This paper has been published in Proceeding of International Conference on Parallel Problem Solving from Nature, 2022.

Direction Vector Selection for R2-based Hypervolume Contribution Approximation

Tianye Shu, Ke Shang(✉), Yang Nan, and Hisao Ishibuchi(✉)

Guangdong Provincial Key Laboratory of Brain-inspired Intelligent Computation, Department of Computer Science and Engineering, Southern University of Science and Technology, Shenzhen, 518055, China
12132356@mail.sustech.edu.cn; kshang@foxmail.com;
nany@mail.sustech.edu.cn; hisao@sustech.edu.cn

Abstract. Recently, an R2-based hypervolume contribution approximation (i.e., R_2^{HVC} indicator) has been proposed and applied to evolutionary multi-objective algorithms and subset selection. The R_2^{HVC} indicator approximates the hypervolume contribution using a set of line segments determined by a direction vector set. Although the R_2^{HVC} indicator is computationally efficient compared with the exact hypervolume contribution calculation, its approximation error is large if an inappropriate direction vector set is used. In this paper, we propose a method to generate a direction vector set for reducing the approximation error of the R_2^{HVC} indicator. The method generates a set of direction vectors by selecting a small direction vector set from a large candidate direction vector set in a greedy manner. Experimental results show that the proposed method outperforms six existing direction vector set generation methods. The direction vector set generated by the proposed method can be further used to improve the performance of hypervolume-based algorithms which rely on the R_2^{HVC} indicator.

Keywords: Evolutionary multi-objective optimization· Hypervolume contribution · Hypervolume contribution approximation.

1 Introduction

In evolutionary multi-objective optimization (EMO), convergence and diversity are two desired properties of a solution set. To address the conflicting nature of the two properties, many indicators are proposed such as hypervolume (HV) [31] [25], generational distance (GD) [28], inverted generational distance (IGD) [5], and R2 [13]. These indicators are used not only for evaluating the performance of evolutionary multi-objective optimization algorithms (EMOAs) but also for designing EMOAs.

Hypervolume is one of the most widely used indicators in EMO since hypervolume is Pareto compliant [30]. Many EMOAs are based on the hypervolume indicator such as SMS-EMOA [1, 11] and FV-EMOA [18]. In these algorithms, hypervolume contribution plays an important role, which is the increment (or

decrement) of the hypervolume of a solution set when a solution is added (or removed). For example, SMS-EMOA discards the solution with the least hypervolume contribution from the population in each generation, so that the hypervolume of the remaining population is maximized. Hypervolume contribution is also crucial in hypervolume subset selection (HSS), which aims to select a subset with the maximum hypervolume from a candidate solution set. Greedy HSS methods [2, 3, 12] usually select (or remove) the solution with the largest (or least) hypervolume contribution iteratively.

One drawback of hypervolume-based algorithms is their expensive computational cost, especially in high-dimensional spaces. This is because the exact calculation of hypervolume and hypervolume contribution is #P-hard [4]. To decrease the computational cost, an R2-based hypervolume contribution approximation method (i.e., R_2^{HVC} indicator) was proposed in [26]. Benefiting from the R_2^{HVC} indicator, an efficient hypervolume-based algorithm R2HCA-EMOA was proposed, which outperforms many state-of-the-art EMOAs on many-objective problems [23]. The R_2^{HVC} indicator was also applied in a greedy approximate HSS algorithm (i.e., GAHSS) whose computational cost is much lower than that of greedy exact HSS algorithms [24].

The basic idea of the R_2^{HVC} indicator is to use different line segments to approximate the hypervolume contribution. Therefore, a set of vectors is needed to determine the directions of these line segments. Nan et al. [21] reported that the performance of the R_2^{HVC} indicator highly depends on the distribution of the used direction vectors, and uniformly distributed direction vectors are not useful for the R_2^{HVC} indicator. However, currently available methods for direction vector set generation are not specially designed for the R_2^{HVC} indicator, and some of them aim to obtain uniformly distributed direction vectors. As a result, these methods are not suitable for the R_2^{HVC} indicator.

In this paper, we propose a direction vector set generation method called the greedy approximation error selection (GAES). Specifically, we formulate the direction vector set generation for the R_2^{HVC} indicator as a subset selection problem. The target is to minimize the approximation error of the R_2^{HVC} indicator, which is defined by the average distance between the ranking of solutions based on the hypervolume contribution and their ranking based on the R_2^{HVC} indicator in a set of training solution sets. The proposed algorithm selects direction vectors one by one from a large candidate direction vector set in a greedy manner. Our experimental results show that the proposed method can achieve the smallest approximation error and the highest correct identification rate among seven direction vector set generation methods. The direction vector set generated by the proposed method can be further used to improve the performance of hypervolume-based algorithms (e.g., GAHSS) which rely on the R_2^{HVC} indicator.

The rest of the paper is organized as follows: In Section 2, we briefly review the hypervolume, the hypervolume contribution, the R2-based hypervolume contribution, six direction vector set generation methods, and subset selection. We propose a direction vector set generation method for the R_2^{HVC} indicator in Sec-

tion 3. The performance of the proposed method is tested in Section 4. Finally, in Section 5, the conclusion is given.

2 Background

2.1 Hypervolume and Hypervolume Contribution

Mathematically, the hypervolume indicator is defined as follows.

Definition 1. *In the objective space R^m with a reference point $\boldsymbol{r} \in R^m$, the hypervolume of a solution set $S \subset R^m$ is defined as*

$$HV(S, \boldsymbol{r}) = \mathcal{L}\left(\bigcup_{\boldsymbol{s} \in S} \{\boldsymbol{a} | \boldsymbol{s} \succeq \boldsymbol{a} \succeq \boldsymbol{r}\}\right), \tag{1}$$

where $\mathcal{L}(\cdot)$ is the Lebesgue measure of a set, and $\boldsymbol{s} \succeq \boldsymbol{a}$ denotes $\boldsymbol{s}$ dominates $\boldsymbol{a}$ (i.e., $s_i \leq a_i \ \forall i \in \{1, 2, .., m\}$ and $s_j < a_j \ \exists j \in \{1, 2, ..., m\}$ in the minimization case).

