



---

# AN APPROACH FOR HEURISTIC PARALLEL LDTW DISTANCE OPTIMIZATION METHOD WITH BIO-INSPIRED STRATEGY

*J. Dai*<sup>\*</sup>, *Y.H. He*<sup>†</sup>, *J.Y. Li*<sup>\*</sup>

---

**Abstract:** Dynamic time warping (DTW) is a classical similarity measure for arbitrary length time series. As an effective improvement of DTW, dynamic time warping under limited warping path length (LDTW) oppresses the long-term pathological alignment problem and allows better flexibility. However, since LDTW increases path lengths, it directly leads to higher time-consuming. In this paper, a new method of similarity sequence measurement (ACO\_LDTW) is proposed by the parallel computing characteristics of ant colony optimization (ACO) algorithm with bio-inspired strategy and the idea of LDTW path restriction. This algorithm searches the optimal path on the restricted distance matrix by simulating the behavior of ant colony parallel foraging. Firstly, the distance matrix is mapped to the 0 – 1 matrix of grid method, and the search range of ants is limited by the warping path in LDTW. Secondly, the state transition probability function, pheromone initialization and update mechanism of ACO algorithm are adapted. On the basis of ensuring the accurate results, the convergence rate can be effectively improved. The validity of ACO\_LDTW is verified by cases. In the 22 data sets of 1NN classification experiment, ACO\_LDTW has the lowest classification error rate in 16 data sets, and it is shorter than the calculation time of LDTW. At the same time, it is applied to the field of mechanical fault diagnosis and has the ability to solve practical engineering applications. Experiments show that ACO\_LDTW is more effective in terms of accuracy and computation time.

Key words: *similarity measure, dynamic time warping (DTW), dynamic time warping under limited warping path length (LDTW), ant colony optimization algorithm (ACO)*

*Received: November 7, 2019*

**DOI:** 10.14311/NNW.2021.31.001

*Revised and accepted: February 28, 2021*

## 1. Introduction

Similarity measurement is an important method to evaluate the approximation between objects. The more consistent objects are in attributes and status, the

---

<sup>\*</sup>Jin Dai; Jiayao Li – Corresponding author; School of Software Engineering, Chongqing University of Posts and Telecommunications, Chongqing, China, E-mail: [daijin@cqupt.edu.cn](mailto:daijin@cqupt.edu.cn), [s181231014@stu.cqupt.edu.cn](mailto:s181231014@stu.cqupt.edu.cn)

<sup>†</sup>Yuhong He; Artificial Intelligence and Big Data College, Chongqing College of Electronic Engineering, Chongqing, China, E-mail: [s161201001@stu.cqupt.edu.cn](mailto:s161201001@stu.cqupt.edu.cn)

greater the similarity is. Moreover, similarity measurement is the core issue of data mining, which combines with mining algorithm to complete different mining tasks, including classification, estimation, prediction, association rules, clustering and so on [1,2]. In literature [3], a sequence clustering algorithm based on weighted sequence pattern similarity is proposed. This method makes the updated center more suitable for the real cluster center, so the clustering quality is improved. Literature [4] proposes a time series regression prediction method based on network links. A large number of studies have shown that the correct measurement of sequence similarity is the key to improve the efficiency and effectiveness of data mining.

Classical similarity measures can be divided into two categories: lock-step measures and elasticity measures. Lock-step measures are one-to-one comparisons of sequences, and the most common one is the Euclidean distance. However, due to the complexity of data, these measurement methods have some limitations. Elastic measures are one-to-many comparisons of sequences. As a very common measure of elasticity, dynamic time warping (DTW) allows points in a sequence to be matched with equal length after self-replication, which overcomes the problem that lock-step measures cannot be matched due to distortion of the sequence [5]. At present, the application of DTW has developed from speech recognition to many other fields, such as bioinformatics [4–6], online signature verification [7–9] and gesture recognition [10–12].

Nowadays, as the number of data increases, the calculation time of the DTW will also increase. At the same time, there has pathological alignment, that is, one point in one sequence maps to the majority of points in another sequence, resulting in a decrease in its accuracy. Many scholars have improved their accuracy and performance. There are many classical improved algorithms, which can be divided into two categories: (1) Consider the trend or shape characteristics of sequences. For example, derivative DTW (DDTW) [13] obtains shape information by considering the first derivative of the sequence, thereby reducing the occurrence of singularities, but does not consider the time and space complexity of the algorithm. (2) Consider sequence path constraints. Fast DTW (FDTW) [14] considers its high time complexity, and effectively deletes a large number of search candidates, thus directly reduce the search cost. But using approximate methods, estimating the time warping distance by using several fragments to represent a rough version of a sequence, it is possible to miss the correct alignment. Quantitative DTW [15, 16] uses the concept of the code word. In speech recognition, it no longer stores multiple templates of the same word, but stores a reference model for each word. Comparing the unknown speech with the center of the reference model, the speed of recognition is improved, but the recognition rate inevitably decreases. Flexible DTW [17] gives extra points to reward one-to-one fragments, avoiding alignment of adjacent points in one sequence with distant points in another. However, due to additional rewards, the calculation time is increased. At present, the applications of DTW and its improved methods have expanded from speech recognition to many fields. For example, in order to solve the problem of inaccurate handwriting recognition on mobile terminals, a dynamic handwriting recognition algorithm based on DTW was proposed by Lei H. W et al [16]. DING J et al. [17] proposed a design scheme based on joint position and joint angle, which can derive appro-

priate size feature vectors for each gesture category of DTW reference template in the database according to the activity characteristics of specific types of gestures. Experiments show that the method using determined features is superior to the traditional method what only considering joint position information.

Dynamic time warping under limited warping path length (LDTW) [18] is a recently proposed method to improve DTW and proposes a new constraint on the warping path. It limits the total number of links during the optimization process of DTW. And LDTW not only oppresses the pathological alignment effectively but also allows more flexibilities when measuring similarities. But LDTW is a recursive calculation, in which an additional dimension (the current length of the warping path) is used in the cumulative cost matrix to record the current number of steps, resulting in a significant increase in computation time. Aiming at this problem, it can be solved by parallel computing. As an effective parallel computing method, ant colony optimization algorithm (ACO) [19] can be introduced into the calculation of similarity between sequences. Therefore, ACO algorithm is utilized to replace the recursive calculation of LDTW, to search for the optimal path in the cumulative cost matrix.

In this paper, an approach for heuristic parallel LDTW distance optimization method with bio-inspired strategy (ACO-LDTW) is proposed. At the same time, in order to make this method faster and more accurate, the state transition probability function, pheromone initialization and update mechanism of ant colony algorithm are improved. The validity of the method is fully illustrated in the case. Through the experiment of 1NN classification of UCR data set, the time is greatly shortened when the accuracy is approximately equal to that of LDTW. Moreover, the algorithm is applied to the field of mechanical fault diagnosis.

## 2. Materials and methods

### 2.1 DTW

In the 1990s, Berndt and Clifford introduced DTW [20] into time-series data mining. By constructing the alignment matrix of the two sequences (Fig. 1(a)), a shortest warping path can be found and matched in the alignment matrix for alignment matching. The shortest path is used as the result of the two-sequence distance metric. The local optimization method is used to obtain the optimal path and simplify the complexity of the process. Consequently, this method can also effectively deal with the similarity measure among the incomplete, abnormal or abrupt data sequences.

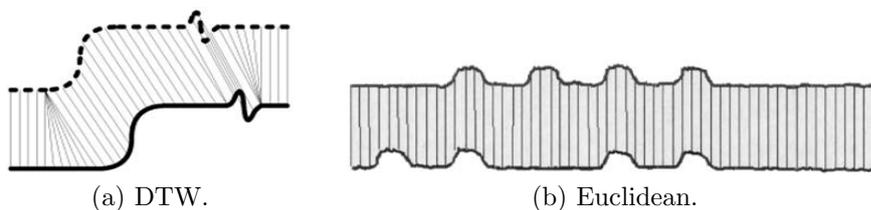


Fig. 1 DTW and Euclidean distance [21].

**Definition DTW** Suppose two sequences  $X = \{x_1, x_2, \dots, x_m, \dots, x_M\}$  and  $Y = \{y_1, y_2, \dots, y_n, \dots, y_N\}$ , the distance matrix between  $X$  and  $Y$  is defined as follows:

$$DM(X, Y) = \begin{bmatrix} dist(x_M, y_1) & \cdots & dist(x_M, y_N) \\ \vdots & \ddots & \vdots \\ dist(x_1, y_1) & \cdots & dist(x_1, y_N) \end{bmatrix}, \quad (1)$$

where  $dist(x_m, y_n)$  is the distance between sequence elements  $x_m$  and  $y_n$ . In this study,  $dist(x_m, y_n) = |x_m - y_n|$ . According to the distance matrix, the DTW distance between the sequences can be defined as follows:

$$D(X_m, Y_n) = dist(x_m, y_n) + \min \left\{ \begin{array}{l} D(x_{m-1}, y_n) \\ D(x_m, y_{n-1}) \\ D(x_{m-1}, y_{n-1}) \end{array} \right\}. \quad (2)$$

In fact, the warping path is generated by certain rules, and different rules result in different distances measurement. From this perspective, Euclidean distance can also be seen as based on the alignment of two sequences, with its own rules for curved paths. As shown in Fig. 1(b), Euclidean distance is strictly one-to-one aligned, and the warping paths consist only of diagonal elements of the alignment matrix.

Set the warping path  $W$ , and the  $k$  element of  $W$  is defined as  $w_k = (m, n)$ , which defines the mapping of sequence  $X$  and  $Y$ . According to Eq. (2), the warping path of DTW must satisfy the following three rules [20]:

1. Boundary constraint:

The starting point of the warping path  $W$  must be  $w_1 = (1, 1)$  and end point  $w_k = (m, n)$ .

2. Monotonicity constraint:

Given  $w_k = (m, n)$ , the next point  $w_{k+1} = (m', n')$  needs to satisfy  $(m' - m) \geq 0$  and  $(n' - n) \geq 0$ . The point is limited in the warping path  $W$  which must be monotonic with time.

