

# A Novel Dynamic Time Wrapping Similarity Algorithm Optimized by Multi-Granularity

Xu Jianfeng<sup>1,2</sup>, Tang Tao<sup>1</sup> and \*Zhang Yuanjian<sup>2</sup>

<sup>1</sup>College of Software, Nanchang University, Nanchang 330047, China;

<sup>2</sup>Department of Computer Science, Tongji University, Shanghai 200000, China)

Email: zhangyj901029@gmail.com

## Abstract

Dynamic time warping algorithm (DTW) is a method of measuring the similarity of time series. Concerning the problem that DTW cannot keep high classification accuracy when the computation speed improved, a FG-DTW method based on the idea of naive granular computing is proposed. In this method, firstly, better temporal granularity is acquired by calculating temporal variance feature and it is used to replace original time series; Secondly, the elastic size of under comparing time series granularity allow dynamic adjustment through DTW algorithm and optimal time series corresponding granularity is obtained; Finally, DTW distance is calculated by optimal corresponding granularity model. At the same time, the early termination strategy of infimum function is introduced to improve the efficiency of FG-DTW algorithm. Experiments show that the proposed algorithm improves the running rate and accuracy effectively.

**Keywords:** DTW; Time sequence; Time grain; Elastic; FG-DTW

## 1. Introduction

Time series data is an important form of big data, widely exists in biological information, stock finance, smart grid, smart city, and many other areas. Time series similar algorithm research is the important foundation of temporal data mining and time series data analysis. Time series similar algorithm is widely used to solve time-series data clustering [1]-[2], time series data classification [3]-[4], similar search [5][6], time series data regression problem *etc.* At present, the representative time series similarity algorithm is based on Euclidean Warping (Distance Dynamic) and time series dynamic time warping distance (Time).

Euclidean distance [7]-[9] in the research of time series similarity comparison, compared with other methods, has the following advantages: easy to realize, no reference, can be applied to any field. But there are also a lot of limitations, which requires participation in the comparison of the timing length and must be equal to the sequence data on the time axis deformation and noise is also very sensitive [10]. Due to the widespread existence of these uncertainties in practical applications, the accuracy of time series analysis based on Euclidean distance is not always ideal.

DTW was firstly applied in speech recognition, which can effectively solve a sequence of deformation on the time axis and the noise interference. DTW uses time warping path with minimum cost matching, allowing time series data extend along the time axis. [11]-[13] shows that the dynamic time warping distance is better than the Euclidean distance in the time series similarity search. Under the background of big data era, analytics services tend to ensure the accuracy and has higher efficiency. However, the computational complexity of DTW is  $O(MN)$ , so that enhancing the efficiency of DTW has important significance.

At present, there are two methods to improve the efficiency of DTW. The first method is using the principle of lower bound function [14]-[16] to fast cut off the extremely

dissimilar time series, leaving part of the candidate set to calculate the DTW point-by-point, hence, achieving fast convergence and improve the efficiency of the algorithm. In [15], four elementary feature vector is extracted from each time series, and then the maximum value of the difference between the two sequences is used as the lower bound function of DTW. This algorithm is often used in subsequence matching problem [17][18]. [18] computed the boundary part that a sequence beyond another sequence  $[qL, qU]$ , as a lower bound on the DTW distance, where  $qL$  and  $qU$  are the sequence strings which composed by the minimum and maximum values of data in the sliding window of  $2w+1$  width, respectively. This method receives much attention in the sequence search [19], and it is applied in many fields [20]. However, due to the shortcomings of the lower bound function asymmetry [18], limiting the in time series clustering and classification applications. Lower bound function method still uses the point calculation, which reflects the time series trend problem undesirably [22]. Under the background of big data, there exists some defects, especially in the data stream processing, through the experiment of comparison algorithm in chapter five, we can conclude that for most time series classification, the lower bound function has some limitations in efficiency of algorithm.

The second method is to reduce the dimension of time series data through data processing, and replace the original time series with low dimensional feature, thus greatly improving the efficiency of DTW computation. For example, [23][26] proposed some algorithms based on the DTW features, the algorithm firstly near the approximation linearly of the original sequence, extract the basic features of each segment, to achieve the dimensionality reduction, then using DTW to calculate the sequence features after segmentation. [23] Proposed the algorithm framework of "dynamic trend of bending", named "trajectory" instead of the original sequence, the distance is calculated with linear approximation after two line segments of the area. But with the approximation of the area between the line segments as the difference will be part of the miscarriage of justice. [26] Used multivariate statistical characteristics of synthetic sequence information, reduced the dimension of multiple time series, thus greatly enhance the computing speed, in accordance with the current stage of the background of the DTW application of large data, improving the efficiency of the operation. But the disadvantages in the similar accuracy are the premise of the algorithm and the original sequence linear approximation by piecewise, did not reflect the characteristics of the original overall sequence, and did not pay attention to inappropriate piecewise destruction of the characteristics of data, which will misjudge the sequence similarity.

