# Research on Modeling and Forecasting Driven by Time Series Stream Data

Xuan Ma
Faculty of Automation and Information Engineering,
Shaanxi Key Laboratory of Complex System Control and
Intelligent Information Processing,
Xi'an University of Technology, Xi'an, China

Guoxin Ma
Faculty of Automation and Information Engineering,
Xi'an University of Technology, Xi'an, China

*Abstract*—**In view of the potentially infinite, fast-reaching, single-scan and noise-related characteristics of time series stream data, a method of data driven modeling and predicting is proposed in this paper. In order to enhance the real-time performance and the modeling accuracy of algorithm to the time series stream, we use a double sliding window to divide the time series stream data. One window is designed as a fixed length window to evaluate the fluctuation of the actual data, the other, as a variable length window, is used to establish a prediction model by GEP algorithm and generate prediction data. And then, the actual data and the prediction data calculated by the prediction model are fused to generate a fusion data. The colony climbing algorithm is applied to improve the population diversity of GEP to improve the prediction accuracy of modeling. The numerical simulation to the four test data sets shows that the proposed algorithm has better prediction accuracy than the Hierarchical Temporal Memory algorithm.**

***Keywords- time series stream data; modeling and forecasting; sliding window***

## I. Introduction

Time series stream data is a collection of data collected in a certain chronological order, which is widely used in the fields of finance, process control and electrical load [1-2]. The time series have two characteristics: Firstly, the collection of each data must be accompanied by time dimension, and the arrival order is arranged in order of time; secondly, the data change according to a certain law within a certain time period. Compared with the static data of the traditional database, the stream data [3] presents the characteristics of fast arrival speed, potentially unlimited, single linear scan and continuous real-time appearance [4]. Therefore, the stream data mining algorithm needs to satisfy: (1) Adapt to complex data stream models; (2) The algorithm should meet the requirements of real-time performance. The processing speed of the algorithm should not be lower than the speed of new data arrives. (3) As the data stream is updated, the predictive model should also be able to quickly follow the regular changes in the data stream.

Actual time series stream data usually contain a lot of noise [5]. Noise is a key issue affecting the modeling of time series stream data. The noise in the data will lead to overfitting or underfitting of the mined time series stream data model and reduce the generalization ability of the predictive model. However, the current research works on how to screen the training sample set and to reduce the influence of the noise in time series stream data is still in its infancy.

The characteristics of time series stream data and the requirements of the algorithm described above, as well as the noise problem in the data, bring a lot of challenges to modeling and forecasting, such as how to reduce the influence of noise in the data, how to mine the inherent laws of the data and to predict the data of future moments accurately.

With the development of prediction technology, the time series stream data prediction method has evolved from the original and classic model-driven method to the data-driven method [6]. The data-driven method does not need to require prior knowledge to preset the model. It constructs an approximate model to approximate the real situation by the past historical data. The data-driven prediction method has attracted people's attention.

At present, the sliding window is generally used to divide stream data in the process of stream data mining, and then mine the model of the data in these windows. For example, the Skylin System developed by Etsy uses the data in the most recent period as the mining object and analyzes the statistical characteristics of the time series stream data by various methods [7]. However, this method cannot effectively mine the internal laws of time series stream data. Some scholars also use wavelet analysis to mine the information of time series in the frequency domain [8]. Such methods are usually computationally intensive and difficult to predict in real time. Literature [9] takes account of the two characteristics of time series described above and uses historical data to establish a predictive model of time series stream data. Literature [10] uses genetic algorithm to optimize the weight and threshold of BP neural network to improve the accuracy of prediction for chaotic time series. In [11], a time series anomaly detection algorithm based on hierarchical temporal memory algorithm is proposed. It learns the pattern law in the time series and establishes the prediction model. Experiment results shows that the algorithm has better prediction ability than the traditional Autoregressive Integrated Moving Average model. However, because the noise in the time series data has not been considered, it results in a larger final prediction error.

In this paper, we propose a method of time series stream data driven modeling and prediction. To the sample data, the time series stream data is divided by the double sliding window, and the GEP algorithm is used to mine the function model of the data in the variable length window. The data fusion method is used

978-1-5386-9490-9/19/$31.00 ©2019 IEEE

to reduce the noise influence on the modeling. We use the colony climbing algorithm to improve the population diversity of GEP to improve the model's ability to respond to the inherent changes of time series stream data.

