# Meta-feature Extraction for Multi-objective Optimization Problems

Xianghua Chu[1,2], Jiayun Wang[1], Shuxiang Li[1], Linya Huang[1], Qiu He[3],
Guodan Bao[4], and Wei Zhao[1](✉)

[1] College of Management, Shenzhen University, Shenzhen, China
x.chu@szu.edu.cn
[2] Institute of Big Data Intelligent Management and Decision,
Shenzhen University, Shenzhen, China
[3] d'Overbroeck's, Oxford OX2 7PL, UK
helen0411@yeah.net
[4] Bethany School, Curtisden Green, Cranbrook TN17 1LB, UK

**Abstract.** Selecting the appropriate meta-features to represent the optimization problems was studied previously. However, the research on the extraction of meta-features for multi-objective problems is lacking. In this paper, a set of meta-features including a unique meta-feature based on Pareto front shape and the combination of meta-features are proposed for the multi-objective optimization problems (MOPs). 25 multi-objective benchmark functions and K-NN algorithm are adopted to realize the algorithm recommendation for MOPs. Experimental results show that the meta-features based on Pareto front can properly represent multi-objective problems and obtain better recommendation performance. The algorithm recommendation accuracy is improved once the combination of meta-features is considered.

**Keywords:** Meta-feature · Meta-learning · Multi-objective optimization problems · Evolutionary algorithms

## 1 Introduction

Multi-objective optimization problems (MOPs) refer to the problems with several objectives to be optimized simultaneously, which widely exist in many fields such as mathematics, physics, engineering and business [1]. Typically, an MOP [2,3] can be modeled as follows:

$$minF(x) = (f_1(x), f_2(x), f_3(x), \cdots, f_n(x)) \tag{1}$$

$$subject\ to : x \in \Omega \tag{2}$$

where $x$ is a decision vector, $\Omega$ refers to the feasible search region, $R^n$ is the objective space, $F(x) : \Omega \rightarrow R^n$ is an m-dimensional objective vector. Since the optimization of one objective often leads to the deterioration of at least one other objective, the optimal solution set $x^*$ is a set of tradeoff solutions

H. Zhang et al. (Eds.): NCAA 2021, CCIS 1449, pp. 432–445, 2021.
https://doi.org/10.1007/978-981-16-5188-5_31

called Pareto optimal solution, where the set of $F(x^*)$ is Pareto front (PF) [4]. Usually, decision makers require an approximation to the PF, so they can select a final solution from the solution set according to her/his preference. Therefore, a number of advanced algorithms have been developed for finding a set of solutions to approximate the PF in a single run. Evolutionary algorithm (EA) is a random search algorithm which simulates biological natural selection and evolution. EA has been proven to be helpful for MOP [5], as they process a set of solutions in parallel, eventually exploiting similarities of solutions by crossover [6]. Various multi-objective evolutionary algorithms (MOEAs) based on different evolutionary mechanisms were constantly designed and applied to solve MOPs successfully, such as decomposition-based MOEA/D [7], NSGA-III [8], domination-based NSGA-II [9], SPEA2 [10] and index-based IBEA [11], EMOEA.

It is known that MOPs have different characteristics [12], while MOEAs perform different search biases [13]. For example, the domination-based NSGA-II, SPEA2 have effective performance when dealing with MOPs with two or three objectives, but the efficiency decreases significantly in tackling many-objective optimization problems (MaOPs). For a complex new MOP instance, one approach is to train the meta-model using a meta-learning (ML) [14] algorithm and adaptively recommend appropriate algorithms for the problem. However, the recommendation process requires prior knowledge about the problem's characteristics and corresponding algorithm performance, which have huge impact on the accuracy of algorithmic recommendation system.

Although many ML literatures have proposed and proved the effectiveness of meta-features [13] for optimization problems, such as meta-features, statistical meta-features and information theoretic meta-features [15,16], no literatures have proposed a set of effective meta-features for MOPs to help algorithm recommendation. Therefore, in this paper, we propose a new set of meta-features for MOPs, which is consist of two components. One is a unique meta-feature based on the shape and properties of the Pareto front, the other is a meta-feature combination based on the target space including common meta-features from statistical features, and geometric measurement features. The proposed meta-feature set is used for the recommendation algorithm for MOPs. Finally, we verify that meta-feature based on Pareto front can represent MOPs and realize algorithm recommendation, and the accuracy of algorithm recommendation will be improved if we consider both meta-features.

This paper is organized as follows: Sect. 2 introduces the background of the meta-features and Pareto front geometrical features of MOPs. Section 3 presents the meta-feature combination proposed in this paper. Section 4 shows the experimental process, setup and result analysis. Section 5 makes conclusions.

