This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TFUZZ.2018.2831640, IEEE Transactions on Fuzzy Systems

Time Series Forecasting based on High-Order Fuzzy Cognitive Maps and Wavelet Transform

Shanchao Yang, Jing Liu, *Senior Member, IEEE*

***Abstract*—Fuzzy cognitive maps (FCMs) have been successfully used to model and predict stationary time series. However, it still remains challenging to deal with large-scale non-stationary time series which have trend and vary rapidly with time. In this paper, we propose a time series prediction model based on the hybrid combination of high-order FCMs with the redundant wavelet transform to handle large-scale non-stationary time series. The model is termed as Wavelet-HFCM. The redundant Haar wavelet transform is applied to decompose original non-stationary time series into multivariate time series, then the high-order FCM is used to model and predict multivariate time series. In learning high-order FCM to represent large-scale multivariate time series, a fast high-order FCM learning method is designed on the basis of ridge regression to reduce the learning time. Finally, summing multivariate time series up yields the predicted time series at each time step. Compared with existing classical methods, the experimental results on eight benchmark datasets show the effectiveness of our proposal, indicating that our prediction model can be applied to various prediction tasks.**

***Index Terms*—Fuzzy cognitive maps, high-order fuzzy cognitive maps, time series prediction, redundant Haar wavelet transform.**

I. Introduction

WITH the ability to handle uncertainty of data, fuzzy modeling approaches can effectively model complex systems, thus having found widespread applications, such as optimal control [1], [2], image retrieval [3], [4], fuzzy reference [5], [6] and time series prediction [7]–[12]. In inheriting the main properties of fuzzy logic and neural networks, fuzzy cognitive maps (FCMs), consisting of a set of nodes and directed edges between nodes, can model states of systems effectively [13]. Due to the powerful inference ability and high interpretability, FCMs have been widely applied in various scientific areas, such as computer vision tasks [14], e-commerce strategic planning [15], artificial emotions forecasting [16], pattern recognition [17]–[19], and gene regulatory network reconstruction [20]–[22].

Time series consist of a sequence of observations ordered in time, such as $\boldsymbol{x} = \{x_1, x_2, \ldots, x_t\}$. Given historical time series, estimating the value of x_{t+1} in the future yields the task of time series prediction. In the combination of fuzzy time series and evolutionary optimization, FCMs have been successfully applied to predict time series [23]–[30]. The original time series are transformed into fuzzy time series, which are used as historical data to develop FCMs. And then evolutionary algorithms (EAs) are used to optimize the parameters of FCMs to model and predict time series, achieving a low prediction error [31]–[35]. Evolution strategies were applied by Koulouriotis *et al.* in [31] to learn FCMs, which can construct the structure of FCMs from zero ground. Parsopoulos *et al.* in [32] introduced an FCM learning algorithm based on the particle swarm optimization method to detect proper weight matrices that lead the FCM to desired steady states. Stach *et al.* in [33] proposed an FCM learning strategy on the basis of a real coded genetic algorithm, which can generate FCM models almost perfectly representing the input data.

The redundant Haar wavelet transform has been well studied for analyzing time series in many works [36]–[38]. Aussem *et al.* in [36] proposed a wavelet-based feature decomposition strategy combined with a recurrent neural network for financial forecasting. Zheng *et al.* in [37] applied a multilayer perceptron to model and forecast the decomposition of the financial signal, demonstrating the advantage of the wavelet transform. Renaud *et al.* in [38] developed a multiresolution methodology for combined noise filtering and signal prediction, showing that wavelet transform can capture short-range and long-term dependencies powerfully.