Based on the definition of the hypervolume, the hypervolume contribution is defined as follows.

Definition 2. *In the objective space R^m with a reference point $\boldsymbol{r} \in R^m$, the hypervolume contribution of a solution $\boldsymbol{s} \in R^m$ to a solution set $S \subset R^m$ is defined as*

$$HVC(\boldsymbol{s}, S, \boldsymbol{r}) = \begin{cases} HV(S, \boldsymbol{r}) - HV(S \setminus \{\boldsymbol{s}\}, \boldsymbol{r}), & \text{if } \boldsymbol{s} \in S, \\ HV(S \cup \{\boldsymbol{s}\}, \boldsymbol{r}) - HV(S, \boldsymbol{r}), & \text{if } \boldsymbol{s} \notin S. \end{cases} \tag{2}$$

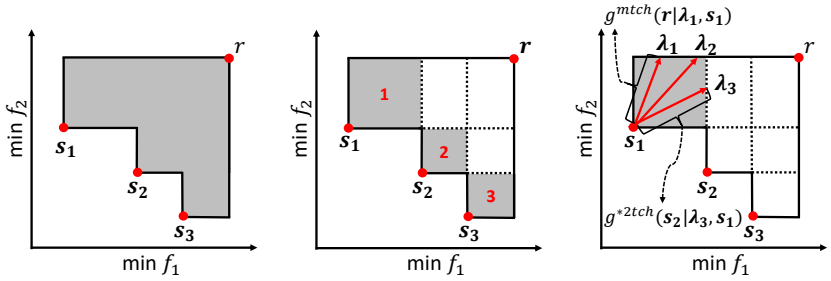
Fig. 1 (a) and (b) illustrate the hypervolume and the hypervolume contribution in the two-objective space.

2.2 R2-based Hypervolume Contribution Approximation

In [26], an R2-based indicator (i.e., R_2^{HVC} indicator) was proposed to approximate the hypervolume contribution. Suppose we have a solution set S in the m-dimensional objective space. To approximate the hypervolume contribution of the solution $\boldsymbol{s}$ to S with the reference point $\boldsymbol{r}$, the lengths of a set of line segments are used. The length L of each line segment is determined by each direction vector $\boldsymbol{\lambda}$ in a given direction vector set V, the solution set $S \setminus \{\boldsymbol{s}\}$ and the reference point $\boldsymbol{r}$. Mathematically,

$$\begin{aligned} R_2^{HVC}(\boldsymbol{s}, S, \boldsymbol{r}, V) &= \frac{1}{|V|} \sum_{\boldsymbol{\lambda} \in V} L(\boldsymbol{s}, S \setminus \{\boldsymbol{s}\}, \boldsymbol{r}, \boldsymbol{\lambda})^m \\ &= \frac{1}{|V|} \sum_{\boldsymbol{\lambda} \in V} min \left\{ \min_{\boldsymbol{s'} \in S \setminus \{\boldsymbol{s}\}} \{g^{*2tch}(\boldsymbol{s'} | \boldsymbol{\lambda}, \boldsymbol{s})\}, g^{mtch}(\boldsymbol{r} | \boldsymbol{\lambda}, \boldsymbol{s}) \right\}^m. \end{aligned} \tag{3}$$

For minimization problems, the $g^{*2tch}(\boldsymbol{s'} | \boldsymbol{\lambda}, \boldsymbol{s})$ function in Eq. (3) is defined as $g^{*2tch}(\boldsymbol{s'} | \boldsymbol{\lambda}, \boldsymbol{s}) = \max\limits_{j \in \{1,2,...,m\}} \left\{ \frac{s_j' - s_j}{\lambda_j} \right\}$, and the $g^{mtch}(\boldsymbol{r} | \boldsymbol{\lambda}, \boldsymbol{s})$ function in Eq. (3) is defined as $g^{mtch}(\boldsymbol{r} | \boldsymbol{\lambda}, \boldsymbol{s}) = \min\limits_{j \in \{1,2,...,m\}} \left\{ \frac{|r_j - s_j|}{\lambda_j} \right\}$. The mechanism of the R2-based hypervolume contribution approximation is illustrated in Fig. 1 (c).



(a) (b) (c)

Fig. 1. Illustration of the hypervolume, the hypervolume contribution and the R2-based hypervolume contribution approximation. The grey area in (a) is the hypervolume of the solution set $\{\boldsymbol{s_1}, \boldsymbol{s_2}, \boldsymbol{s_3}\}$. Each grey area (1, 2 and 3) in (b) is the hypervolume contribution of the corresponding solution ($\boldsymbol{s_1}$, $\boldsymbol{s_2}$ and $\boldsymbol{s_3}$) to the solution set $\{\boldsymbol{s_1}, \boldsymbol{s_2}, \boldsymbol{s_3}\}$, respectively. The red lines in (c) illustrate the mechanism of the R2-based hypervolume contribution approximation.

2.3 Direction Vector Set Generation Methods

Six existing direction vector set generation methods are briefly explained. The first three methods are space filling methods. Every generated weight vector $\boldsymbol{w}$ is normalized to obtain the direction vector $\boldsymbol{\lambda}$ (i.e., $\boldsymbol{\lambda} = \frac{\boldsymbol{w}}{||\boldsymbol{w}||_2}$).

