TSINGHUA SCIENCE AND TECHNOLOGY
ISSN 1007-0214 04/24 pp343–355
DOI: 10.26599/TST.2023.9010036
Volume 29, Number 2, April 2024

Evolutionary Multi-Tasking Optimization for High-Efficiency Time Series Data Clustering

Rui Wang, Wenhua Li*, Kaili Shen, Tao Zhang, and Xiangke Liao

Abstract: Time series clustering is a challenging problem due to the large-volume, high-dimensional, and warping characteristics of time series data. Traditional clustering methods often use a single criterion or distance measure, which may not capture all the features of the data. This paper proposes a novel method for time series clustering based on evolutionary multi-tasking optimization, termed i-MFEA, which uses an improved multifactorial evolutionary algorithm to optimize multiple clustering tasks simultaneously, each with a different validity index or distance measure. Therefore, i-MFEA can produce diverse and robust clustering solutions that satisfy various preferences of decision-makers. Experiments on two artificial datasets show that i-MFEA outperforms single-objective evolutionary algorithms and traditional clustering methods in terms of convergence speed and clustering quality. The paper also discusses how i-MFEA can address two long-standing issues in time series clustering: the choice of appropriate similarity measure and the number of clusters.

Key words: time series clustering; evolutionary multi-tasking; multifactorial optimization; clustering validity index; distance measure

1 Introduction

Time series data arises regularly in many fields such as finance, traffic, and health[1, 2]. How to dig out the value behind these data has become a research hotspot. However, the feature of raw time series data cannot be easily detected due to the large-volume, high-dimensional, and warping characteristics. As is known, clustering methods, e.g., k-means[3], DBSCAN[4], and hierarchical clustering[5], can help extract typical and abnormal patterns. Nevertheless, existing methods are not specially designed for dealing with time series data with such high-dimensional and complex datasets, which therefore may fail when performing on large-scale time series data.

Clustering methods aim to partition data into appropriate clusters based on certain objectives or rules[6, 7], such as minimization of the within-classes-variance or maximization of the between-classes-variance. Since clustering can be regarded as an optimization problem, Evolutionary Algorithms (EAs) as powerful optimizers[8, 9], are often applied, e.g., Particle Swarm Optimization (PSO) algorithm[10, 11] and Genetic Algorithm (GA)[12, 13].

Despite the popularity of traditional clustering algorithms and single-objective EA-based clustering methods, we argue that these methods can achieve clustering results by optimizing one objective, either

• Rui Wang is with Xiangjiang Laboratory, Changsha 410205, China, and also with College of Systems Engineering, National University of Defense Technology (NUDT), Changsha 410073, China. E-mail: ruiwangnudt@gmail.com.
• Wenhua Li and Tao Zhang are with College of Systems Engineering, National University of Defense Technology, Changsha 410073, China. E-mail: liwenhua1030@aliyun.com; zhangtao@nudt.edu.cn.
• Kaili Shen is with Ant Group Co., Ltd., Hangzhou 310000, China. E-mail: xiaowo.skl@alibaba-inc.com.
• Xiangke Liao is with College of Computer Science and Technology, NUDT, Changsha 410073, China. E-mail: xkliao@nudt.edu.cn.
∗ To whom correspondence should be addressed.
Manuscript received: 2023-03-21; revised: 2023-04-10; accepted: 2023-04-19

implicitly or explicitly[14, 15], may be ineffective. For instance, the k-means merely minimizes the Sum of the Square Distance (SSD) between data points and cluster centroids, though there are various clustering validity indexes, such as compactness, separation, and connectivity[16]. Evidently, a single index cannot capture all the characters of certain datasets, and thus often results in poor performance when it is not appropriately used. Moreover, optimization based on different clustering validity indexes simultaneously can lead to different results satisfying various preferences, which, to some extent, could be more appealing to decision-makers. To do so, clustering methods that adopt the idea of multi-objective optimization have been proposed. For example, Handl and Knowles[17] proposed a two-objective model, to minimize the overall deviation and connectivity for clustering problems by EAs. Wang et al.[18] considered optimizing the number of clusters and the SSD, which can obtain a set of clustering results corresponding to different k. Although multiple clustering results can be found via Evolutionary Multi-objective Optimization (EMO)[18, 19], the use of EMO requires optimization objectives to be conflicted with one another. The clustering validity indexes, however, are not always conflicting. For example, minimizing the within-variance would not cause an increase of between-classes-variance exactly. Thus, when obtaining multiple clustering results by EMO, extra transformation needs to be performed on objective functions. To avoid such transformation, the recent Evolutionary Multi-Tasking (EMT) framework[20] can be applied. EMT is inspired by the human brain, which aims to improve problem-solving ability through implicit synergy among different (optimization) tasks. By assigning certain populations to different tasks, the solutions to multiple tasks can be obtained simultaneously. Specifically, the genetic segments of different tasks, which are similar to an extent, can be shared during the evolution, thus, accelerating the search process. An application of EMT in real-world problems can be found in Ref. [21].

As for complex high-dimensional time series, clustering tasks with different objective functions (corresponding to different clustering validity indexes) can be constructed based on the EMT framework. There are many useful validity indexes that can measure the compactness and connectivity of clustering results, e.g., the Total-Within-Cluster-Variation (TWCV)[22], Davies-Bouldin Index (DBI)[23], and Silhouette Index (SI). Also, the distance/similarity measures are essential in objective function construction, e.g., Euclidean distance, cosine wavelets, Dynamic Time Warping (DTW), and Pearson correlation coefficient[24]. Since evolutionary clustering with different validity indexes usually produce different clustering results, integrating these validity indexes into the EMT framework may result in more comprehensive clustering results, which is the major motivation of this work. Overall, the main contributions of this study are as follows:

(1) This work proposes a novel method based on EMT optimization, which can optimize multiple clustering tasks with different validity indexes or distance measures simultaneously. This enables the generation of diverse and robust clustering solutions that can satisfy various preferences of decision-makers.

(2) An improved multifactorial EA is introduced, which enhances the vertical cultural transmission strategy and avoids the loss of high-quality solutions during the evolution process.

(3) This work demonstrates the effectiveness and superiority of the proposed method on two artificial datasets, and shows how it can address two long-standing issues in time series clustering: the choice of appropriate similarity measure and the number of clusters.

(4) The paper also discusses how the proposed method can be applied to anomaly detection by clustering. The paper thus provides a new perspective and a powerful tool for time series clustering and analysis.

The rest of this study is organized as follows. In Section 2, the background of clustering methods and evolutionary multi-tasking is introduced. In Section 3, the improved MFEA (i-MFEA) for multi-clustering of time series data is elaborated. Experimental results and discussion are presented in Sections 4 and 5, respectively. Section 6 concludes the paper.

2 Related Work

2.1 Time series clustering

Time series clustering refers to the partition of a dataset consisting of time series curves according to certain validity indexes[1, 25]. Given a dataset $\mathcal{D} = \{x_1, x_2, \dots x_n\}$ composed of n unmarked samples, each sample $x_i = (x_{i1}, x_{i2}, \dots, x_{im})$ is an m-dimensional vector. Clustering is to divide the dataset $\mathcal{D}$ into k disjoint subsets $\{C_l | l = 1, 2, \dots, k\}$. Meanwhile, it ensures that samples in the same cluster are more similar than samples in different clusters. Generally, m_l is the centroid of the l-th cluster, and each sample x_i is assigned to the nearest cluster centroid. The partition matrix u_{il}, describing the membership of samples to centroids, is as follows:

$$u_{il} = \begin{cases} 1, & \text{if } l = \operatorname{argmin} F(i); \\ 0, & \text{otherwise}, \end{cases} \tag{1}$$

$$F(i) = \operatorname{dist}(x_i, m_l) \tag{2}$$

where $\operatorname{dist}(x_i, m_l)$ is the distance function that measures the similarity between x_i and m_l.