Despite of the success of FCMs in modeling time series, these prediction methods based on FCMs still have two major limitations. First, the strategy of fuzzy time series suitable for stationary time series has limitations in dealing with non-stationary time series, due to the fact that non-stationary time series change rapidly with time and often have trends. Fuzzy time series cannot capture the patterns of change and trend, unless these two patterns exist in the training dataset, which means that re-learning the parameters of FCMs is necessary when new data come in. The same problem is also observed by Salmeron *et al.* in [23], where they recommended to use this forecasting approach only for time series that are linear and tend to be trend stationary. Unfortunately, most time series in reality are non-stationary. Second, the widely used FCM learning methods based on EAs are inefficient to handle large-scale time series, due to the fact that EAs optimize problems iteratively and have to evaluate fitness functions repeatedly, resulting in a slow learning process. Thus, existing FCM learning methods can only handle small-scale time series, cannot guarantee the solution's quality and demonstrate inefficiency when dealing with large-scale time series. Thus, it is necessary to seek for a different method to transform time series into multivariate time series and develop

This work was supported in part by the Outstanding Young Scholar Program of National Natural Science Foundation of China (NSFC) under Grant 61522311, in part by the General Program of NSFC under Grant 61773300, and in part by the Key Program of Fundamental Research Project of Natural Science of Shaanxi Province, China under Grant 2017JZ017.

The authors are with the Key Laboratory of Intelligent Perception and Image Understanding of Ministry of Education, Xidian University, Xi'an 710071, China. For additional information regarding this paper, please contact Jing Liu (corresponding author, email: neouma@163.com).

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TFUZZ.2018.2831640, IEEE Transactions on Fuzzy Systems

a fast FCM learning algorithm to deal with large-scale time series.

To overcome these two limitations of FCMs in modeling time series described above, we develop a forecasting framework by the hybrid combination of FCMs with the redundant Haar wavelet transform, which is termed as Wavelet-HFCM. Our proposed forecasting framework consists mainly of two steps. The wavelet transform is used to decompose the original time series into multivariate time series, instead of the usage of strategy of fuzzy time series, then these multivariate time series serve as historical data to develop the structure of FCM. And then, to deal with large-scale time series, a fast FCM learning method based on ridge regression [39] is proposed, which can optimize the parameters of FCMs efficiently. Once the structure of FCM is identified, the FCM can be used to predict future values. Eight publicly available datasets are used to validate the performance of Wavelet-HFCM and the experimental results indicate that Wavelet-HFCM can model and predict large-scale non-stationary time series with high accuracy and efficiency. The comparison with existing algorithms shows the good performance of our proposal.

The proposed Wavelet-HFCM exhibits the originality in several ways. First, to the best of our knowledge, Wavelet-HFCM is the first prediction model that combines FCMs with wavelet transform, achieving high prediction accuracy by taking the advantages of fuzzy logic and wavelet theory. Second, a fast FCM learning algorithm based on ridge regression is proposed, which can be used to optimize parameters of FCMs effectively when dealing with large-scale time series. Compared with the existing FCM learning methods that can efficiently handle time series with no more than one hundred data points, Wavelet-HFCM can learn FCMs to represent large-scale time series with hundreds of or even thousands of data points efficiently.

The remainder of this paper is organized as follows. Section II presents the basic knowledge of FCMs and the related applications on time series forecasting. Section III introduces the details of the proposed prediction model and the learning algorithm for FCMs based on ridge regression. Section IV describes the comprehensive experimental evaluation and the discussion about the obtained results. Finally, Section V gives the conclusions and future research directions.

II. Related Work

Forecasting time series with high accuracy has been a challenging problem. In this section, we first introduce the basic knowledge of FCMs, and then give a brief overview of existing methods based on FCMs for time series prediction.