3. Continuity constraint:

Given  $w_k = (m, n)$ , the next point  $w_{k+1} = (m', n')$  needs to satisfy  $(m' - m) \geq 1$  and  $(n' - n) \geq 1$ . It cannot cross a point to match, only aligning with its adjacent points. It is to ensure that every point in the sequence  $X$  and  $Y$  appears in the warping path  $W$ .

In DTW, dynamic programming is designed to search the shortest distance. It is a multi-stage decision-making process. The local shortest distance in the process of solving can be regarded as several interrelated stages, then each stage needs to make a decision (depend on the distance  $dist(x, y)$  of the current two data points and the minimum of the local optimal solution in the previous stage), so as to achieve the best activity effect (the shortest distance) in the whole process. Generally speaking, the decision-making in each stage has a sequence. The decision-making depends

on the current state, and then causes the status to transfer. A decision-making sequence is produced in the changing state, so it has the meaning of “dynamic”. Therefore, the solution sought by DTW is a dynamic process in which the global solution is achieved through the superposition of several local optima.

## 2.2 LDTW

However, there are some faults in DTW, such as pathological alignment, as shown in Fig. 2, which illustrates a typical example of pathological alignment generated by DTW, and several singularities (the red triangle in Fig. 2) can be found in the graph. The singularity can be understood as a data point in one time series that connects the large subsection of other time series. Obviously, the shortcoming of pathological alignment (pathological matching) has a great impact on the accuracy of similarity measurement. The restriction of LDTW [18] on the total number of links in DTW optimization can effectively avoid the generation of singularity. Unlike rigid constraints such as window constraints and the feature DTW [22–25], LDTW can determine the number of links assigned to each data point and where to place the links, allowing greater flexibility and thus reducing pathological alignment risk.

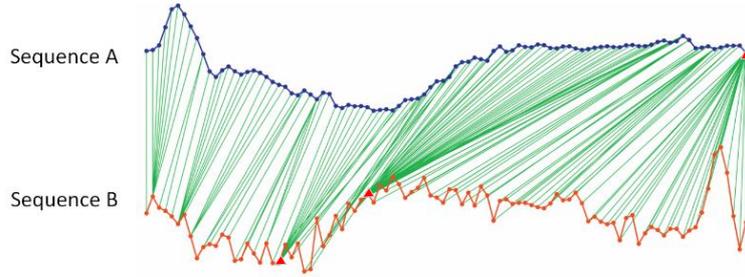


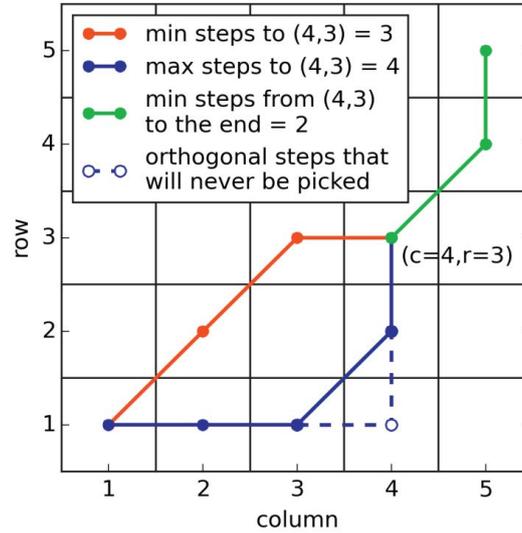
Fig. 2 A typical example of pathological alignment generated by DTW [18].

**Definition LDTW** In LDTW method, the constraint of the total link number is used to limit the warping path length. To describe the method more intuitively, we use the term “step count” (path length  $-1$ , denoted by  $s$ ). Suppose two sequences  $X = \{x_1, x_2, \dots, x_r, \dots, x_R\}$  and  $Y = \{y_1, y_2, \dots, y_c, \dots, y_C\}$ , the LDTW distance between the sequences can be defined as follows:

$$D(X_r, Y_c, s) = \text{dist}(x_r, y_c) + \min \left\{ \begin{array}{l} D(X_r, Y_{c-1}, s-1) \\ D(X_{r-1}, Y_c, s-1) \\ D(X_{r-1}, Y_{c-1}, s-1) \end{array} \right\}. \quad (3)$$

The minimum number of steps is clearly  $r - 1$  or  $c - 1$ , so the minimum step is  $\max(r, c) - 1$ . If the cell is in the bottom row  $r = 1$  ( $r$  is the row index of the cumulative cost matrix) or the left-most column  $c = 1$  ( $c$  is the column index of the cumulative cost matrix), then the maximum step is also equal to the larger one between  $r-1$  and  $c-1$ . Elsewhere, the maximum step is equal to  $(c-2)+1+(r-2) =$

$r + c - 3$ . As shown in Fig. 3, we can see clearly that the shortest path to (4, 3) by red line is 3 and the longest path to (4, 3) by blue line is 4.



**Fig. 3** An example of maximum and minimum steps to reach a certain cell by LDTW [18].

The only parameter of LDTW is the upper bound of warping path length  $L_{UB}$ . The effective range of  $L_{UB}$  is finite according to the rules of DTW. If the lengths of the two time series are  $R$  and  $C$ , then  $\max(R, C) + 1 \leq L_{UB} \leq R + C - 2$ . In our experiment, taking the 50 words dataset as an example, the length of the time series length is 270, so the search space of  $L_{UB}$  is  $[271, 538(270 + 270 - 2)]$ . Paper [18] learns an appropriate  $L_{UB}$  on the training set through LOOCV (leave one method for cross validation). During LOOCV, each candidate results in a 1NN classification error count. When there is only one candidate with the least number of errors, it is  $L_{UB}$ . If the least number of errors is an area, the median of the optimal zone is chosen as the optimal result  $L_{UB}$ . It can be found that LDTW not only limits the length of the warping path, but also keeps the flexibility of DTW.

Furthermore, each cell retains the only value in the cumulative cost matrix of DTW, which is the global minimum cumulative cost between all possible path lengths. In LDTW conversely, each cell must hold multiple values, each of which is the minimum cumulative cost at the corresponding warping path length. Therefore, its solution is to calculate the results of all warping paths and then select the smallest of these results, but at the same time, it requires a higher computational cost.

### 2.3 ACO

The method of grid map has the characteristics of small computation and easy realization. It subdivides the search area into several cells of equal size, so that the complexed problems of environment can be broken down into small problems that

can be easily solved. As a new bio-inspired algorithm, the ant colony algorithm has better parallelism, positive feedback and robustness. It's inspired by the idea of ants foraging for food. Ants will secrete the chemicals called pheromones during their moving, and the behavior of the ant colony is regulated by this chemical because they can sense the concentration of this substance in the environment and they prefer to move in the direction of higher concentration, thus dominating the whole colony.

The ACO algorithm based on the grid map uses the 0-1 matrix simulation environment ("0" represents the free grids, and "1" represents the obstacle grids). More specifically, (1) the algorithm is required to map the distance matrix into the 0-1 matrix of the grid. The distance value between the corresponding points for each grid is marked. (2) In order to ensure that the steps of searching for the shortest path do not exceed the range of the corresponding steps. We restrict the search area of ACO, where the grids are set to "1" as the obstacle grids, the search area can be limited to a hexagonal region. This not only satisfies the steps limit of LDTW but also greatly reduces the search range of ants, making the search efficiency improved. (3) In the ACO algorithm, the direction of ants walking is not constrained, but each step in LDTW can only follow three directions (as shown in Fig. 4). Therefore, three restriction rules in LDTW are introduced to restrict the area where ants can walk in three directions at each step, and the convergence speed is also accelerated.

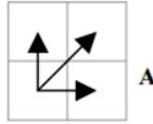
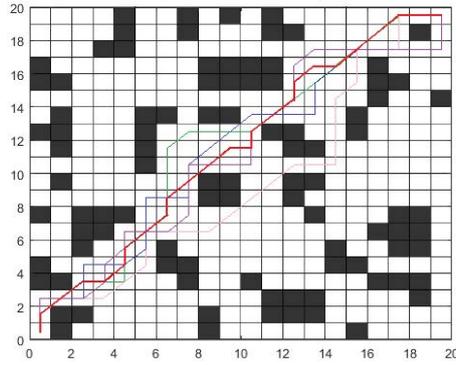


Fig. 4 The direction of each step.

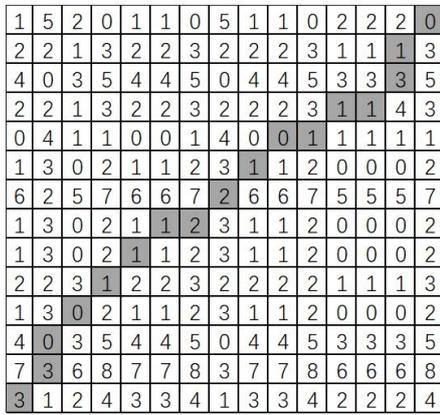
### 3. The proposed method

#### 3.1 Analysis

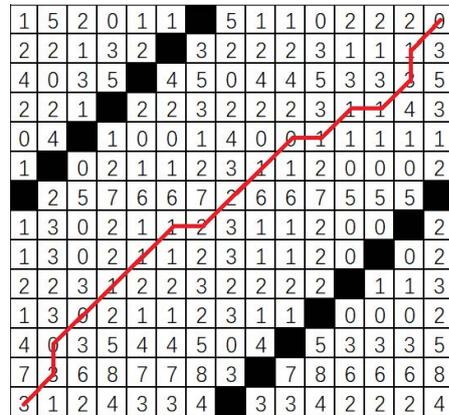
Based on related works, Eq. (3) can be seen that the current step count is introduced to limit the warping path. Each cell must contain multiple values, and each value is the minimum distance cumulative cost within the step range. This results in a large time cost ( $O(L_{UB} \cdot R \cdot C)$ , where  $R$  is the length of the sequence  $X$ ;  $C$  is the length of the sequence  $Y$ ) for calculating LDTW. As shown in Fig. 5(a, b), the optimal path of LDTW is similar to the optimal path of ACO in the grid map [26]. The former searches for the optimal path (the sum of unit values) in the distance matrix (the corresponding value of each unit); the latter searches for the optimal path (the shortest distance between the starting point and the ending point) in the two-dimensional environment (0-1 matrix simulation environment) simulated by the grid map. Because LDTW adopts the recursive linear idea, the computation time is too much. However, ACO has the characteristics of concurrent computing, information positive feedback and random search, which can contribute to reducing the computation time.



