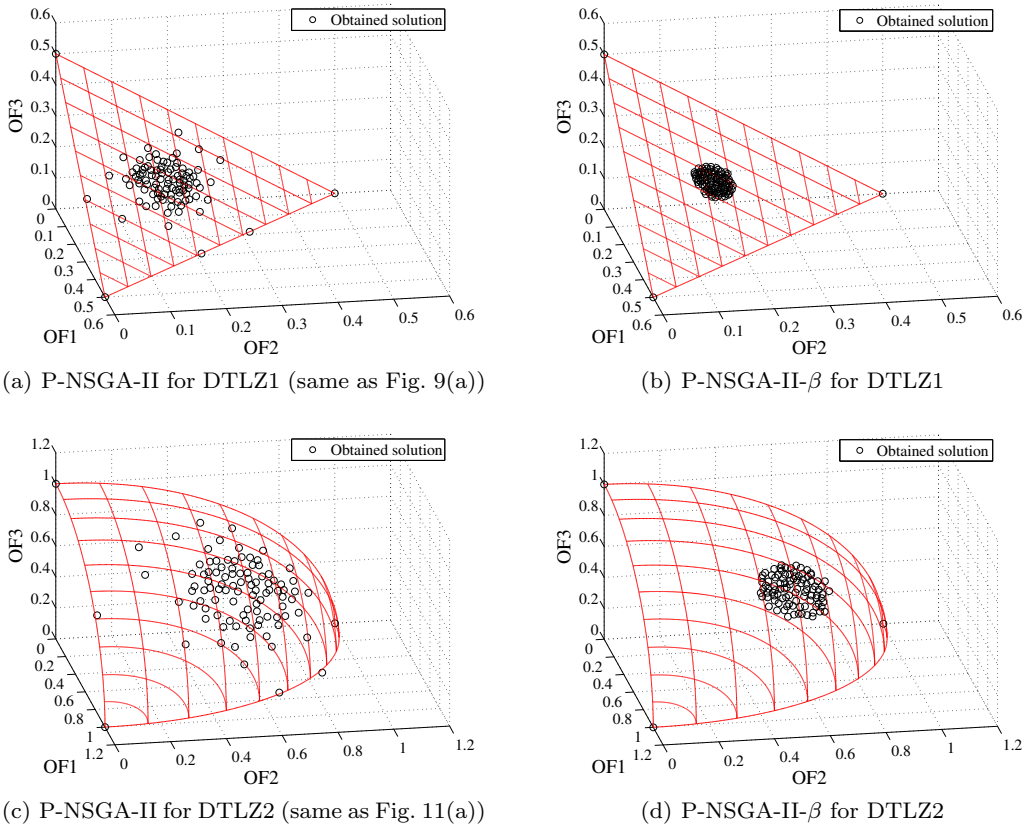
m	*IGD*	DTLZ1			DTLZ2		
		$\sigma = 0.5$	$\sigma = 1.0$	$\sigma = 1.5$	$\sigma = 0.5$	$\sigma = 1.0$	$\sigma = 1.5$
3	Mean	0.00175863	0.00632922	0.0138842	0.0057934	0.0220078	0.0603654
	S. D.	0.00056554	0.00269953	0.00625094	0.00218685	0.00913901	0.0221896
	Max	0.00307013	0.0109115	0.028303	0.00968681	0.037154	0.104445
	Min	0.00095532	0.00121733	0.00181872	0.00213589	0.00750099	0.0155799
6	Mean	0.279844	0.0741858	0.106501	0.100377	0.351112	0.433202
	S. D.	0.315845	0.0194134	0.0203351	0.0170997	0.0870452	0.0753267
	Max	0.94044	0.112587	0.144383	0.132555	0.445146	0.552965
	Min	0.0154173	0.042988	0.0563864	0.0625668	0.145358	0.217334
9	Mean	0.24333	0.263654	0.209867	0.341699	0.561845	0.652656
	S. D.	0.459102	0.235989	0.169136	0.150097	0.091093	0.0952315
	Max	2.17592	1.00077	0.862267	0.604532	0.71405	0.755006
	Min	0.0284503	0.0770086	0.101666	0.0270013	0.344374	0.342719
12	Mean	0.213644	0.294761	0.290069	0.487111	0.624975	0.713958
	S. D.	0.145053	0.217745	0.208247	0.156632	0.10952	0.10264
	Max	0.637266	0.985317	0.990439	0.842665	0.790547	0.841917
	Min	0.0895942	0.107507	0.0863938	0.264613	0.358866	0.514292
m	*IGD*	DTLZ3			DTLZ4		
		$\sigma = 0.5$	$\sigma = 1.0$	$\sigma = 1.5$	$\sigma = 0.5$	$\sigma = 1.0$	$\sigma = 1.5$
3	Mean	0.00529837	0.0296807	0.0442294	0.171906	0.158109	0.204684
	S. D.	0.00168033	0.0118831	0.0206389	0.293515	0.270133	0.275594
	Max	0.00845259	0.0606698	0.0821234	0.919402	0.919402	0.919402
	Min	0.0023318	0.0138009	0.00104259	0.00051721	0.00902537	0.0218671
6	Mean	26.5069	3.26407	4.45209	0.38606	0.453815	0.512454
	S. D.	11.9289	4.69948	6.17632	0.242543	0.162358	0.112435
	Max	48.9407	22.6148	30.1371	0.7654	0.919402	0.765405
	Min	11.242	0.220903	0.348566	0.0712117	0.146252	0.395453
9	Mean	27.2526	5.59493	7.35756	0.484691	0.491961	0.517954
	S. D.	11.8035	4.77109	5.5712	0.142158	0.087893	0.115953
	Max	52.5801	17.0533	24.2766	0.753022	0.652962	0.817233
	Min	11.1635	0.75546	1.57854	0.226236	0.348362	0.355192
12	Mean	14.5904	11.7174	12.9306	0.453481	0.594122	0.628919
	S. D.	6.82313	12.7745	8.00948	0.136384	0.0850342	0.0771816
	Max	33.1188	42.7623	34.8484	0.761975	0.735554	0.875277
	Min	3.75524	1.14262	2.68637	0.298992	0.406936	0.507199

Table 11 Preference specifications in Fig. 17.