A. Fuzzy Cognitive Maps

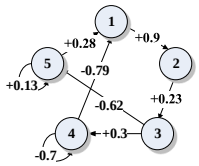
Using a set of concepts and fuzzy causal relations among concepts to model states of systems, FCMs can be represented in the form of weighted directed graphs. An FCM includes N_c concept nodes, and state values of these concepts are denoted as a vector $\mathbf{A}$,

$$\mathbf{A} = [A_1, A_2, \ldots, A_{N_c}] \tag{1}$$

where $A_i \in [0, 1]$ (or $A_i \in$[-1, 1]), $i = 1, 2, \ldots, N_c$. The state value of each node means the activation degree of the node, and the relationships between nodes are defined as an $N_c \times N_c$ weight matrix W.

$$\mathrm{W} = \begin{bmatrix} w_{11} & w_{12} & \cdots & w_{1N_c} \\ w_{21} & w_{22} & \cdots & w_{2N_c} \\ \vdots & \vdots & \ddots & \cdots \\ w_{N_c1} & w_{N_c2} & \cdots & w_{N_cN_c} \end{bmatrix} \tag{2}$$

where w_{ij} stands for the relationship originating from the jth node and pointing to the ith node, and takes the value from the range of [-1, 1], where $i, j = 1, 2, \ldots, N_c$.



(a)

$$\mathbf{W} = \begin{bmatrix} 0 & 0 & 0 & -0.79 & 0.28 \\ 0.9 & 0 & 0 & 0 & 0 \\ 0 & 0.23 & 0 & 0 & 0 \\ 0 & 0 & 0.3 & -0.7 & 0 \\ 0 & 0 & -0.62 & 0 & 0.13 \end{bmatrix}$$

(b)

Fig. 1. A toy example FCM with 5 nodes: (a) graphic structure, (b) weight matrix.

Fig. 1 shows a simple example FCM with five nodes. For example, $w_{43} = 0.3$ indicates node 3 has positive impact on node 4 with 0.3 strength. Similarly, $w_{14} = -0.79$ means node 4 has negative impact on node 1 with 0.79 strength. The state value of the node at the (t+1)th iteration is determined by the weight matrix and state values of all connected nodes at the tth iteration. Thus, the dynamics of FCMs can be described by the following equation,

$$A_i(t+1) = \psi(\sum_{j=1}^{N_c} w_{ij} A_j(t)) \tag{3}$$

where $A_i(t)$ is the activation level of node i at the tth iteration, and ψ is the transfer function mapping the activation level into unit range.

In practice, among various transfer functions used in FCMs, the sigmoid function works well in most cases. However, when the node's value can be negative, it is necessary to use the $\tanh$ function defined as follows,

$$\tanh(x) = \frac{e^x - e^{-x}}{e^x + e^{-x}} \tag{4}$$

In FCMs, the state value of each node at the (t+1)th iteration only depends on state values of all connected nodes at the tth iteration, which makes FCMs cannot model the long temporal dependencies and limits the modeling ability of FCMs. Thus, high order fuzzy cognitive maps (HFCM) have been proposed to enhance the approximation ability of FCMs [40]. A k–order HFCM is expressed as follows,

$$A_i(t{+}1){=}\psi\Big(\sum_{j=1}^{N_c} w_{ij}^1 A_j(t){+}w_{ij}^2 A_j(t{-}1){+}\cdots{+}w_{ij}^k A_j(t{-}k{+}1){+}w_{i0}\Big) \tag{5}$$

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TFUZZ.2018.2831640, IEEE Transactions on Fuzzy Systems

where w_{ij}^k stands for the relationship originating from the jth node and pointing to the ith node at time step $t+1-k$, and w_{i0} is the constant bias related to the ith node.