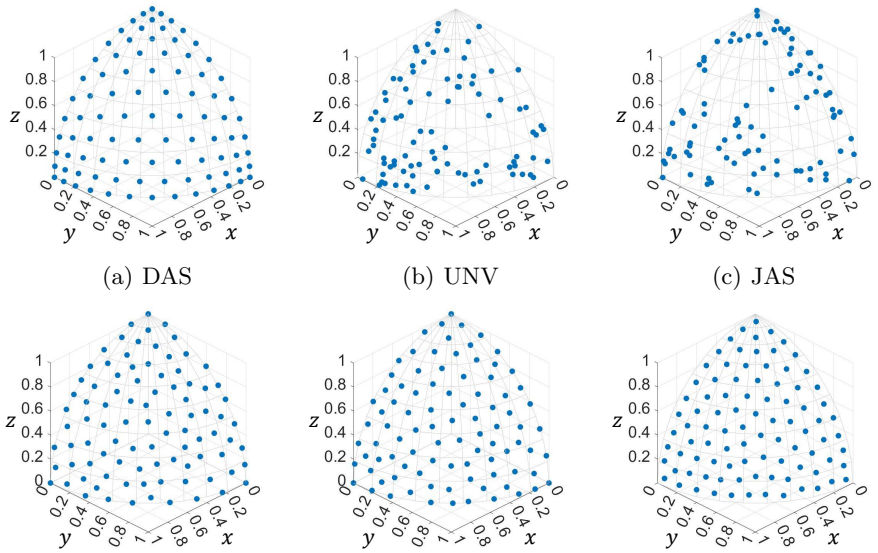
- **Das and Dennis (DAS) method [6]**: In the m-dimensional space, the DAS method generates a weight vector $\boldsymbol{w} = (w_1, w_2, ..., w_m)$ by uniformly dividing each dimension into H parts. The value of w_j is selected from $\{0, \frac{1}{H}, \frac{2}{H}, ..., 1 - \sum_{i=1}^{j-1} w_i\}$. Totally, C_{H+m-1}^{m-1} weight vectors are generated.
- **Unit normal vector (UNV) method [10]**: In the UNV method, a set of weight vectors is randomly sampled from the m-dimensional normal distribution (i.e., $\boldsymbol{w} \sim N_m(0, I_m)$).
- **JAS method [17]**: The JAS method randomly generates a weight vector $\boldsymbol{w} = (w_1, w_2, ..., w_m)$ with the uniform probability distribution [17]. For $k < m$, w_k is sampled by $w_k = (1 - \sum_{j=1}^{k-1} w_j)(1 - \sqrt[m-k]{\mu})$ where μ is randomly sampled from $[0, 1]$. For the last dimension, $w_m = 1 - \sum_{j=1}^{m-1} w_j$.

The other three methods select the desired direction vector set from a large candidate direction vector set generated by one of the first three methods.

- **Maximally spare selection method with DAS (MSS-D [7])**: A large candidate direction vector set U is generated by the DAS method. Then, m extreme direction vectors $(1, 0, ..., 0)$, $(0, 1, ..., 0)$, ..., $(0, 0, ..., 1)$ are selected as the initial direction vector set V. The direction vector $\boldsymbol{\lambda} \in U$ with the largest distance to the vector set V is selected (i.e., moved from U to V). This step repeats until V reaches the desired size.
- **Maximally spare selection method with UNV (MSS-U [7])**: The only difference between MSS-U and MSS-D is that the candidate direction vector set U in MSS-U is generated by UNV instead of DAS.

– **Kmeans-U [15]**: The method starts with a large candidate direction vector set U generated by the UNV method. Then the k-means clustering [20] is used to obtain a direction vector subset V from U.

In Fig. 2, a direction vector set of size 91 is generated by each method, and the size of the candidate direction vector set is 49,770 for the MSS-D, MSS-U and Kmeans-U methods. In Fig. 2, the direction vector sets generated by the DAS, MSS-D, MSS-U and Kmeans-U methods are more uniform than those generated by the UNV and JAS methods.



(a) DAS (b) UNV (c) JAS

(d) MSS-D (e) MSS-U (f) Kmeans-U

Fig. 2. Illustration of the direction vector sets generated by the six methods outlined in Section 2.3. Each direction vector set contains 91 direction vectors.

2.4 Subset Selection

Subset selection is to select a subset from a large candidate set to optimize a given metric [22]. Formally, given a set $U = \{e_1, e_2, ..., e_N\}$, a metric f (to be maximized) and a positive integer k, subset selection aims to find a subset $V \subseteq U$ with $|V| = k$ for maximizing $f(V)$. That is, $V^* = argmax_{V \subseteq U, |V|=k} f(V)$. When the target is to minimize $f(V)$, the problem can be written as $V^* = argmin_{V \subseteq U, |V|=k} f(V)$. To solve the subset selection problem, the greedy inclusion is a simple yet widely used method. For example, hypervolume subset selection (HSS) aims to maximize the hypervolume of the selected solution subset. The greedy inclusion for the HSS [3, 12] selects the solution with the largest hypervolume contribution iteratively.

3 Proposed Method for Selecting Direction Vector Set

Since the performance of the R_2^{HVC} indicator strongly depends on the used direction vector set, we propose a simple greedy inclusion algorithm called the greedy approximation error selection (GAES) for obtaining a high-quality direction vector set for the R_2^{HVC} indicator, and analyze its time complexity.

3.1 Approximation Error

Equipped with a good direction vector set, the R_2^{HVC} indicator is supposed to be consistent with the hypervolume contribution. That is, the approximation error should be small. Usually, we are interested in the ranking of solutions based on the hypervolume contribution values in hypervolume-based algorithms. Therefore, we define the approximation error between the ranking based on the hypervolume contribution and the ranking based on the R_2^{HVC} indicator. The hypervolume contribution is calculated by the WFG algorithm [29], and the R_2^{HVC} indicator is calculated by Eq. (3). Suppose we have a solution set $S_i = \{\boldsymbol{s_1}, \boldsymbol{s_2}, ..., \boldsymbol{s_t}\}$. We denote the ranking based on the hypervolume contribution by $\sigma_H(\boldsymbol{s_1})$, $\sigma_H(\boldsymbol{s_2})$, ..., $\sigma_H(\boldsymbol{s_t})$, where $\sigma_H(\boldsymbol{s_i})$ is the rank of the solution $\boldsymbol{s_i}$ among the t solutions in S_i. In the same manner, we denote the ranking based on the R_2^{HVC} indicator by $\sigma_R(\boldsymbol{s_1})$, $\sigma_R(\boldsymbol{s_2})$, ..., $\sigma_R(\boldsymbol{s_t})$. Spearman's footrule is one of the most well-known distances between rankings [19], which can be described as follows:

$$D(S_i, \sigma_H, \sigma_R) = \sum_{j=1}^{t} |\sigma_H(\boldsymbol{s_j}) - \sigma_R(\boldsymbol{s_j})|. \tag{4}$$

Small approximation error means that the distance between the two rankings is small. Based on the distance in Eq. (4), we can measure the approximation error. For a set of solution sets $S = \{S_1, S_2, ...S_T\}$, the approximation error is defined as follows:

$$AE(S, V, \boldsymbol{r}) = \frac{1}{T} \sum_{i=1}^{T} D(S_i, \sigma_H, \sigma_R), \tag{5}$$

where $\boldsymbol{r}$ is the reference point, σ_H is the ranking of the solutions in S_i based on the hypervolume contribution, and σ_R is the ranking of these solutions based on the R_2^{HVC} indicator with the direction vector set V.