(a) ACO



(b) LDTW



(c) ACO.LDTW

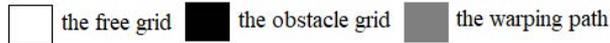


Fig. 5 The optimal path of the three algorithms.

In general, ACO has certain advantages in computational efficiency and has been widely used in mobile robot obstacle avoidance [27]. Therefore, this paper considers using the search mechanism of ACO to replace the computer mechanism of LDTW, and proposes a novel similarity measure ACO.LDTW, to improve the defect of excessive computation time.

In summary, ACO.LDTW consists of two modules, ACO module and LDTW algorithm module. Inspired by the autonomous foraging behavior of ants, ACO.LDTW uses ant colony optimization to obtain the shortest warping path between sequences, which has the characteristics of heuristic and parallelization. In this algorithm, the black barrier grid in the grid map is set up to restrict the ant search path and induces the ant to find the optimal path under the pheromone's positive feedback strategy, so as to choose the optimal solution. The problem of

high computation time cost of LDTW has been effectively solved in this algorithm, which greatly reduces the computational overhead on the premise of ensuring the accuracy of ACO\_LDTW calculation.

### 3.2 ACO\_LDTW algorithm

Based on the above analysis, ACO\_LDTW algorithm is proposed. Fig. 6 illustrates the technical route of ACO\_LDTW.

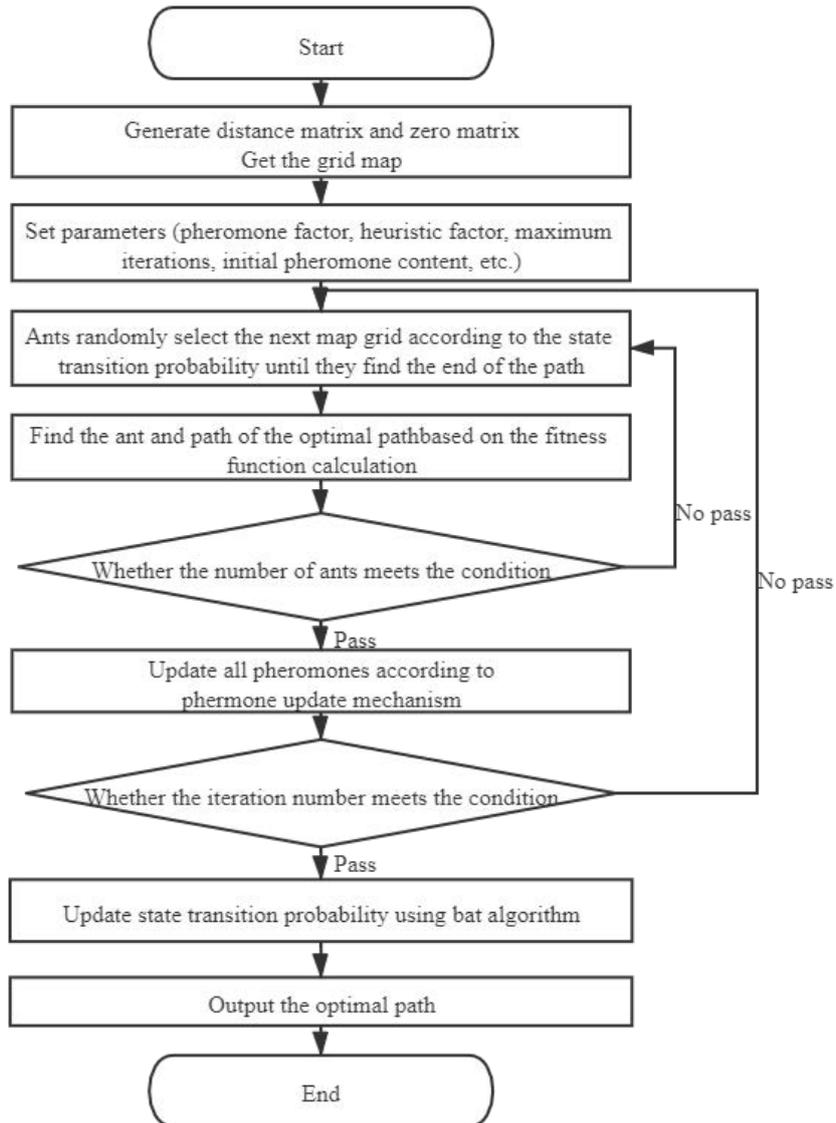


Fig. 6 The technical route of ACO\_LDTW.

Fitness function, state transition probability, limitation of the search area and pheromone initialization and update mechanism are involved in the process of the algorithm. ACO\_LDTW calculates distance matrix between sequences, and generates zero matrix, and obtains corresponding grid map. After setting the initial parameters, the ant colony calculates the optimal path through fitness function and state transition probability, then updates pheromone and iterates several returned results.

### 3.2.1 State transition probability

Furthermore, considering the three constraint rules of LDTW, we limit the direction of each ant walking area. As shown in Fig. 4, each ant can only select the next access point in three grids at most. Traditional ACO algorithm usually uses the following state transition probability to select the next access point [27]:

$$p_{ij}^k(t) = \begin{cases} \frac{[\tau_{ij}(t)]^\alpha \cdot [\eta_{ij}(t)]^\beta}{\sum_{s \in \text{allowed}_k} [\tau_{is}(t)]^\alpha \cdot [\eta_{is}(t)]^\beta}, & j \in \text{allowed}_k \\ 0, & \text{otherwise} \end{cases}, \quad (4)$$

where  $p_{ij}^k(t)$  is the probability of the ant  $k$  going to  $(i, j)$  in the  $t$  iteration;  $\text{allowed}_k$  is a set of three grids that ants can choose in the next step.  $\tau_{ij}(t)$  is the pheromone, and  $\eta_{ij}(t)$  is the heuristic factor. In this algorithm, the factor is the reciprocal of the distance between the corresponding points in the two sequences. The  $\alpha$  and  $\beta$  are constants. The former is the weighted value of pheromone factor and the latter is the weighted value of heuristic factor. In this research, the state transition probability of the original ACO algorithm is improved to the dynamic state transition probability. LDTW searches for the optimal path according to the minimum of the three adjacent grids. If only the original state transition probability is used, the optimal path cannot be found effectively. Therefore, by generating a 0-1 uniform random number  $\text{rand}()$  and a parameter  $p$ , when  $\text{rand}() > p$ , the next access point is selected with the original state transition probability from Eq. (8). Otherwise, the next access point is selected with a smaller value in the accessible grid. The parameter  $p$  increases with time, and this paper mainly adopts the incremental operator in bat algorithm [28]:

$$p^{t+1} = p^0 [1 - \exp(-\gamma t)], \quad (5)$$

where here  $\gamma$  is a constant and takes 0.9;  $p^0$  is usually limited to the range  $[0, 1]$ . It should be noted that with the passing of computing time, the parameter  $p$  will get closer to  $p^0$ . In the early stage, the cells are selected mainly according to the original state transition probability, which ensures that the previous search has randomness. And in the later stage, the cells with smaller values in the adjacent grid are selected, which accelerates the convergence speed.

### 3.2.2 Fitness function

There are some differences in the application background between the ACO and LDTW. Based on the grid map, the fitness function of ACO is usually to find the shortest path from the starting point to the end point. Therefore, when the

algorithm searches for a path in the grid map, the length of the path is converted into the corresponding grid value. The fitness function is designed as follows:

$$F(S_k) = \sum_{i=1}^n dist(i), (1 < n \leq L_{UB} + 1), \quad (6)$$

where  $S_k$  is a solution formed by the ant  $k$  going to the target position,  $dist(i)$  is the value of the  $i$ -th grid passed by the ant  $k$ , and the value is the distance between the corresponding points in the two sequences.  $n$  is the total number of grids in the path solution.

### 3.2.3 Limitation of the search area

The step length in LDTW limits the range of bend path and avoids pathological alignment problem. Eq. (3) shows that LDTW is calculated recursively from the point  $(r, c)$  to  $(1, 1)$ , and the step length cannot exceed the maximum step size. In this study, the distance matrix is mapped to a 0-1 matrix in a grid map, and then the optimal path is searched by the algorithm proposed in this research. When the dimension is too large, the calculation efficiency of LDTW will be significantly reduced. In order to improve the computation efficiency, this algorithm combines the step length to restrict the search area of ants in a grid map. Therefore, for data with a large number of dimensions, the search area can be restricted to a hexagon by Eq. (7).

$$L = \text{ceil}[(N_{\text{step\_max}} - N_{\text{dim}}) \times \varepsilon], \quad (7)$$

where  $N_{\text{step\_max}}$  is the maximum step,  $N_{\text{dim}}$  is the dimension,  $\text{ceil}[]$  is rounded towards plus infinity,  $L$  is the two sides of the search area. As shown in Fig. 7, the searchable area is limited to a hexagon by a black barrier.