	P-NSGA-II				
	Center vector	Spread vector		Center vector	Spread vector
(a)	Generations 0-1000		(b)	Generations 1001-1025	
	$\mathbf{w}$ = (1/3, 1/3, 1/3)	$\mathbf{s}$ = (1.0, 1.0, 1.0)		$\mathbf{w}$ = (1/3, 1/3, 1/3)	$\mathbf{s}$ = (0.5, 0.5, 0.5)
(c)	Generations 1026-1125		(d)	Generations 1126-1225	
	$\mathbf{w}$ = (1/5, 1/5, 3/5)	$\mathbf{s}$ = (0.5, 0.5, 0.5)		$\mathbf{w}$ = (1/5, 1/5, 3/5)	$\mathbf{s}$ = (1.0, 1.0)
				$\mathbf{w}$ = (1/5, 3/5, 1/5)	$\mathbf{s}$ = (1.0, 1.0)
				$\mathbf{w}$ = (3/5, 1/5, 1/5)	$\mathbf{s}$ = (1.0, 1.0)
(e)	Generations 1226-1325		(f)	Generations 1326-1425	
	$\mathbf{w}$ = (1/9, 1/9, 7/9)	$\mathbf{s}$ = (0.5, 0.5, 0.5)		$\mathbf{w}$ = (1/9, 1/9, 7/9)	$\mathbf{s}$ = (0.3, 0.3, 0.3)
	$\mathbf{w}$ = (1/9, 7/9, 1/9)	$\mathbf{s}$ = (0.5, 0.5, 0.5)		$\mathbf{w}$ = (1/9, 7/9, 1/9)	$\mathbf{s}$ = (0.3, 0.3, 0.3)
	$\mathbf{w}$ = (7/9, 1/9, 1/9)	$\mathbf{s}$ = (0.5, 0.5, 0.5)		$\mathbf{w}$ = (7/9, 1/9, 1/9)	$\mathbf{s}$ = (0.3, 0.3, 0.3)

Table 12 Preference specifications in Fig. 18.

	P-NSGA-II				
	Center vector	Spread vector		Center vector	Spread vector
(a)	Generations 0-200		(b)	Generations 201-400	
	$\mathbf{w}$ = (1.0, 1.0)	$\mathbf{s}$ = (1.0, 1.0)		$\mathbf{w}$ = (0.1, 0.9)	$\mathbf{s}$ = (1.0, 1.0)
(c)	Generations 401-600		(d)	Generations 601-800	
	$\mathbf{w}$ = (0.9, 0.1)	$\mathbf{s}$ = (1.0, 1.0)		$\mathbf{w}$ = (0.9, 0.1)	$\mathbf{s}$ = (0.5, 0.5, 0.5)
	$\mathbf{w}$ = (0.5, 0.5)	$\mathbf{s}$ = (1.0, 1.0)		$\mathbf{w}$ = (0.5, 0.5)	$\mathbf{s}$ = (0.5, 0.5, 0.5)
	$\mathbf{w}$ = (0.1, 0.9)	$\mathbf{s}$ = (1.0, 1.0)		$\mathbf{w}$ = (0.1, 0.9)	$\mathbf{s}$ = (0.5, 0.5, 0.5)



(a) P-NSGA-II for DTLZ1 (same as Fig. 9(a))

(b) P-NSGA-II-β for DTLZ1

(c) P-NSGA-II for DTLZ2 (same as Fig. 11(a))

(d) P-NSGA-II-β for DTLZ2

Fig. 16 Solutions obtained by P-NSGA-II and its former version P-NSGA-II-β for three-objective DTLZ1 and DTLZ2. The results of P-NSGA-II (i.e., (a) and (c)) are copied from the previous subsection for a comparison purpose.

multiple center vectors of Gaussian functions during optimization. This means that P-NSGA-II handles a situation where a DM has multiple preferences or multiple DMs have their preferences, that can be updated during optimization.

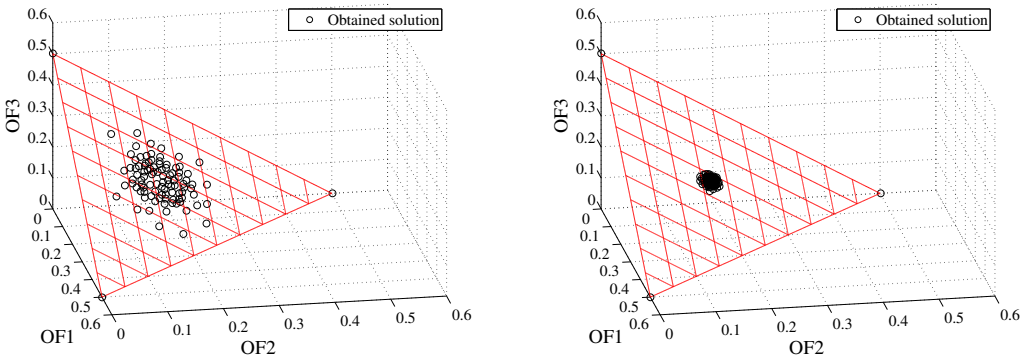
The adaptability of P-NSGA-II in Fig. 17 is measured by the convergence, the relative hypervolume, and IGD in Fig. 19. From Fig. 19(a), it can be observed that the convergence is improved through generations regardless of the preference specification. For the convergence of solutions to the Pareto front and their diversity in the objective space, larger values of the relative hypervolume are achieved between generations 1126 and 1225 in Fig. 19(b). Through the generations, the value of σ in the spread vector $\mathbf{s} = (\sigma, \sigma, \sigma)$ is specified as a comparatively large value (i.e., $\sigma = 1.0$). The specification made the distribution of solutions wider, resulting in larger values of the relative hypervolume. For the IGD measure, the decrease in the value of IGD (i.e., the improvement) is identified at the beginning of each range, b, c, d, e, and f in Fig. 19(c). This means that P-NSGA-II is capable of handling the variable preference during optimization.