## II. Algorithm Mechanism

The design idea of the algorithm is to divide the time series stream data by the double sliding window, then use GEP [12] to perform function discovery on the data in the variable length window and generate an optimal prediction model. Then substitute the next timestamp into the prediction model to generate the predicted value. When the new data arrives, the data fusion method is used to fuse the predicted value with the new data to generate a fusion value, so as to reduce the influence of noise on the GEP algorithm searching prediction model. Adding the fusion value to the time series stream, and then divide the fusion time series stream data by the double sliding window. Before doing function discovery in the variable length window, using the colony climbing algorithm to improve the population diversity and the global search ability of the GEP algorithm. Using the data fusion method to fuse the predicted and actual values, and then repeat the above operations to realize the modeling and forecasting of the time series stream data.

### A. Double sliding window

Considering the characteristics of stream data with potential infinity, fast arrival speed and single scanning, which made scanning data not to be repeated, and the modeling and prediction to be real-time, in this paper, we use a double sliding window to divide the time series stream data. The double sliding window is shown as in Figure.1.

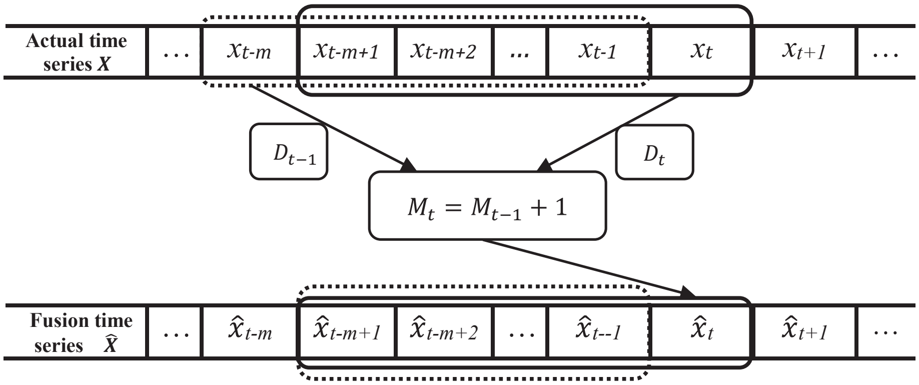


Figure 1. Double sliding window

Given a time series stream $X = \{x_1, x_2, x_3 \dots\}$, shown as in Figure 1, one window is applied to the actual time series, which is fixed length, the other is applied to the fusion time series, which is variable length. Take the fixed length $m$, there are $m$ data from data $x_t$ to data $x_{t-m+1}$ in the fixed length window, and calculating the variance $D_t$ of the data. In the variable length window, take the length $M$, the length of the variable length window at the time $t$ is represented as $M_t$, which is determined according to the variance of the data in the fixed length window, shown as below.

$$M_t = \begin{cases} M_{t-1} + 1 & if\ D_t > D_{t-1} \\ M_{t-1} - 1 & otherwise \end{cases} \tag{1}$$

In the fixed length window, if the fluctuation of the data at time $t$ is larger than the fluctuation of the data at time $t$-$1$, then increasing the length of the variable length window, otherwise reducing the length of the variable length window. The length of the variable length window should be set to the upper and lower limits, and the upper limit is set to $M_{big}$ , the lower limit is $M_{lit}$.If $M_t > M_{big}$ , then $M_t = M_{big}$ ,if $M_t < M_{lit}$ , then $M_t = M_{lit}$ .The length of the variable length window should not be too large. If the size is too large, the data in the window will be too much, resulting in the GEP spending too much time in finding function, which may be not satisfied with the real-time of the stream data. If the window length is too small, it will result in insufficient sample size, making it difficult for the GEP algorithm to accurately mine the inherent laws of the stream data.

### B. GEP algorithm

The GEP (Gene Expression Programming), through the genetic manipulation, such as natural selection, crossover and mutation, improves the evolution of the population and the individual fitness of the gene. GEP has the advantages of simple coding, flexible processing and fast convergence. It has achieved good results in function discovery, association rule mining, neural network optimization and classification [13].

In this paper, the GEP algorithm is used to establish the function model from the data and its timestamp in the variable length window. Then, the function model is used to predict. At the same time, the fitting root mean square error $h$ of the function model can be calculated. By substituting the timestamp of the next moment into the prediction model, the prediction value for the next moment can be calculated.