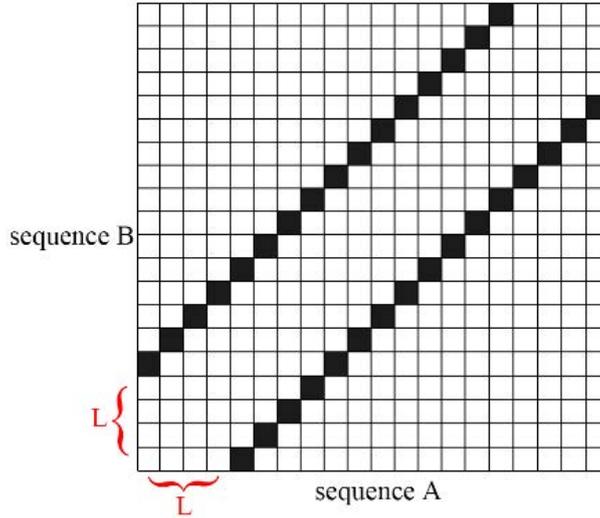
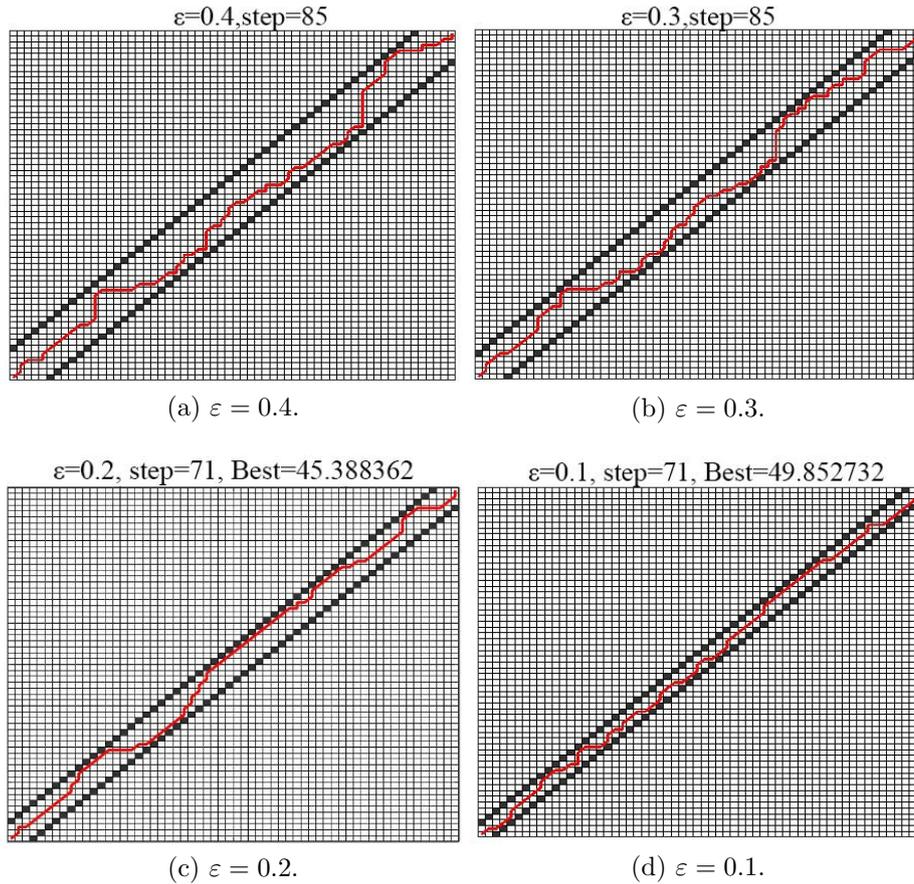


Fig. 7 Mapping from a distance matrix to a grid map.

The coefficient  $\varepsilon$  controls the scope of the search area, and the increase or decrease of the coefficient leads to the increase or decrease of the search range. In order to discuss the value range of the coefficient, the first dataset of Synthetic Control with a dimension of  $60 \times 60$  in the fifth part is selected, and the maximum step is 72. Set  $\varepsilon$  to a different value, and search with ACO\_LDTW. After calculating the iteration 100 times, the effect obtained is shown in Fig. 8.



**Fig. 8** Influence diagram of  $\varepsilon$  on search area.

The searchable range is limited to a narrow area in the middle by a black grid, the red line represents a search solution, which is 0.4, 0.3, 0.2, and 0.1 in Fig. 8(a–d), respectively. It can be seen that when the value of  $\varepsilon$  is large, the search range is also large, the algorithm cannot effectively converge to the optimal solution, and in the obtained result, the step size has exceeded the maximum steps; and the value of  $\varepsilon$  is too small, the search range getting smaller, leading to the omission of some good solutions. Therefore,  $\varepsilon$  is 0.2 in this research, which not only meets the limited steps of LDTW but also improves the computational efficiency of the ACO algorithm by narrowing the search area.

### 3.2.4 Pheromone initialization and update mechanism

In addition, the ACO algorithm itself has some defects, such as falling into local optimum, resulting in slow convergence. These defects are mainly reflected in the related operations of pheromone. In the initialization of pheromone, traditional ACO algorithm such as the MAX-MIN Ant System (MMAS) [29] initializes the pheromone of each location to a constant  $c$ , which results in the decrease of the search rate in the early stage. Therefore, in order to make the algorithm search on the diagonal line as far as possible in the early stage, the pheromone on the diagonal line is initialized to a constant  $c$ , and the pheromone content of the rest is randomly selected in  $[0, c]$ .

However, when the algorithm is running, the ant colony prefers to choose areas with high pheromone, and it is easy to ignore areas with low pheromone. Especially, the updating operation of the original ACO algorithm will make pheromone difference in different areas become larger and larger, leading to partial optimality and even stagnation.

In order to solve this problem, this paper adopts an optimal path update operation proposed by Ning J [30]. In this method, the hybrid enhanced pheromone updating mechanism is used to reduce the local optimal solution, and the pheromone smoothing mechanism is used to solve the stagnation problem. Subsequently, the pheromone values on this path are processed as follows:

$$\tau_2 = \rho * (\tau_{\max} - \tau_1), \quad (8)$$

where  $\tau_1$  is the current pheromone value on the stagnant path;  $\tau_2$  is the pheromone value processed by the smoothing mechanism;  $\rho$  is the pheromone evaporation coefficient, taking  $\rho = 0.1$ . If the algorithm is in the middle of the calculation, there is a stagnation phenomenon, which uses Eq. (8) to smooth the path.

The hybrid enhanced pheromone update mechanism is introduced to update pheromones:

$$\tau_{ij} \leftarrow (1 - \rho)\tau_{ij} + \Delta\tau_{ij}^{\text{best}} + \Delta\tau_{ij}^{\text{ibest}}, \quad (9)$$

where  $\Delta\tau_{ij}^{\text{best}}$  denotes the total number of pheromones on the current optimal path and  $\Delta\tau_{ij}^{\text{ibest}}$  denotes the pheromone increment generated by the new optimal path, which expands the search space. When the new path is superior to the current optimal path in this iteration, we add  $\Delta\tau_{ij}^{\text{ibest}}$  to the pheromone update equation; otherwise, the pheromone update is performed according to the original pheromone update equation.

$$\Delta\tau_{ij}^{\text{ibest}} = \begin{cases} \frac{1}{F(S_{\text{ibest}})} & \text{if } \langle i, j \rangle \in S_{\text{ibest}} \\ 0 & \text{otherwise} \end{cases}, \quad (10)$$

where  $F()$  is the fitness function and  $S_{\text{ibest}}$  is the iterative optimal solution set. When the algorithm approaches a defined stagnation state, the pheromone matrix is reinitialized by a new pheromone smoothing mechanism to balance the concentration difference of pheromones. To balance the pheromone concentration difference of the path, it helps ants to continue the search as it approaches stagnation.

The concrete steps of the ACO\_LDTW algorithm are as follows.

---

**Algorithm 1** Algorithm ACO\_LDTW.

---

**Input**sequence  $X = \{x_1, x_2, \dots, x_i, \dots, x_I\}$ sequence  $Y = \{y_1, y_2, \dots, y_j, \dots, y_J\}$ Fitness evaluation function:  $F()$ Incremental factor:  $p = p^0$ Initial iterations:  $iter = 1$ Maximum iterations:  $K$ **Output** $S_{\text{best}}$ : the optimal solution $L_{\text{step}}$ : the length of the optimal solution**Begin**

Computing grid map: Converting sequence  $X$  and  $Y$  into distance matrix shown in (1), generating zero matrix of the same dimension as distance matrix, using (7) limiting the zero matrix to 0-1 matrix, the grid map is obtained.

ACO algorithm parameter initialization:  $\eta_{ij}$  is the heuristic factor. The constant  $\alpha$  is the weighted value of the pheromone factor, and the constant  $\beta$  is the weighted value of the heuristic factor. Pheromone content on the counter-diagonal in grid map is  $\tau_{ij}^* = c$ , and the pheromone content in other locations is between 0 and  $c$ :  $\tau_{ij} = c \times rand()$ .

```

while  $iter \leq K$  do
    //  $iter$  is the number of iterations,  $K$  is the maximum number
    // of iterations.
    for  $k$  do do
        // ant  $k$ 
        if  $rand() > p$  then
            //  $rand()$  is the uniform random number generated between
            // 0 and 1.
            Ants choose the location to be transferred on a grid map by (4).
        else
            Select the corresponding minimum grid in the grid map as the next transfer
            location.
        end if
        Record the transfer position with the array  $Path[]$  until it moves to the target
        position, forming a complete solution  $S_k$  and a complete route  $Path_k$ 
        if  $F(S_k) < F(S_{\text{best}})$  then
             $S_{\text{best}} \leftarrow S_k$ 
             $Path_{\text{best}} \leftarrow Path_k$ 
        end if
    end for
     $\tau_{ij} \leftarrow$  update pheromone generated by (9) and (10)
    if the search process approaches a defined stagnation state [29] then
        Smoothing the pheromone on the stagnant path by (8)
    end if
    Increase  $p$  by (5)
     $iter = iter + 1$ 
end while
return  $S_{\text{best}}$ ,  $Step = \text{length}(Path_{\text{best}}) - 1$ 
End

```

---

## 4. Results and discussion

Grey incidence analysis (GIA) [31] is an important research area in grey theory. It can effectively deal with a little sample, incomplete information. Grey incidence degree (GID) is the calculation basis of GIA, which can decide the incidence according to the geometry curve between sequences. The closer each sequence curve approaches, the bigger the incidence degree of the corresponding sequence is. The main purpose of DTW is to measure the similarity between numerical sequences. Consequently, in this section, case experiments are used to evaluate the performance of the LDTW method as a distance measurement by calculating the GID.