It should be noted that the center vector used to calculate the IGD measure is changed at the generation 1026. Furthermore, the number of center vectors is increased to three at the generation 1126. As can be observed from Fig. 17, smaller values of σ in the spread vector(s) result in smaller values of IGD, meaning good convergence to the region(s) preferred by a DM.

Finally, P-NSGA-II is applied to a 7-objective hybrid car controller optimization problem (HCCOP) to demonstrate its usefulness in practice. The same parameters specification as in Table 1 is used with $n = 11$ being the number of decision variables. For different settings of the preference in P-NSGA-II, finally obtained solutions are plotted in Figs. 20 to 27 using a parallel coordinate plot. The center vector $\mathbf{w}$ and the spread vector $\mathbf{s} = (\sigma, \sigma, ..., \sigma)$ used in Gaussian functions are noted in the figures.

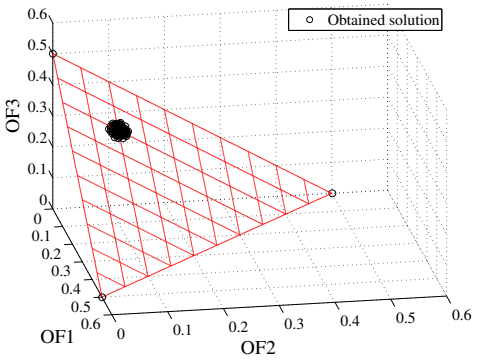
In Fig. 20(a), a pattern of lines through objectives is identified. With an increase in the value of σ, the width of the pattern becomes thick in Fig. 20. As the center vector is specified as $(1/7, 1/7, 1/7, 1/7, 1/7, 1/7, 1/7)$, all the objectives are equally handled in Figs. 20.

It should be noted that P-NSGA-II can focus the search on preferred regions. Let us assume that a de-