There are some researches combined with the two algorithms, [27] proposed a query method named Fast Search under Distance Time Wrapping. First of all, the time series is divided into small blocks, and the lower bounds of Keogh distance can be calculated, the time series of arbitrary length can be queried. It uses multi-granularity method of similarity measure for the coarse granularity sequence of similarity measure firstly, using the idea of function lower bound, namely  $D_{lb}(o_i, o_j) > \varepsilon \Rightarrow D(o_i, o_j) > \varepsilon$ , where  $D_{lb}(o_i, o_j)$  is coarse-grained LB Keogh distance. If  $D_{lb}(o_i, o_j) \leq \varepsilon$ , in accordance with the same method, sliming the sequence of refinement points until the end of point-by-point calculation. In this way, according to  $D_{lb}(o_i, o_j) \geq \varepsilon$ , avoiding the unnecessary sequence calculation to improve the running speed. However, FTW scans the sequence set repeatedly, without considering the effect of particle size on time series similarity, so that the efficiency of FTW is not improved effectively. In this paper, we learn from the idea of FTW segmentation.

This paper propose a sequential morphological variance algorithm for coarse timing granularity selection based on the above researches. In the calculation of the coarse-grained DTW, allows on the coarse timing particle elastic preferentially to obtain optimal matching. Finally, combined with the early termination of the lower bound of the

function, improving the running speed under the premise of the higher classification accuracy.

## 2. Dynamic Time Warping Similarity Algorithm

The idea of DTW is as follows: Given two time series  $p$  and  $Q$ ,  $P = (p_1, p_2, \dots, p_M)$  and  $Q = (q_1, q_2, \dots, q_N)$ . The distance matrix between two series data points is  $D_{M \times N} = \{d(i, j)\}_{M \times N}$ , where  $1 \leq i \leq M$ ,  $1 \leq j \leq N$ , the values of  $d(i, j)$  is determined by the square of Euclidean distance between  $p_i$  and  $q_j$ , namely  $d(i, j) = (p_i - q_j)^2$ . The matrix stores the distance between two time series of data at different time points. As shown in Figure 1 and 2, each square in the figure is equal to the value of the elements, so that DTW is to find a continuous path  $W = w_1, w_2, \dots, w_c$  from the matrix, which minimizes the accumulate value of the elements on the path.

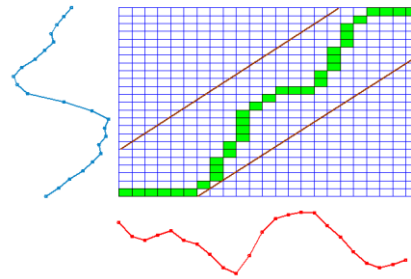


Figure 1. Structuring Bending Matrix and Calculating the Bending Distance

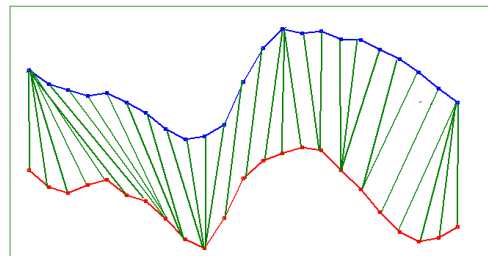


Figure 2. Two Series after DTW Alignment

In the computation of DTW, the bending path  $W$  needs to meet some characters as follows:

Endpoint alignment. The bending path must start at the beginning of the two sequence, end at the end of the two sequence, namely  $w_1 = d(p_1, q_1)$  and  $w_N = d(p_M, q_N)$ .

Adjacent point continuous. The adjacent points in a bending path must be derived from the adjacent squares of the matrix (which can be a diagonal or adjacent to the boundary), namely

$$\begin{cases} w_k = d(p_i, q_j) \\ w_{k+1} = d(p_{i'}, q_{j'}) \end{cases}; i' \leq i+1, j' \leq j+1$$

Interval increment. The bending path extends over time, namely

$$\begin{cases} w_k = d(p_i, q_j) \\ w_{k+1} = d(p_{i'}, q_{j'}) \end{cases}; i' \geq i, j' \geq j$$

### 3. Multi-granularity DTW Model

The main process of the algorithm in this paper is divided into two steps. The first step: Volatility granularity granulated according to the characteristics of the variance of the time series. The second step: Using the early termination of the lower bound function and elastic particle size adjustment to calculate the distance of DTW. According to the characteristics of the algorithm, the algorithm is called Flexible Granularity Dynamic Time Warping (FG-DTW).

We make a brief explanation of the mathematical notation in this paper, as shown in the Table 1.