## 2 Related Background

### 2.1 Meta-feature

Meta-features are a set of data to characterize problem properties and their relations with algorithm performance [14]. Identifying the appropriate set of meta-features is a key challenge and a crucial step for meta-learning task. Limited literatures [17,18] have proposed the formal definition of meta-feature for single objective optimization problem. Meta-features are defined as a function $f : D \rightarrow R_k$, calculated by a set of $k$ values extracted from a dataset $D$. The function $f$ detailed as

$$f(D) = \sigma(m(D, h_m), h_s) \tag{3}$$

According to this function, we can know that the extraction of meta-features is divided into two steps. The first step $m : D \rightarrow R_k^{'}$ is a characterization measure [18], it extracts useful fitness information values from a dataset $D$, the second step $\sigma : R_k^{'} \rightarrow R_k$ is a summarization function [18], such as mean, minimum, maximum, skewness and so on.

In the field of single objective optimization, the extraction technology of meta-features is very mature. So far, several types of meta-features are proposed to characterize problem, including simple meta-features, statistical meta-features, information theoretic meta-features [15,16], model based meta-features [16,19] , landmarking meta-features [20], and so on. However, the focus of the research is still to choose a set of meta-features suitable for a certain kind of problem, so that the algorithm recommendation effect of meta-learning is the best. Many literatures have successfully extracted meta-features and developed the algorithm recommendation model for the single objective optimization problem. Fabio Pinto et al. [21] presented a framework to systematically generate meta-features which are more informative than the non-systematic ones. Adriano Rivolli et al. [17,18] proposed a tool MFE to solve the problem that the meta-learning experiment is difficult to reproduce. Jorge Kanda et al. [13] studied the four groups of meta-features of TSP problem, such as the edge and vertex measures, the result shows a good solution with a well meta-feature set, though TSP problem under the different scene; Xianghua Chu et al. [12] proposed an adaptive algorithm recommendation system (ARM) based on meta-learning, which extracted three meta-features, including statistical features, geometric measurement features and landscape features, to represent the target space, and the experimental results showed high recommendation accuracy.

In spite of the technology and application of meta-feature extraction for single objective optimization problem are very mature at present, the technique of feature extraction may not be suitable for multi-objective problems, because MOPs have more complex characteristics. The optimal solution of the MOP is not one, but a Pareto solution set composed of many solutions. Therefore, it is very important to understand the problem from the perspective of Pareto fronts and Pareto solutions. The existing research on the characteristics of multi-objective optimization problems focuses on the construction of benchmark functions and the description of some characteristics of these functions [5,22].

The purpose is to test the multi-objective algorithm on standard test functions with various characteristics. Different algorithms have different performance in solving multi-objective problems with different characteristics. The current research on the characteristics of multi-objective problems only stops at the description of language, No literatures describe the systematic meta-feature extraction and meta-model construction method of MOP. Nevertheless, some studies [22] have shown that the performance of decomposition-based MOEA is closely related to the shape of the Pareto front, indicating that there is also a mapping between the performance of MOEA and some properties specific to MOP. In this paper, we propose the Pareto geometric features peculiar to MOPs as meta-features besides the traditional ones.

### 2.2 Pareto Front Geometrical Features of MOPs

Different from the Pareto optimal front of the single objective problem is a single point, the Pareto optimal front of the MOPs is a plane mapped by the Pareto optimal solution set, which can have a wide variety of geometric shapes.

The geometrical features of MOPs' Pareto Front include convex, concave, mixed, degenerate, connected [5]. A convex front is one that covers its convex hull. A convex front is a front that is covered by its convex hull. The linear front is both convex and concave. A front is mixed if the front has connected subsets that contain at least two of the three properties strictly convex, strictly concave and linear. A degenerate front is one that the dimension of it is one dimension less than that of the objective space, for instance, a front that is a point in a two objective problem is degenerate.

In this paper, we name the Pareto front geometric feature of MOPs using in ML as PF-based meta-feature, which is used to represent a unique meta-feature of multi-objective problems in the meta-learning task.

## 3 Proposed Meta-features for MOPs

In order to comprehensively capture the characteristics of the problem and highlight the characteristics of the solution space of the MOPs, we consider two kinds of meta-features: the first one is the target space-based features applied in meta-learning studies [12,23] including statistical features and geometric measurement features. The aim is to characterize the fitness space of MOPs. The other one is a proposed new meta-feature based on the Pareto front to characterize the shape and properties of the Pareto front of MOPs.

### 3.1 Target Space-Based Features

As we all know, the algorithm carries out random search in the objective space of the optimization problem and approaches the optimal solution step by step according to the specified mechanism. Therefore, the distribution of the objective space can provide important information about the most appropriate search strategy for a particular space. Consistent with the single-objective problem, the objective function of the MOPs also has statistical characteristics that can be used to describe each objective space, so we consider the statistical features which can provide the statistical information of the problem's target space and are relatively simple to be extracted. We take $N$ data points as a sample, the characterization measure is fitness value $f(x_i)$, which is calculated by the correspondent objective function $f(x)$ at the point $i$. Table 1 shows a set of summarization function that can almost comprehensively capture the statistical characteristics of MOPs' objective space. The mean of fitness value reflects the average level of it, and represents the average height of fitness space to a certain extent. The standard deviation of fitness values evaluates the degree to which the fitness value deviates from the mean and the bumpiness of the surface. Skewness and kurtosis of fitness values evaluates the symmetry of the surface and its flatness relative to the normal distribution.