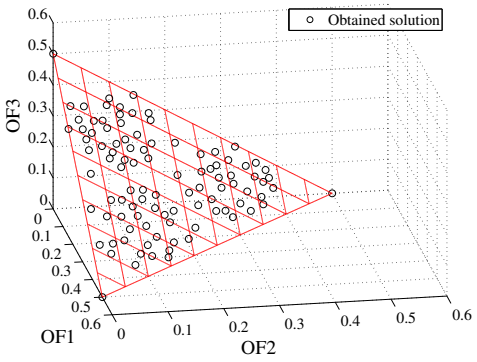


(a) 1000th generation

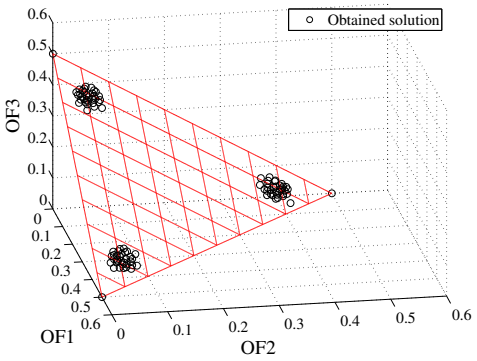
(b) 1025th generation



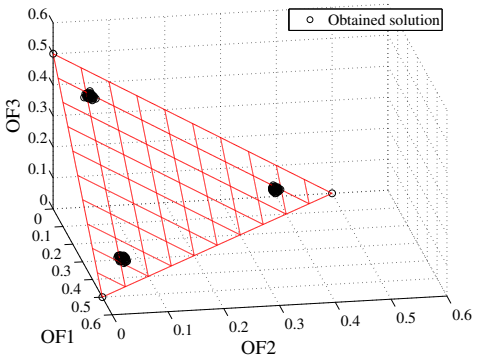
(c) 1125th generation



(d) 1225th generation



(e) 1325th generation



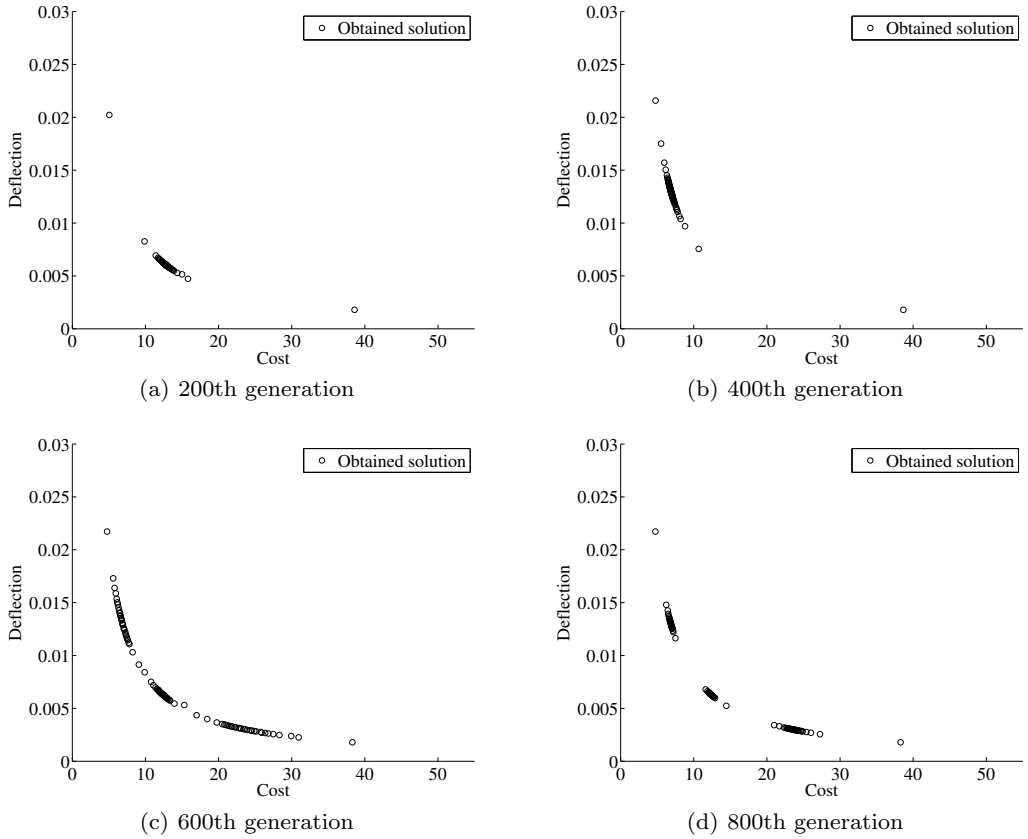
(f) 1425th generation

Fig. 17 Solutions obtained by P-NSGA-II for three-objective DTLZ1. Preference is specified in Table 11.

cision maker prefers the first objective of HCCOP ten times as much as each of the other objectives. In this case, the preference on the objectives can be written by a vector $\mathbf{u} = (10, 1, 1, 1, 1, 1, 1)$[7]. Then, each element in $\mathbf{u}$ is inversed and the normalized vector is used as a center vector $\mathbf{w}$ of Gaussian functions in P-NSGA-II[8]. This preference specification is used in Fig. 21. In the same manner, the second objective of HCCOP is preferred ten times as much as each of the other objectives in Fig. 22, the third objectives in Fig. 23, and so on. As can be seen from Figs. 21 to 27, solutions are often con-

[7] Larger values at the i-th element in $\mathbf{u}$ indicate that the i-th objective is preferred higher.

[8] For maximization problems, the normalized $\mathbf{u}$ is directly used as a center vector of Gaussian functions in P-NSGA-II. On the other hand, each element in $\mathbf{u}$ is inversed and the normalized vector is used as the center vector for minimization problems.



(a) 200th generation (b) 400th generation

(c) 600th generation (d) 800th generation

Fig. 18 Solutions obtained by P-NSGA-II for two-objective WBDP. Preference is specified in Table 12.

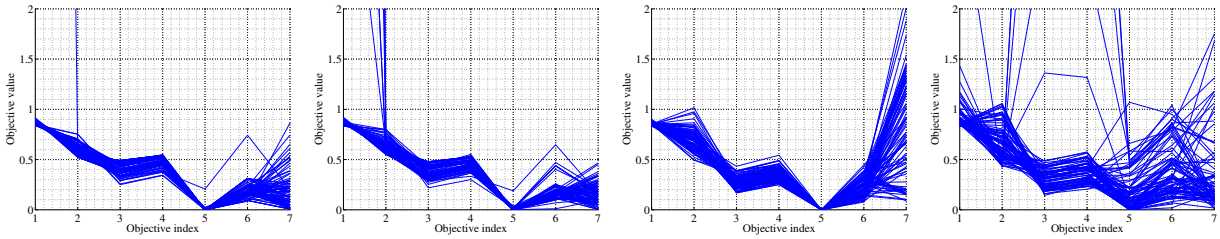
verged to the preferred objective and have a diversity with respect to the objective.

6 Concluding remarks

We demonstrated the performance of P-NSGA-II, where the preference of a DM is represented using Gaussian functions on a hyperplane and the representation is integrated into the framework of NSGA-II. This paper further modified our previous work [33,39] by considering an allowable radius and a shortest distance for each solution with the same rank. Through computational experiments for DTLZ problems, each with 3, 6, 9, and 12 objectives, a two-objective welded beam design problem, and a 7-objective hybrid car controller optimization problem, the performance of P-NSGA-II to focus the search on regions preferred by a DM has been demonstrated. Considering the allowable radius and the shortest distance results in dense solutions around center vectors and sparse solutions away from the center vectors on a hyperplane. The adaptability of P-NSGA-II to the change of preference during optimization has also been examined. The application of P-NSGA-II to more many-objective optimization real-world scenarios is left for future works.

