A Fast Algorithm for Network Forecasting Time Series

FAN LIU¹, YONG DENG²

¹Yingcai Honors of school&School of Mathematical Sciences , University of Electronic Science and Technology of China, Chengdu, 610054, China (e-mail: liufanuestc@gmail.com)
²Institute of Fundamental and Frontier Science, University of Electronic Science and Technology of China, Chengdu, 610054, China (e-mail: dengentropy@uestc.edu.cn)

ABSTRACT  Time series has a wide range of applications in various fields. Recently, a new math tool, named as visibility graph, is developed to transform the time series into complex networks. One shortcoming of existing network-based time series prediction methods is time consuming. To address this issue, this paper proposes a new prediction algorithm based on visibility graph and markov chains. Among the existing network-based time series prediction methods, the main step is to determine the similarity degree between two nodes based on link prediction algorithm. A new similarity measure between two nodes is presented without the iteration process in classical link prediction algorithm. The prediction of Construct Cost Index (CCI) shows that the proposed method has the better accuracy with less time consuming.

INDEX TERMS  Time series, Visibility graph, Markov chains, Similarity measure, Link prediction, Construct Cost Index.

I. INTRODUCTION

Time series analysis and prediction is an important topic which is used in the fields of traffic [1], engineering [2], [3] complex networks [4]–[6] and so on [7]–[10]. Time series analysis can help analyze the characteristics of data and explore potential information. For example, predicting project costs can help individuals and organizations reduce costs and schedule projects. Construction cost index (CCI) is the weighted sum of the average price of labor, standard structural steel, Portland cement and wood, which is widely used in housing construction [11].

There are a lot of traditional time series forecasting methods including the stochastic [12], support vector machines (SVM) [13] and neural network methods [14]. Autoregressive Integrated Moving Average (ARIMA) [12] is a typical method in stochastic methods. ARIMA has several types, such as Autoregressive Moving Average (ARMA) [15]–[17], Autoregressive (AR) [18]–[20], Moving Average (MA) [21]–[23]. Neural Network [12], [14], [24] is a data-driven approach. SVM [13], [25], [26] has many applications in time series, such as regression, signal processing and time series analysis.

Recently, complex networks [27]–[31] have also been used for time series analysis [32], [33]. Many studies have shown that complex networks can help predict time series, and the effective information is mined by researching the complex networks [34]–[37]. Lacasa et al. [27] proposed a method to convert time series into a network, which can effectively present the internal structure of the network. Liu et al. [38] proposed a method to measure the similarity of two nodes, some scholars use this method for time series [39]–[41]. However, this method needs to iteratively calculate the similarity of two nodes, which is very time-consuming.

To address this issue, a fast method is proposed to calculate the similarity of two nodes without iteration. The proposed method is mainly divided into four steps. Firstly, the time series is converted into complex networks. Then the new method is used to calculate the similarity between the nodes. Finally the k nodes which are the most similar to the last node are determined, and the weights are calculated to make a prediction.

The structure of this article is as follows: preliminaries will be introduced in the Section 2. Section 3 will introduce the proposed method. The experiments will be carried out in the Section 4, and finally the summary will be given in Section 5.

II. PRELIMINARIES

In this section, some basic knowledge will be introduced including visibility graph [42] and transition probability [38]. A visibility graph [43], [44] algorithm is an algorithm that converts a time series into complex networks [27], [45]. The
constructed graph inherits several properties of the series. The periodic sequence is converted into a regular graph, and the random sequence is converted into a random graph.

**Definition 1** A time series is defined as \( Y = \{y_1, y_2, \ldots, y_N\} \), where \( i \Delta T \) and \( T \) is the index of time [46].

**Definition 2** Connectivity in time series is defined as follows [27].

\[
y_k < y_j + (y_i - y_j) \frac{j - k}{j - i} \quad \& \quad i < k < j
\] (1)

If the points \((i, y_i)\) and \((j, y_j)\) satisfy the above formula, then the two points are connected.

Note that the visibility graph has a total of three properties.
- Connected: Each adjacent point is connected.
- Undirected: the visibility is non-directional.
- Mapping in-variance: visibility criterion is invariant for both horizontal and vertical scaling and horizontal and vertical translation.

The transition probability is the key parameter in link prediction.