**Table 1.** Meta-features based on objective space statistical information.

| Meta-feature | Description |
|---|---|
| $\overline{f} = \frac{1}{N}\sum_{i=1}^{N} f_i$ | Mean of fitness values |
| $SD(f) = \sqrt{\frac{1}{N-1}\sum_{i=1}^{N}(f_i - \overline{f})^2}$ | Standard deviation of fitness values |
| $\gamma_1(f(x)) = E\left\{\left[(f_i - \overline{f})/Std.(f_i)\right]^3\right\}$ | Skewness of fitness values |
| $\gamma_2(f(x)) = E\left[(f_i - \overline{f})^4\right]/(E\left[(f_i - \overline{f})^2\right])^2$ | Kurtosis of fitness values |
| $\nabla f = lg(abs(max(f_i) - min(f_i)))$ | Altitude of search space |
| $Q_1 = 25\% quartile\ of\ response\ values$ | The lower quartile of fitness values |
| $Q_2 = 50\% quartile\ of\ response\ values$ | The median quartile of fitness values |
| $Q_3 = 75\% quartile\ of\ response\ values$ | The upper quartile of fitness values |

However, simple statistical features may not be able to capture important problem surfaces characteristics which are very complex, so we consider another set of meta-features that can also describe the objective space surfaces characteristics, the geometric measurement features. We uses the gradient value $G_i$ of the $ith$ data point as the characterization measure, $G_i$ is calculated as:

$$G_i = f(x_i) - f(x_i + \Delta x_i), i = 1, \cdots, N \tag{4}$$

In this equation, $x_i$ refers to the position of the point $i$ in $D$ dimension, $f(x_i)$ is the fitness value and $\Delta x_i$ is 1% of the domain of the function. As is shown in the Table 2, we select 5 summarization functions [12] to extract meta-features, including the gradient-based features and outlier ratio. The first meta-feature mean of gradient of fitness surface evaluates the steepness and roughness of the fitness surface based on its rate of change around the sampled data points. The standard deviation of gradient of fitness surface evaluates the changes in the rate of change of sample data. Max of gradient of fitness surface is a measure of the maximum degree of surface mutation. The outlier ratio evaluated by the Grubbs Test measures the percentage of extreme values in all response values.

**Table 2.** Meta-features based on objective space surfaces characteristics.

| Meta-feature | Description |
|---|---|
| $\overline{\lvert G\rvert} = \frac{1}{N}\sum_{i=1}^{N}\lvert G_i\rvert$ | Mean of gradient of fitness surface |
| $M(\lvert G\rvert)$ | Median of gradient of fitness surface |
| $SD(\lvert G\rvert) = \sqrt{\frac{1}{N-1}\sum_{i=1}^{N}(\lvert G_i\rvert - \overline{\lvert G\rvert})^2}$ | Standard deviation of gradient of fitness surface |
| $\lvert G\rvert_{max} = max\{\lvert G_1\rvert, \lvert G_2\rvert, \cdots, \lvert G_N\rvert\}$ | Max of gradient of fitness surface |
| $OR$ | Outlier ratio |

### 3.2 PF-Based Features

The meta-features from the previous section are the features of MOPs' objective space. However, for MOPs, the shape and properties of the Pareto front are also ever-changing and can provide important information for the recommendation of the optimal solution algorithm [23]. Unfortunately, the mapping of the Pareto front can be one-to-one or many-to-one, and the complexity of Pareto front increases with the increase of targets. So it's difficult to extract features from a point of information in the Pareto front. In another way, we can select the classification attribute as the feature of the Pareto front.

The meta-feature is extracted through One-Hot Encoding. Firstly, we determine the feature types of the MOPs' PF, and then the features are transformed into digital features by the One-Hot Encoding. The reason for this method is that it can extend the discrete feature value to Euclidean space, and the code composed of several discrete features corresponds to a point in Euclidean space, so it will be more reasonable to calculate the distance between features. Meta-features based on Pareto front geometrical characteristics are shown in Table 3, the shapes of the Pareto front include concave, convex, linear and mixed. The continuity of the Pareto front indicates whether the Pareto front is disconnected, and the dimensional consistency of Pareto front mapping reveals whether degenerate solutions are exist in the Pareto front.