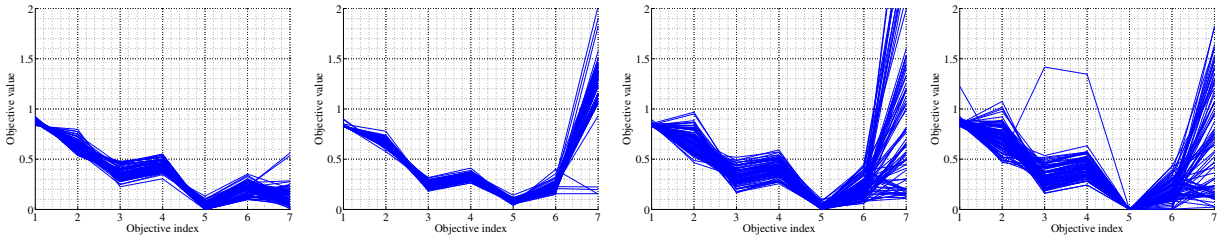
References

1. Allmendinger, R., Knowles, J.: 'Hang on a minute': Investigations on the effects of delayed objective functions in multiobjective optimization. In: R. Purshouse, P. Fleming, C. Fonseca, S. Greco, J. Shaw (eds.) Evolutionary Multi-Criterion Optimization: EMO 2013, *Lecture Notes in Computer Science*, vol. 7811, pp. 6–20. Springer (2013)
2. Auger, A., Bader, J., Brockhoff, D., Zitzler, E.: Articulating user preferences in many-objective problems by sampling the weighted hypervolume. In: Proc. of 2009 Genetic and Evolutionary Computation Conference, pp. 555–562. ACM (2009)
3. Beume, N., Naujoks, B., Emmerich, M.: SMS-EMOA: Multiobjective selection based on dominated hypervolume. European Journal of Operational Research **181**(3), 1653–1669 (2007)
4. Branke, J., Deb, K., Dierolf, H., Osswald, M.: Finding knees in multi-objective optimization. In: Parallel Problem Solving from Nature - PPSN VIII, pp. 722–731. Springer (2004)
5. Branke, J., Deb, K., Miettinen, K., Slowinski, R.: Multiobjective optimization: Interactive and evolutionary approaches. Springer (2008)
6. Censor, Y.: Pareto optimality in multiobjective problems. Applied Mathematics and Optimization **4**(1), 41–59 (1977)



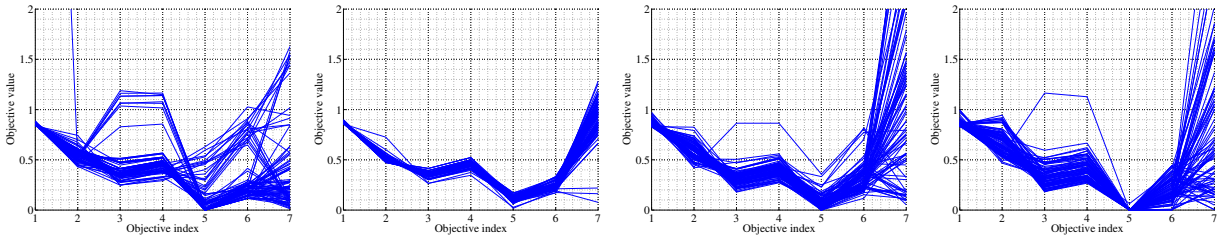
(a) $\sigma = 0.1$. (b) $\sigma = 0.5$. (c) $\sigma = 1.0$. (d) $\sigma = 1.5$.

Fig. 20 Finally obtained solutions by using a center vector $\mathbf{w} = \frac{1}{7}(1, 1, 1, 1, 1, 1, 1)$.



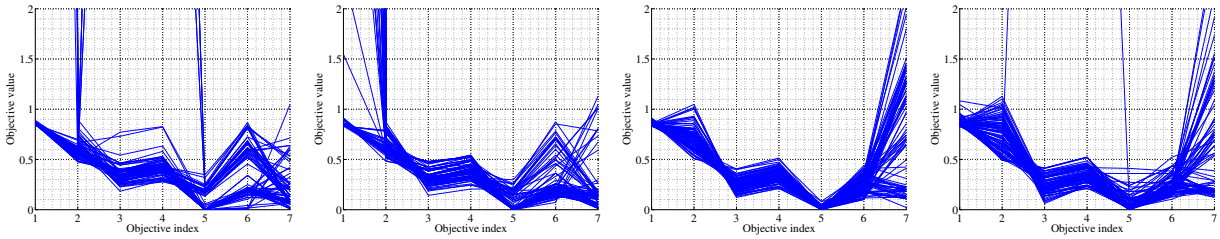
(a) $\sigma = 0.1$. (b) $\sigma = 0.5$. (c) $\sigma = 1.0$. (d) $\sigma = 1.5$.

Fig. 21 Finally obtained solutions by using a center vector $\mathbf{w} = \frac{1}{61}(1, 10, 10, 10, 10, 10, 10)$.



(a) $\sigma = 0.1$. (b) $\sigma = 0.5$. (c) $\sigma = 1.0$. (d) $\sigma = 1.5$.

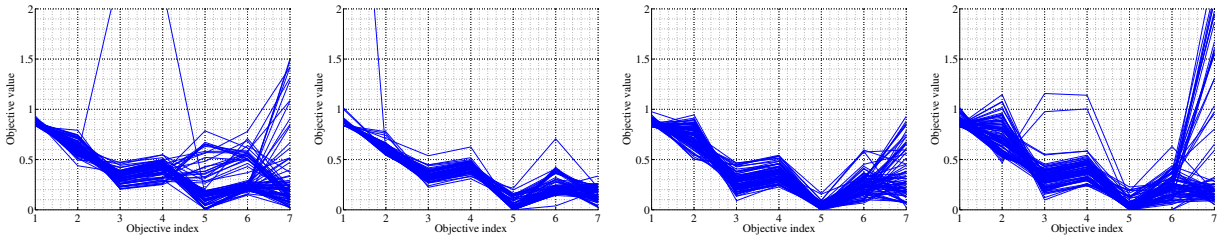
Fig. 22 Finally obtained solutions by using a center vector $\mathbf{w} = \frac{1}{61}(10, 1, 10, 10, 10, 10, 10)$.



(a) $\sigma = 0.1$. (b) $\sigma = 0.5$. (c) $\sigma = 1.0$. (d) $\sigma = 1.5$.