3.2 Problem Formulation

Now we can formulate the problem of generating a good direction vector set for the R_2^{HVC} indicator as a subset selection problem. Given a large candidate direction vector set $U = \{\boldsymbol{\lambda_1}, \boldsymbol{\lambda_2}, ..., \boldsymbol{\lambda_N}\}$, a set of training solution sets S, and a reference point $\boldsymbol{r}$, the problem is to find a direction vector subset $V \subset U$ with $|V| = n$ $(n < N)$ to minimize the approximation error $AE(S, V, \boldsymbol{r})$ in Eq. (5).

3.3 Greedy Inclusion Algorithm

To solve the above problem, a simple greedy inclusion algorithm called the greedy approximation error selection (GAES) is proposed (Algorithm 1). Firstly, a set of training solution sets S should be prepared in advance. Then, a large candidate direction vector set U is generated by some methods. The direction vector set V is empty initially. Iteratively, the direction vector $\boldsymbol{\lambda}^* \in U$ which minimizes $AE(S, V \cup \{\boldsymbol{\lambda}\}, \boldsymbol{r})$ is selected (i.e., moved from U to V) until V reaches the desired size.

We analyze the time complexity of Algorithm 1. The most time-consuming step is to calculate the approximation error AE. Let us consider a single training solution set S_i with size t. In Eq. (4) and (5), we have to obtain two rankings of the solutions in S_i: One is based on the hypervolume contribution and the other is based on the R_2^{HVC} indicator. The hypervolume contribution can be calculated in advance. Therefore, we only need to calculate the R_2^{HVC} indicator with the direction vector set $V \cup \{\boldsymbol{\lambda}\}$ for every $\boldsymbol{\lambda} \in U$. It is worth noting in Eq. (3) that the R_2^{HVC} indicator with a direction vector set V is basically the average length of the line segment determined by the direction vector $\boldsymbol{\lambda}$ for every $\boldsymbol{\lambda} \in V$. Thus, we can calculate the length L for each solution in S_i with each direction vector in U in advance, which requires $O(Nt^2m)$ time. With these lengths, we can update the R_2^{HVC} indicator for each solution in S_i in $O(t)$ time. Sorting these solutions requires $O(t \log t)$ time, which is performed for each $\boldsymbol{\lambda}$ in U in each iteration in Algorithm 1. The total time complexity is $O(T(nNt \log t + Nt^2m))$.

Algorithm 1 Greedy Approximation Error Selection

Input: S (a set of training solution sets), N (size of a candidate direction vector set), n (size of a desired direction vector set), $\boldsymbol{r}$ (reference point)
Output: V (desired direction vector set)
Generate a candidate direction vector set U of size N by some methods.
$V \leftarrow \emptyset$
while $|V| < n$
  $\boldsymbol{\lambda}^* = \underset{\boldsymbol{\lambda} \in U}{argmin}\ AE(S, V \cup \{\boldsymbol{\lambda}\}, \boldsymbol{r})$
  Move $\boldsymbol{\lambda}^*$ from U to V
end while

4 Experiments and Discussions

4.1 Direction Vector Selection

Experimental Settings. The first experiment is to illustrate the direction vector selection process of the GAES algorithm (Algorithm 1). We generate 100 training solution sets of size 100, and the hypervolume contribution of each solution in each training solution set is calculated in advance. More specifically, to generate each training solution set S_i with size 100, we first determine the shape (triangular or inverted triangular) and the curvature (linear, convex or concave)

of the Pareto front. Then, 100 solutions in S_i are randomly sampled from this Pareto front. The triangular Pareto front $\sum_{i=1}^{m} f_i^p = 1, f_i \geq 0$ for $i = 1, 2, ..., m$ is used in the first 50 training solution sets. The remaining 50 training solution sets use the inverted triangular Pareto front $\sum_{i=1}^{m}(1 - f_i)^p = 1, 0 \leq f_i \leq 1$ for $i = 1, 2, ..., m$. The p value in the two formulas controls the curvature. To make the curvature more diverse, the p value is determined by $p = 2^x$ where x is uniformly sampled from $[-1, 1]$ (i.e., the range of p value is $[0.5, 2]$). The candidate direction vector set of size 10,000 is generated by the UNV method in Section 2.3. The size of the desired direction vector set is set as 91, 105 and 120 for 3, 5 and 8-objective cases, respectively. The reference point $\boldsymbol{r}$ is set as (1.2, 1.2, ..., 1.2). The proposed method with UNV is denoted as GAES-U in our experiments.

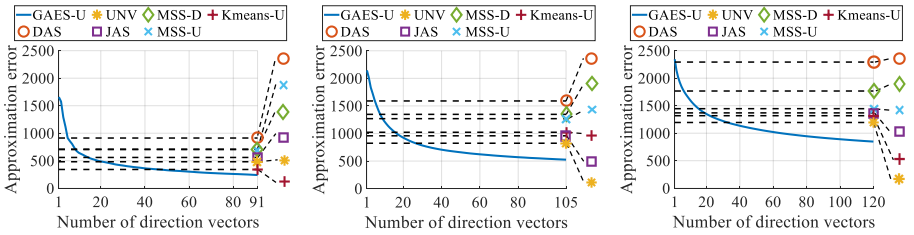
The six direction vector set generation methods in Section 2.3 are used as the baselines. For the MSS-U, MSS-D, and Kmeans-U methods, the size of the candidate direction vector set is set as 49,770, 46,376 and 31,824 for 3, 5 and 8-objective cases, respectively. For each of the six methods and the GAES-U method, 21 direction vector sets are generated from 21 independent runs. We conduct the experiments on a virtual machine equipped with two ADM EPYC 7702 64-Core CUP@2.4GHz, 256GB RAM and Ubuntu Operating System. All codes are implemented in MATLAB R2021b and available from `https://github.com/HisaoLabSUSTC/GAES`.