Given sequences  $X_0 = \{x_0(1), x_0(2), \dots, x_0(m)\}$  and  $X_i = \{x_i(1), x_i(2), \dots, x_i(n)\}$ ,  $\text{ACO\_LDTW}(X_0, X_i)$  is LDTW distance between  $X_0$  and  $X_i$ . Then GID-LDTW is defined as follow:

$$\gamma(X_0, X_i) = \frac{\min_m \min_n |x_0(t_0) - x_i(t_i)| + \xi \max_m \max_n |x_0(t_0) - x_i(t_i)|}{\text{ACO\_LDTW}(X_0, X_i) / \lambda + \xi \max_m \max_n |x_0(t_0) - x_i(t_i)|}. \quad (11)$$

In Eq. (11),  $1 < t_i \leq m, 1 < t_j \leq n$  and  $\xi$  is the distinguishing coefficient and  $\lambda$  is the path length corresponds to  $\text{ACO\_LDTW}(X_0, X_i)$ . In general, usually set  $\xi \in [0, 1]$ , this paper uses  $\xi = 0.5$ .

In the case, China's GDP of various industries from 2001 to 2005 (Tab. I,  $X_0$  represents the GDP value,  $X_1$  represents the primary industry value,  $X_2$  represents the secondary industry value,  $X_3$  represents the tertiary industry value) is adopted.

Sequence	2001 (¥100 billion)	2002	2003	2004	2005
$X_0$	109.7	120.3	135.8	159.9	183.1
$X_1$	15.5	16.2	17.1	21	23.1
$X_2$	49.5	53.9	62.4	73.9	87
$X_3$	44.6	50.2	56.3	65	73

**Tab. I** China's GDP and the different industry output from 2001 to 2005 (unit, ¥100 billion).

### 4.1 Equal length sequences

GID models between  $X_0$  and  $X_1, X_2, X_3$  is calculated, respectively and the results of GID-ACO\_LDTW are compared with other GIDs. The steps are as follows:

1. Calculate the initial images of the original and zero images of start point:

$$X_0' = (0, 0.0966, 0.2379, 0.4576, 0.6691),$$

$$X_1' = (0, 0.0452, 0.1032, 0.3548, 0.4903),$$

$$X_2' = (0, 0.0444, 0.2606, 0.4929, 0.7576),$$

$$X_3' = (0, 0.1256, 0.2915, 0.4574, 0.6368).$$

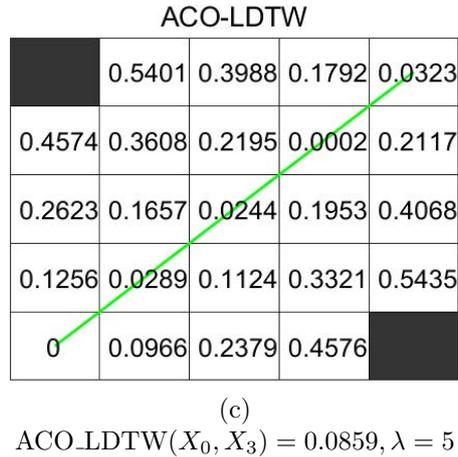
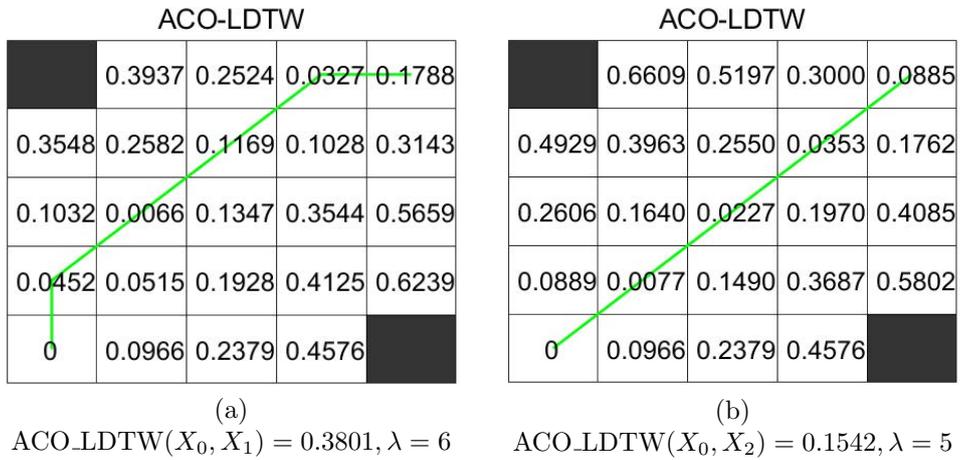
- Calculate the distance matrix between sequences. The corresponding LDTW distance and warping path length ( $\lambda$ ) are calculated. The result is shown in Fig. 9(a–c).

According to Fig. 9(a–c) and Eq. (11), we can get the GID of the sequences based on ACO-LDTW ( $\xi = 0.5$ ):

$$\gamma(X_0, X_1) = \frac{0 + 0.5 \times 0.6691}{0.3802/6 + 0.5 \times 0.6691} = 0.8407,$$

$$\gamma(X_0, X_2) = \frac{0 + 0.5 \times 0.7576}{0.1542/5 + 0.5 \times 0.7576} = 0.9250,$$

$$\gamma(X_0, X_3) = \frac{0 + 0.5 \times 0.6691}{0.0859/5 + 0.5 \times 0.6691} = 0.9510.$$



**Fig. 9** Distance matrix of sequences with equal length and ACO-LDTW distance.

3. Performance comparison. Compared with five GID models, it can be clearly shown that the results of Deng's GID, Relative GID, GID-DTW, and GID-ACO LDTW are more similar, but Absolute GID has an obvious mistake, as is shown in Tab. II.

Sequence	Deng's GID	Absolute GID	Relative GID	GID-DTW	GID-ACO LDTW
$(X_0, X_1)$	0.5664	0.5488	0.8837	0.8071	0.8407
$(X_0, X_2)$	0.7875	0.7456	0.9725	0.9250	0.9250
$(X_0, X_3)$	0.8546	0.7112	0.9889	0.9510	0.9510

**Tab. II** The GID result using different methods between same-length sequences with  $X_0$ .

From Tab. II, Deng's GID, Relative GID, GID-DTW, and GID-LDTW are consistent with the conclusion of qualitative analysis (according to the ranking of industrial value from high to low, it is the third industry, the second industry and the first industry in turn.). Thus, it can be obtained that GID-ACO LDTW has reliable performance when the sequence length is consistent.

## 4.2 Unequal length sequences

In order to evaluate the performance of different GID models when there are missing sequences data (or unequal length sequences), partial missing processing (shown in Tab. III) is carried out on the sequence data (from Tab. I).

Sequence	2001 (¥100 billion)	2002	2003	2004	2005
$X_0$	109.7	120.3	135.8	159.9	183.1
$X_1$	15.5	16.2	17.1	21	23.1
$X_2$	49.5	-	62.4	73.9	87
$X_3$	44.6	50.2	-	65	73

**Tab. III** China's GDP and the different industry output from 2001 to 2005 (partial missing, unit: ¥100 billion).

1. Calculate the initial images of the original and zero images of start point:

$$X_0^{\prime 0} = (0, 0.0966, 0.2379, 0.4576, 0.6691),$$

$$X_1^{\prime 0} = (0, 0.0452, 0.1032, 0.3548, 0.4903),$$

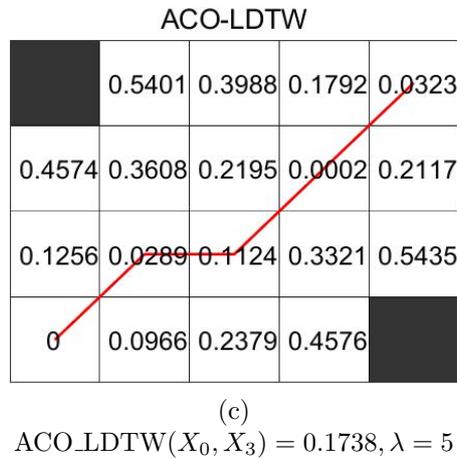
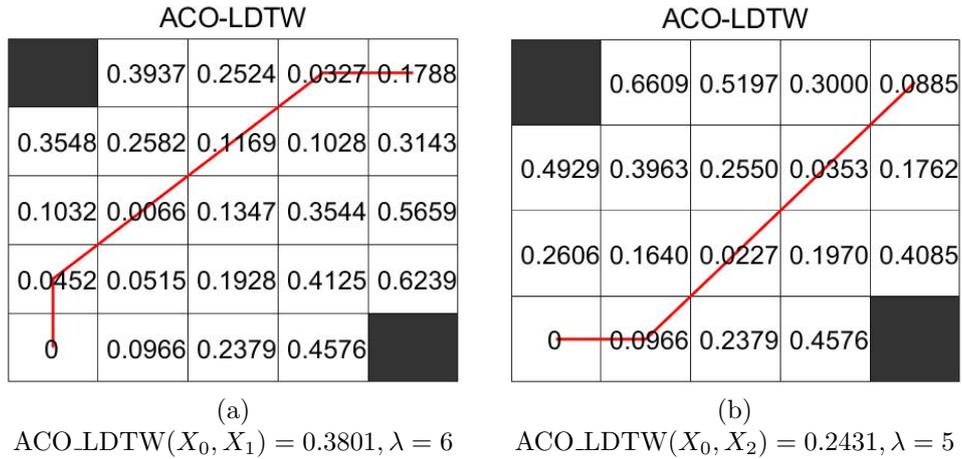
$$X_2^{\prime 0} = (0, 0.2606, 0.4929, 0.7576),$$

$$X_3^{\prime 0} = (0, 0.1256, 0.4574, 0.6368).$$

- Calculate the distance matrix between sequences; the numbers marked on each cell indicates the number of possible steps to that cell (shown in brackets) and calculate the corresponding ACO.LDTW distance and warping path length ( $\lambda$ ). The result is shown in Fig. 10(a–c).

According to Fig. 10 and Eq. (11), we can obtain GID of the sequences based on ACO.LDTW ( $\xi = 0.5$ ):

$$\gamma(X_0, X_1) = 0.8407; \gamma(X_0, X_2) = 0.8863; \gamma(X_0, X_3) = 0.9058$$



**Fig. 10** Distance matrix of sequences with unequal length and ACO.LDTW distance.

- Performance comparison. Because Deng's GID, Absolute GID, and Relative GID require sequences with the same length, the mean value method is used to fill in the missing data.