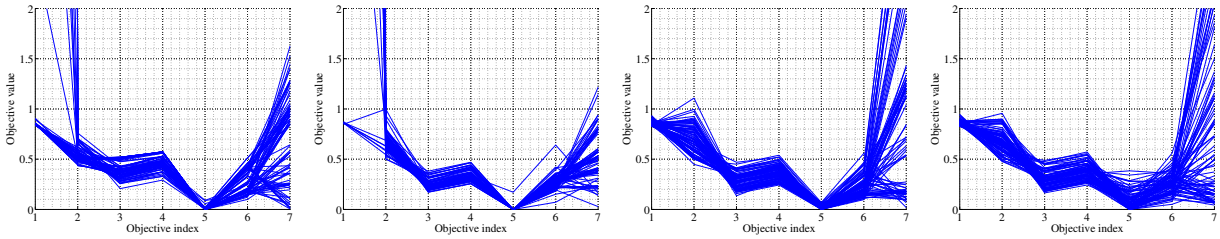
Fig. 23 Finally obtained solutions by using a center vector $\mathbf{w} = \frac{1}{61}(10, 10, 1, 10, 10, 10, 10)$.

7. Chankong, V., Haimes, Y.: Multiobjective decision making: theory and methodology. Dover Publications (1983)
8. Coello, C.: Handling preferences in evolutionary multiobjective optimization: A survey. In: Proc. of 2000 IEEE Congress on Evolutionary Computation, pp. 30–37. IEEE (2000)
9. Coello, C., Lamont, G., Van Veldhuizen, D.: Evolutionary algorithms for solving multi-objective problems. Genetic and evolutionary computation. Springer (2007)
10. Deb, K.: Multi-objective optimization using evolutionary algorithms. Wiley (2001)
11. Deb, K., Jain, H.: An evolutionary many-objective optimization algorithm using reference-point based nondominated sorting approach, part I: solving problems with box constraints. IEEE Trans. on Evolutionary Computation **18**(4), 577–601 (2013)
12. Deb, K., Kumar, A.: Light beam search based multiobjective optimization using evolutionary algorithms. In: Proc. of 2007 IEEE Congress on Evolutionary Computation, pp. 2125–2132. IEEE (2007)
13. Deb, K., Pratap, A., Agarwal, S., Meyarivan, T.: A fast and elitist multiobjective genetic algorithm: NSGA-II. IEEE Trans. on Evolutionary Computation **6**(2), 182–197 (2002)
14. Deb, K., Sundar, J., Udaya Bhaskara Rao, N., Chaudhuri, S.: Reference point based multi-objective optimization using evolutionary algorithms. International Jour-



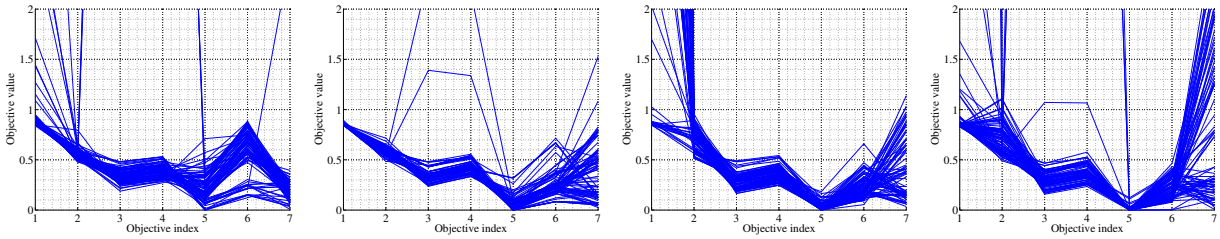
(a) $\sigma = 0.1$. (b) $\sigma = 0.5$. (c) $\sigma = 1.0$. (d) $\sigma = 1.5$.

Fig. 24 Finally obtained solutions by using a center vector $\mathbf{w} = \frac{1}{61}(10, 10, 10, 1, 10, 10, 10)$.



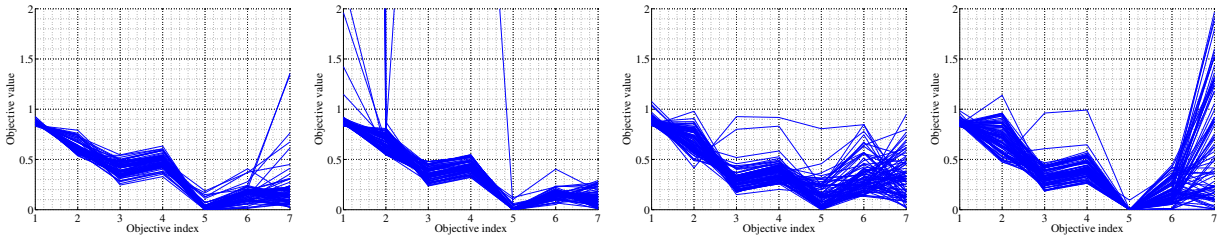
(a) $\sigma = 0.1$. (b) $\sigma = 0.5$. (c) $\sigma = 1.0$. (d) $\sigma = 1.5$.

Fig. 25 Finally obtained solutions by using a center vector $\mathbf{w} = \frac{1}{61}(10, 10, 10, 10, 1, 10, 10)$.



(a) $\sigma = 0.1$. (b) $\sigma = 0.5$. (c) $\sigma = 1.0$. (d) $\sigma = 1.5$.

Fig. 26 Finally obtained solutions by using a center vector $\mathbf{w} = \frac{1}{61}(10, 10, 10, 10, 10, 1, 10)$.

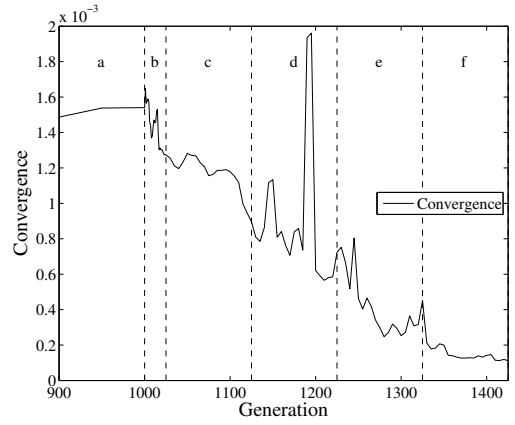


(a) $\sigma = 0.1$. (b) $\sigma = 0.5$. (c) $\sigma = 1.0$. (d) $\sigma = 1.5$.