Many distance functions have been proposed to measure the degree of similarity between two vectors, e.g., Euclidean distance, DTW, Pearson correlation, and cosine wavelets[26]. Let x_i and x_j be m-dimensional vectors, four similarity/distance functions are described here. Note that the symbol "↓" indicates that the smaller the distance, the more similar the vectors. The symbol "↑" refers to the opposite.

Euclidean distance (↓): It is the most widely used metric to measure the similarity of two vectors, and can be defined as follows:

$$D_E(x_i, x_j) = \sqrt{\sum_{r=1}^{m} \left|x_{ir} - x_{jr}\right|^2} \tag{3}$$

DTW (↓)[27]: DTW is an improved version of Euclidean distance. Two time series may be similar but not aligned along the same time axis. DTW can be used to twist one of the sequences on the time axis for better alignment to obtain the shortest warping distance between two series. Based on dynamic programming, the match starts from (x_{i1}, x_{j1}), and the distance is accumulated at each point until (x_{in}, x_{jm}). The cumulative distance can be calculated as follows:

$$\begin{aligned} D_{\mathrm{DTW}}(n,m) = {} & D_E(x_{in}, x_{jm}) + \\ & \min\Big\{D_{\mathrm{DTW}}(n-1,m), D_{\mathrm{DTW}}(n,m-1), \\ & D_{\mathrm{DTW}}(n-1,m-1)\Big\} \end{aligned} \tag{4}$$

Cosine wavelets (↑)[28]: Its value is within the range of $[-1, 1]$. It has no relationship with the magnitude of the vectors but only is relevant to the direction of the vectors, which is shown as follows:

$$D_{\text{cosine}}(x_i, x_j) = \frac{\sum_{r=1}^{m} x_{ir} x_{jr}}{\sqrt{\sum_{r=1}^{m} x_{ir}^2}\sqrt{\sum_{r=1}^{m} x_{jr}^2}} \tag{5}$$

Pearson correlation coefficient (↑)[29]: It measures the degree of correlation between two vectors. Its range is within $[-1, 1]$. When x_i is linearly correlated with x_j, the correlation coefficient is 1 (perfect positive correlation) or –1 (perfect negative correlation),

$$D_{\text{pearson}}(x_i, x_j) = \frac{\sum_{r=1}^{m}[(x_{ir} - u_{x_i})(x_{jr} - u_{x_j})]}{S_{x_i} S_{x_j}} \tag{6}$$

$$u_{x_i} = \frac{1}{m}\sum_{r=1}^{m} x_{ir},\ u_{x_j} = \frac{1}{m}\sum_{r=1}^{m} x_{jr} \tag{7}$$

$$S_{x_i} = \sqrt{\sum_{r=1}^{m}\left(x_{ir} - u_{x_i}\right)^2},\ S_{x_j} = \sqrt{\sum_{r=1}^{m}\left(x_{jr} - u_{x_j}\right)^2} \tag{8}$$

where u_{x_i} and S_{x_i} are the mean and standard deviation of x_i, respectively, and u_{x_j} and S_{x_j} are the mean and standard deviation of x_j, respectively.

2.2 Clustering validity index

Concerning the performance evaluation of different clustering methods, several clustering validity indexes have been proposed[1]. Next, we briefly introduce some representative ones.

SSD[18]: It measures the intra-variance of data points with their nearest centroid. Minimizing SSD leads to compactness, convex clusters, and representative cluster centroids. It is defined as follows:

$$\mathrm{SSD} = \sum_{r=1}^{k}\sum_{x_i \in C_r} \operatorname{dist}(x_i, m_r) = \sum_{r=1}^{k}\sum_{x_i \in C_r} \|x_i - m_r\|^2 \tag{9}$$

SI[30]: It is used to evaluate the effectiveness of data clustering. The larger the SI value is, the better the clustering result is, which is defined as follows:

$$\mathrm{SI} = \frac{1}{n}\sum_{i=1}^{n} \frac{b_{x_i} - a_{x_i}}{\max(a_{x_i}, b_{x_i})} \tag{10}$$

where a_{x_i} is the mean distance of x_i to all other samples in the same cluster, and b_{x_i} is the mean distance of x_i to other samples in different clusters.

DBI[23]: The index combines both the cluster cohesion (within-variance) and separation (distance between cluster centroids) information. It is defined as follows:

$$\mathrm{DBI} = \frac{1}{k}\sum_{i,j=1}^{k} \max_{i \neq j} R_{ij} \tag{11}$$

where R_{ij} represents the similarity between two clusters C_i and C_j.

In addition, the Adjusted Rand Index (ARI) and Adjusted Mutual Information (AMI)[16] are also commonly used. Specifically, the ARI measures the similarity of the two assignments, and AMI measures the consistency of two assignment distributions, ignoring the absolute value of the labels. Both ARI and AMI are positive indexes and standardized, with values range $[-1, 1]$. The larger the index value the better the clustering results.

2.3 Evolutionary multi-tasking

Given multiple optimization tasks, EMT[31, 32] can optimize them simultaneously through the unification of solution encoding (search space) and implicit genetic transfer. For example, in Fig. 1, the multi-tasking environment consists of two-tasks with objective function f_1 and f_2. The two-objective functions have different landscapes, however, their optimal solutions are similar. The gene segments of f_1 can help f_2 to jump out of local optima, thus accelerating the optimization process. Overall, when proper optimization tasks are constructed, the EMT framework has the potential to speed up the convergence, and obtain higher quality solutions thanks to the information exchange between different landscapes[20].

The MultiFactorial Optimization (MFO) algorithm is a paradigm of EMT[20]. Assuming that there are N_T tasks, each task is labeled as T_i $(1 \leqslant i \leqslant N_T)$. The objective function of each task is denoted as f_i $(1 \leqslant i \leqslant N_T)$.

The MultiFactorial EA (MFEA)[33, 34] has been demonstrated effective for MFO. Moreover, it is reported that the more similar the tasks are, the better performance MFEA would obtain[20]. The tasks can be discrete, continuous, mixed, single-objective, or multi-objective[35, 36]. For example, in Ref. [37], MFEA is applied to solve permutation-based combinatorial optimization problems. Experimental results show that the travel salesman problem, quadratic assignment problem, linear ordering problem, and job-shop scheduling problem can be solved simultaneously with both efficiency and effectiveness through the EMT framework by MFEA. In Ref. [37], an improved MFEA was proposed and applied to twelve multi-tasking capacitated vehicle routing problems. Permutation-based unified solution representation and split-based decoding were proposed to adapt to the problem-specific space. Nevertheless, there are abundant practical applications of MFEA in clustering tasks.

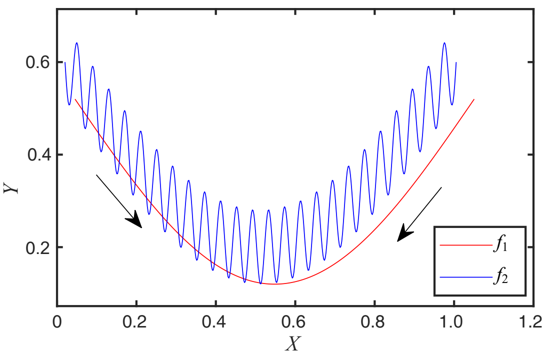


Fig. 1 Illustration of the evolutionary multi-tasking, where solution information of f_1 can be transferred to help the searching process of f_2.