Results and Discussions. The performance of the selected direction vectors by GEAS-U is shown in Fig. 3 at each iteration (i.e., after selecting a single direction vector, two direction vectors, ..., n direction vectors). The blue curve shows that the approximation error of the R_2^{HVC} indicator on the training solution sets decreases monotonically as more direction vectors are selected by the GAES-U method. The GAES-U method (i.e., the rightmost point of the blue curve) has a better approximation error than the other six methods. With the same number of direction vectors, the approximation error by each method increases as the number of objectives increases. This is because more direction vectors are needed for the R_2^{HVC} indicator to approximate the hypervolume contribution precisely in a higher-dimensional space. The advantage of the GAES-U method is clear in the 8-objective case. Only one direction vector selected by the GAES-U method (i.e., the leftmost point of the blue curve) has a similar approximation error as 120 direction vectors generated by the DAS method (i.e., the top dash line). The 40 direction vectors selected by the GAES-U method have a better approximation error than 120 direction vectors generated by the other methods.

The direction vector sets generated by the GAES-U method are shown in Fig. 4. In the 3-objective case, the direction vector set generated by the GAES-U method in Fig. 4 (a) is less uniform than those generated by the DAS, MSS-D, MSS-U and Kmean-U methods in Fig. 2 (a), (d), (e) and (f), and is more uniform than those generated by the UNV and JAS methods in Fig. 2 (b) and (c).

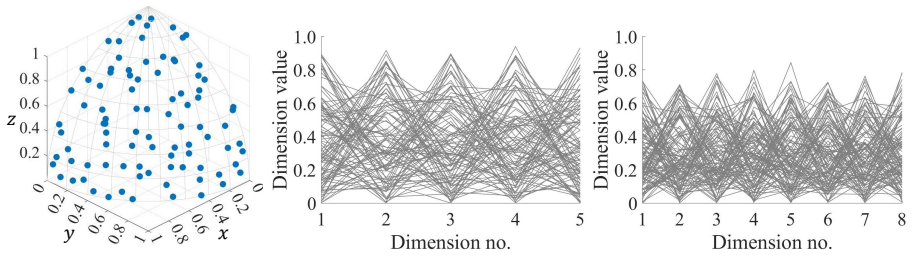
Fig. 5 (a) shows the computational time for the training solution sets generation including the hypervolume contribution calculation, which increases severely as the number of the objectives increases. However, this part only needs to be

performed once, and the generated training solution sets and the calculated hypervolume contribution can be used for multiple runs of the GAES-U method. The runtime of the GAES-U method increases slightly as the number of the objectives increases as shown in Fig. 5 (b).



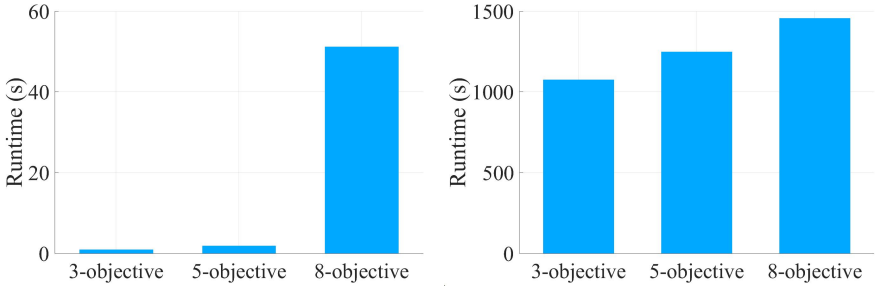
(a) 3-objective (b) 5-objective (c) 8-objective

Fig. 3. Approximation errors of the R_2^{HVC} indicator with different direction vector set generation methods on the training solution sets (average results over 21 runs).



(a) 3-objective (b) 5-objective (c) 8-objective

Fig. 4. The direction vector sets generated by the GAES-U method.



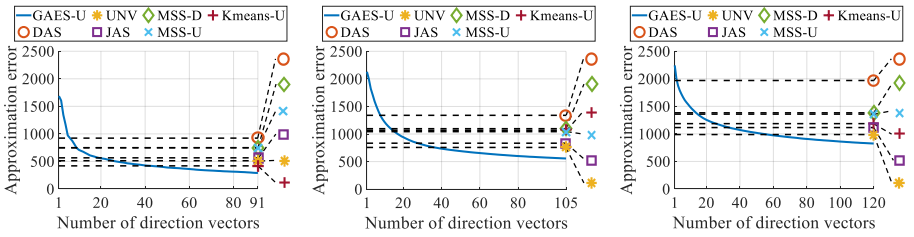
(a) (b)

Fig. 5. Runtime (a) for generating the training solution sets and pre-calculating the hypervolume contribution of each solution in each training solution set. Runtime (b) for selecting direction vectors by the GAES-U method averaged over 21 runs.

4.2 Test on Six Regular Pareto Fronts

Experimental Settings. In the previous experiment, we have obtained different direction vector sets. Then, in this experiment, we compare the performance of the R_2^{HVC} indicator with these direction vector sets on testing solution sets. To generate testing solution sets, six regular Pareto fronts are considered, which are linear triangular, concave triangular and convex triangular Pareto fronts and their corresponding inverted Pareto fronts. For each type of the front, 100 testing solution sets of size 100 are randomly sampled from the front. Firstly, we calculate the approximation error defined in Eq. (4) on the testing solution sets using different direction vector sets. Then, the correct identification rate (CIR) of the R_2^{HVC} indicator with different direction vector set generation methods is calculated. For a testing solution set, the correct identification means that the solution with the least hypervolume contribution is correctly identified by the R_2^{HVC} indicator, and the CIR implies how many testing solution sets are correctly handled.