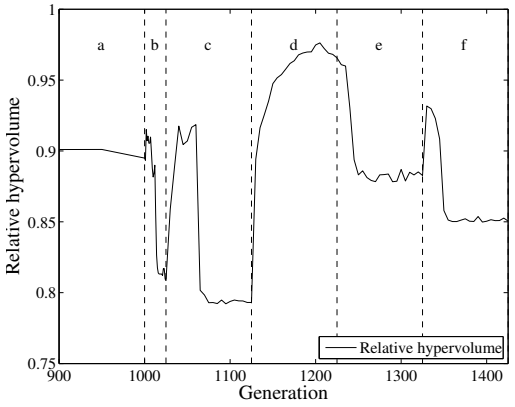
Fig. 27 Finally obtained solutions by using a center vector $\mathbf{w} = \frac{1}{61}(10, 10, 10, 10, 10, 10, 1)$.

nal of Computational Intelligence Research **2**(3), 273–286 (2006)

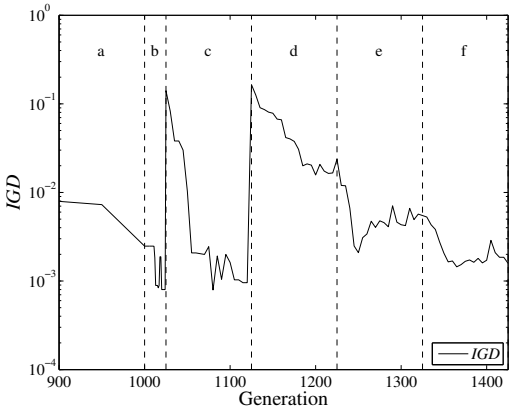
15. Deb, K., Thiele, L., Laumanns, M., Zitzler, E.: Scalable test problems for evolutionary multi-objective optimization. Tech. Rep. 2001001, Kanpur Genetic Algorithms Laboratory (KanGAL), Indian Institute of Technology Kanpur (2001)
16. Deb, K., Thiele, L., Laumanns, M., Zitzler, E.: Scalable multi-objective optimization test problems. In: Proc. of 2002 IEEE Congress on Evolutionary Computation (World Congress on Computational Intelligence), pp. 825–830. IEEE (2002)
17. Deb, K., Thiele, L., Laumanns, M., Zitzler, E.: Scalable test problems for evolutionary multiobjective optimization. In: A. Abraham, L. Jain, R. Goldberg (eds.) Evolutionary Multiobjective Optimization: Theoretical Advances and Applications, Advanced Information and Knowledge Processing, pp. 105–145. Springer (2005)
18. Fishburn, P.: Lexicographic orders, utilities and decision rules: a survey. Management Science **20**(11), 1442–1471 (1974)
19. Haimes, Y., Ladson, L., Wismer, D.: On a bicriterion formulation of the problems of integrated system identification and system optimization. IEEE Trans. on Systems, Man, and Cybernetics **1**(3), 296–297 (1971)



(a) Convergence



(b) Relative hypervolume



(c) *IGD*

Fig. 19 Performance indicators on solutions obtained by P-NSGA-II for three-objective DTLZ1. Preference is specified in Table 11 and the obtained solutions are depicted in Fig. 17

20. Handl, J., Knowles, J.: Evidence accumulation in multiobjective data clustering. In: R. Purshouse, P. Fleming, C. Fonseca, S. Greco, J. Shaw (eds.) Evolutionary Multi-Criterion Optimization: EMO 2013, *Lecture Notes in Computer Science*, vol. 7811, pp. 543–557. Springer (2013)
21. Ishibuchi, H., Narukawa, K.: Comparison of evolutionary multiobjective optimization with reference solution-based single-objective approach. In: Proc. of 2005 Genetic and Evolutionary Computation Conference, pp. 787–794. ACM (2005)
22. Ishibuchi, H., Tsukamoto, N., Nojima, Y.: Evolutionary many-objective optimization: A short review. In: Proc. of 2008 IEEE Congress on Evolutionary Computation (World Congress on Computational Intelligence), pp. 2419–2426. IEEE (2008)
23. Ishibuchi, H., Tsukamoto, N., Sakane, Y., Nojima, Y.: Indicator-based evolutionary algorithm with hypervolume approximation by achievement scalarizing functions. In: Proc. of 2010 Genetic and Evolutionary Computation Conference, pp. 527–534. ACM (2010)
24. Jaszkiewicz, A., Słowiński, R.: The light beam search approach - an overview of methodology and applications. European Journal of Operational Research **113**(2), 300–314 (1999)
25. Knowles, J., Corne, D.: Approximating the nondominated front using the Pareto archived evolution strategy. Evolutionary Computation **8**(2), 149–172 (2000)
26. Korhonen, P., Wallenius, J.: Visualization in the multiple objective decision-making framework. In: Multiobjective optimization, *Lecture Notes in Computer Science*, vol. 5252, pp. 195–212. Springer (2008)
27. Lotov, A., Miettinen, K.: Visualizing the Pareto frontier. In: Multiobjective optimization, *Lecture Notes in Computer Science*, vol. 5252, pp. 213–243. Springer (2008)
28. Miettinen, K.: Nonlinear multiobjective optimization. Springer (1999)
29. Molina, J., Santana, L., Hernández-Díaz, A., Coello, C., Caballero, R.: g-dominance: Reference point based dominance for multiobjective metaheuristics. European Journal of Operational Research **197**(2), 685–692 (2009)
30. Morino, H., Obayashi, S.: Knowledge extraction for structural design of regional jet horizontal tail using multi-objective design exploration (MODE). In: R. Purshouse, P. Fleming, C. Fonseca, S. Greco, J. Shaw (eds.) Evolutionary Multi-Criterion Optimization: EMO 2013, *Lecture Notes in Computer Science*, vol. 7811, pp. 656–668. Springer (2013)
31. Narukawa, K.: Effect of dominance balance in many-objective optimization. In: R. Purshouse, P. Fleming, C. Fonseca, S. Greco, J. Shaw (eds.) Evolutionary Multi-Criterion Optimization: EMO 2013, *Lecture Notes in Computer Science*, vol. 7811, pp. 276–290. Springer (2013)
32. Narukawa, K., Rodemann, T.: Examining the performance of evolutionary many-objective optimization algorithms on a real-world application. In: Proc. of 2012 IEEE International Conference on Genetic and Evolutionary Computing, pp. 316–319. IEEE (2012)
33. Narukawa, K., Tanigaki, Y., Ishibuchi, H.: Evolutionary many-objective optimization using preference on hyperplane. In: Proc. of 2014 Genetic and Evolutionary Computation Conference, pp. 91–92. ACM (2014)
34. Okabe, T., Jin, Y., Sendhoff, B.: A critical survey of performance indices for multi-objective optimisation. In: Proc. of 2003 IEEE Congress on Evolutionary Computation, vol. 2, pp. 878–885. IEEE (2003)
35. Pedro, L., Takahashi, R.: Modeling decision-maker preferences through utility function level sets. In: R. Takahashi, K. Deb, E. Wanner, S. Greco (eds.) Evolutionary Multi-Criterion Optimization: EMO 2011, *Lecture Notes in Computer Science*, vol. 6576, pp. 550–563. Springer (2011)