B. Related Work on Time Series Forecasting using FCMs

Existing prediction methods based on FCMs [24]–[30], [41], [42] share the same pattern: first the given numerical time series are transformed into fuzzy values via various fuzzification processes, then FCMs serve as prediction models. For example, Stach *et al.* in [26] exploited FCMs with granular, fuzzy-set-based model of inputs. Membership values were calculated based on the predefined linguistic descriptors, then these values were used to learn FCMs by real-coded genetic algorithms. Finally, the predicted fuzzy values were defuzzified to obtain numerical values. Pedrycz in [27] proposed an algorithmic way to predict time series based on FCMs. Fuzzy clustering algorithm was used to form the architecture of FCMs, then parameters of FCMs were optimized by the particle swarm optimization method. Song *et al.* in [28] proposed a fuzzy neural network to enhance the learning ability of FCMs. Lu *et al.* in [29] developed a framework to model and predict time series based on HFCMs and fuzzy c-means clustering algorithm. Salmeron *et al.* in [23] proposed an approach to optimize the structure of FCMs dynamically for time series prediction and recommended to use their FCM-based approach for forecasting linear and trend stationary time series. Pedrycz *et al.* in [30] introduced a mechanism to represent a numeric time series in terms of information granules and optimized information granules at the parametric level with the use of particle swarm optimization. Froelich *et al.* in [41] proposed a new approach for granular modeling of time series. The time series were approximated as the sequence of granules, forming information granules that were predicted by the FCM. Papageorgiou *et al.* in [42] proposed a two-stage time series prediction model on the basis of FCMs and artificial neural networks and demonstrated the usefulness of the new two-stage approach in time series prediction. Vanhoenshoven *et al.* in [43] proposed a hybrid approach for time series forecasting based on neuro-fuzzy prediction models and FCMs.

III. Wavelet-HFCM

A. Overall Design Process

In HFCM, each node represents a meaningful concept, and all nodes function together to model the system by interacting with each other. However, original time series are numeric with one dimension, which correspond to only one node in HFCM. By using the redundant Haar wavelet transform to decompose original time series into multivariate time series, a collection of meaningful concepts (nodes) are obtained, forming the structure of HFCM. The transformed multivariate time series are used by HFCM to predict $x(t+1)$, instead of the raw past observations $\{x_1, x_2, \ldots, x_t\}$.

Fig. 2 shows the procedure using Wavelet-HFCM for time series forecasting. First, the original time series are normalized into the range of [-1, 1], then the redundant Haar wavelet transform is used to transform numeric data into a set of

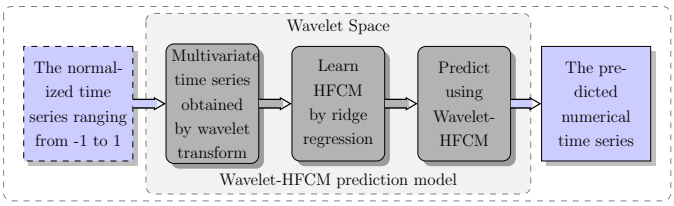


Fig. 2. Procedure of time series prediction using Wavelet-HFCM.

nodes of HFCM. And then, a fast HFCM learning algorithm is proposed to optimize the weight matrix of HFCM. The developed HFCM can predict future values based on historical data available. Finally, at each time step, summing values of all nodes up determines the predicted value of original time series. The details of Wavelet-HFCM are summarized in the following subsections.

B. Wavelet Representation of Time Series

A wavelet decomposition provides analysis of a signal both in time and frequency, allowing decision-makers to observe time series in different resolution levels. The decimation effect of discrete wavelet transform makes it not suitable for forecasting time series, owing to different length of wavelet coefficient sequences in different resolution levels. However, this problem can be addressed by the redundant Haar wavelet transform, where the length of wavelet coefficients at each resolution level is the same with the length of input time series. Since the calculation of wavelet coefficients is updating with time, the transform can cope with the problems of modeling non-stationary time series discussed previously.

Given decomposition level p, the input time series $c_0(k)$ can be expressed in the addition form of smoothed residual (c_p) and wavelet coefficients (d_j) .

$$c_0(k) = c_p(k) + \sum_{j=1}^{p} d_j(k) \tag{6}$$

where k is the time point index and j is the resolution level.