3 Multifactorial Evolutionary Algorithm for Time Series Clustering

This section elaborates on time series clustering via i-MFEA. First, the general framework of MFEA is introduced. Then, the clustering tasks based on different clustering criteria are constructed. Third, the encoding strategy in i-MFEA is introduced. Lastly, an improved vertical cultural transmission is proposed, which is the critical step of i-MFEA and plays the role of multifactorial inheritance.

3.1 General framework

The framework of i-MFEA is shown in Algorithm 1, which contains five main steps. First, N_T clustering tasks with different objective functions are designed for the dataset $\mathcal{D}$. Then, multi-populations with the same encoding strategy and skill factors are initialized. Subsequently, offspring is generated through crossover and mutation operators, during which vertical cultural transmission is applied. The knowledge transfer among populations helps individuals to jump out of local optima. An improved vertical cultural transmission is proposed here, see Algorithm 2. After that, the elitist strategy is designed for population maintenance. Finally, the best chromosome of each task is selected and translated into cluster centroids, and time series curves are assigned to the nearest centroids.

In MFEA, the skill factor τ_i is used to split individuals into different skill groups. It denotes one of the tasks, and individuals with the same skill factor τ_i constitute a population. The elitist strategy[38] is employed in each population with the same skill factor $\tau_i = t$ to keep the number of individuals in each population the same. At each generation, the parent population and offspring population with the same skill factor are combined to form a joint population $R_{\tau=t}$. The individuals in the

Algorithm 1 General framework of i-MFEA.

Input: Maximum generation MaxGen, population size N, dataset $\mathcal{D}$, number of clusters k, number of tasks N_T

Output: Clustering results $C = \{C_1, C_2, \dots, C_k\}$, centroids $\{m_1, m_2, \dots, m_l\}$

Pop $\leftarrow$ Initialization (N, N_T);
Design T clustering tasks for dataset $\mathcal{D}$;
Pop $\leftarrow$ AssignSkillFactor (τ_i);
while gen $\leqslant$ MaxGen **do**
  Off $\leftarrow$ Variation (Pop); /* Generate new solutions. */
  while $t \leqslant T$ **do**
    uPop$_t$ $\leftarrow$ MergePop (Pop, Off, t); /* Merge the Population and Offspring with the same skill factor $\tau_i = t$.*/
    φ_τ $\leftarrow$ CalFitness (uPop$_t$); /* Calculate the scalar fitness.*/
    P_{gen} $\leftarrow$ Select (uPop$_t$, m, φ_τ); /* Select m individuals with the highest fitness. */
    Pop = Pop $\cup$ P_{gen};
    t+=1;
  end while
  gen+=1;
end while
while $t \leqslant T$ **do**
  BestSol $\leftarrow$ argmax uPop$_t$;
  Get centroids $\{m_1, m_2, \dots, m_k\}$ from BestSol;
  Assign samples to the nearest centroids;
end while

Algorithm 2 Offspring generation and vertical cultural transmission

Input: Parents P

Output: Offspring O

$O \leftarrow \phi$;
for P_a and P_b in P **do**
  Generate a random number r;
  if parents' skill factor $\tau_{P_a} = \tau_{P_b}$ or $r < 0.3$ **then**
    $C_a, C_b \leftarrow$ Variation (P_a, P_b);
    if skill factor $\tau_{P_a} = \tau_{P_b} = \tau$ **then**
      offspring's skill factor $\tau_{C_a} = \tau_{C_b} = \tau$;
    else
      offspring owns two skill factors, $\tau^1_{C_a} = \tau^1_{C_b} = \tau_{P_a}, \tau^2_{C_a} = \tau^2_{C_b} = \tau_{P_b}$;
    end if
  else
    $C_a \leftarrow$ Mutate (P_a);
    $\tau_{C_a} \leftarrow \tau_{P_a}$;
  end if
  $O \leftarrow O \cup C_a \cup C_b$;
end for

joint population are evaluated based on the objective function $f_{\tau=t}$ of the task $\tau_i = t$ and are ranked together. The factorial rank $r_{\tau=t}$ of individual p denotes the index of p in the list of population members $R_{\tau=t}$ sorted in ascending order with respect to $f_{\tau=t}$. The fitness $\varphi_{\tau=t} = 1/r_{\tau=t}$. Finally, certain individuals with the highest fitness are selected as new parents.

3.2 Multi-tasking construction

Three types of clustering task sets are constructed. By using different objective functions, the landscapes of tasks in each set are different. However, the optimal solutions for these tasks (optimization problems) are consistent. Thus, by co-evolving all tasks with the EMT framework, the search efficiency could be enhanced.

3.2.1 Clustering validity indexes

Three different cluster validity indexes are used as objective functions of three tasks,

$$\begin{cases} \min \text{SSD}, \\ \min \text{DBI}, \\ \min\ (1 - \text{SI}) \end{cases} \tag{12}$$

Euclidean distance is adopted as the similarity measure for all the tasks for fairness. Different objective functions have different landscapes but should in principle result in the same optimal solution.

3.2.2 Distance measures

The distance/similarity measure is essential in high-dimensional time series clustering. Thus, objective functions based on different distance measures are constructed as the optimization objectives of tasks. Considering both the calculation complexity and clustering performance, the SSD measure is selected as the objective function. On the one hand, different tasks can learn from each other to accelerate the optimization process and obtain diverse results. On the other hand, different results reveal the characteristics of time series and help to determine the most suitable distance/similarity measure. Overall, we choose the following four distance calculation methods to measure the SSD: $D_E(x_i, x_j)$, $D_{\text{DTW}}(x_i, x_j)$, $1 - D_{\text{cosine}}(x_i, x_j)$, and $1 - D_{\text{Pearson}}(x_i, x_j)$.

3.2.3 Clustering numbers *k*

As is known, determining an appropriate number of clusters k is a long-standing issue in clustering methods. Through the EMT framework, the MFEA can be applied to find clustering results under different k in a single run. The best k can thus be determined according to the clustering results in an ad-hoc manner.

3.3 Encoding strategy

There are two typical encoding strategies when performing clustering by EAs, i.e., point-based encoding and prototype-based encoding[39]. Although the dimension of the time series is high, the length of dimension m and the number of clusters k are usually much smaller than the number of samples n. Thus, centroid-based encoding (a type of prototype-based encoding) is chosen here, which also enables to obtain typical time series patterns. In centroid-based encoding, each chromosome describes the clustering centroid of the dataset. The length of the chromosomes is $k \times m$ when clustering an m-dimensional dataset into k subsets[18]. Assuming that $k = 3$ and $m = 2$, a chromosome is encoded as $\{0.2, 0.5, 0.3, 0.6, 0.4, 0.1\}$, then the three clustering centroids are $\{(0.2, 0.5), (0.3, 0.6), (0.4, 0.1)\}$. Furthermore, if k is variable, then the length of all the chromosomes should be $m \times k_{\max}$. The first $m \times k$ points will be taken finally and the others will be abandoned. The chromosome is initialized within the range [0, 1]. Then, all the samples are then assigned to the nearest clustering centroids in the same way as the k-means method. Finally, k clusters and their corresponding representative time series patterns are obtained.

3.4 Offspring generation and improved vertical cultural transmission

The simulated binary crossover (namely SBX) and Polynomial Mutation (PM) operators are adopted to generate offspring[38, 40], since the chromosome comprises only real numbers. There are two parameters in each of the operator which determine the probability and magnitude of the operation. The probability parameters, P_c in SBX and P_m in PM, affect the number of points on the chromosome to perform crossover or mutation at each time. The distribution indexes, η_c in SBX and η_m in PM, determine the variability of the offspring from their parents. Specifically, given a large distribution value, offspring solutions are more likely to be close to their parents. Besides, the chromosome of the centroid-based encoding could be long, and the two-point crossover operator which exchanges gene segments between random intersections of two chromosomes is incorporated to further enhance the exploration ability of the algorithm[39].