### C. Colony climbing algorithm

Considering the GEP algorithm is easy to fall into the local optimum, to overcome its shortcoming, we adopt the colony climbing algorithm [13], which has the advantage of effectively changing the current population framework and rapidly expanding the search space to adapt to the rapid changes of the time series stream data. The colony climbing algorithm uses the colony search strategy of the evolutionary computing to ensure the global searching. The method of using the colony climbing algorithm is described as follows.

When the new data arrives and the data in the variable length window alternates, the strategy of eliminating the inferiors is adopted. The individuals of the population are ranked according to the fitness from large to small, and the individuals with better fitness in the first half are retained, and the individuals with poor fitness in the latter half are eliminated. Then the individuals in the latter half are reinitialized. The new population is used as the initial population of the GEP algorithm to find function from the data of the next variable length window. The colony climbing algorithm not only improves the diversity of the population, but also retains the predictive model individuals with better adaptability.

### *D. Data Fusion*

Due to the time series stream data including noise, in order to reduce the influence of noise data, a fusion method is proposed in this paper. The fusion method is, when the new data $Z$ arrives, according to the best prediction model in the variable window of the previous time mined by the GEP algorithm and its fitting root mean square error $h$, substituting the current timestamp, and calculating the predicted value $P$ of the current time.

The root mean square error of the predicted data $Q$ is calculated as follows,

$$Q = \sqrt{h^2 + H^2} \tag{2}$$

Here, $H$ is the error of the fusion value at the previous moment, which is calculated by equation (5).

In actual, the noise mainly comes from the detection error of the sensor, so the noise mean square error is known. We set it as $R$, and the error of the actual value $Z$ obeys Gaussian distribution $N(0, R^2)$. The initial value of $H$ can be assigned a value greater than $R$ and less than $2R$, and the influence of the initial value will gradually disappear during the error transfer process.

$$K = Q^2/(Q^2 + R^2) \tag{3}$$

$$\hat{x} = (Z - P)K + P \tag{4}$$

In equation (3), $K$ is the weight gain, and in equation (4), $\hat{x}$ is the fusion value.

$$H = \sqrt{(Q^2(1-K))} \tag{5}$$

In equation (5), $H$ is the root mean square error of the fusion value. According to the predicted value $P$ of the model based on historical data mining and its prediction root mean square error $Q$, the current actual data $Z$ and its root mean square error $R$ are corrected. The fusion value $\hat{x}$ is then added to the fused time series stream.

## III. Algorithm Description

| | Algorithm step |
|---|---|
| 1 | Initialize parameters, variable length window data and population; |
| 2 | using the GEP algorithm (coding, decoding, natural selection, two-point cross, mutation) to model the data in the variable length window, if the search iteration time is less than or equal to the sampling interval，and calculating the next time prediction value $P$ according to the elite model； |
| 3 | Calculating the variable length sliding window length $M$ according to equation (1); |
| 4 | Calculating the fusion value $\hat{x}$ by the equation (2)(3)(4); |
| 5 | Calculating the root mean square error $H$ of the fusion value according to equation (5); |
| 6 | Dividing the fusion time series stream by the variable length window after the current time fusion value enters the fusion time series stream; |
| 7 | useing the GEP algorithm to model the data in the variable length window, if the search iteration time is less than or equal to the sampling interval; calculating the predicted value $P$ and the fitting root mean square error $h$ according to the elite model; |
| 8 | Using the group climbing algorithm; |
| 9 | If new data arrives, return step 3; else, algorithm end. |

## IV. Simulation Results

### *A. Test data and experiments*

We use the method of generating test data in literature [11], shown in Table1, to generate four data sets.

TABLE 1. TEST DATA SET

| set | function | noise |
|---|---|---|
| $S_1$ | $s_i = 0.5 * \sin(i * \pi/10) + 0.5 * sin(i * \pi/5) + N(\mu, \sigma)$<br>$i \in [1,4920]$ | μ=0，<br>σ =0.01 |
| $S_2$ | | μ=0，<br>σ =0.03 |
| $S_3$ | | μ=0，<br>σ =0.06 |
| $S_4$ | | μ=0，<br>σ =0.09 |

To verify the validity of the algorithm proposed in this paper, we compare it with the Hierarchical Temporal Memory (HTM) used in the literature [11]. The algorithm proposed in this paper is programmed by C++ in Microsoft Visual Studio 2017, and run by CPU Intel CoreTM$^{i5}$-3230M CPU@ 2.6 GHz, Memory 4GB, Hybrid Hard Disk 500GB, Windows 10 Professional operating system.