**Definition 3** The transition probability is defined by [38]

\[
P_{ij} = \frac{I_{ij}}{D_i}
\] (2)

Where \(P_{ij}\) is the probability that \(i\) turns to \(j\), \(D_i\) is the degree of point \((i, y_i)\), and

\[
I_{ij} = \begin{cases} 
1, & y_k < y_j + (y_i - y_j) \frac{i - k}{j - i} \quad \& \quad i < k < j \\
0, & \text{otherwise}
\end{cases}
\]

**III. PROPOSED METHOD**

This paper proposes a new method that uses the past \(N\) data points to predict the \((N+1)th\) data point.

The flow chart of the proposed method is shown in Figure 1.

**STEP 1** Convert time series \(Y = \{y_1, y_2, \ldots, y_N\}\) into complex networks according to Eq.(1). The visibility algorithm is an algorithm that converts time series into a visibility graph [27].

**STEP 2** Calculate the probability transfer matrix according to Eq.(2). The transition probability of two nodes can be regarded as a similar measure of two nodes. Zhang et al. [39] explained in detail why the transition probability can be used to measure the similarity of two nodes. However, the method proposed by Zhang et al. [39] is very time consuming. It is necessary to present a faster way.

**STEP 3** A method of similarity of new computational nodes is proposed based on stationary distribution of Markov chains in this step. The stationary distribution of the Markov chain is determined by the transition matrix and the initial distribution. Note that we assume that the time series \(Y = \{y_1, y_2, \ldots, y_N\}\) is a Markov chain.

In the method of time series prediction based on complex networks, a key step is to calculate the similarity between the \(Nth\) node and the previous \(N - 1\) nodes. Then, the value of the \((N + 1)th\) node is determined by the \(Nth\) node and the \(Jth\) node. Where the \(Jth\) node is the node most similar to the \(Nth\) node, which can be seen in Figure 2.

![Figure 2: Predict time series based on complex networks](image)

To the best of our knowledge, most current methods of calculating similarity are based on link prediction. The advantage of this method is that it can calculate the similarity of any two nodes in a complex network. However, isn’t the similarity of such compute nodes fully applicable to time series? In fact, it is only necessary to calculate the similarity between the \(Nth\) node and the previous \(N - 1\) nodes. This method uses the value of the stationary distribution of the probability transfer matrix as the similarity, which is presented as follows.

\[
V_{Nj}^T(j) = P_i^T V_0
\] (3)

Where \(V\) is the \(N\)-dimensional column vector, \(V_{Nj}^T(j)\) is the similarity between the \(Nth\) node and the \(Jth\) node, and \(V_0\) is the probability distribution of the initial nodes. In the proposed method, we set \(T\) ensuring that the \(\|V^{T+1} - V^T\|\) are less than \(10^{-5}\). In the link prediction, the similarity of any two nodes is calculated as follows [39].

\[
S_{ij}^{SRW} = \sum_{t=0}^{T} k_{i} P_t^T \pi_i + \frac{k_{j}}{2|E|} P_t^T \pi_j
\] (4)

Where \(S_{ij}^{SRW}\) is the similarity between node \(i\) and node \(j\), \(k_i\) and \(k_j\) are the degrees of nodes \(i\) and node \(j\), \(|E|\) is the number of edges in the network. Note that \(\pi\) is different from \(V\). \(\pi_i\) is an \(N\)-dimensional column vector, where the value...
Author et al.: Preparation of Papers for IEEE TRANSACTIONS and JOURNALS

of the $i$-th element is 1, and the value of the other elements is 0. \( V \) is a column vector of \( N \), representing the probability distribution of all nodes, where the values of all elements are between 0 and 1.

Normally, calculating the similarity between the \( N \)-th node and the previous \( N-1 \) nodes requires \( N-1 \) calculations based on link prediction. However, in the proposed method, it only needs to be calculated once [39].

It is noted that one property of the Markov chain is utilized in calculating the similarity of nodes: the limit distribution is the same as the stationary distribution. The process of certification is provided in Section VI.

**STEP 4** This step first looks for the top \( k \) maximum values in \( V \). Here, it is considered that the top \( k \) maximum values points are most similar to the \( N \)-th point. The final prediction can be calculated as follows.

\[
y_{N+1} = \sum_{r=1}^{k} \frac{V_r}{\sum_{j=1}^{k} V_j} (y_N - y_r) \left( \frac{(N + 1) - N}{V_j} + y_N \right)
\]

Where \( V_{Nj}^T (j) \) is recorded as \( V_j \).