Given the resolution level j, the smoothed data $c_j(k)$ at iteration k can be determined by the following equation.

$$c_j(k) = \frac{1}{2}[c_{j-1}(k) + c_{j-1}(k - 2^{j-1})] \tag{7}$$

The wavelet transformation d_j for the resolution level j is obtained by calculating the signal difference between two successive resolution levels:

$$d_j(k) = c_{j-1}(k) - c_j(k) \tag{8}$$

Equation (6) provides a reconstruction formula for the original signal. At each scale j, a set $\{d_j\}$ called a wavelet scale can be obtained. The wavelet scale has the same number of sample points as the signal. At one time point k, we never use information after k to calculate the wavelet coefficients. This means that it will be not necessary to re-compute the wavelet transform coefficients of the full signal, if new measurements come in.

The redundant wavelet transform of original time series yields the historical time series data to learn HFCMs. For

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TFUZZ.2018.2831640, IEEE Transactions on Fuzzy Systems

example, given wavelet decomposition level $N_c - 1$, the original time series are transformed into total N_c sets of data ($N_c - 1$ sets of d_j plus one set of residual c_p). Each set of data corresponds to each distinct node in the HFCM.

To illustrate the internal process of wavelet transform, Fig. 3 shows a clear process of calculating the last wavelet coefficients of redundant Haar wavelet transform in each scale, with the decomposition level being four.

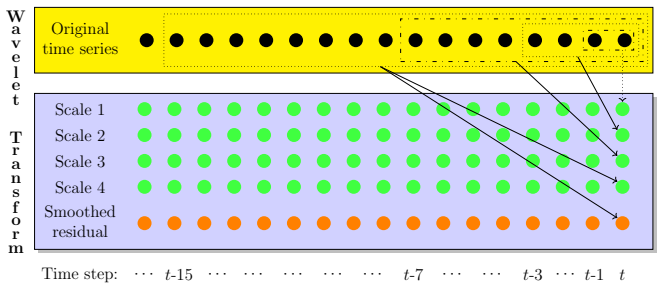


Fig. 3. Redundant Haar wavelet transform with the decomposition level being 4. This diagram shows which time steps of original time series are used to calculate the last wavelet coefficient in each scale.

C. Learning HFCM via Ridge Regression

With the multivariate time series obtained by using the redundant Haar wavelet transform, the weight matrix of HFCM can be learned from multivariate time series. Without loss of generality, the 2-order HFCM is taken as an example to show how to learn HFCM via our proposed learning method. This method can be easily extended to learn HFCM with arbitrary order. Wu *et al.* [44], [45] pointed out that the problem of learning weight matrix of FCMs can be simplified into learning local connections of nodes individually, as shown in Fig. 4. The same decomposition strategy is also used in our HFCM learning methods. After decomposing an HFCM into local connections, the weights of these connections are optimized using the following proposed method. Finally, with local connections of all nodes known, they can be combined to form the whole FCM weight matrix.

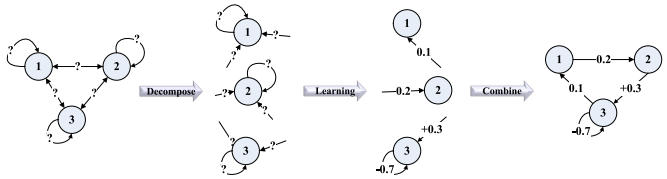


Fig. 4. Procedure of learning a toy example FCM with three nodes.

The non-linear dynamic equation (5) of HFCM can be linearized by the following inverse transformation,

$$\psi^{-1}(A_i(t+1)) = \sum_{j=1}^{N_c} \left(w_{ij}^1 A_j(t) + w_{ij}^2 A_j(t-1) + w_{i0} \right) \quad (9)$$

where ψ^{-1} is the inverse function of the transfer function ψ.