Vertical cultural transmission is presented in Algorithm 2, which plays the role of implicit genetic transfer among solutions during the offspring generation. Offspring can be categorized into three types based on different associative mating strategies: (1) offspring are generated through crossover and mutation on two parents from the same population, thus both their skill factors are the same as their parents, (2) offspring are generated from one parent through clone and mutation and inherit its skill factor, and (3) offspring are generated through crossover and mutation on two parents with different skill factors with a certain probability, which leads to vertical cultural transmission. In the original MFEA, the skill factor of a child would inherit one of its parents randomly[20]. It is, however, found in the experiment that such random task skill assignment deteriorates the performance of MFEA. To alleviate this issue, i-MFEA is proposed as depicted in Fig. 2. Specifically, traditional MFEA generates solutions and then assign these solutions with a single certain task according to their parents. For i-MFEA, both tasks from parents are assigned to the offspring. The offspring produced in i-MFEA would be accompanied by two skill factors and be evaluated on both tasks.

The offspring should be evaluated concerning all the tasks according to the definition of EMT. However, this may cause large calculations. Therefore, the original MFEA evaluates offspring only on one task. This clearly may eliminate some good offspring. Therefore, i-MFEA makes a trade-off between the computational cost and good solution retention. When the parents come from two different task populations, offspring would be evaluated on both tasks. Only when the parents are from the same population, offspring are evaluated on one task.

4 Experiment

4.1 Experimental setting

4.1.1 Competitor algorithms

To verify the effectiveness of i-MFEA in solving

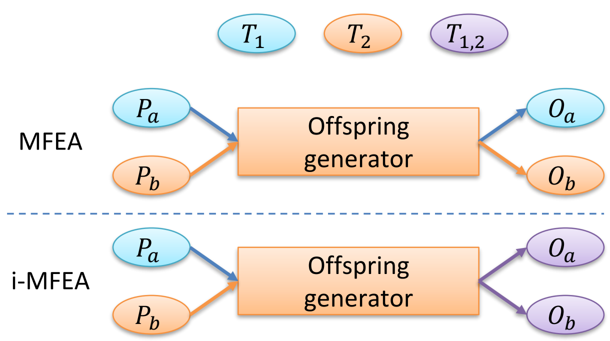


Fig. 2 Illustration of vertical cultural transmission comparison between MFEA and i-MFEA. Notably, $T_{1,2}$ means that the new solutions O_a and O_b are suitable for both Tasks 1 and 2.

time series clustering problems, Single-Objective EA (SOEA), MFEA, and traditional clustering methods are chosen as competitor algorithms. To be specific, we choose a recently proposed genetic algorithm in Ref. [41] to represent the SOEA, in which the algorithm is executed multiple times to obtain all clustering results. The Euclidean distance is used as the similarity measure for fairness. To ensure fair comparison, all algorithms adopt the SBX and PM operators, where $P_c = 1$ and $P_m = 10/n$. For all the algorithms, we set the population size $N = 50$, and the maximum generation $G = 200$. All experiments are implemented based on a PC configured with an Intel i9-9900X @ 3.50 GHz and 64 G RAM. To be statistical, 30 independent runs are conducted for each experiment.

4.1.2 Benchmark

Two simulated artificial time series datasets based on the sine function are generated in the experiment. As described in Table 1 and Fig. 3, dataset DS-S (easily separable) and DS-NS (not easily separable) both contain four obvious clusters with 200 time series data. The length of the sequence is 12, which can be described as follows:

$$X_i = \left[x_i^1, x_i^2, \ldots, x_i^{12}\right], \forall i \in [1, 200] \tag{13}$$

The cluster centroid $m_i = \sin(X_i)$ and n time series data Y_i around the centroid are generated through the following function:

$$\begin{aligned} y_i^j &= \sin(x_i^j \times (1 + r_i)) + p_i^j, \\ Y_i &= \left[y_i^1, y_i^2, \ldots, y_i^{12}\right], \quad i = 1, 2, \ldots, n \end{aligned} \tag{14}$$

where p is a random perturbation with a normal distribution (mean $= 0$ and variance $= \delta_p$) to each data point of time series (at the y-axis), r (mean$= 0$ and variance $=\delta_r$) is a random delay percentage to the sequence X (at the x-axis). Moreover, the $\min - \max$ normalization is conducted before clustering, leading the value to be within [0, 1],

$$y_i^j = \frac{y_i^j - y_i^{\min}}{y_i^{\max} - y_i^{\min}} \tag{15}$$

The difference between dataset DS-S and DS-NS is the values of δ_r and δ_p. By increasing the values of δ_r and δ_p, the bounds of clusters in the dataset would become more indistinct.

4.2 Results

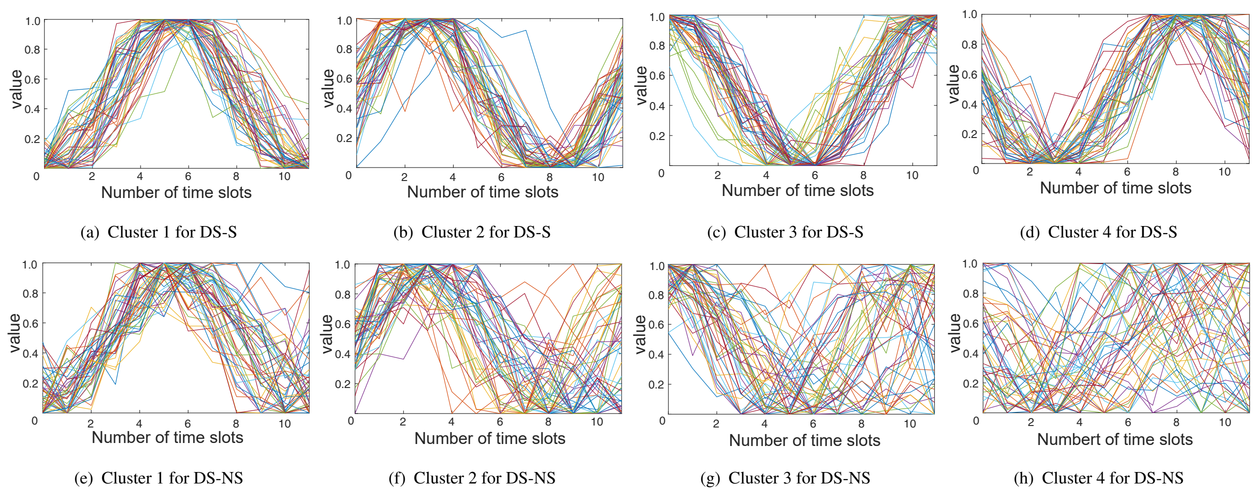
The clustering performance is evaluated from two aspects, i.e., convergence speed and the quality of the results. Based on the true labels, ARI and AMI (mentioned in Section 2.2) are used as partition metrics.

4.2.1 Performance comparison

In this experiment, three optimization tasks are set through objective functions SSD, SH, and DBI. In addition, Fig. 4 shows the convergence trends on each task based on DS-S and DS-NS. Note that each point in the curve is averaged over 30 algorithm runs. It can be observed that all three tasks optimized by i-MFEA, SOEA, and MFEA converge quickly, however, i-MFEA in general performs the best according to the convergence curves and the median results. Table 2 lists the best results obtained by i-MFEA, MFEA, and

Table 1 Overall information of the series datasets used in this study.