**IV. APPLICATION IN CCI PREDICTION**

This section verifies the effectiveness of the method through CCI data. The CCI data set is the engineering cost data [47], [48], with a total of 295 data points. From January of 1990 to July of 2014. The flow chart of the whole experiment is shown below.

In order to assess the accuracy of the forecast, a total of 5 evaluation criteria are chosen including normalized root mean square error (NRMSE), mean absolute percentage error (MAPE), symmetric mean absolute percent error (SMAPE), mean absolute difference (MAD), Root Mean Squared Error (RMSE) [49].

**A. STEP BY STEP PREDICTION**

\[
NRMSE = \sqrt{\frac{1}{N} \sum_{i=1}^{N} \left| \hat{y}_i - y_i \right|^2}
\]

\[
MAPE = \frac{1}{N} \sum_{i=1}^{N} \left| \frac{\hat{y}_i - y_i}{\hat{y}_i} \right| \times 100
\]

\[
SMAPE = \frac{2}{N} \sum_{i=1}^{N} \left| \frac{\hat{y}_i - y_i}{\hat{y}_i + y_i} \right|
\]

\[
MAD = \frac{1}{N} \sum_{i=1}^{N} |\hat{y}_i - y_i|
\]

\[
RMSE = \sqrt{\frac{1}{N} \sum_{i=1}^{N} \left| \hat{y}_i - y_i \right|^2}
\]

The flow chart of the prediction is shown in Figure 3.

**Table 1** shows a comparison of the three methods including SMA [50], Zhang et al.’s [39] method and proposed method. It can be seen that the proposed method performs better than the previous method on three indicators.

![Flow chart of the prediction](image)

**Table 1: Comparison in three methods**

<table>
<thead>
<tr>
<th>Method</th>
<th>MAD</th>
<th>RMSE (%)</th>
<th>MAPE (%)</th>
<th>SMAPE (%)</th>
<th>NRMSE (%)</th>
</tr>
</thead>
<tbody>
<tr>
<td>SMA(k=1) [50]</td>
<td>21.59</td>
<td>32.73</td>
<td>0.3110</td>
<td>0.6350</td>
<td></td>
</tr>
<tr>
<td>Zhang et al. [39]</td>
<td>20.05</td>
<td>29.33</td>
<td>0.2889</td>
<td>0.5690</td>
<td></td>
</tr>
<tr>
<td>Proposed method</td>
<td>19.88</td>
<td>28.06</td>
<td>0.2899</td>
<td>0.5455</td>
<td></td>
</tr>
</tbody>
</table>

Figure 4 shows a comparison of real and predicted values in the CCI data set. It can be seen that the overall predicted value is very consistent with the real, but there is a partial error.
B. OUT-OF-SAMPLE PREDICTIONS

To further verify the effectiveness of the proposed method, the entire experimental process can be seen on Algorithm 1. The sliding window method is used to test. In the experiment, the sliding window was set to \( L = 3, L = 6, L = 12 \).

Algorithm 1

Require: Dataset, length of \( L \);
Ensure: Average error;
for \( i = 1 \) to \( (N - L - 2) \) do
    \( \text{Train dataset} = \{x|x(i), i = 1, \ldots, k + 1\} \);
    for \( j \) to \( L \) do
        \( \hat{y} = \text{Proposed method}(\text{Train dataset}(k + j + 1)) \);
    end for
    \( \text{Error} \);
end for
\( \text{Average error} \);

Table 2 shows the error of non-synchronization. It can be seen that the MAD, MAPE, SMAPE, and RMSE gradually increase with the increase of the step size, but the NRMSE decreases with the increase of the step size.

<table>
<thead>
<tr>
<th>( L )</th>
<th>MAD (%)</th>
<th>MAPE (%)</th>
<th>SMAPE (%)</th>
<th>RMSE</th>
<th>NRMSE</th>
</tr>
</thead>
<tbody>
<tr>
<td>3</td>
<td>30.8937</td>
<td>0.4522</td>
<td>0.4531</td>
<td>34.0824</td>
<td>185.618</td>
</tr>
<tr>
<td>6</td>
<td>42.1445</td>
<td>0.6222</td>
<td>0.6222</td>
<td>48.1535</td>
<td>73.4600</td>
</tr>
<tr>
<td>12</td>
<td>56.4889</td>
<td>0.8377</td>
<td>0.8414</td>
<td>65.6824</td>
<td>38.9931</td>
</tr>
</tbody>
</table>

Table 2: Errors comparison in three step size

Figure 4: The real values and predicted values of CCI

Figure 5: MAPE in three step size

Figure 5 shows that each error of the error MAPE is in three step size.