In order to evaluate the prediction accuracy of the proposed algorithm, the Mean Absolute Percentage Error (MAPE) value is used to measure the prediction accuracy of the algorithm. The MAPE value is defined as:

$$MAPE = \frac{\sum_{i=1}^{N} \left| s_i - \hat{s}_i \right|}{\sum_{i=1}^{N} \left| s_i \right|} \tag{6}$$

where $\hat{s}_i$ is the predicted value, $s_i$ is the actual value.

### B. Algorithm parameters

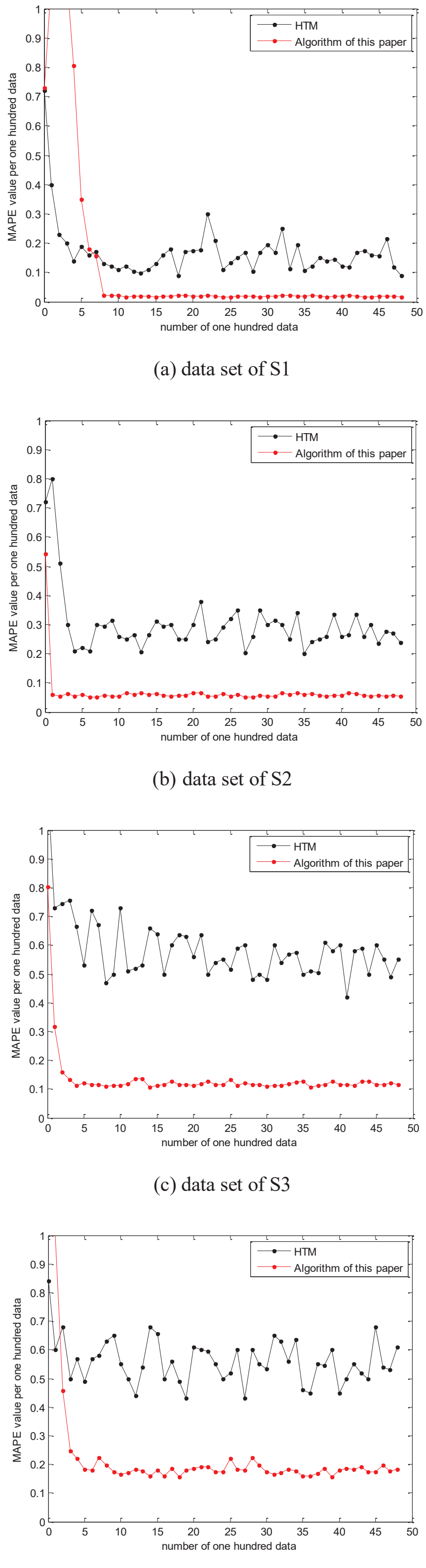
The parameter list of the algorithm is shown in Table 2. The $N$ in MAPE is 100, except for 20 data in the initial window, a total of 49 MAPE values. The interval between each data incoming is 1 second.

TABLE 2. ALGORITHM PARAMETER TABLE

| parameter | value | parameter | value |
|---|---|---|---|
| Fixed length window *m* | 20 | Variable length window initial *M* | 20 |
| Variable length window upper limit $M_{big}$ | 30 | Variable length window lower limit $M_{lit}$ | 10 |
| Population size | 100 | Gene length | 21 |
| Mutation rate | 0.1 | Two-point cross rate | 0.6 |
| Symbol set | *+,-,*,/,sin,cos, 1og10,sqrt,* timestamp *x* | Constant set | 0.1,0.2,0.5, 2,π,10 |

### C. Experiment results and analysis

The MPAE values of the two algorithms on the four test data sets is shown in Figure 2. From (a) and (b) of Figure.2, we can see that, the two algorithms have higher MAPE values in the initial, but both algorithms can effectively learn the internal law of data over time. However, because the HTM algorithm has not taken into account the noise in the data in time, its MAPE values fluctuate in a higher range, while the MAPE values of the proposed algorithm fluctuate in a lower range. From (c) and (d) of Figure.2, we can see that, in the high noise environment, the MAPE values of the HTM algorithm fluctuate in a higher range due to the inability to process the noise of the data in time, and the MAPE values of the proposed algorithm is still lower and fluctuate in a narrower range. Obviously, the proposed algorithm has better prediction performance than the HTM algorithm in the prediction of time series stream data with noise.