**Table 3.** Meta-features based on Pareto front Geometrical characteristics.

| Meta-feature | Description |
|---|---|
| $SPF$ | The shapes of Pareto front |
| $CPF$ | The continuity of the Pareto front |
| $DCPF$ | The dimensional consistency of Pareto front mapping |

# 4 Experimental Validation and Results

## 4.1 Experimental Process

To verify the two types of meta-features we used can well capture the characteristics of MOPs, we apply the framework of adaptive recommendation system from Chu's study [12], which has been proved that it can provide better ranking success rate and optimization problem efficiency in the case of extracting the features based on the target space and using the K-NN learning algorithm to learn. Figure 1 shows this framework.

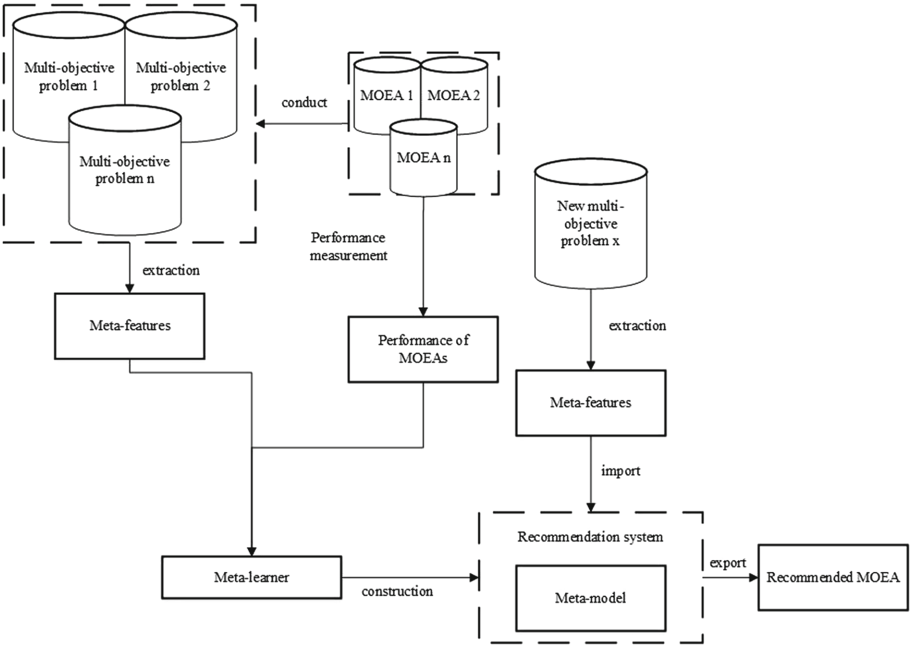

**Fig. 1.** Adaptive recommendation system framework.

In this study, the problem repository is composed of four problem suites with 25 benchmark functions: DTLZ1-9, WFG1-8, ZDT1-6 and MW1-2. Six representative MOEAs of three types, MOEA/D, NSGA-III, NSGA-II, SPEA2, eMOEA, and IBEA were selected as the algorithm repository. The performance of the MOEAs we selected is measured by the algorithm ideal ranking for each multi-objective benchmark function. The algorithm ideal ranking of a multi-objective benchmark function is determined by the average result of Inverted Generational Distance (IGD), which is a performance indicator obtained by each algorithm runs 10 times on this problem. The smaller the value of IGD, the higher the ranking. Since the more meta-features extracted does not mean the better effect, the combination of meta-features is considered in this paper. Firstly, we test the PF-based meta-feature separately, and use the meta-classifier to train meta-model which can recommend the best algorithm ranking for a new problem. Then the recommendation results of its combination with target space-based meta-features in the Sect. 3 are studied. Besides, we also consider the influence of the combination of meta-features on the recommendation results of different problem dimensions (30-dimension, 40-dimension and 50-dimension). Spearman's rank correlation coefficient (SRCC) and hit ratio are used as indexes to evaluate the performance of the meta-model by leave-one-out cross validation method. SRCC can evaluates the consistency between the recommended ranking and the ideal ranking. The hit ratio can measure the percentage of exact matches between the ideal and recommended best performance in all problems.

Since K-NN has shown its high efficiency in algorithm selection problems [12,13], we choose KNN algorithm for training meta-model. K-NN is based on some distance measurement to find the k examples closest to the target in the training set, and uses "voting method" to classify the new examples based on the type of k nearest neighbor examples. Many literatures [12,13] show that K = 3 has the best effect, so 3-NN is finally selected for classification prediction.

### 4.2 Experimental Setup

To ensure fairness and unbiased, the following measures are taken in this paper:

1) All the experiments are implemented in MATLAB 2017b with Intel Core i7 2.6 GHz and 16.0 GB RAM.
2) the parameter setting of the MOEAs is the default in the PlatEMO [25].
3) The sample size of the benchmark function is set as 1000, the objective number of each benchmark function is set as 2.
4) The population number and number of iterations of each algorithm are set to 100 and 10,000 respectively, and each algorithm runs 10 times independently.
5) The ideal ranking of algorithm performance of the six algorithms on 25 benchmark functions is obtained through experiments on the PlatEMO [25].