Sequence	Deng's GID	Absolute GID	Relative GID	GID-DTW	GID-ACO LDTW
$(X_0, X_1)$	0.5664	0.5488	0.8837	0.8071	0.8407
$(X_0, X_2)$	0.8226	0.7367	0.9851	0.8863	0.8863
$(X_0, X_3)$	0.7297	0.7164	0.9805	0.9058	0.9058

**Tab. IV** The gid result using different methods with unequal length sequences and  $X_0$ .

As shown in Tab. IV, when the length of the sequence is not equal, artificial complement has a great impact on the final analysis conclusion. In this situation, the result of GID-ACO\_LDTW is consistent with the conclusion of qualitative analysis. However, all of Deng's GID, Absolute GID and Relative GID all have made erroneous judgement. Therefore, GID-ACO\_LDTW is more adaptive than other methods.

### 4.3 1NN (1-Nearest Neighbor, 1NN) classification

In this section, based on the 1NN classification experiment [17], the performance of ACO\_LDTW as a distance measure is evaluated to effectively solve the problem that the calculation time of LDTW is too long.

In order to compare with LDTW conveniently, the same data sets as those in the literature [18] are also selected. Experiments are carried out on twenty-two datasets from UCR time series classification archive [32]. It mainly includes real-time series datasets and some synthetic datasets collected from various fields. Tab. V summarizes the basic information of all the datasets, such as the number of classes, the size of the training set, the size of the test set, and the length of the time sequence. The number of classes is from 2 (such as Gun-Point and Wafer) to 50 (50Words). The length of time series ranges from 60 (Synthetic Control) to 637 (Lightning-2).

The relevant parameters of ACO\_LDTW are set as follows:  $\alpha$  represents the weight of the pheromone factor of the state transition probability, and is usually set to 2;  $\beta$  represents the weight of the heuristic factor, and is usually set to 8.  $\rho$  is the pheromone evaporation coefficient, and it is usually limited to the range  $[0, 1)$ . According to the parameter experiment in reference [30],  $\rho$  is set to 0.02.

In Tab. VI, the 1NN error rate of LDTW and the upper limit of the optimal warping path length [17] have been given. We compare ACO\_LDTW with LDTW, and the four most widely used distance metrics: Euclidean distance, naive DTW, DTW under best Sakoe-Chiba band and DTW under best Itakura parallelogram. From Tab. VI, it can be seen that ACO\_LDTW has the lowest error rate of 16 datasets among 22 datasets. In addition to Gun-Point, Face (all) and Lightning-7, other datasets are consistent with LDTW. And the calculation time is shorter than LDTW, which will be described in detail in the time complexity section. These results prove the practicability of ACO\_LDTW. Because Euclidean distance is based on a strict one-to-one matching method, there are some deviations in its

Dataset	Number of classes	Size of training set	Size of testing set	Time series length
Synthetic Control	6	300	300	60
Gun-Point	2	50	150	150
CBF	3	30	900	128
Face (all)	14	560	1,690	131
OSU Leaf	6	200	242	427
Swedish Leaf	15	500	625	128
50Words	50	450	455	270
Trace	4	100	100	275
Two Patterns	4	1,000	4,000	128
Wafer	2	1,000	6,174	152
Face (four)	4	24	88	350
Lightning-2	2	60	61	637
Lightning-7	7	70	73	319
ECG	2	100	100	96
Adiac	37	390	391	176
Yoga	2	300	3,000	426
Fish	7	175	175	463
Plane	7	105	105	144
Car	4	60	60	577
Beef	5	30	30	470
Coffee	2	28	28	286
OliveOil	4	30	30	570

**Tab. V** *The basic information of 22 datasets.*

measurement results for time series. Time series are extended and shortened in DTW to calculate the similarity between the two-time series. DTW allows points in time series to be matched in equal length after self-replication, which overcomes the problem that Euclidean distance cannot be matched because of time series distortion. But DTW itself also has some problems. When matching a sequence, one point in the sequence corresponds to several points in the other sequence, which leads to deviation from the diagonal line too far, so there are deviations in the application results of 1NN. Aiming at this problem, DTW under best Sakoe-Chiba band and DTW under best Itakura parallelogram are the two most widely used constrained window shape methods, the range of curved paths is rigidly limited in them, thereby the curved path from deviating is avoided too far. Although the problem of pathological alignment is avoided in this mandatory setting, there are cases where the correct match is missed, resulting in unsatisfactory results in the operation of 1NN. Unlike the two methods mentioned above, LDTW specifies how many connections are allocated to each data point and where these links are placed by a softer constraint than windowing constraint, instead of setting a rigid limit. As a result, it allows more flexibilities and avoids the risk of missing the correct alignment. From the experimental results, it can be seen that the application of

1NN classification works well. However, the computation time cost of LDTW is very high. On the premise of ensuring the accuracy of 1NN classification, the linear recursive computing mechanism of ACO\_LDTW is replaced by the positive feedback mechanism of information and the parallel computation of ant colony optimization based on a grid map, so as to reduce the computation time cost.

A pairwise comparison of the 1NN error rate between ACO\_LDTW and the other five methods in Fig. 11(a–e). One polyline is formed by the 1NN error rate of ACO\_LDTW on different datasets, and the other represents other comparison methods. Among them, the closer the point is to the center of the circle, the smaller the error rate is, and vice versa. As we can see, most of the points in the ACO\_LDTW method are closer to the center point of the circle, indicating that its effect is still better. On each graph, the number of ACO\_LDTW closer to the center of the circle and the number of other contrast methods closer to the center of the circle is depicted, and the rest are the same number of distances from the center of the circle. For example, in Fig. 11(a), ACO\_LDTW is superior to Euclidean distance in 18 of 22 datasets, and is equal on 4 datasets, and is not inferior to Euclidean distance on any dataset. These charts show intuitively the advantages of ACO\_LDTW on various datasets.

#### 4.3.1 Time complexity

Next, the time complexity of ACO\_LDTW is analyzed and the running time is compared with other methods.  $L_{UB}$  is the upper bound of the warping path in ACO\_LDTW. The number of iterations of ACO\_LDTW is  $K$ , the number of ants is  $M$ , and the time complexity is  $O(K \cdot M \cdot L_{UB})$ .

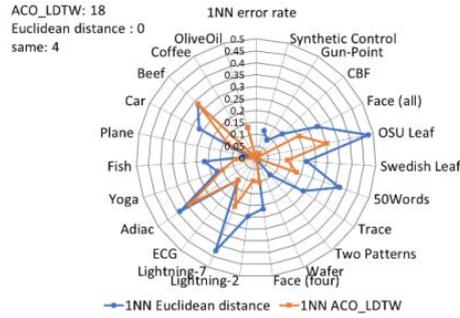
In order to compare the running time of different methods, on the basis of the 1NN classification experiment, five different datasets are recorded for the corresponding calculation time, and Fig. 12 shows the running time. It can be seen that ACO\_LDTW has shorter computation time than LDTW. Because the dynamic adjustment of state transition probability ensures that the previous search is random, the convergence rate is accelerated at the later stage. And the ants do not need to find the optimal path in the whole grid map. By limiting the search scope, the smaller the scope, the shorter the calculation time, it can greatly save computing time.

#### 4.4 Mechanical fault diagnosis

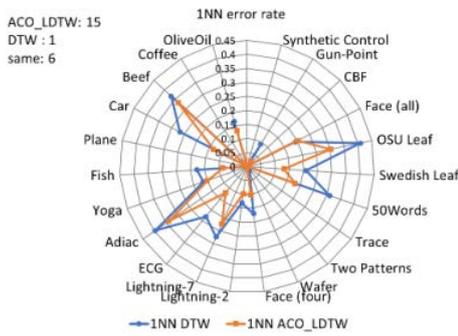
Furthermore, ACO\_LDTW is applied to the field of mechanical fault diagnosis to verify the practical application ability. The bearing fault data of Case Western Reserve University [33] is used to simulate single point faults at different parts of the bearing by electric discharge machining grooving. This data is mainly through the setting of different groove width: 0.007, 0.014, 0.021 (1 inch = 2.54 cm), to simulate bearing slight fault, moderate fault, and serious fault. Six kinds of fault vibration signals of the driving end vibration bearing with spindle speed 1772 rpm, load 1 HP and sampling frequency 12000 Hz are selected for analysis in this experiment, including slight faults of outer ring, slight faults of inner ring, slight faults of rolling body, moderate faults of inner ring, serious faults of inner ring and serious faults of rolling body. The main experimental steps are as follows:

Dataset	INN	INN DTW	INN DTW	INN DTW	INN LDTW	INN ACO_LDTW
	Euclidean distance	(best Sakoe-Chiba band( $r\%$ ) <sup>a</sup> )	(best Itakura parallelogram (d) <sup>b</sup> )	(best upper bound of warping path length)	(best upper bound of warping path length)	(best upper bound of warping path length)
Synthetic Control	0.120	0.017 (6)	<b>0.007 (6)</b>	<b>0.007 (72)</b>	<b>0.007 (72)</b>	<b>0.007 (72)</b>
Gun-Point	0.087	0.087 (0)	0.027 (5)	<b>0.020 (158)</b>	<b>0.023 (158)</b>	<b>0.023 (158)</b>
CBF	0.148	<b>0.003 (11)</b>	0.004 (55)	<b>0.003 (198)</b>	<b>0.003 (198)</b>	<b>0.003 (198)</b>
Face (all)	0.286	0.192 (3)	<b>0.179 (8)</b>	0.198 (147)	0.201 (147)	0.201 (147)
OSU Leaf	0.479	0.388 (7)	0.355 (50)	<b>0.301 (468)</b>	<b>0.301 (468)</b>	<b>0.301 (468)</b>
Swedish Leaf	0.211	0.154 (2)	0.150 (6)	<b>0.131 (134)</b>	<b>0.131 (134)</b>	<b>0.131 (134)</b>
50Words	0.369	0.242 (6)	0.255 (29)	<b>0.180 (301)</b>	<b>0.180 (301)</b>	<b>0.180 (301)</b>
Trace	0.240	0.010 (3)	0.010 (7)	<b>0.000 (416)</b>	<b>0.000 (416)</b>	<b>0.000 (416)</b>
Two Patterns	0.090	0.002 (4)	0.000 25 (10)	<b>0.000 (194)</b>	<b>0.000 (194)</b>	<b>0.000 (194)</b>
Wafer	0.005	0.005 (1)	0.005 (3)	<b>0.003 (155)</b>	<b>0.003 (155)</b>	<b>0.003 (155)</b>
Face (four)	0.216	0.114 (2)	0.170 (14)	<b>0.102 (386)</b>	<b>0.102 (386)</b>	<b>0.102 (386)</b>
Lightning-2	0.246	0.131 (5)	0.131 (35)	<b>0.098 (778)</b>	<b>0.098 (778)</b>	<b>0.098 (778)</b>
Lightning-7	0.425	0.288 (5)	0.247 (24)	<b>0.205 (355)</b>	0.224 (355)	0.224 (355)
ECG200	<b>0.120</b>	<b>0.120 (0)</b>	<b>0.120 (1)</b>	<b>0.120 (98)</b>	<b>0.120 (98)</b>	<b>0.120 (98)</b>
Adiac	0.389	0.391 (3)	0.376 (15)	<b>0.338 (177)</b>	<b>0.338 (177)</b>	<b>0.338 (177)</b>
Yoga	0.170	0.155 (2)	<b>0.135 (8)</b>	0.149 (467)	0.149 (467)	0.149 (467)
Fish	0.217	0.154 (4)	0.206 (9)	<b>0.086 (474)</b>	<b>0.086 (474)</b>	<b>0.086 (474)</b>
Plane	0.038	<b>0.000 (6)</b>	<b>0.000 (5)</b>	<b>0.000 (218)</b>	<b>0.000 (218)</b>	<b>0.000 (218)</b>
Car	0.267	0.233 (1)	0.283 (1)	<b>0.133 (598)</b>	<b>0.133 (598)</b>	<b>0.133 (598)</b>
Beef	0.333	0.333 (0)	<b>0.267 (12)</b>	0.333 (471)	0.333 (471)	0.333 (471)
Coffee	<b>0.000</b>	<b>0.000 (0)</b>	<b>0.000 (1)</b>	<b>0.000 (307)</b>	<b>0.000 (307)</b>	<b>0.000 (307)</b>
OliveOil	0.133	<b>0.133 (0)</b>	<b>0.133 (1)</b>	<b>0.133 (581)</b>	<b>0.133 (581)</b>	<b>0.133 (81)</b>

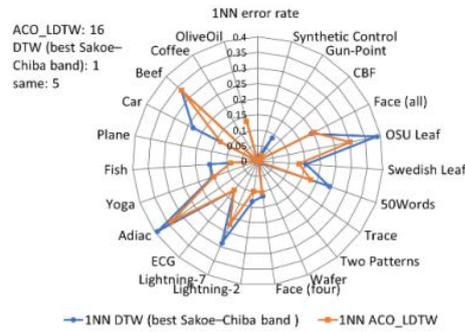
Tab. VI Comparisons on INN error rate.



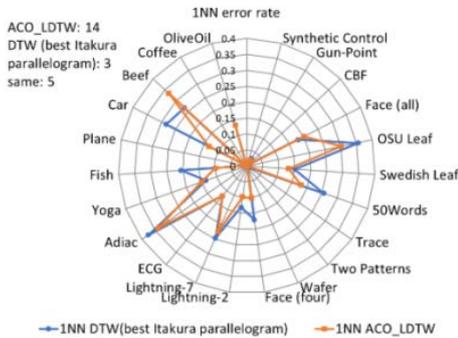
(a) Comparison with Euclidean distance.



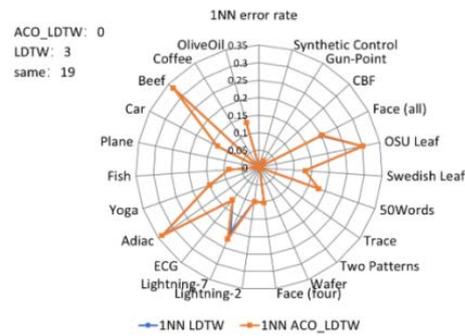
(b) Comparison with DTW.



(c) Comparison with DTW (best Sakoe-Chiba band).



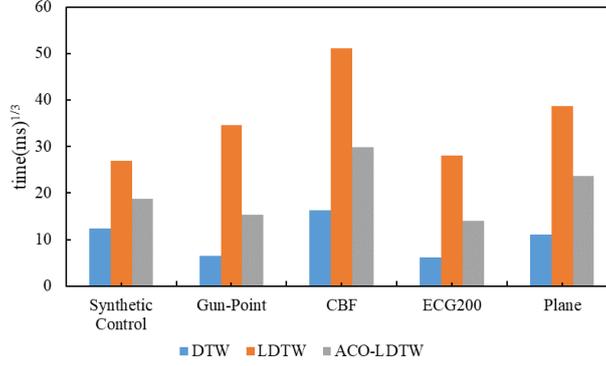
(d) Comparison with DTW (best Itakura parallelogram).



(e) Comparison with LDTW.

**Fig. 11** Comparison of 1NN error rates between ACO\_LDTW and five competitors.

**Step 1:** In each fault state, 50 sets of samples are taken, each sample with 2048 points, 20 of which are used as training samples, and the remaining 30 are used as test samples;



**Fig. 12** Running time of 1NN classification on five different datasets.

**Step 2:** 24 time-frequency domain statistical characteristics are extracted from the vibration signal of each sample [34].

**Step 3:** Use the support vector machine (SVM) to classify the above samples. First, the Gauss kernel function (RBF) is used as the kernel function in SVM, and its expression is as follows:

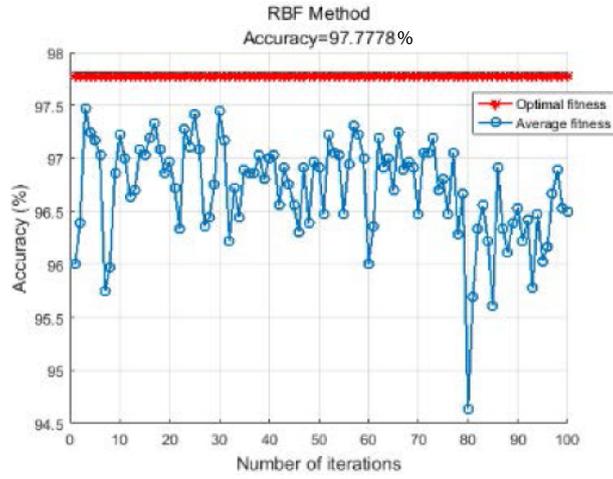
$$K(X_i, X_j) = \exp \left[ \frac{-\|X_i - X_j\|^2}{2\sigma^2} \right]. \quad (12)$$

**Step 4:**  $\|X_i - X_j\|^2$  used to calculate the similar distance between samples in Eq. (12). To verify the validity of the algorithm  $\|X_i - X_j\|^2$  is replaced by ACO\_LDTW( $X_i, X_j$ ) to calculate the similar distance;

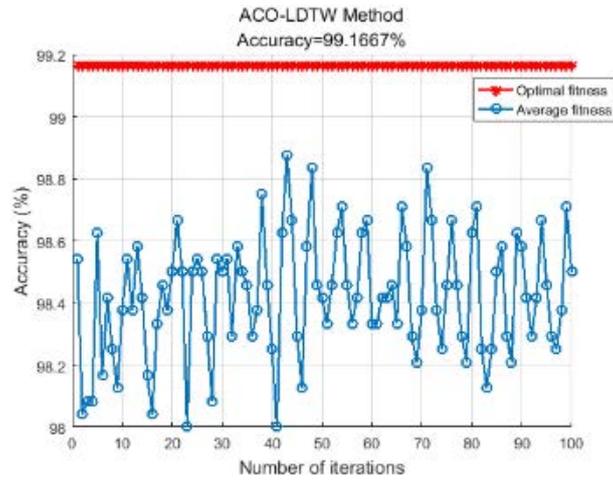
**Step 5:** Because the kernel parameter ( $\delta$ ) and penalty factor ( $C$ ) have great influence on its classification performance, the algorithm is used to optimize the parameters of SVM [35]. The original Gaussian kernel function and the replaced kernel function are used to adjust the parameters to get the optimal classification accuracy.

The convergence curves of classification accuracy for two kernel functions are shown in Fig. 13(a, b).

As can be seen from the above diagram, compared with RBF kernel function, the improved kernel function can effectively improve the classification accuracy, the average accuracy rate reached 99.1667%, reflecting that the algorithm has a useful application prospect in practical engineering problems. The similarity distance in the improved kernel function brings the idea of LDTW into the grid map. The most important feature of LDTW is that it limits the length of the warping path, effectively reduces pathological alignment, and ensures the accuracy. But when using this improved kernel function to classify, we need to use ACO\_LDTW calculate the similarity distance between sequences first. Compared with using RBF kernel function, it brings some calculation costs.



(a) RBF kernel function.



(b) Improved kernel function.

**Fig. 13** Comparison of convergence curves.

## 4.5 Discussion

From the above experiments, it can be seen that ACCO\_LDTW, as a basic method of similarity metrics, maintains better processing accuracy in data prediction (equal or unequal length sequences) and classification processing after combining with other data mining models. In addition, in terms of time complexity, although there is a certain increase compared with DTW, and a significant decrease compared with LDTW, which is conducive to the further application of LDTW.

## 5. Conclusions

In this paper, the intelligent heuristic search mechanism of ant colony algorithm is introduced to LDTW to divide the complex data mining tasks to several parallel sub tasks. Compared with LDTW and other similarity measurement methods, ACO-LDTW has the same accuracy with LDTW on most datasets and is superior to other methods. At the same time, the calculation time is less than LDTW. ACO-LDTW is suitable for data mining application scenarios that need large-scale data analysis, such as industrial manufacturing and Internet of things. It can meet the requirements of high time efficiency.

In the future, as ACO-LDTW may fall in the local optimal solution, the optimal solution obtained by ACO-LDTW will may be not the best. Enhancing the algorithm's global searching capability will be the main goal of the next stage.

## Acknowledgement

This work is supported by National Natural Science Foundation of China (No. 61936001).