**Table 1. Mathematical Notation and Explanation**

Notation	Explanation
$T_{(t_i, a_x)}$	Size of time series partitioning
$*T_{(t_i, a_x)}$	Feature of the extracted time series
$d()$	Distance function
$\arg \min_{\theta, \delta}$	The minimum distance between two particle sizes
M, N	The length of original time series
$t_i$	The begin time point of time series
$a_x$	The number of particle size
$\theta, \delta$	The size of adjustable particle size
m, n	The number of particle size after dividing

For convenience, the time windows based on timing variance is called timing particle, represented as  $T_{(t_i, a_x)}$ , where  $t_i$  is the begin time point of the timing particle,  $a_x$  is the data point of the timing particle. After the division of timing particle, we extract the characteristics of time series (five partial position method, linear approximation) to calculate it, represented as  $*T_{(t_i, a_x)}$ .

#### 3.1. Granulation Partition Based on Time Series Variance

Given a time series T with length M,  $T = \{T_{(t_1, 1)}, T_{(t_2, 1)}, \dots, T_{(t_m, 1)}\}$ , which is the unit time series, can be represented as:

$$Var(seg_m^n) = E(seg_m^n - E(seg_m^n))^2 = \sum_{i=m}^n (\Delta T_i - E(seg_m^n))^2$$

where  $seg_m^n$  is the selected difference from m to n in the unit particle size,  $E(seg_m^n)$  is the expected value from m to n in the unit particle size,  $\Delta T_i = \|\Delta T_{(t_{i+1}, 1)} - \Delta T_{(t_i, 1)}\|$ . We select coarse grain based on sequence variation fluctuation characteristics, the Specific

algorithms are as follows:

Step 1: Obtaining  $\Delta T = \{\Delta T_1, \Delta T_2, \dots, \Delta T_{M-1}\}$  from  $T = \{T_{(t_1,1)}, T_{(t_2,1)}, \dots, T_{(t_M,1)}\}$ ,  
 where  $\Delta T_j = \|\Delta T_{(t_{i+1},1)} - \Delta T_{(t_i,1)}\|$ ,  $1 < i < M$ ,  $1 \leq j \leq M-1$ .

Step 2: Incorporated gradually into the difference sequence elements based on the difference of each unit difference sequence. In the case of a given threshold  $\varepsilon$ , dividing the time series according to  $Var(seg_m^n)$ .

Assuming a division of coarse particle size, we obtain a sequence of length K, which we denote as  $T = \{T_{(t_1, a_{i1})}, \dots, T_{(t_k, a_{ik})}\}$ , if  $t_i$  and  $b_i$  is the begin and end of i order particle, respectively, where  $b_i = t_i + a_{ii}$ . So that each part meeting such relation:

$$Var(seg_{t_i}^{b_i}) = \sum_{j=t_i}^{b_i-1} (\Delta T_j - E(seg_{t_i}^{b_i-1}))^2 \leq \varepsilon$$

We give an example to explain the algorithm. Given a time series  $T = \{5, 7, 6, 8, 14, 16, 17, 19, 4, 20, 5, 19, 3\}$ , we can obtain the difference sequence  $\Delta T = \{2, 1, 2, 6, 2, 1, 2, 15, 16, 15, 14, 16\}$ , assuming a threshold  $\varepsilon = 2$ . The variance of  $\Delta T_1$  and  $\Delta T_2$  is 0.5 less than 2, so merge into a particle size, other sequences are similar to above principle. So that we obtain the variance equals to 0.67, when merged into  $\Delta T_4$ , the variance become 15.9, hence, the new particle size is started at  $\Delta T_5$ . Followed on above processing, we obtain the result of coarse granularity selection as follows:

$T_{(t_1,4)} = \{5, 7, 6, 8\}$ ,  $\Delta T_{(t_5,4)} = \{14, 16, 17, 19\}$ ,  $\Delta T_{(t_9,5)} = \{4, 20, 5, 19, 3\}$ . As shown in the Figure 3.

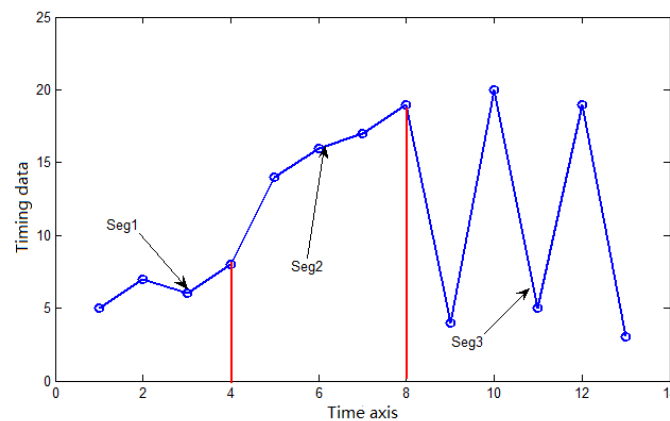


Figure 3. Sketch Map of Variance Segment

The algorithm is as follows:

Input: Time series T, a threshold  $\varepsilon$

Output: The index of Time series granularity partition index []