### 4.3 Experimental Results and Evaluation

In this paper, we obtained the ideal ranking of 6 MOEAs for each MOP benchmark function of 30 dimensions, 40 dimensions and 50 dimensions respectively to measure MOEAs performance. Take the 50-dimensional result in Table 4 as an example, the optimal algorithm data of each benchmark function is indicated in bold. For experimental results from 50-dimension, NSGA-II algorithm has the best performance for benchmark functions DTLZ 2, 3, 5, 6, 9, WFG 4, 5, 6, 7, 8 and ZDT 2, which may be related to the feature that their Pareto fronts are convex and continuous. For the test problems DTLZ 1, DTLZ 8 and WFG 3, where the Pareto front shape is linear and continuous, the MOEAS based on decomposition have the best performance. According to the data, many such relationships can be found, which further indicates that the Pareto front shape can provide information about the optimal performance algorithm.

**Table 4.** IGD of the six MOEAs on 25 benchmark functions for 50-dimension.

| Benchmark function | MOEAD | NSGA-II | NSGA-III | IBEA | SPEA2 | eMOEA |
|---|---|---|---|---|---|---|
| DTLZ1 | 1.3820e+2 | 1.3419e+2 | 2.4903e+2 | **1.2311e+2** | 1.2921e+2 | 1.7297e+2 |
| DTLZ2 | 2.8348e−2 | **1.7646e−2** | 3.5810e−2 | 2.5657e−2 | 2.1053e−2 | 9.0678e−2 |
| DTLZ3 | 3.7114e+2 | **3.3828e+2** | 6.2927e+2 | 3.6210e+2 | 3.4927e+2 | 5.3858e+2 |
| DTLZ4 | 5.2775e−1 | 3.0988e−1 | **4.1815e−2** | 3.8568e−1 | 9.4148e−2 | 2.8794e−1 |
| DTLZ5 | 3.0556e−2 | **1.7965e−2** | 3.4749e−2 | 2.5436e−2 | 2.0218e−2 | 1.0335e−1 |
| DTLZ6 | 1.6959e+1 | **8.8287e+0** | 1.0816e+1 | 9.0258e+0 | 1.0243e+1 | 1.5907e+1 |
| DTLZ7 | 8.1595e−1 | 7.8510e−2 | 2.0468e−1 | **5.7889e−2** | 1.2362e−1 | 7.2861e−1 |
| DTLZ8 | NaN (NaN) | 1.6596e−1 | **1.5116e−1** | NaN (NaN) | NaN (NaN) | NaN (NaN) |
| DTLZ9 | 1.6629e+1 | **6.4314e+0** | 7.7523e+0 | 7.1369e+0 | 7.2025e+0 | 1.1439e+1 |
| WFG1 | 1.5047e+0 | 1.0463e+0 | 1.1879e+0 | **9.4770e−1** | 1.0947e+0 | 1.2010e+0 |
| WFG2 | 3.1789e−1 | **1.0846e−1** | 1.1682e−1 | 1.1242e−1 | 1.1807e−1 | 1.9480e−1 |
| WFG3 | 2.9008e−1 | 1.1325e−1 | 1.4112e−1 | **9.6788e−2** | 1.2120e−1 | 1.5086e−1 |
| WFG4 | 2.4521e−1 | **8.2535e−2** | 9.3848e−2 | 9.4766e−2 | 8.5879e−2 | 1.4315e−1 |
| WFG5 | 1.6671e−1 | **5.5367e−2** | 6.8498e−2 | 6.7162e−2 | 5.9289e−2 | 1.0669e−1 |
| WFG6 | 2.3195e−1 | **8.8982e−2** | 1.1974e−1 | 9.9508e−2 | 1.0070e−1 | 1.3485e−1 |
| WFG7 | 3.7602e−1 | **4.2815e−2** | 6.0185e−2 | 5.9616e−2 | 4.6995e−2 | 1.0329e−1 |
| WFG8 | 2.6282e−1 | **1.2818e−1** | 1.4241e−1 | 1.3408e−1 | 1.3208e−1 | 1.7850e−1 |
| ZDT1 | 4.8669e−1 | 5.0774e−2 | 1.0124e−1 | **3.1166e−2** | 5.7299e−2 | 1.6325e−1 |
| ZDT2 | 6.8341e−1 | **9.1844e−2** | 2.2581e−1 | 4.6176e−1 | 9.4580e−2 | 1.2176e+0 |
| ZDT3 | 4.8648e−1 | 5.5911e−2 | 9.4073e−2 | **2.6406e−2** | 7.2564e−2 | 1.6750e−1 |
| ZDT4 | 7.6528e+1 | 6.1339e+1 | 8.2208e+1 | 6.7255e+1 | **5.4032e+1** | 2.0289e+2 |
| ZDT5 | 1.0041e+1 | 2.3197e−1 | 5.3910e−1 | 2.8128e+0 | **1.6152e−1** | 6.8829e−1 |
| ZDT6 | 3.7091e+0 | 3.0061e+0 | 3.9115e+0 | **2.7027e+0** | 3.4060e+0 | 5.2363e+0 |
| MW1 | NaN (NaN) | NaN (NaN) | NaN (NaN) | NaN (NaN) | NaN (NaN) | NaN (NaN) |
| MW2 | 5.5339e−1 | 3.7704e−1 | 5.0171e−1 | 3.6892e−1 | **1.7557e−1** | 9.6832e−1 |