## References

- [1] CHUI K.T., GUPTA B.B., VASANT P. A genetic algorithm optimized RNN-LSTM model for remaining useful life prediction of turbofan engine[J]. *Electronics*, 2021, 10(3), pp. 285, doi: [10.3390/electronics10030285](https://doi.org/10.3390/electronics10030285).
- [2] BHATTI M.H., KHAN J., KHAN U.G., IQBAL R., GUPTA B. Soft computing-based EEG classification by optimal feature selection and neural networks[J]. *IEEE Transactions on Industrial Informatics*, 2019, 15(10), pp. 5747–5754.
- [3] KOTSIFAKOS A., ATHITSOS V., PAPAPETROU P. Query-sensitive distance measure selection for time series nearest neighbor classification[J]. *Intelligent Data Analysis*, 2016, 20(1), pp. 5–27, doi: [10.3233/IDA-150791](https://doi.org/10.3233/IDA-150791).
- [4] PARMEZAN A.R.S., BATISTA G.E. A study of the use of complexity measures in the similarity search process adopted by knn algorithm for time series prediction. In: *IEEE 14th International Conference on Machine Learning and Applications (ICMLA)*, Miami, Florida, U.S. Piscataway, NJ: IEEE, 2015, pp. 45–51.
- [5] SAKOE H., CHIBA S. Dynamic programming algorithm optimization for spoken word recognition[J]. *IEEE transactions on acoustics, speech, and signal processing*, 1978, 26(1), pp. 43–49, doi: [10.1109/TASSP.1978.1163055](https://doi.org/10.1109/TASSP.1978.1163055).
- [6] RAIMONDI D., ORLANDO G., MOREAU Y., VRANKEN W.F. Ultra-fast global homology detection with discrete cosine transform and dynamic time warping[J]. *Bioinformatics*, 2018, 34(18), pp. 3118–3125, doi: [10.1093/bioinformatics/bty309](https://doi.org/10.1093/bioinformatics/bty309).
- [7] MARTÍN-YEBRA A., LANDREANI F., CASELLATO C., PAVAN E., MIGEOTTE P.F., FRIGO C., MARTÍNEZ J.P., CAIANI E.G. Evaluation of respiratory-and postural-induced changes on the ballistocardiogram signal by time warping averaging [J]. *Physiological measurement*, 2017, 38(7), pp. 1426, doi: [10.1088/1361-6579/aa72b0](https://doi.org/10.1088/1361-6579/aa72b0).
- [8] SHARMA A., SUNDARAM S.. On the exploration of information from the DTW cost matrix for online signature verification[J]. *IEEE transactions on cybernetics*, 2017, 48(2), pp. 611–624, doi: [10.1109/TCYB.2017.2647826](https://doi.org/10.1109/TCYB.2017.2647826).
- [9] TOLOSANA R., VERA-RODRIGUEZ R., FIERREZ J., ORTEGA-GARCIA J. Exploring Recurrent Neural Networks for On-Line Handwritten Signature Biometrics[J]. *IEEE Access*, 2018, 6, pp. 5128–5138, doi: [10.1109/ACCESS.2018.2793966](https://doi.org/10.1109/ACCESS.2018.2793966).

- [10] TANG J., CHENG H., ZHAO Y., GUO H. Structured Dynamic Time Warping for Continuous Hand Trajectory Gesture Recognition \*[J]. *Pattern Recognition*, 2018, 80, pp. 21–31, doi: [10.1016/j.patcog.2018.02.011](https://doi.org/10.1016/j.patcog.2018.02.011).
- [11] CHENG H., DAI Z., LIU Z., ZHAO Y. An image-to-class dynamic time warping approach for both 3D static and trajectory hand gesture recognition[J]. *Pattern recognition*, 2016, 55, pp. 137–147, doi: [10.1016/j.patcog.2016.01.011](https://doi.org/10.1016/j.patcog.2016.01.011).
- [12] BRANDEJSKY T. Qualitative behaviours similarity measure[J]. *Neural Network World*, 2002, 12(2), pp. 105–112.
- [13] KEOGH E. J., PAZZANI M. J. Scaling up dynamic time warping for datamining applications. In: *Proceedings of the sixth ACM SIGKDD international conference on Knowledge discovery and data mining*, New York, U.S. New York: Association for Computing Machinery, 2000, pp. 285–289.
- [14] TSINASLANIDIS P., ALEXANDRIDIS A., ZAPRANIS A., LIVANIS E. Dynamic time warping as a similarity measure: applications in finance. In: *13th Annual Conference of Hellenic Finance and Accounting Association (HFAA)*, Volos, Greece. 2014.
- [15] HSU C. J., HUANG K. S., YANG C. B., GUO Y. P. Flexible Dynamic Time Warping for Time Series Classification[J]. *Procedia Computer Science*, 2015, 51, pp. 2838–2842, doi: [10.1016/j.procs.2015.05.444](https://doi.org/10.1016/j.procs.2015.05.444).
- [16] LEI H.W., LIU R.R., LIU X. A Dynamic Handwriting Recognition Algorithm Based on DTW [J]. *Computing Technology and Automation*, 2018, 37(1), pp. 83–88, doi: [10.16339/j.cnki.jsjsyzdh.201801017](https://doi.org/10.16339/j.cnki.jsjsyzdh.201801017).
- [17] DING J., CHANG C. W. Feature design scheme for Kinect-based DTW human gesture recognition[J]. *Multimedia Tools and Applications*, 2016, 75(16), pp. 9669–9684, doi: [10.1007/s11042-015-2782-3](https://doi.org/10.1007/s11042-015-2782-3).
- [18] ZHANG Z., TAVENARD R., BAILLY A., TANG X., TANG P., CORPETTI T. Dynamic time warping under limited warping path length[J]. *Information Sciences*, 2017, 393, pp. 91–107, doi: [10.1016/j.ins.2017.02.018](https://doi.org/10.1016/j.ins.2017.02.018).
- [19] Akka K., Khaber F. Mobile robot path planning using an improved ant colony optimization[J]. *International Journal of Advanced Robotic Systems*, 2018, 15(3), doi: [10.1177/1729881418774673](https://doi.org/10.1177/1729881418774673).
- [20] KEOGH E. J., PAZZANI M. J. Derivative dynamic time warping. In: *Proceedings of the 2001 SIAM international conference on data mining*. Society for Industrial and Applied Mathematics, 2001, pp. 1–11.
- [21] KEOGH E., RATANAMAHATANA C. A. Exact indexing of dynamic time warping[J]. *Knowledge and information systems*, 2005, 7(3), pp. 358–386, doi: [10.1007/s10115-004-0154-9](https://doi.org/10.1007/s10115-004-0154-9).
- [22] BAILLY A., MALINOWSKI S., TAVENARD R. Bag-of-temporal-sift-words for time series classification. In: *ECML/PKDD workshop on advanced analytics and learning on temporal data*, Würzburg, Germany, 2015 .
- [23] ZHANG Z., TANG L., TANG P. Local feature based dynamic time warping. In: *2014 International Conference on Data Science and Advanced Analytics (DSAA)*. Shanghai, China. Piscataway, NJ : IEEE, 2014, pp. 425–429.
- [24] ZHANG Z., TANG P., DUAN R. Dynamic time warping under pointwise shape context[J]. *Information sciences*, 2015, 315, pp. 88–101, doi: [10.1016/j.ins.2015.04.007](https://doi.org/10.1016/j.ins.2015.04.007).
- [25] CANDAN K.S., ROSSINI R., SAPINO M.L., WANG X. sDTW: computing DTW distances using locally relevant constraints based on salient feature alignments[J]. *Proceedings of the VLDB Endowment(PVLDB)*, 2012, 5(11), pp. 1519–1530, doi: [10.14778/2350229.2350266](https://doi.org/10.14778/2350229.2350266).
- [26] ELFES A. Using occupancy grids for mobile robot perception and navigation[J]. *Computer*, 1989, 22(6), pp. 46–57, doi: [10.1109/2.30720](https://doi.org/10.1109/2.30720).
- [27] AKKA K., KHABER F. Mobile robot path planning using an improved ant colony optimization[J]. *International Journal of Advanced Robotic Systems*, 2018, 15(3), doi: [10.1177/1729881418774673](https://doi.org/10.1177/1729881418774673).

- [28] YANG X. S., GANDOMI A. H. Bat algorithm: a novel approach for global engineering optimization[J]. *Engineering computations*, 2012, 29(5), pp. 464–483, doi: [10.1108/026444401211235834](https://doi.org/10.1108/026444401211235834).
- [29] STÜTZLE T., HOOS H. H. MAX–MIN ant system[J]. *Future generation computer systems*, 2000, 16(8), pp. 889–914, doi: [10.1016/S0167-739X\(00\)00043-1](https://doi.org/10.1016/S0167-739X(00)00043-1).
- [30] NING J., ZHANG Q., ZHANG C., ZHANG B. A best-path-updating information-guided ant colony optimization algorithm[J]. *Information Sciences*, 2018, 433, pp. 142–162, doi: [10.1016/j.ins.2017.12.047](https://doi.org/10.1016/j.ins.2017.12.047).
- [31] DENG J. L., *The Foundation of Grey System*. Wuhan, Hubei, China. Wuhan: HUST, 2002.
- [32] CHEN Y., KEOGH E., HU B., BEGUM N., BAGNALL A., MUEEN A., BATISTA G. *The UCR Time Series Classification Archive*, available online: [www.cs.ucr.edu/~eamonn/time\\_series\\_data/](http://www.cs.ucr.edu/~eamonn/time_series_data/)
- [33] The Case Western Reserve University, Bearing Data Center, 2007, <http://csegroups.case.edu/bearingdatacenter/pages/download-data-file>.
- [34] LEI Y., HE Z., ZI Y. A new approach to intelligent fault diagnosis of rotating machinery[J]. *Expert Systems with applications*, 2008, 35(4), pp. 1593–1600, doi: [10.1016/j.eswa.2007.08.072](https://doi.org/10.1016/j.eswa.2007.08.072).
- [35] THARWAT A., HASSANIEN A. E., ELNAGHI B. E. A BA-based algorithm for parameter optimization of support vector machine[J]. *Pattern Recognition Letters*, 2017, 93, pp. 13–22, doi: [10.1016/j.patrec.2016.10.007](https://doi.org/10.1016/j.patrec.2016.10.007).