1: Algorithm VarianceSection (T,  $\varepsilon$ )

2:  $M = \text{length}(T)$ ;

3: for i=1 to M-1 do

4:  $\Delta T[i] = |T[i+1] - T[i]|$ ;

5: endfor

```

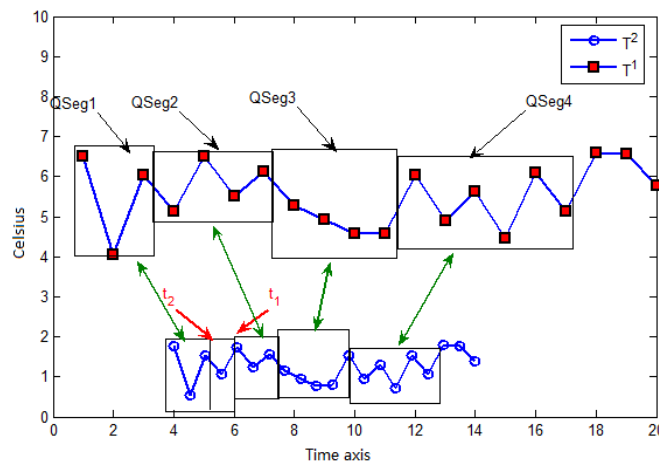
6:j:=1;
7:start:=1;
8:for i=2 to M-1 do
9:  end=I;
10: Var=VAR (start, end);
11: if Var>ε
12:  index[j]:=I;
13:  j++;
14:  start:=i+1;
15: end if
16:end for
17:return index[];
    
```

Based on above algorithm, we can know that the time complexity of the algorithm is  $O(M)$ .

### 3.2. The Optimal Coarse Grain Size Division Model

#### 3.2.1. Problem Description

Through the timing variance algorithm, we get the division of coarse granularity, but coarse granularity partitioning results do not have to be the most effective appropriate granularity, because of the influence by threshold, the granularity error appears. We make the follow explanations about this phenomenon. Each particle has a "flexible", that is, a good time may not be the most similar between two corresponding compared sequences. We assume that there are two temperature sequences  $T^1$  and  $T^2$ , as shown in Figure 4:



**Figure 4. Schematic Diagram of Grain Size Adjustment**

Sequence  $T^2$  is compressed by  $T^1$ , if we divide the coarse granularity by the same threshold  $\epsilon$ , we may get the results as shown in Figure 4. We learn from Figure 4 that  $d(PSeg1, QSegt_1) > d(PSeg1, QSegt_2)$ , so we propose the elastic adjustment of coarse grain size produced by the algorithm of time series variance.

### 3.2.2. Adjustment of Particle Size

According to the time series variance, we use the following method to adjust the time series.

Given two time series  $T^1$  and  $T^2$ ,  $T^2_{(t_i, a_i)}$  is the time series to be adjusted form  $T^2$ ,  $T^2_{(t_i, a_i)^\theta}$  is the solution space of  $T^2_{(t_i, a_i)}$ ,  $T^2_{(t_i, a_i)} \in T^2_{(t_i, a_i)^\theta}$  and  $T^2_{(t_i, a_i)^\theta} = T^2_{(t_i \pm \theta, a_i \pm \delta)}$ , where  $\theta$  and  $\delta$  belong to minimal variables.  $T^1_{(t_j, a_j)}$  is the reference sequence of time series, keep constant,  $T^2_{(t_i, a_i)}$  is the adjustable time series, keep various. Assuming that the optimal time series is  $T^2_{(t_j, a_j)_{best}}$ , so that we need to adjust the rule to minimize the evaluation function F.

$$F = \arg \min_{\theta, \delta} = \sum_{i, j=1}^k \| a - b \|^2$$

$$\text{where } a = d(*T^1_{(t_i, a_i)}, *T^2_{(t_j, a_j)_{best}}), b = d(*T^1_{(t_i, a_i)}, *T^2_{(t_j + \theta, a_j + \delta)})$$

### 3.2.3. Algorithm of the Adjustable Particle Size

The algorithm of the adjustable particle size is as follows.

Input: Time series T1 and the timing particle T<sub>(t1 at1)</sub>, time series T2 to be adjustable and the timing particle

T<sub>(tj atj)</sub>.

Output: The optimal distance between two time series dbase (\*T<sub>(t1 at1)</sub>), \*T2<sub>(tj atj)</sub>).