First, PF-based features are used to extract meta-features from multi-objective benchmark functions to help build the meta-model, Table 5 lists the average SRCC and hit ratio for three different problem dimensions. SRCC [26] is a metric used to assess the consistency between a recommended ranking and a real ranking. SRCC is defined as:

$$\rho_i = 1 - 6\left[\left(\sum_{i=1}^{N} d_{i,a}^2\right) / (N/(N^2-1))\right] \tag{5}$$

where $d_{i,a}$ is the Manhattan distance between the recommended rank and ideal rank of algorithms $a$; $N$ is the number of algorithms. If $\rho$ is equal to 1, that means the results are very consistent, the prediction is more accurate. Hit ratio is the ratio between the number of correctly predicted labels in test cases and the number of all test cases. The best result is shown in bold. From the experimental results, the overall average recommendation accuracy is more than 60%, indicating that the MOEAs recommendation method based on Pareto front features has achieved initial success. In Table 5, the highest SRCC 72% occurred in algorithm recommendation for 40-dimensional problems, and the highest hit ratio 76% also occurred in algorithm recommendation for 40-dimensional problems, that is, 19 optimal algorithms could be selected from 25 benchmark functions.

**Table 5.** The SRCC and hit ratio results of using only PF-based features

| Performance indicators | 30-D | 40-D | 50-D |
|---|---|---|---|
| SRCC | 0.62 ± 0.04 | **0.72** ± 0.02 | 0.67 ± 0.03 |
| Hit ratio | 0.60 ± 0.02 | **0.76** ± 0.02 | 0.68 ± 0.02 |

**Table 6.** The SRCC and hit ratio results of using two types of features

| Performance indicators | 30-D | 40-D | 50-D |
|---|---|---|---|
| SRCC | 0.62 ± 0.03 | 0.72 ± 0.02 | **0.74** ± 0.02 |
| Hit ratio | 0.64 ± 0.02 | **0.76** ± 0.02 | 0.72 ± 0.02 |

In addition to Pareto-based features, whether combining other features, such as those based on target space, can contribute to the improvement of accuracy remains to be studied. Therefore, we used the target space-based features in the Sect. 3 combined with PF-based features to extract meta-features for the multi-objective benchmark function. The results are shown in Table 6, its overall average recommendation accuracy is also over 60%. Figure 2 shows the line chart