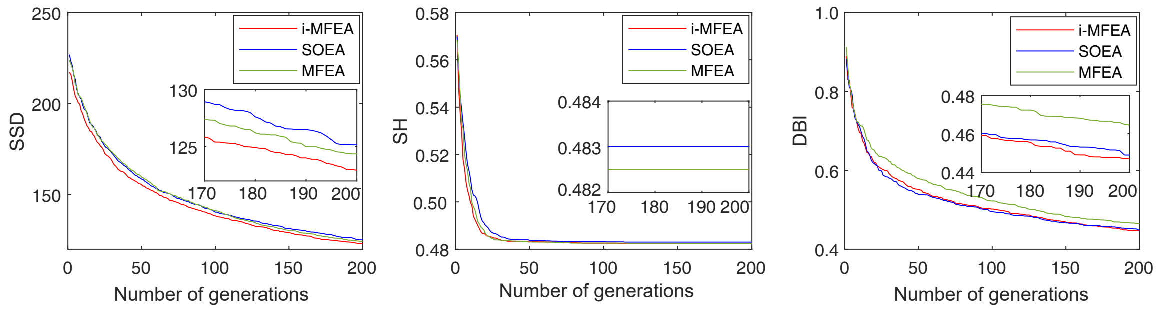
Dataset	Size	Dimension	X	δ_r	δ_p
DS-S	200	12	$X_1 = [-0.5\pi, 1.5\pi]$, $X_2 = [0, 2\pi]$, $X_3 = [0.5\pi, 2.5\pi]$, $X_4 = [\pi, 3\pi]$	0.2	0.1
DS-NS	200	12	$X_1 = [-0.5\pi, 1.5\pi]$, $X_2 = [0, 2\pi]$, $X_3 = [0.5\pi, 2.5\pi]$, $X_4 = [\pi, 3\pi]$	0.3	0.2



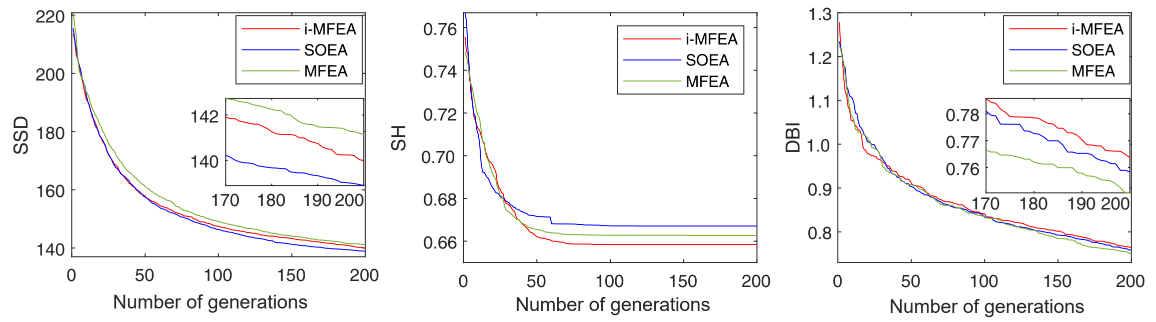
(a) Cluster 1 for DS-S (b) Cluster 2 for DS-S (c) Cluster 3 for DS-S (d) Cluster 4 for DS-S

(e) Cluster 1 for DS-NS (f) Cluster 2 for DS-NS (g) Cluster 3 for DS-NS (h) Cluster 4 for DS-NS

Fig. 3 Distribution of two series datasets used in this study, where different colors are used to distinguish data.



(a) Convergence trends on DS-S dataset



(b) Convergence trends on DS-NS dataset

Fig. 4 Convergence trends of SSD, SH, and DBI averaged across 30 independent runs for i-MFEA, SOEA, and MFEA, respectively.

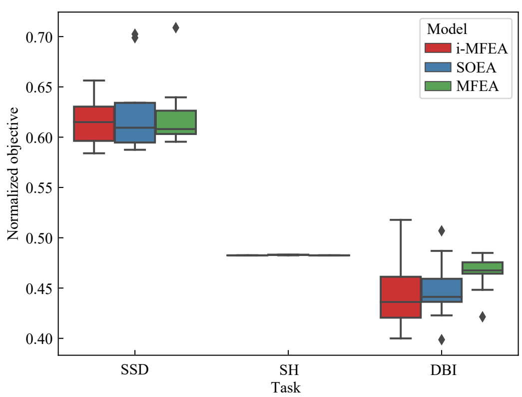
Table 2 Clustering performance of i-MFEA compared to SOEA and MFEA, where the best result is highlighted with bold fonts.

Algorithm	DS-S		DS-NS	
	ARI	AMI	ARI	AMI
i-MFEA	**0.8965**	**0.8734**	0.4059	**0.4307**
SOEA	0.8843	0.8609	**0.4209**	0.4253
MFEA	0.8958	0.8717	0.4036	0.4238

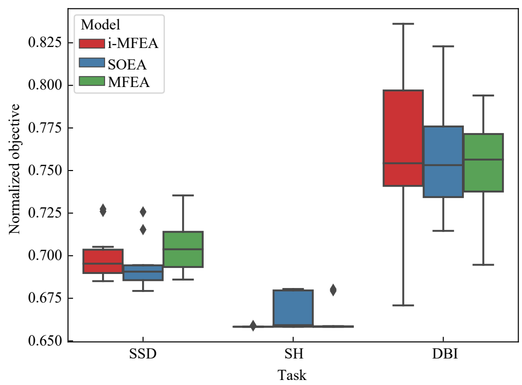
SOEA, which also demonstrate the superiority of i-MFEA. Figure 5 boxplots the optimized objective values of 30 independent runs. It is observed clearly that i-MFEA obtains the best results (minimum value) on most of the tasks for both DS-S and DS-NS datasets. However, the results obtained by the original MFEA are sometimes unstable even if the dataset is separable.

4.2.2 Comparison of i-MFEA with traditional clustering methods

In this section, i-MFEA is compared with three traditional clustering algorithms, i.e., k-means, agglomerative clustering, and DBSCAN. Table 3 shows the results obtained by all the algorithms from which we can observe that i-MFEA outperforms the three traditional clustering algorithms according to the RIA and MIA metrics. Table 3 shows the final clusters of time series curves. It can be observed that DBSCAN performs poorly while other methods achieve acceptable



(a) DS-S



(b) DS-NS

Fig. 5 Boxplots of SSD, SH, and DBI results obtained by i-MFEA, SOEA, and MFEA, respectively.

Table 3 Clustering performance of i-MFEA compared with k-means, agglomerative clustering, and DBSCAN.

Algorithm	DS-S		DS-NS	
	ARI	AMI	ARI	AMI
i-MFEA	**0.8965**	**0.8734**	**0.4059**	**0.4307**
k-means	0.8843	0.8609	0.363	0.393
AGC	0.8568	0.8472	0.4039	0.4171
DBSCAN	0.6186	0.6551	0.099	0.2881

solutions for the datasets. Furthermore, as the structure of the time series dataset is often unknown beforehand, traditional single-objective-based clustering methods require many experiments to determine which validity index (or the number of clusters) to use. The results confirm the advantage of i-MFEA, being capable of finding various clustering results simultaneously.