- 1: Algorithm Adjust (\*T<sub>(t1 at1)</sub>, \*T<sub>(tj atj)</sub>)
- 2: if (a<sub>ij</sub>-1)
- 3: for k=1 to a<sub>ij</sub>-1
- 4: b1= dbase(\*T<sub>(t1 at1)</sub>, T2<sub>(tj-k+1 atj)</sub>);
- 5: b2= dbase(\*T<sub>(t1 at1)</sub>, \*T2<sub>(tj-k atj)</sub>);
- 6: c1=best(b1,b2);
- 7: end for ;
- 8: end if;
- 9: for k= 1to a<sub>ij</sub>-1
- 10: b1= dbase(\*T<sub>(t1 at1)</sub>, \*T2<sub>(tj+k-1 atj)</sub>);
- 11: b2= dbase(\*T<sub>(t1 at1)</sub>, \*T2<sub>(tj+k atj)</sub>);
- 12: c2=best(b1,b2);
- 13: end for;
- 14:for k=1 to a<sub>ij</sub>-1
- 15: b1= dbase(\*T<sub>(t1 at1)</sub>, \*T2<sub>(tj atj-k+1)</sub>);
- 16: b2= dbase(\*T<sub>(t1 at1)</sub>, \*T2<sub>(tj atj-k)</sub>);
- 17: c3=best(b1,b2);
- 18: end for;
- 19:if(a<sub>ij</sub>+1)

```

20: for k=1 to aj+1
21:  b1= dbase(*T(t1 at1),T2(tj atj-k-1));
22:  b2= dbase(*T(t1 at1),*T2(tj-k atj-k));
23:  c4=best(b1,b2);
24: end for ;
25:end if
26:return min(c1,c2,c3,c4);
    
```

Based on above algorithm, we can know that the time complexity of the algorithm is  $O(a_{ii})$ , where  $a_{ii}$  is the size of particle, owing to the very small value of the size, a single grain size adjustment hardly consumes the time cost.

### 3.3. FG-DTW Synthesis Algorithm

Early termination is a method used in the calculation of limited distance, where the limitation is that the distance calculation is only one means to solve the problem. However, the similarity search of time series can only be satisfied by  $d(p, q) \leq \varepsilon$ , if we have  $d(p', q') > \varepsilon (p' < p, q' < q)$ , then there is no need to continue the calculation. In this paper, we combine with the early termination algorithm, the FG-DTW algorithm model is as follows:

Input: The time series TS1, TS2 to be adjustable, the thresholds  $\mu$ .

Output: If DTW (TS1, TS2)  $\leq \mu$ , return true, otherwise, return false.

```

1: Algorithm GetSimilarity (TS1, TS2)
   TS1 = {T(t1,a1), T(t1,a2), ..., T(tk,amm)} TS2 = {T(t1,a1), T(tj,a2), ..., T(tn,an)}
2: Index1 []:= VanrianceSection (TS1, ε);
3: Index2 []:= VanrianceSection (TS2, ε);
4: TS1= Getsequence (TS1, Index1 []);
5: TS2= Getsequence (TS2, Index2 []);
6: a=1;
7: M[1][1]= Adjust(*T(t1, at1), *T2(t1, at1));
8:  if (M [1] [1]> μ)
9:   return false;
10: end if;
11:  M[i][1]=M[i-1][1]+d(*T(ti, ati),*T2(t1, at1));
12: M[1][i]=M[1][i-1]+ d(*T(t1, at1),*T2(ti, ati));
13: for i=2 to m do
14: overflow=true;
15: for j=2 to n do
16: temp=min (M [i-1] [j-1], M [i-1] [j], M[i] [j-1]);
17: if (temp>μ) M[i] [j] =DOUBLE_MAX;
18: end if
    
```



```

19: else M[i][j]=temp+Adjust(d(*T(ti,ati), *T(tja,atj)));
20:   end if
21:   overflow=false;
22:   end for
23:   if (overflow) break;
24: end for
25:   if (overflow) return false;
26: return (M[m] [n] ≤ μ)
    
```

Based on above algorithm, we can know that the time complexity of the algorithm is  $O(mna_i)$ . Since  $n \leq N$ ,  $m \leq M$  and the early termination algorithm, we can improve the efficiency of DTW effectively. In principle, the efficiency of the algorithm is comparable to that based on feature algorithm, but the fine granularity of the algorithm increases the complexity of the algorithm. The actual experiment is slightly inferior to the characteristic DTW algorithm.

## 4. Experimental Program

### 4.1. Introduction of Experiment

Sequential clustering and classification are two common techniques in time series data mining, and their results are very good to illustrate the performance of the algorithm. This experiment uses time series classification to examine the performance of DTW, subsection DTW, early termination of DTW, these algorithms were recorded as DTW, Piecewise-DTW, Stop-DTW Early, and FG-DTW. The experiments bases on K-NN proximity method, and the stability, efficiency and classification accuracy of these algorithms are investigated. Under the background of big data, how to classify on the basis of small errors on the rapid classification of time series, is to examine the similarity method of the standard. The experimental data is from 10 data sets of UCR database, and the detailed information of the data set is shown in Table 2.