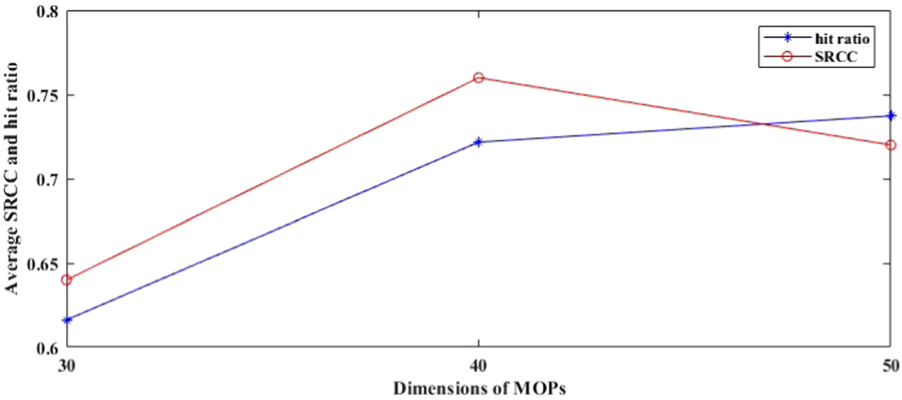

**Fig. 2.** The performance of the recommendation using three types of features in three dimensions.

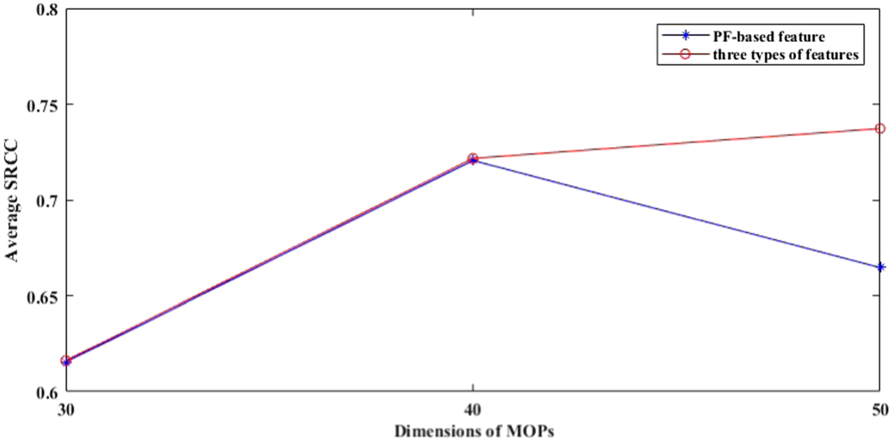

**Fig. 3.** SRCC values under different meta-features.

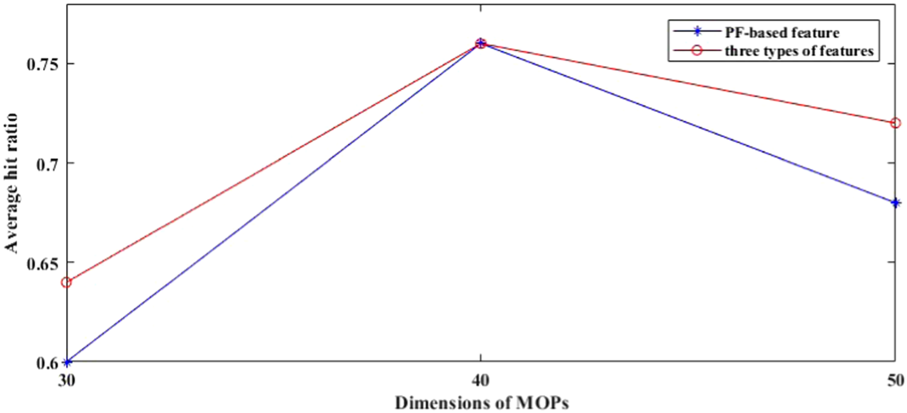

**Fig. 4.** Hit ratio values under different meta-features.

of average SRCC and hit ratio values under different dimensions. It can be seen that the change of problem dimension will affect the final recommendation result of the problem, and this influence may be positive.

In Fig. 3, when the dimensions are in 30 and 40 dimensions, the combination of Pareto-based features and other problem features as meta-features will not have much impact on the final prediction accuracy, but for 50 dimensions, it improves the prediction accuracy. In Fig. 4, the combined meta-features help improve the hit ratio of the predicted problem. In general, although the Pareto-based meta-features can produce better results for problem recommendation, the combined meta-features perform better than the Pareto-based meta-features alone, which indicates that the Pareto-based and the target space-based meta-features should be considered simultaneously when extracting features for multi-objective problems.

# 5 Conclusion

Meta-feature extraction is a challenging frontier topic in algorithm selection and recommendation. Its goal is to select the features that can represent the problem and map them with the algorithm performance that can solve the problem, so as to improve the accuracy of recommendation. In the research of single-objective problem, feature extraction and algorithm recommendation have been very mature, and there is no systematic literature to extract features and recommend appropriate algorithms for MOPs. In this paper, we propose a unique meta-feature based on Pareto shape and properties for MOPs, carry out experiment on 25 multi-objective benchmark functions and use K-NN algorithm to realize the algorithm recommendation of extracting PF-based features for MOPs. In the experimental process, we also considered the combination of features based on PF and target space.

The contributions of this paper summarized as follows: 1) We introduce the meta-learning of machine learning field to the algorithm selection of multi-objective problem, which expands the new perspective of solving multi-objective problems; 2) We verify that the shape of the Pareto fronts can provide effective information for the algorithm recommendation of multi-objective problems; 3) We propose the PF-based features peculiar to MOPs for meta-learning processes, which can reduce the space-time complexity of feature extraction under the premise of no significant reduction in recommendation accuracy; 3) The result of our experiment proves that PF-based meta-features can represent MOPs and realize algorithm recommendation. Simultaneously, if the combination of PF-based meta-features and target space-based meta-features is considered, the accuracy of algorithm recommendation will be improved, indicating that the combination of those features can more comprehensively characterize the features of MOPs. This provides more possibilities for multi-objective feature extraction engineering field.

Although the prospect of feature extraction for multi-objective problems is good at present, there is still room for progress. For example, the target space of the MOP also has some features that the single-objective problem does not have, such as multimodal, deceptive. How to represent these features and whether they can be used to extract meta-features remains to be studied.

**Acknowledgements.** This work was partially supported by the National Natural Science Foundation of China (Grant No. 71971142 and 71701079).