36. Pedro, L., Takahashi, R.: Decision-maker preference modeling in interactive multiobjective optimization. In: R. Purshouse, P. Fleming, C. Fonseca, S. Greco, J. Shaw (eds.) Evolutionary Multi-Criterion Optimization: EMO 2013, *Lecture Notes in Computer Science*, vol. 7811, pp. 811–824. Springer (2013)
37. Rachmawati, L., Srinivasan, D.: Multiobjective evolutionary algorithm with controllable focus on the knees of the pareto front. IEEE Trans. on Evolutionary Computation **13**(4), 810–824 (2009)
38. Sato, H., Aguirre, H., Tanaka, K.: Controlling dominance area of solutions and its impact on the performance of MOEAs. In: S. Obayashi, K. Deb, C. Poloni, T. Hiroyasu, T. Murata (eds.) Evolutionary Multi-Criterion Optimization: EMO 2007, *Lecture Notes in Computer Science*, vol. 4403, pp. 5–20. Springer (2007)
39. Tanigaki, Y., Narukawa, K., Nojima, Y., Ishibuchi, H.: Preference-based NSGA-II for many-objective knapsack problems. In: Proc. of Joint 7th International Conference on Soft Computing and Intelligent Systems and 15th International Symposium on Advanced Intelligent Systems. IEEE (2014)
40. Wagner, T., Beume, N., Naujoks, B.: Pareto-, aggregation-, and indicator-based methods in many-objective optimization. In: S. Obayashi, K. Deb, C. Poloni, T. Hiroyasu, T. Murata (eds.) Evolutionary Multi-Criterion Optimization: EMO 2007, *Lecture Notes in Computer Science*, vol. 4403, pp. 742–756. Springer (2007)
41. Wang, Y., Yang, Y.: Particle swarm optimization with preference order ranking for multi-objective optimization. Information Sciences **179**(12), 1944–1959 (2009)
42. Wickramasinghe, U., Li, X.: Integrating user preferences with particle swarms for multi-objective optimization. In: Proc. of 2008 Genetic and Evolutionary Computation Conference, pp. 745–752. ACM (2008)
43. Wickramasinghe, U., Li, X.: A distance metric for evolutionary many-objective optimization algorithms using user-preferences. In: AI 2009: Advances in Artificial Intelligence, pp. 443–453. Springer (2009)
44. Wierzbicki, A.: A mathematical basis for satisficing decision making. Mathematical Modelling **3**(5), 391–405 (1982)
45. Woźniak, P.: Preferences in multi-objective evolutionary optimisation of electric motor speed control with hardware in the loop. Applied Soft Computing **11**(1), 49–55 (2011)
46. Zhang, Q., Li, H.: MOEA/D: A multiobjective evolutionary algorithm based on decomposition. IEEE Trans. on Evolutionary Computation **11**(6), 712–731 (2007)
47. Zitzler, E., Künzli, S.: Indicator-based selection in multiobjective search. In: X. Yao, E. Burke, J. Lozano, J. Smith, J. Guervós, J. Bullinaria, J. Rowe, P. Tiño, A. Kabán, H. Schwefel (eds.) Parallel Problem Solving from Nature: PPSN VIII, *Lecture Notes in Computer Science*, vol. 3242, pp. 832–842. Springer (2004)
48. Zitzler, E., Laumanns, M., Thiele, L.: SPEA2: Improving the strength pareto evolutionary algorithm for multiobjective optimization. In: Proc. of Evolutionary Methods for Design, Optimisation and Control with Application to Industrial Problems, pp. 95–100 (2001)
49. Zitzler, E., Thiele, L.: Multiobjective evolutionary algorithms: A comparative case study and the strength Pareto approach. IEEE Trans. on Evolutionary Computation **3**(4), 257–271 (1999)
50. Zitzler, E., Thiele, L., Laumanns, M., Fonseca, C., Fonseca, V.: Performance assessment of multiobjective optimizers: An analysis and review. IEEE Trans. on Evolutionary Computation **7**(2), 117–132 (2003)