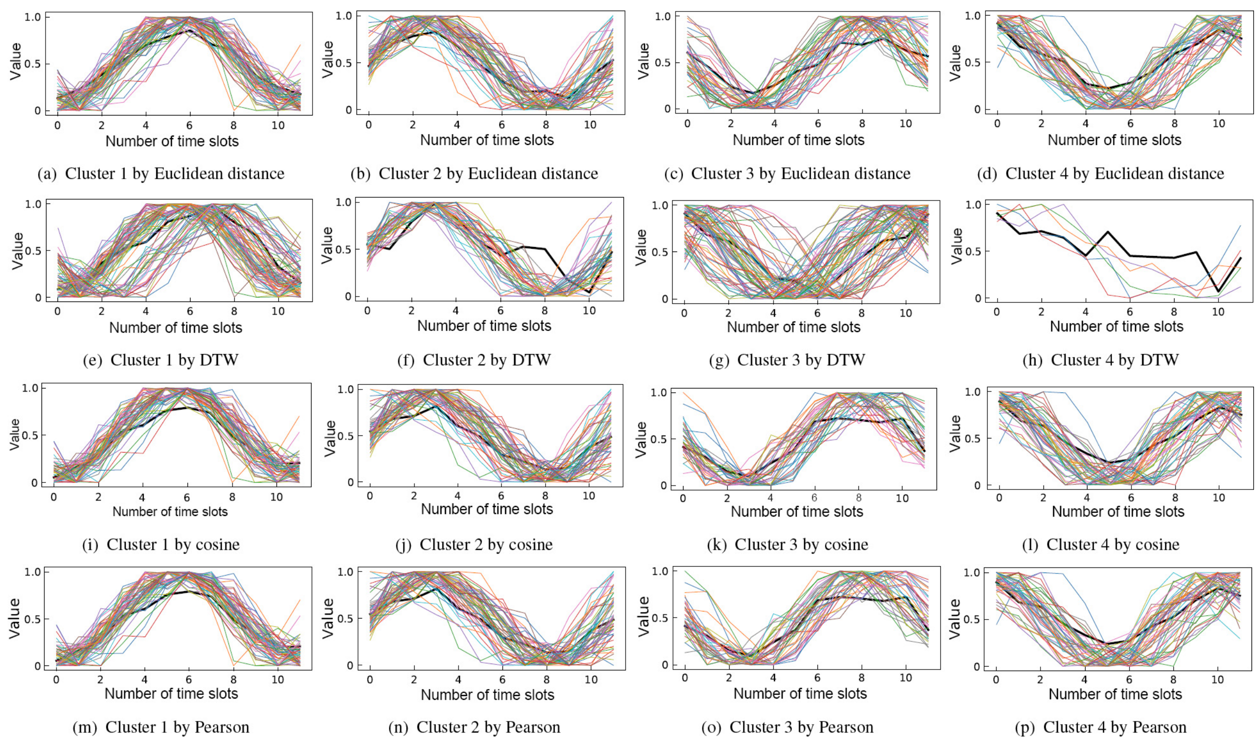
Concerning the clustering centroids, the clustering task with the objective function SSD can obtain representative clustering centroids. Although the cluster centroids obtained by using DBI and SH can be used to separate time series curves, they cannot represent the typical patterns of the clusters. The k-means can obtain smooth clustering centroids. DBSCAN and hierarchical clustering are not centroid-based clustering and thus obtain no clustering centroid.

As for objective functions of the tasks, although clustering tasks with objective function SH have good convergence and stability of solutions as observed in Figs. 4 and 5, the results show that the curves are divided into three distinct clusters rather than four. The clustering tasks with objective function SSD evenly divide the dataset, and the dissimilarity among the clusters is relatively obvious. The performance of the DBI is between the use of SH and SSD.

5 Discussion

5.1 Effect of different distance measures

This section investigates i-MFEA for clustering based on different distance measures. Amongst all tasks with different distance measures, the task built on the DTW distance performs the worst according to the clustering results shown in Fig. 6 and the convergence curves shown in Fig. 7. The reason may be that time series data in the experiment not only have time delays at the x-axis, but also own large perturbations at the y-axis. Pearson correlation coefficient has the best effect and can capture time-varying characteristics. (It can be attributed that the datasets here are generated using the sine function, and the trend of increase and decrease is predominant). The Euclidean distance and the cosine wavelets have medium effects. Overall, from the results, we can conclude that i-MFEA can obtain clustering solutions of time series corresponding to different distance measures



(a) Cluster 1 by Euclidean distance (b) Cluster 2 by Euclidean distance (c) Cluster 3 by Euclidean distance (d) Cluster 4 by Euclidean distance

(e) Cluster 1 by DTW (f) Cluster 2 by DTW (g) Cluster 3 by DTW (h) Cluster 4 by DTW

(i) Cluster 1 by cosine (j) Cluster 2 by cosine (k) Cluster 3 by cosine (l) Cluster 4 by cosine

(m) Cluster 1 by Pearson (n) Cluster 2 by Pearson (o) Cluster 3 by Pearson (p) Cluster 4 by Pearson

Fig. 6 Time series clustering by i-MFEA using different distance measures (Euclidean distance, DTW, cosine, and Pearson) as the objective function, where different colors are used to distinguish data.

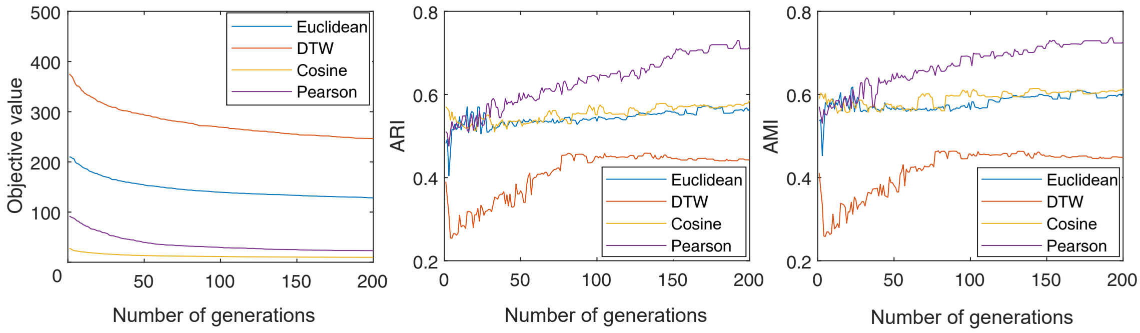


Fig. 7 Convergence trends (averaged across 30 independent runs) of objective functions, ARI, and AMI obtained by i-MFEA based on different similarity measures.

simultaneously. This is helpful for users to capture the features of the dataset.

5.2 Anomaly detection by clustering

i-MFEA can find multiple clustering results under different settings of k in a single algorithm run. The best clustering results (the best k and the centroids) are therefore can be identified through an ad-hoc validation. These results can further be used for anomaly detection. Here, we investigated multi-tasking clustering with different k. The Pearson correlation coefficient based SSD is used as the objective function. The clustering results for DS-S and DS-NS are shown in Table 3.

From the results, we can see that for DS-S, the best k is 4. For DS-NS, though the best k is not evident, decision-makers can choose amongst the clustering results according to their preferences. Moreover, anomaly detection can be achieved when clustering against different k. As the number of clusters k increases, the abnormal data would be detected as a separate cluster for dataset DS-S, while the four well-clustered subsets will not be further divided. For the DS-NS, the artificial abnormal curve and curves that are similar to it are identified and grouped into a separate cluster on the tasks with a large k.

6 Conclusion

Time series data clustering is a long-standing and challenging issue[42, 43]. Since the feature of time series data is complicated and diverse, the dataset structure and clustering objective are fuzzy. EMT can achieve various clustering results under different clustering criteria in a single run, which provides great convenience for decision-makers. Motivated by this, this study investigates time series clustering via the EMT framework. Since there are similarities between tasks, under the EMT framework a better solution would be obtained with the help of the information exchange amongst neighboring solutions (tasks) during the evolution process.