(a) data set of S1

(b) data set of S2

(c) data set of S3

(d) data set of S4

Figure 2. Comparison of the predicted MAPE values between two algorithms

To examine the real-time performance of the algorithm, we set the data incoming interval of the data set $S_3$ to 1 second, 0.8 second, and 0.6 second, respectively. The results are shown in Figure 3. From Figure 3, we can see that, in the early stage, the algorithm has larger prediction error for the data incoming interval of 0.6s and 0.8s compared to the data incoming interval of 1s, but in the later stage, the algorithm can accurately mine the prediction model and reduce the prediction error. The reason is that, with the data incoming interval being reduced, the time of the GEP running is shortened, which makes it difficult to find an accurate prediction model in a short time, resulting in a larger early prediction error.

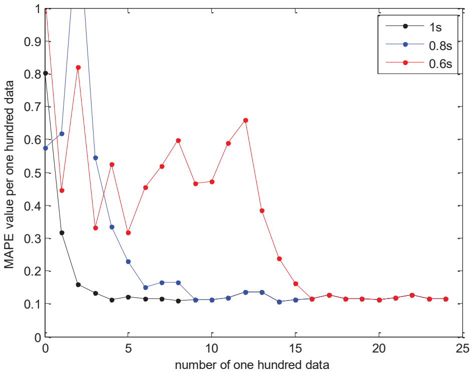


Figure 3. The real-time performance of algorithm

## V. Conclusion

In this paper, the algorithm of modeling and forecasting driven by time series stream data is proposed. The double sliding window is designed to divide the time series stream data to cope with the real-time requirement of processing the stream data. The GEP algorithm is used to find function model, and the data fusion method is used to reduce the influences of noise data on the modeling for enhancing the accuracy of model. The colony climbing algorithm is used to rapidly expand the search space of the population to respond to the rapid change of the stream data and to improve the prediction accuracy. The results of numerical simulation show that the proposed algorithm has better prediction performance than the HTM algorithm.

## References

[1] Araujo R, Ferreira T A E. An intelligent hybrid morphological-rank-linear method for financial time series prediction[J]. Neurocomputing, 2009, 72(10): 2507-2524.

[2] Cao Y P, Tian X M. Nonlinear system fault prognosis based on SVM and Kalman predictor[J]. Control and Decision (in Chinese), 2009, 24(3): 477-480.

[3] Che-Qing J , Wei-Ning Q , Ao-Ying Z . Analysis and Management of Streaming Data: A Survey[J]. Journal of Software (in Chinese), 2004, 15(8):1172-1181.

[4] LI Wen-zhong, ZUO Wan-li, HE Feng-ling. Entropy-based Algorithmin for Noise Deletction in Multi-dimensional Stream Data[J]. Computer science (in Chinese), 2012, 39(2):195-197.

[5] Yang H Q, Huang K Z, King I, et al. Localized support vector regression for time series prediction[J].Neurocomputing, 2009, 72(10): 2659-2669.

[6] Karlaftis M G, Vlahogianni E I. Statistical methods versus neural networks in transportation research: Differences, similarities and some insights [J]. Transportation Research Part C: Emerging Technologies, 2011, 19(3): 387-399.

[7] Etsy. Skyline[CP]. https: / /github. com /etsy / skyline, 2017.

[8] Rajagopalan V, Ray A. Symbolic time series analysis via wavelet-based partitioning[J]. Signal Processing, 2006, 86(11):3309-3320.

[9] Esling P, Agon C. Time-Series Data Mining[J]. ACM Computing Surveys, 2012, 45(1):1-34.

[10] Li S, Liu L J, Xie Y L. Chaotic prediction for short-term traffic flow of optimized BP neural network based on genetic algorithm [J], Control and Decision (in Chinese), 2011, 26(10): 1581-1585.

[11] Zeng Wei-ru, Wu Jia, Yan Fei. Time Series Anomaly Detection Model Based on Hierarchical Temporal Memory[J]. Acta Electronica Sinica (in Chinese), 2018, 46(2): 325-332.

[12] Ferreira C. Gene Expression Programming: a New Adaptive Algorithm for Solving Problems[J]. Complex Systens, 2001, 13(2):87-129.

[13] Li T , Tang C , Wu J , et al. GEP-NFM: Nested Function Mining Based on Gene Expression Programming[C]. International Conference on Natural Computation. IEEE, 2008.

[14] LU Xin-wei,CAI Zhi-hua. Application of a novel GEP algorithm in evolutionary modeling and forecasting[J]. Computer Applications (in Chinese), 2005, 25(12): 2783-2786.