# References

1. Li, K., Wang, R., Zhang, T.: Evolutionary many-objective optimization: a comparative study of the state-of-the-art. IEEE Access **6**, 26194–26214 (2018)
2. Ma, X., Yang, J., Wu, N.: A comparative study on decomposition-based multi-objective evolutionary algorithms for many-objective optimization. In: 2016 IEEE Congress on Evolutionary Computation (CEC), pp. 2477–2483 (2016)

3. Wang, R., Zhou, Z., Ishibuchi, H.: Localized weighted sum method for many-objective optimization. IEEE Trans. Evol. Comput. **22**(1), 3–18 (2018)
4. Tanabe, R., Ishibuchi, H.: A review of evolutionary multimodal multiobjective optimization. IEEE Trans. Evol. Comput. **24**(1), 193–200 (2020)
5. Huband, S., Hingston, P., Barone, L.: A review of multiobjective test problems and a scalable test problem toolkit. IEEE Trans. Evol. Comput. **10**(5), 477–506 (2006)
6. Zitzler, E., Thiele, L.: An evolutionary algorithm for multiobjective optimization: the strength pareto approach. TIK-report, vol. 43 (1998)
7. Zhang, Q., Li, H.: MOEA/D: a multiobjective evolutionary algorithm based on decomposition. IEEE Trans. Evol. Comput. **11**(6), 712–731 (2007)
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 Trans. Evol. Comput. **18**(4), 577–601 (2013)
9. Deb, K., Pratap, A., Agarwal, S.: A fast and elitist multiobjective genetic algorithm: NSGA-II. IEEE Trans. Evol. Comput. **6**(2), 182–197 (2002)
10. Zitzler, E., Laumanns, M., Thiele, L.: SPEA2: improving the strength Pareto evolutionary algorithm. TIK-report, vol. 103 (2001)
11. Zitzler, E., Künzli, S.: Indicator-based selection in multiobjective search. In: Yao, X., et al. (eds.) PPSN 2004. LNCS, vol. 3242, pp. 832–842. Springer, Heidelberg (2004). https://doi.org/10.1007/978-3-540-30217-9_84
12. Chu, X., Cai, F., Cui, C.: Adaptive recommendation model using meta-learning for population-based algorithms. Inf. Sci. **476**, 192–210 (2019)
13. Kanda, J., De Carvalho, A., Hruschka, E.: Meta-learning to select the best meta-heuristic for the traveling salesman problem: a comparison of meta-features. Neurocomputing **205**, 393–406 (2016)
14. Brazdil, P., Carrier, C.G., Soares, C.: Metalearning: Applications to Data Mining. Springer, Heidelberg (2008). https://doi.org/10.1007/978-3-540-73263-1
15. Peng, Y., Flach, P.A., Soares, C., Brazdil, P.: Improved dataset characterisation for meta-learning. In: Lange, S., Satoh, K., Smith, C.H. (eds.) DS 2002. LNCS, vol. 2534, pp. 141–152. Springer, Heidelberg (2002). https://doi.org/10.1007/3-540-36182-0_14
16. Vilalta, R., Giraud-Carrier, C.G., Brazdil, P.: Using meta-learning to support data mining. Int. J. Comput. Sci. Appl. **1**(1), 31–45 (2004)
17. Alcobaca, E., Siqueira, F., Rivolli, A.: MFE: towards reproducible meta-feature extraction. J. Mach. Learn. Res. **21**, 1–5 (2020)
18. Rivolli, A., Garcia, L. P., Soares, C.: Towards reproducible empirical research in meta-learning, pp. 32–52 (2018)
19. Reif, M., Shafait, F., Goldstein, M., Breuel, T., Dengel, A.: Automatic classifier selection for non-experts. Pattern Anal. Appl. **17**(1), 83–96 (2012). https://doi.org/10.1007/s10044-012-0280-z
20. Balte, A., Pise, N., Kulkarni, P.: Meta-learning with landmarking: a survey. Int. J. Comput. Appl. **105**, 8 (2014)
21. Pinto, F., Soares, C., Mendes-Moreira, J.: Towards automatic generation of metafeatures. In: Bailey, J., Khan, L., Washio, T., Dobbie, G., Huang, J.Z., Wang, R. (eds.) PAKDD 2016. LNCS (LNAI), vol. 9651, pp. 215–226. Springer, Cham (2016). https://doi.org/10.1007/978-3-319-31753-3_18
22. Bossek, J.: smoof: single-and multi-objective optimization test functions. R J. **9**(1), 103 (2017)
23. Ishibuchi, H., Setoguchi, Y., Masuda, H.: Performance of decomposition-based many-objective algorithms strongly depends on pareto front shapes. IEEE Trans. Evol. Comput. **21**(2), 169–190 (2017)

24. Cui, C., Hu, M.Q., Weir, J.D.: A recommendation system for meta-modeling: a meta-learning based approach. Expert Syst. Appl. **46**, 33–44 (2016)
25. Tian, Y., Cheng, R., Zhang, X.: PlatEMO: a MATLAB platform for evolutionary multi-objective optimization [educational forum]. IEEE Comput. Intell. Mag. **12**(4), 73–87 (2017)
26. Neave, H., Worthington, P.: Distribution-free tests. Contemp. Sociol. **19**(3), 137–153 (1990)