Results and Discussions. Fig. 6 shows the approximation error of the R_2^{HVC} indicator with different direction vector set generation methods on the testing solution sets. The GAES-U method (i.e., the rightmost point of the blue curve in each figure in Fig. 6) has the smallest approximation error on the testing solution sets. This observation shows the good generalization ability of the GAES-U method. The CIR of the R_2^{HVC} indicator is shown in Table 1. The best CIR (i.e., 58.5%) is obtained by the GAES-U method in Table 1. The UNV and JAS method obtain 51.1% and 50.4% CIR, respectively. The worst one is the Kmeans-U method with the CIR of 33.7%. From these CIR results of the R_2^{HVC} indicator, we can see that the proposed GAES-U method is clearly better than the other six methods.



(a) 3-objective (b) 5-objective (c) 8-objective

Fig. 6. Approximation errors of the R_2^{HVC} indicator with different direction vector set generation methods on the testing solution sets (average results over 21 runs).

4.3 Application: GAHSS

Experimental Settings. The direction vector sets generated by the GAES-U method are tested by the greedy approximate hypervolume subset selection

Table 1. Correct identification rate (CIR) of the R_2^{HVC} indicator with different direction vector set generation methods on different solution sets. The rank of each method is in the parenthesis, and a small value means a better rank.

Solution Set		GAES-U	DAS	UNV	JAS	MSS-D	MSS-U	Kmeans-U
Linear Triangular	3	**76.4%(1)**	59.0%(7)	66.8%(4)	67.0%(3)	68.0%(2)	66.2%(5)	63.4%(6)
	5	**55.4%(1)**	15.0%(7)	46.5%(2)	46.0%(3)	17.0%(6)	18.7%(5)	27.6%(4)
	8	29.6%(5)	23.0%(6)	36.4%(2)	34.9%(3)	**42.0%(1)**	30.0%(4)	14.3%(7)
Linear Inverted Triangular	3	**73.4%(1)**	53.0%(6)	64.0%(3)	63.2%(5)	67.0%(2)	63.5%(4)	50.0%(7)
	5	**62.1%(1)**	14.0%(7)	51.8%(2)	49.2%(3)	27.0%(6)	30.9%(4)	28.6%(5)
	8	**51.0%(1)**	0.0%(6.5)	37.0%(2)	24.4%(4)	0.0%(6.5)	6.9%(5)	28.1%(3)
Concave Triangular	3	**67.4%(1)**	32.0%(7)	57.0%(3)	58.8%(2)	38.0%(5)	36.8%(6)	49.4%(4)
	5	**61.4%(1)**	16.0%(6)	50.7%(2)	49.8%(3)	21.0%(5)	26.7%(4)	13.3%(7)
	8	**65.9%(1)**	53.0%(6)	64.8%(3)	61.7%(4)	56.0%(5)	65.3%(2)	42.3%(7)
Concave Inverted Triangular	3	**73.8%(1)**	41.0%(7)	59.6%(3)	60.7%(2)	50.0%(5)	50.7%(4)	46.9%(6)
	5	40.1%(3)	26.0%(4)	**42.7%(1)**	42.1%(2)	23.0%(5)	15.9%(6)	12.1%(7)
	8	**55.5%(1)**	0.0%(7)	38.6%(3)	36.4%(4)	10.0%(6)	43.9%(2)	33.0%(5)
Convex Triangular	3	**66.5%(1)**	55.0%(6)	55.1%(5)	57.2%(4)	60.0%(3)	60.3%(2)	36.0%(7)
	5	26.2%(2)	17.0%(6)	25.0%(3)	**30.4%(1)**	19.0%(4)	18.8%(5)	4.5%(7)
	8	7.0%(5)	**33.0%(1)**	14.2%(4)	18.8%(3)	30.0%(2)	4.0%(6)	1.3%(7)
Convex Inverted Triangular	3	**65.5%(1)**	32.0%(6)	50.0%(2)	49.5%(3)	38.0%(4)	35.1%(5)	27.9%(7)
	5	88.5%(2)	**90.0%(1)**	76.6%(5)	79.3%(4)	66.0%(6)	80.8%(3)	42.6%(7)
	8	87.7%(2)	**100.0%(1)**	83.1%(4)	77.6%(6)	61.0%(7)	81.7%(5)	84.8%(3)
Avg. Rank		**1.72**	5.42	2.94	3.28	4.47	4.28	5.89
Avg. CIR		**58.5**%	36.6%	51.1%	50.4%	38.5%	40.9%	33.7%

(GAHSS) algorithm [24]. The difference between the GAHSS algorithm and the greedy HSS algorithm mentioned in Section 2.4 is that the R_2^{HVC} indicator is used to approximate the hypervolume contribution in GAHSS. A part of the subset selection benchmark test suite proposed in [27] is used to test the performance of the GAHSS algorithm equipped with the direction vector sets generated by the GAES-U method and the other six methods. Specifically, the candidate solution sets consist of the nondominated solutions after 100,000 function evaluations when NSGA-III [8] is run on DTLZ1 [9], DTLZ2 [9], Minus-DTLZ1 [16], Minus-DTLZ2 [16], DTLZ7 [9] and WFG3 [14] problems for 3, 5 and 8-objective cases. Thus, 18 candidate solution sets are used. We select 91, 210 and 156 solutions from the candidate solution sets for 3, 5 and 8-objective cases, respectively. The hypervolume of the selected solution subset is used as the performance metric. The reference point is set as 1.2 times the nadir point of the true Pareto front for each candidate solution set. GAHSS is performed 21 runs with each direction vector set generation method, and the Wilcoxon rank sum test is used to compare the hypervolume performance.

Results and Discussions. Table 2 shows the hypervolume of the solution subset selected by the GAHSS algorithm with different direction vector set generation methods. The best result is obtained by the proposed GAES-U method. One interesting observation is that the Kmeans-U method, which has poor performance in the CIR experiment in Table 1, shows competitive performance in the GAHSS experiment in Table 2. Future examinations on this interesting observation is needed.

Table 2. Hypervolume of the solution subset selected by GAHSS with different direction vector set generation methods on different candidate solution sets. The rank of each method is in the parenthesis, and a small value means a better rank. The Wilcoxon rank sum test is used to compare the performance. The symbols "+", "-" and "≈" mean the GAES-U method "is significantly better than", "is significantly worse than" and "has no significant difference with" the corresponding method, respectively.