MFEA, an effective method for multi-tasking optimization, however, shows unstable performance for time series clustering. This study, therefore, proposes a novel vertical cultural transmission strategy that can greatly improve the performance of MFEA for time series clustering. To examine the efficiency and effectiveness of the improved MFEA, namely, i-MFEA, several experiments are conducted. The results show that i-MFEA shows better or at least competitive performance compared to the single-objective EA-based clustering method and the three state-of-the-art traditional methods (k-means, agglomerative clustering, and DBSCAN). Moreover, clustering results based on different distance measures, or different number of clusters k can be obtained by i-MFEA in a single run.

For future studies, experiments based on real-world time series datasets would be conducted. In addition, other encoding strategies, e.g., polynomial coding, would be studied for higher-dimensional time series data to make clustering more effective. Moreover, self-paced learning would be considered to improve the robustness of the algorithm[44]. Lastly, it is reported that the optimization of data clustering shows apparently multi-modality[45–48]. Therefore, focusing on obtaining more global optimal solutions by multi-tasking optimization is an interesting topic.

Acknowledgment

This work was supported by the Open Project of Xiangjiang Laboratory (No. 22XJ02003) and the National Natural Science Foundation of China (No. 62122093).

References

[1] S. Aghabozorgi, A. S. Shirkhorshidi, and T. Y. Wah, Time-series clustering–a decade review, *Inf. Syst.*, vol. 53, pp.

16–38, 2015.

[2] B. Yang, Y. Yang, Q. Li, D. Lin, Y. Li, J. Zheng, and Y. Cai, Classification of medical image notes for image labeling by using MinBERT, *Tsinghua Science and Technology*, vol. 28, no. 4, pp. 613–627, 2023.

[3] M. Ahmed, R. Seraj, and S. M. S. Islam, The k-means algorithm: A comprehensive survey and performance evaluation, *Electronics*, vol. 9, no. 8, p. 1295, 2020.

[4] E. Schubert, J. Sander, M. Ester, H. P. Kriegel, and X. Xu, DBSCAN revisited, revisited: Why and how you should (still) use DBSCAN, *ACM Trans. Database Syst.*, vol. 42, no. 3, p. 19, 2017.

[5] P. Contreras and F. Murtagh, Hierarchical clustering, in *Handbook of Cluster Analysis*, C. Hennig, M. Meila, F. Murtagh, and R. Rocci, eds. New York, NY, USA: Chapman and Hall/CRC, 2015, pp. 124–145.

[6] U. Maulik, S. Bandyopadhyay, and A. Mukhopadhyay, *Multiobjective Genetic Algorithms for Clustering*: *Applications in Data Mining and Bioinformatics*. Berlin, Germany: Springer, 2011.

[7] F. Wang, X. Wang, and S. Sun, A reinforcement learning level-based particle swarm optimization algorithm for large-scale optimization, *Inf. Sci.*, vol. 602, pp. 298–312, 2022.

[8] K. Zhu, J. Li, and H. Baoyin, Trajectory optimization of the exploration of asteroids using swarm intelligent algorithms, *Tsinghua Science and Technology*, vol. 14, no. S2, pp. 7–11, 2009.

[9] W. Li, G. Zhang, X. Yang, Z. Tao, and H. Xu, Sizing a hybrid renewable energy system by a coevolutionary multiobjective optimization algorithm, *Complexity*, vol. 2021, p. 8822765, 2021.

[10] L. Li, L. Jiao, J. Zhao, R. Shang, and M. Gong, Quantum-behaved discrete multi-objective particle swarm optimization for complex network clustering, *Pattern Recogn.*, vol. 63, pp. 1–14, 2017.

[11] W. Li, R. Wang, T. Zhang, M. Ming, and K. Li, Reinvestigation of evolutionary many-objective optimization: Focus on the Pareto knee front, *Inf. Sci.*, vol. 522, pp. 193–213, 2020.

[12] U. Maulik and S. Bandyopadhyay, Genetic algorithm-based clustering technique, *Pattern Recogn.*, vol. 33, no. 9, pp. 1455–1465, 2000.

[13] W. Li, T. Zhang, R. Wang, B. Wang, Y. Song, and X. Li, A knee-point driven multi-objective evolutionary algorithm for flexible job shop scheduling, in *Proc. 2019 IEEE Symp. Series on Computational Intelligence*, Xiamen, China, 2019, pp. 1716–1722.

[14] J. Handl and J. Knowles, Evolutionary multiobjective clustering, in *Proc.* 8^{th} *Int. Conf. on Parallel Problem Solving from Nature*, Birmingham, UK, 2004, pp. 1081–1091.

[15] E. Jiang, L. Wang, and J. Wang, Decomposition-based multi-objective optimization for energy-aware distributed hybrid flow shop scheduling with multiprocessor tasks, *Tsinghua Science and Technology*, vol. 26, no. 5, pp. 646–663, 2021.

[16] O. Arbelaitz, I. Gurrutxaga, J. Muguerza, J. M. Pérez, and I. Perona, An extensive comparative study of cluster validity indices, *Pattern Recogn.*, vol. 46, no. 1, pp. 243–256, 2013.

[17] J. Handl and J. Knowles, Multi-objective clustering and cluster validation, in *Multi-Objective Machine Learning*, Y. Jin, ed. Berlin, Germany: Springer, 2006, pp. 21–47.

[18] R. Wang, S. Lai, G. Wu, L. Xing, L. Wang, and H. Ishibuchi, Multi-clustering via evolutionary multi-objective optimization, *Inf. Sci.*, vol. 450, pp. 128–140, 2018.

[19] W. Li, T. Zhang, R. Wang, S. Huang, and J. Liang, Multimodal multi-objective optimization: Comparative study of the state-of-the-art, *Swarm Evol. Comput.*, vol. 77, p. 101253, 2023.

[20] A. Gupta, Y. S. Ong, and L. Feng, Multifactorial evolution: Toward evolutionary multitasking, *IEEE Trans. Evol. Comput.*, vol. 20, no. 3, pp. 343–357, 2016.

[21] W. Li, R. Wang, T. Zhang, M. Ming, and H. Lei, Multi-scenario microgrid optimization using an evolutionary multi-objective algorithm, *Swarm Evol. Comput.*, vol. 50, p. 100570, 2019.

[22] Y. Liu, T. Özyer, R. Alhajj, and K. Barker, Integrating multi-objective genetic algorithm and validity analysis for locating and ranking alternative clustering, *Informatica*, vol. 29, no. 1, pp. 33–40, 2005.

[23] M. Kim and R. S. Ramakrishna, New indices for cluster validity assessment, *Pattern Recogn. Lett.*, vol. 26, no. 15, pp. 2353–2363, 2005.

[24] G. Gan, C. Ma, and J. Wu, *Data Clustering*: *Theory*, *Algorithms*, *and Applications*, 2nd ed. Alexandria, VA, USA: SIAM, 2007.

[25] H. Li, X. Wu, X. Wan, and W. Lin, Time series clustering via matrix profile and community detection, *Adv. Eng. Inf.*, vol. 54, p. 101771, 2022.

[26] A. Blázquez-García, A. Conde, U. Mori, and J. A. Lozano, A review on outlier/anomaly detection in time series data, *ACM Comput. Surv.*, vol. 54, no. 3, p. 56, 2021.

[27] S. Salvador and P. Chan, Toward accurate dynamic time warping in linear time and space, *Intelligent Data Analysis*, vol. 11, no. 5, pp. 561–580, 2007.