**Table 2. The Information of Data Set**

Name	Number of class	Size of training set	Size of testing set	Time series length
S-Control	6	300	300	60
Gun-Point	2	50	150	150
CBF	3	30	900	128
Trace	4	100	100	275
Fish	7	175	175	463
50_word	50	450	455	270
Adiac	37	390	391	176
Beef	5	30	30	470
Coffee	2	28	28	286
FaceAll	14	560	1690	131

Two sets of experiments were designed in this paper. Experiment one is the same nature, different size of the time sequence of the performance of different methods. Experiment two is to examine the performance of different methods in different time series.

#### 4.2. Experiment One

In this paper, we select two data sets from 10 data sets as the experimental data of this experiment. We select CBF, 50\_word data sets. Table 2.2 shows CBF data set test set a total of 900, with 100 as the interval, from the angle of efficiency and sequential classification accuracy, testing the performance of different algorithm the experiment get time efficiency, as shown in Figure 5:

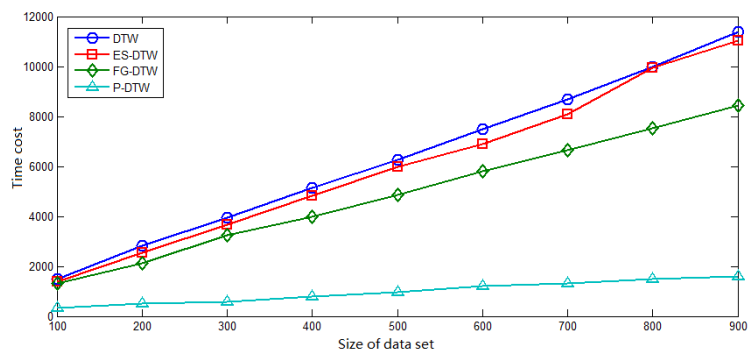


Figure 5. Comparison of Time Efficiency

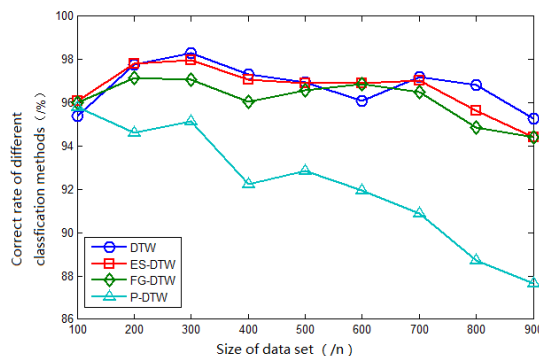


Figure 6. Comparison of Correct Rate

As shown in Figure 6, although P-DTW in efficiency is good, but with the increase in the data set, the correct rate of P-DTW relative decline, and finally almost into an exponential decline. For large data sets, the decline in the correct rate is not allowed. FG-DTW is relatively stable. The difference between the former two algorithms is not obvious.

Based on Table 2, we can know 50 word data sets have multiple categories, examining the different methods of discrimination performance, from Figure 7, it can be seen that FG-DTW method is similar to ES-DTW method on the time efficiency at the early stage, the later time growth is still linear, which did not increase with the rapid decrease of time efficiency.

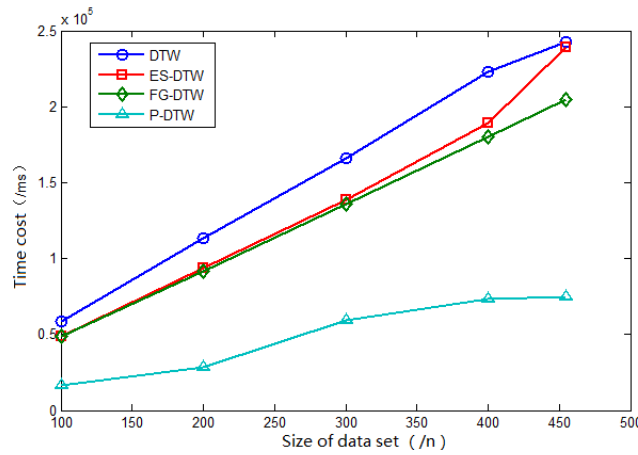


Figure 7. Comparison of Time Efficiency

### 4.3. Experiment Two

In order to examine the stability of the different algorithms, for different data sets, we used histogram to reflect the performance of different methods at the time efficiency and accuracy. 10 data sets properties are different, due to the different data sets, the gap between actual times of running algorithm is large, which can't be represented in the same chart. So that for different data sets, the running time of the algorithm are normalized,

dealing with the following formula  $a_i / \sum_{j=1}^4 a_j$ , where  $a_j$  is the running time of different methods on the same data set.