Candidate Solution Set		GAES-U	DAS	UNV	JAS	MSS-D	MSS-U	Kmeans-U
DTLZ1	3	**1.90E-1(1)**	1.89E-1(7,+)	1.90E-1(4,+)	1.89E-1(6,+)	1.90E-1(3,+)	1.90E-1(5,+)	1.90E-1(2,≈)
	5	**7.68E-2(1)**	7.56E-2(5,+)	7.67E-2(3,+)	7.67E-2(4,+)	7.54E-2(6,+)	7.53E-2(7,+)	7.67E-2(2,+)
	8	**1.67E-2(1)**	1.20E-2(7,+)	1.67E-2(2,+)	1.67E-2(4,+)	1.46E-2(6,+)	1.67E-2(5,+)	1.67E-2(3,+)
DTLZ2	3	**1.15E+0(1)**	9.30E-1(7,+)	1.15E+0(3,+)	1.15E+0(4,+)	1.01E+0(6,+)	1.05E+0(5,+)	1.15E+0(2,≈)
	5	**2.21E+0(1)**	1.59E+0(7,+)	2.20E+0(3,+)	2.20E+0(4,+)	1.62E+0(6,+)	2.09E+0(5,+)	2.21E+0(2,≈)
	8	4.16E+0(2)	2.21E+0(6,+)	4.15E+0(3,+)	4.14E+0(4,+)	2.08E+0(7,+)	4.12E+0(5,+)	**4.16E+0(1,-)**
Minus-DTLZ1	3	8.87E+7(2)	8.74E+7(7,+)	8.85E+7(3,+)	8.84E+7(5,+)	8.81E+7(6,+)	8.84E+7(4,+)	**8.88E+7(1,-)**
	5	4.48E+12(2)	4.07E+12(7,+)	4.46E+12(3,+)	4.40E+12(4,+)	4.11E+12(6,+)	4.23E+12(5,+)	**4.48E+12(1,≈)**
	8	**1.36E+19(1)**	8.23E+18(7,+)	1.27E+19(3,+)	1.11E+19(4,+)	8.41E+18(6,+)	9.05E+18(5,+)	1.35E+19(2,+)
Minus-DTLZ2	3	4.47E+1(2)	4.00E+1(7,+)	4.45E+1(3,+)	4.44E+1(4,+)	4.31E+1(5,+)	4.31E+1(6,+)	**4.47E+1(1,≈)**
	5	**2.51E+2(1)**	2.31E+2(7,+)	2.47E+2(3,+)	2.45E+2(4,+)	2.31E+2(6,+)	2.40E+2(5,+)	2.50E+2(2,+)
	8	**1.15E+3(1)**	6.55E+2(7,+)	1.10E+3(3,+)	1.08E+3(4,+)	8.17E+2(6,+)	1.04E+3(5,+)	1.14E+3(2,+)
DTLZ7	3	2.81E+0(2)	2.54E+0(7,+)	2.80E+0(3,+)	2.79E+0(4,+)	2.72E+0(6,+)	2.73E+0(5,+)	**2.81E+0(1,≈)**
	5	**5.08E+0(1)**	4.56E+0(7,+)	5.04E+0(3,+)	5.03E+0(4,+)	4.79E+0(6,+)	4.87E+0(5,+)	5.07E+0(2,+)
	8	**7.56E+0(1)**	6.05E+0(6,+)	7.40E+0(5,+)	7.41E+0(3,+)	5.50E+0(7,+)	7.40E+0(4,+)	7.55E+0(2,≈)
WFG3	3	3.85E+1(2)	3.55E+1(5,+)	3.84E+1(3,≈)	3.83E+1(4,≈)	3.55E+1(6,+)	3.54E+1(7,+)	**3.85E+1(1,-)**
	5	**1.47E+4(1)**	1.37E+4(7,+)	1.46E+4(3,+)	1.47E+4(2,+)	1.37E+4(6,+)	1.46E+4(4,+)	1.46E+4(5,+)
	8	1.01E+8(3)	6.42E+7(7,+)	1.01E+8(4,≈)	**1.01E+8(1,-)**	9.09E+7(6,+)	1.01E+8(2,-)	9.90E+7(5,+)
Avg. Rank		**1.44**	6.67	3.17	3.83	5.89	4.94	2.06
+/-/≈			18/0/0	16/0/2	16/1/1	18/0/0	17/1/0	8/3/7

5 Conclusion

In this paper, we formulated the problem of generating a good direction vector set for the R_2^{HVC} indicator as a subset selection problem to minimize the approximation error. A greedy inclusion method called the greedy approximation error selection (GAES) was proposed to solve this problem. Experimental results showed that the GAES method outperforms other available methods for direction vector set generation for the R_2^{HVC} indicator. The direction vector set generated by the GAES method was applied to the greedy approximate hypervolume subset selection, and good performance was demonstrated in comparison with the other direction vector set generation methods. One future research topic is to examine the performance of the GAES method in hypervolme-based evolutionary multi-objective algorithms.

Acknowledgements. This work was supported by National Natural Science Foundation of China (Grant No. 62002152, 61876075), Guangdong Provincial Key Laboratory (Grant No. 2020B121201001), the Program for Guangdong Introducing Innovative and Enterpreneurial Teams (Grant No. 2017ZT07X386), The Stable Support Plan Program of Shenzhen Natural Science Fund (Grant No. 20200925174447003), Shenzhen Science and Technology Program (Grant No. KQTD2016112514355531).