Figure 6: MAPE error \( L = 12 \) comparations with other methods

Figure 6 shows the comparison of different methods on MAPE [39]. Where the time consumption is smaller, the accuracy of the proposed method is the same as that of the Zhang et al.’ [39] method.

V. CONCLUSIONS

Based on the stationary distribution of visibility graph and markov chains [51], a more efficient method is proposed to predict the time series.

The proposed method is divided into four steps. The first step is to convert the time series into a visibility graph. The second step uses the visibility graph to calculate the probability transfer matrix. The third step uses the properties of the probability transfer matrix to solve the stationary distribution. The fourth step is based on finding the point with the highest \( \text{pro-}k \) probability as the most similar point to the \( Nth \), and the value of the (\( N + 1th \)) point is predicted according to the weighted coefficient.

The contribution of this paper is that a faster method to calculate similarity between \( Nth \) and previous \( N-1th \) nodes
is proposed. In predicting that the value of the N+1th node is jointly determined by the first k nodes, it seems to improve the accuracy.

VI. APPENDIX

Definition 4 The stationary distribution \( V = \{v_1, v_2, \ldots, v_N\} \) is defined as follows.

- \( \sum_{j \in E} v_j = 1 \).
- \( v_j = \sum_{i \in E} v_i P_{ij} \).

Where \( PV = V \) is the matrix form of the above formula.

Definition 5 If there is a limit, for all states \( i \) and \( j \), the limit distribution is defined as follows.

\[
\lim_{n \to \infty} P^n_{ij} = \Pi_j \quad \text{and} \quad \sum_{j \in E} \Pi_j = 1 \quad (11)
\]

Theorem 1 The stationary distribution of Markov processes is equal to the limit distribution:

Proof Assume that the limit distribution of the Markov chain \( X = \{x(n), n = 1, 2, \ldots\} \) is \((\Pi_1, \Pi_2, \ldots)\).

\[
\lim_{n \to \infty} P^n_{ij} = \Pi_j \quad (12)
\]

because \( P^n_{ij} = \sum_{r \in E} \Pi_r \cdot P_{rj} \). So when \( n \) tends to infinity.

\[
\Pi_j = \sum_{r \in E} \Pi_r \cdot P_{rj} \quad (13)
\]

References


FAN LIU is a Junior student at the Yingcai Honors college of Mathematics and Physics Basic Science(Academic Talent Program) at the University of Electronic Science and Technology of China. In December 2017, Fan was appointed as research assistant in Institute of Fundamental and Frontier Science at the University of Electronic Science and Technology of China. Fan worked with Prof. Deng here in Evidence theory. Fan’s research direction is in evidence theory, and he is interested in machine learning, quantum computing, and uncertainty analysis. Fan received a national scholarship and outstanding student scholarship in 2018.

YONG DENG received the Ph.D. degree in Precise Instrumentation from Shanghai Jiao Tong University, Shanghai, China, in 2003. From 2005 to 2011, he was an Associate Professor in the Department of Instrument Science and Technology, Shanghai Jiao Tong University. From 2010, he was a Professor in the School of Computer and Information Science, Southwest University, Chongqing, China. From 2012, he was a Visiting Professor in Vanderbilt University, Nashville, TN, USA. From 2016, he was also a Professor in School of Electronic and Information Engineering, Xi’an Jiaotong University, Xi’an, China. From 2017, he is the full professor of Institute of Fundamental and Frontier Science, University of Electronic Science and Technology of China, Chengdu, China. Professor Deng has published more than 200 papers in referred journals such as IEEE Trans. On Fuzzy Systems, IEEE Trans. On Cybernetics, Decision Support Systems, European Journal of Operational Research and Scientific Reports. His research interests include evidence theory, decision making, information fusion and complex system modelling. He severed as the program member of many conferences such as International Conference on Belief Functions. He severed as many editorial board members such as Academic Editor of the PLOS ONE and Applied Intelligence. He severed as the reviewer for more than 30 journals such as IEEE Transaction on Fuzzy Systems. Professor Deng has received numerous honors and awards, including the Elsevier Highly Cited Scientist in China in 2014 to 2018.

***