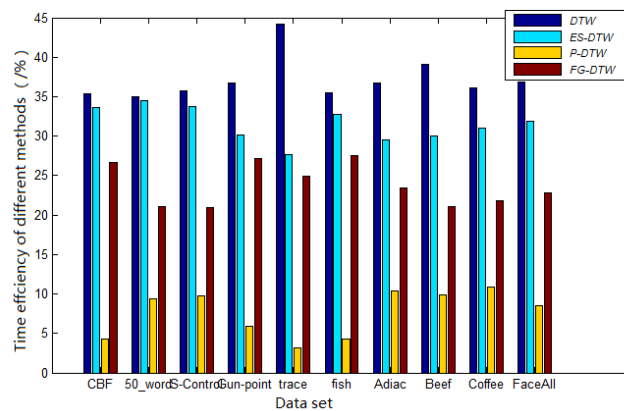
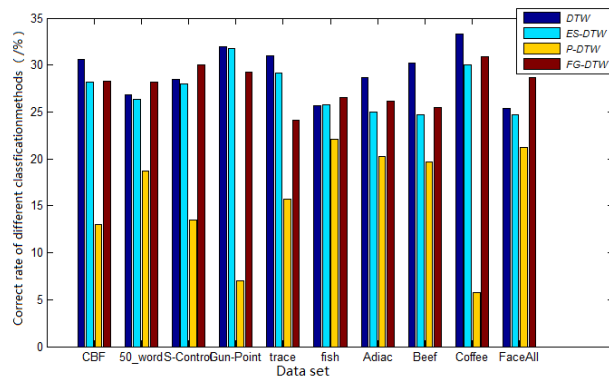


Figure 8. Time Efficiency Comparison on Different Data Sets



**Figure 9. Correct Rate of Classification**

As shown in Figure 8, for different data sets, the relative time of the four methods spend a relatively stable proportion. Overall speaking, the time efficiency of FG-DTW is still relatively good, has been around 0.25, the other two methods are basically in more than 0.3, the best performance is still P-DTW.

Based on Figure 9, we can obviously see that P-DTW exists much shortage in correct rate, which reduce the rate of correct to improve the speed of for some application limitations. FG-DTW is even higher than DTW and ES-DTW in most data sets. Although some are lower than the two, but the difference is small, which shows that the FG-DTW method is not based on the loss of the correct rate of efficiency. On the whole, FG-DTW balances the time efficiency and the correct rate, and is adapted to the different needs better.

## 5. Conclusion

DTW is an important method to judge the similarity of time series data in recent years, especially in the era of big data, time series data will be involved in almost every field. But the data point based distance operation is bound to affect the application of DTW in time sequence. Based on particle size, the feature extraction method is a new way, but there is a great influence on the noise and drift.

In this paper, based on the time series variance algorithm, the elastic strategy is given with the threshold constraint. This design is the further optimization, equal to the DTW based on the time grain as the benchmark to determine the corresponding relationship of the rough adjustment. The effectiveness of the experimental results illustrates the validity of this strategy.

Some deficiencies still exist in the design of the algorithm FG-DTW. First of all, variance threshold adjustment is the key factor affecting the performance of, only through the experimental debugging, have some limitations. Secondly, the time efficiency also needs to be improved. In addition, the lack of effective mechanism in the process of the solution space search of elastic time particles makes the search lack of direction, the search process will speed up if some information that reduce the search space can be found.

## Acknowledgements

Authors would like to thank the anonymous referees for their constructive comments and valuable suggestions. This work is supported by the National Science Foundation of China (Nos.61273304) and Specialized Research Fund for the Doctoral Program of Higher Education of China (Nos.20130072130004).