References

1. Beume, N., Naujoks, B., Emmerich, M.: SMS-EMOA: Multiobjective selection based on dominated hypervolume. European Journal of Operational Research **181**(3), 1653–1669 (2007)
2. Bradstreet, L., Barone, L., While, L.: Maximising hypervolume for selection in multi-objective evolutionary algorithms. In: Proceedings of IEEE Congress on Evolutionary Computation (CEC). pp. 1744–1751 (2006)
3. Bradstreet, L., While, L., Barone, L.: Incrementally maximising hypervolume for selection in multi-objective evolutionary algorithms. In: Proceedings of IEEE Congress on Evolutionary Computation (CEC). pp. 3203–3210 (2007)
4. Bringmann, K., Friedrich, T.: Approximating the volume of unions and intersections of high-dimensional geometric objects. Computational Geometry: Theory and Applications **43**(6), 601–610 (2010)
5. Coello, C.A.C., Sierra, M.R.: A study of the parallelization of a coevolutionary multi-objective evolutionary algorithm. In: Mexican International Conference on Artificial Intelligence. pp. 688–697. Springer (2004)
6. Das, I., Dennis, J.E.: Normal-boundary intersection: A new method for generating the Pareto surface in nonlinear multicriteria optimization problems. SIAM Journal on Optimization **8**(3), 631–657 (1998)
7. Deb, K., Bandaru, S., Seada, H.: Generating uniformly distributed points on a unit simplex for evolutionary many-objective optimization. In: Proceedings of International Conference on Evolutionary Multi-Criterion Optimization. pp. 179–190 (2019)
8. 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 Transactions on Evolutionary Computation **18**(4), 577–601 (2014)
9. Deb, K., Thiele, L., Laumanns, M., Zitzler, E.: Scalable test problems for evolutionary multiobjective optimization. In: Evolutionary Multiobjective Optimization, pp. 105–145. Springer (2005)
10. Deng, J., Zhang, Q.: Approximating hypervolume and hypervolume contributions using polar coordinate. IEEE Transactions on Evolutionary Computation **23**(5), 913–918 (2019)
11. Emmerich, M., Beume, N., Naujoks, B.: An EMO algorithm using the hypervolume measure as selection criterion. In: Proceeding of International Conference on Evolutionary Multi-Criterion Optimization. pp. 62–76. Springer (2005)
12. Guerreiro, A.P., Fonseca, C.M., Paquete, L.: Greedy hypervolume subset selection in low dimensions. Evolutionary Computation **24**(3), 521–544 (2016)
13. Hansen, M.P., Jaszkiewicz, A.: Evaluating the quality of approximations to the non-dominated set. IMM Technical Report, Institute of Mathematical Modelling, Technical University of Denmark (1998)
14. Huband, S., Hingston, P., Barone, L., While, L.: A review of multiobjective test problems and a scalable test problem toolkit. IEEE Transactions on Evolutionary Computation **10**(5), 477–506 (2006)
15. Ishibuchi, H., Imada, R., Setoguchi, Y., Nojima, Y.: Reference point specification in inverted generational distance for triangular linear Pareto front. IEEE Transactions on Evolutionary Computation **22**(6), 961–975 (2018)
16. Ishibuchi, H., Setoguchi, Y., Masuda, H., Nojima, Y.: Performance of decomposition-based many-objective algorithms strongly depends on Pareto front shapes. IEEE Transactions on Evolutionary Computation **21**(2), 169–190 (2017)

17. Jaszkiewicz, A.: On the performance of multiple-objective genetic local search on the 0/1 knapsack problem - a comparative experiment. IEEE Transactions on Evolutionary Computation **6**(4), 402–412 (2002)
18. Jiang, S., Zhang, J., Ong, Y.S., Zhang, A.N., Tan, P.S.: A simple and fast hypervolume indicator-based multiobjective evolutionary algorithm. IEEE Transactions on Cybernetics **45**(10), 2202–2213 (2015)
19. Kumar, R., Vassilvitskii, S.: Generalized distances between rankings. In: Proceedings of the 19th International Conference on World Wide Web. pp. 571–580 (2010)
20. MacQueen, J., et al.: Some methods for classification and analysis of multivariate observations. In: Proceedings of the Fifth Berkeley Symposium on Mathematical Statistics and Probability. vol. 1, pp. 281–297 (1967)
21. Nan, Y., Shang, K., Ishibuchi, H.: What is a good direction vector set for the R2-based hypervolume contribution approximation. In: Proceedings of the Genetic and Evolutionary Computation Conference (GECCO). p. 524–532 (2020)
22. Qian, C.: Distributed Pareto optimization for large-scale noisy subset selection. IEEE Transactions on Evolutionary Computation **24**(4), 694–707 (2020)
23. Shang, K., Ishibuchi, H.: A new hypervolume-based evolutionary algorithm for many-objective optimization. IEEE Transactions on Evolutionary Computation **24**(5), 839–852 (2020)
24. Shang, K., Ishibuchi, H., Chen, W.: Greedy approximated hypervolume subset selection for many-objective optimization. In: Proceedings of the Genetic and Evolutionary Computation Conference (GECCO). pp. 448–456 (2021)
25. Shang, K., Ishibuchi, H., He, L., Pang, L.M.: A survey on the hypervolume indicator in evolutionary multiobjective optimization. IEEE Transactions on Evolutionary Computation **25**(1), 1–20 (2021)
26. Shang, K., Ishibuchi, H., Ni, X.: R2-based hypervolume contribution approximation. IEEE Transactions on Evolutionary Computation **24**(1), 185–192 (2020)
27. Shang, K., Shu, T., Ishibuchi, H., Nan, Y., Pang, L.M.: Benchmarking subset selection from large candidate solution sets in evolutionary multi-objective optimization. arXiv preprint arXiv:2201.06700 (2022)
28. Van Veldhuizen, D.A.: Multiobjective evolutionary algorithms: Classifications, analyses, and new innovations. Ph.D. Dissertation, Air Force Institute of Technology (1999)
29. While, L., Bradstreet, L., Barone, L.: A fast way of calculating exact hypervolumes. IEEE Transactions on Evolutionary Computation **16**(1), 86–95 (2012)
30. Zitzler, E., Brockhoff, D., Thiele, L.: The hypervolume indicator revisited: On the design of Pareto-compliant indicators via weighted integration. In: Proceeding of International Conference on Evolutionary Multi-Criterion Optimization. pp. 862–876. Springer (2007)
31. Zitzler, E., Thiele, L., Laumanns, M., Fonseca, C.M., Da Fonseca, V.G.: Performance assessment of multiobjective optimizers: An analysis and review. IEEE Transactions on Evolutionary Computation **7**(2), 117–132 (2003)