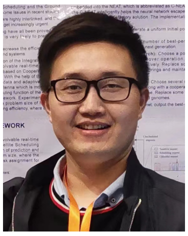
[28] J. Hu, Y. Pan, T. Li, and Y. Yang, Tw-co-MFC: Two-level weighted collaborative fuzzy clustering based on maximum entropy for multi-view data, *Tsinghua Science and Technology*, vol. 26, no. 2, pp. 185–198, 2021.

[29] R. A. Armstrong, Should Pearson's correlation coefficient be avoided? *Ophthalmic Physiol. Opt.*, vol. 39, no. 5, pp. 316–327, 2019.

[30] Y. Zhang, Y. Li, J. Song, X. Chen, Y. Lu, and W. Wang, Pearson correlation coefficient of current derivatives based pilot protection scheme for long-distance LCC-HVDC transmission lines, *Int. J. Electr. Power Energy Syst.*, vol. 116, p. 105526, 2020.

[31] S. Liu, Q. Lin, L. Feng, K. C. Wong, and K. C. Tan, Evolutionary multitasking for large-scale multiobjective optimization, *IEEE Trans. Evol. Comput.*, doi: 10.1109/TEVC.2022.3166482.

[32] E. Osaba, J. Del Ser, A. D. Martinez, and A. Hussain, Evolutionary multitask optimization: A methodological overview, challenges, and future research directions, *Cogn. Comput.*, vol. 14, no. 3, pp. 927–954, 2022.
[33] K. K. Bali, Y. S. Ong, A. Gupta, and P. S. Tan, Multifactorial evolutionary algorithm with online transfer parameter estimation: MFEA-II, *IEEE Trans. Evol. Comput.*, vol. 24, no. 1, pp. 69–83, 2020.
[34] P. T. H. Hanh, P. D. Thanh, and H. T. T. Binh, Evolutionary algorithm and multifactorial evolutionary algorithm on clustered shortest-path tree problem, *Inf. Sci.*, vol. 553, pp. 280–304, 2021.
[35] A. Gupta, Y. S. Ong, L. Feng, and K. C. Tan, Multiobjective multifactorial optimization in evolutionary multitasking, *IEEE Trans. Cybern.*, vol. 47, no. 7, pp. 1652–1665, 2017.
[36] Y. S. Ong and A. Gupta, Evolutionary multitasking: A computer science view of cognitive multitasking, *Cogn. Comput.*, vol. 8, no. 2, pp. 125–142, 2016.
[37] Y. Yuan, Y. S. Ong, A. Gupta, P. S. Tan, and H. Xu, Evolutionary multitasking in permutation-based combinatorial optimization problems: Realization with TSP, QAP, LOP, and JSP, in *Proc. 2016 IEEE Region 10 Conf.*, Singapore, 2016, pp. 3157–3164.
[38] L. Zhou, L. Feng, J. Zhong, Y. S. Ong, Z. Zhu, and E. Sha, Evolutionary multitasking in combinatorial search spaces: A case study in capacitated vehicle routing problem, in *Proc. 2016 IEEE Symp. Series on Computational Intelligence*, Athens, Greece, 2016, pp. 1–8.
[39] K. Deb, A. Pratap, S. Agarwal, and T. Meyarivan, A fast and elitist multiobjective genetic algorithm: NSGA-II, *IEEE Trans. Evol. Comput.*, vol. 6, no. 2, pp. 182–197, 2002.
[40] A. Mukhopadhyay, U. Maulik, and S. Bandyopadhyay, A survey of multiobjective evolutionary clustering, *ACM Comput. Surv.*, vol. 47, no. 4, p. 61, 2015.
[41] W. Gong, Z. Cai, and D. Liang, Adaptive ranking mutation operator based differential evolution for constrained optimization, *IEEE Trans. Cybern.*, vol. 45, no. 4, pp. 716–727, 2015.
[42] J. Zhang and L. Xing, An improved genetic algorithm for the integrated satellite imaging and data transmission scheduling problem, *Comput. Oper. Res.*, vol. 139, p. 105626, 2022.
[43] R. Lu, H. Shen, Z. Feng, H. Li, W. Zhao, and X. Li, HTDeT: A clustering method using information entropy for hardware Trojan detection, *Tsinghua Science and Technology*, vol. 26, no. 1, pp. 48–61, 2021.
[44] X. Li and H. Liu, Greedy optimization for K-means-based consensus clustering, *Tsinghua Science and Technology*, vol. 23, no. 2, pp. 184–194, 2018.
[45] X. Chang, D. Tao, and X. Chao, Multi-view self-paced learning for clustering, in *Proc. 24th Int. Conf. on Artificial Intelligence*, Buenos Aires, Argentina, 2015, pp. 3974–3980.
[46] W. Li, T. Zhang, R. Wang, and H. Ishibuchi, Weighted indicator-based evolutionary algorithm for multimodal multiobjective optimization, *IEEE Trans. Evol. Comput.*, vol. 25, no. 6, pp. 1064–1078, 2021.
[47] W. Li, X. Yao, T. Zhang, R. Wang, and L. Wang, Hierarchy ranking method for multimodal multiobjective optimization with local Pareto fronts, *IEEE Trans. Evol. Comput.*, vol. 27, no. 1, pp. 98–110, 2023.
[48] X. Yao, W. Li, X. Pan, and R. Wang, Multimodal multi-objective evolutionary algorithm for multiple path planning, *Comput. Ind. Eng.*, vol. 169, p. 108145, 2022.



Rui Wang received the BEng degree from National University of Defense Technology (NUDT), China in 2008, and the PhD degree from the University of Sheffield, UK in 2013. He is working currently at National University of Defense Technology, China. His current research interests include multi-objective optimization and machine learning methods. He has authored more than 40 referred papers including those published in *IEEE Transactions on Evolutionary Computation* and *IEEE Transactions on Cybernetics*.

He serves as an associate editor of the *IEEE Trans. on Evolutionary Computation*, *Swarm and Evolutionary Computation*, *Expert System with Applications*, etc. He is the recipient of the Operational Research Society PhD Prize in 2014, the Funds for Distinguished Young Scientists from the Natural Science Foundation of Hunan province in 2016, the Wu Wen-Jun Artificial Intelligence Outstanding Young Scholar in 2017, and the National Science Fund for Outstanding Young Scholars in 2021.

Wenhua Li received the BEng and MEng degrees from NUDT, China in 2018 and 2020, respectively. He is now a PhD candidate in management science and technology at NUDT, China. He has authored more than 20 papers, including those published in *IEEE Transactions on Evolutionary Computation* and *Swarm and Evolutionary Computation*. His current research interests include multi-objective evolutionary algorithms, energy management in microgrids, and artificial intelligence.

Kaili Shen received the MEng degree from NUDT, China in 2020. She is now a software engineer at Ant Group Co., Ltd., China. Her current research interests include data science and clustering methods.

Tao Zhang received the BEng, MEng, and PhD degrees from NUDT, China in 1998, 2001, and 2004, respectively. He is currently a full professor at College of Systems Engineering, NUDT, China. He is also the director of the Hunan Key Laboratory of Multi-energy System Intelligent Interconnection Technology (HKL-MSI2T). His current research interests include optimal scheduling, data mining, and optimization methods on the energy internet. He is the recipient of the Science and Technology Award of Provincial Level (first places in 2020 and 2021, and second places in 2015 and 2018).

Xiangke Liao received the BEng degree from Tsinghua University, China in 1985, and the MEng degree from NUDT, China in 1988. He is currently a full professor at College of Computer Science and Technology, NUDT. He is the principle investigator and chief designer of Tianhe-2 supercomputer. His current research interests include high performance computer system software and general-purpose operating system.