## References

- [1] T. W. Liao, "Clustering of time series data-a survey", *J. Pattern Recognition*, vol. 38, no. 11, (2005), pp. 1857-1874.
- [2] X. Zhang, J. Liu, Y. Du and T. Lv, "A novel clustering method on time series data", *J. Expert Systems with Applications*, vol. 38, no. 9, (2011), pp. 11891-11900.
- [3] Y. S. Jeong, M. K. Jeong and O. A. Omitaomu, "Weighted Dynamic Time Warping For Time Series Classification", *J. Pattern Recognition*, vol. 44, no. 9, (2011), pp. 2231-2240.
- [4] P. P. Shen, H. Wang, Z. Q. Meng, Z. Y. Yang, Z. P. Zhi, R. Jin and A. M. Yang, "An Improved Parallel Bayesian Text Classification Algorithm", *J. International Information and Engineering Technology Association*, vol. 3, no. 1, (2016), pp. 6-10.
- [5] F. Gullo, G. Ponti, A. Tagarelli and S. Greco, "A time series representation model for accurate and fast similarity detection", *J. Pattern Recognition*, vol. 42, no. 11, (2009), pp. 2998-3014.
- [6] W. Qiang and V. Megalooikonomou, "A dimensionality reduction technique for efficient time series similarity analysis", *Information Systems*, vol. 33, no. 1, (2008), pp. 115-132.
- [7] C. Faloutsos, M. Baganathan and Y. Manolopoulos, "Fast subsequence matching in time series databases", *J. Association for Computing Machinery Sigmod Record*, vol. 23, no. 2, (2000), pp. 19-429.
- [8] D. Rafiei and A. Mendelzon, "Similarity — Based Queries for Time Series Data", *J. Association for Computing Machinery Sigmod Record*, vol. 26, no. 2, (1998), pp. 13-25.
- [9] K. P. Chan and W. C. Fu, "Efficient Time Series Matching by Wavelets, C//Institute of Electrical and Electronics Engineers 29th International Conference on Data Engineering", Washington: IEEE Computer Society, (1999), pp. 126-133.
- [10] H. Li and L. Yang, "Extensions and relationships of some existing lower—bound function for dynamic time warping", *J. Journal of Intelligent Information Systems*, vol. 43, no. 1, (2014), pp. 1-21.
- [11] B. J. Jain, "Generalized gradient learning on time series", *J. Machine Learning*, vol. 100, no. 2, (2015), pp. 587-608.
- [12] H. Izakian, W. Pedrycz and I. Jamal, "Fuzzy clustering of time series data using dynamic time warping distance", *J. Engineering Applications of Artificial Intelligence*, vol. 39, no. 3, (2015), pp. 235-244.
- [13] E. Keogh, "Exact indexing of dynamic time warping", *Proceedings of the 28th International Conference on Very Large Databases Conference*. Hong Kong: Very Large Data Base Endowment, (2002), pp. 406-417.
- [14] E. Keogh and C. A. Ratanamahatana, "Exact indexing of dynamic time warping", *J. Knowledge & Information Systems*, vol. 7, no. 3, (2005), pp. 358-386.
- [15] S. W. Kim, S. Park and W. W. Chu, "An index-based approach for similarity search supporting time warping in large sequence databases", *Data Engineering, 2001 Proceedings 17<sup>th</sup> International Conference .Heidelberg, Germany: IEEE Computer Society, (2001), pp. 607-614.*
- [16] B. K. Yi, H. V. Jagadish and C. Faloutsos, "Efficient Retrieval of Similar Time Sequences Under Time Warping", *Data Engineering, 1998 Proceedings 14th International Conference on Orlando, FL: IEEE Computer Society, (1998), pp. 201-208.*
- [17] S. W. Kim, J. Yoon, S. Park and T. H. Kim, "Shape-based retrieval of similar subsequences in time-series databases", *Proceedings of the 2002 Association for Computing Machinery symposium on Applied computing*. New York: ACM, (2002), pp. 438-445.
- [18] S. Park, S. W. Kim and W. W. Chu, "Segment-based approach for subsequence searches in sequence databases", *Proceedings of the 2001 Association for Computing Machinery symposium on Applied computing*. New York: ACM, (2001), pp. 248-252.
- [19] H. Kremer, S. Günemann, A. M. Lyanescu, I. Assent and T. Seidl, "Efficient processing of multiple DTW queries in time series databases", *Scientific and Statistical Database Management, Springer Berlin Heidelberg, (2011), pp. 150-167.*
- [20] T. Rakthanmanon, B. Campana, A. Mueen, G. Batista and B. Westover, "Searching and mining trillions of time series subsequences under dynamic time warping", *Proceedings of the 18th ACM SIGKDD international conference on Knowledge discovery and data mining New York: ACM, (2012), pp. 262-270.*
- [21] J. Zhang, "Research on similarity of time series", D. Wu Han:Huazhong University of Science and Technology, (2008).
- [22] Z. Zhang, P. Tang and R. Duan, "Dynamic time warping under point wise shape context", *J. Information Sciences*, vol. 315, no. 10, (2015), pp. 88-101.
- [23] L. Sun, Y. Yang and W. Liu, "Trended DTW Based on Piecewise Linear Approximation for Time Series Mining", *Proceedings of the 2011 IEEE 11th International Conference on Data Mining Workshops*. Vancouver, BC: IEEE Computer Society, (2011), pp. 877-884.
- [24] S. S. Jambhale and A. Khaparde, "Gesture recognition using DTW & piecewise DTW", *Electronics and Communication Systems, 2014 International Conference on*. Coimbatore: IEEE, (2014), pp. 1-5.
- [25] J. Hills, J. Lines, E. Baranauskas, J. Mapp and A. Bagnall, "Classification of time series by shapelet transformation", *J. Data Mining & Knowledge Discovery*, vol. 28, no. 4, (2014), pp. 851-881.
- [26] Q. Cai, L. Chen and J. Sun, "Piecewise statistic approximation based similarity measure for time series", *J. Knowledge—Based Systems*, vol. 85, no. 10, (2015), pp. 181-195.

- [27] Y. Sakurai, M. Yoshikawa and C. Faloutsos, "Fast Similarity Search under the Time Wrapping Distance", Proceedings of 24th Association for Computing Machinery SIGMOD-SIGACT-SIGART Symposium on Principles of Database Systems (PODS), Maryland, USA: ACM Press, (2005), pp. 326-337.