Once the time series with length L at different time step t are available, the transformed dynamic equation (9) can be rewritten in the vector form:

$$Y_i = \mathbf{X} W_i \quad (10)$$

where Y_i is a vector containing $\psi^{-1}(A_i(t+1))$ through all time, $\mathbf{X}$ is the state matrix concerning all state $A_j(t)$ of nodes at different t, and W_i is the weight vector between all nodes and the ith node. The details of these three variables are expressed as follows,

$$Y_i = \begin{bmatrix} \psi^{-1}(A_i(3)) \\ \psi^{-1}(A_i(4)) \\ \vdots \\ \psi^{-1}(A_i(L)) \end{bmatrix} \quad (11)$$

$$\mathbf{X} = \begin{bmatrix} A_1(2) & A_1(1) & A_2(2) & A_2(1) & \dots & A_{N_c}(2) & A_{N_c}(1) & 1 \\ A_1(3) & A_1(2) & A_2(3) & A_2(2) & \dots & A_{N_c}(3) & A_{N_c}(2) & 1 \\ \vdots & \vdots & \vdots & \vdots & \ddots & \vdots & \vdots & \vdots \\ A_1(L-1) & A_1(L-2) & A_2(L-1) & A_2(L-2) & \dots & A_{N_c}(L-1) & A_{N_c}(L-2) & 1 \end{bmatrix} \quad (12)$$

$$W_i^T = \begin{bmatrix} w_{i1}^1 & w_{i1}^2 & w_{i2}^1 & w_{i2}^2 & \cdots & w_{iN_c}^1 & w_{iN_c}^2 & w_{i0} \end{bmatrix} \quad (13)$$

Different from the FCM learning method proposed by Wu *et al.* [44], [45] with the usage of lasso regression, ridge regression is adopted to solve the following optimization problem to make Wavelet-HFCM be more capable of generalization, determining the local connections of the ith node.

$$\min_{W_i} \{ \frac{1}{2L} \| Y_i - \mathbf{X} W_i \|_2^2 + \alpha \| W_i \|_2 \} \quad (14)$$

where $\|W_i\|_2 = \sqrt{\sum_k W_{ik}^2}$, and α is a non-negative regularization parameter.

Solving (14) determines the weight vector W_i between all nodes and the ith node. Lots of free software can solve this problem efficiently. Here, the ridge regression algorithm provided by the Python library Scikit-learn is adopted, and more details can be found in [46]. Repeating the process above N_c times for total N_c nodes yields all weight vectors, and the latter can be combined to form the weight matrix. For a k-order HFCM with N_c nodes, the learning process takes $O(Lk^2N_c^2)$ for each node, so learning an HFCM to achieve a high-quality solution needs $O(Lk^2N_c^3)$ in total.

IV. Experiments and Discussion

The experiments consist of two aspects —— analyze the effect of three hyper-parameters on the performance of Wavelet-HFCM and validate Wavelet-HFCM's performance.

A. Data Sets

We use eight benchmark time series with different characteristics to demonstrate the effectiveness of proposed model. The first one is the sunspot time series that records the annual number of sunspots, which consists of total 289 observations from 1700 to 1987 and was often used by other researchers [29], [47]–[51]. The second dataset concerns daily open prices of S&P 500 stock index from June 1, 2016 to June 1, 2017, total 251 observations on record, which has been used in related publications [52]–[57], and can be obtained from the Yahoo finance website [58]. The third dataset records the monthly milk production in pound from January 1962 to

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TFUZZ.2018.2831640, IEEE Transactions on Fuzzy Systems

December 1975, consisting of 168 observations totally [59]. The fourth dataset records the monthly closing of the Dow-Jones industrial index from August 1968 to August 1981, concerning 291 observations in total [60]. The fifth dataset consists of 240 observations reflecting the highest radio frequency that can be used for broadcasting in Washington, D.C over the period of May 1934 to April 1954 [61]. The sixth dataset concerning 192 observations [62] and the seventh dataset concerning 600 observations [63], record CO_2 (ppm) at Mauna Loa from 1965 to 1980 and monthly Lake Erie level from 1921 to 1970, respectively. The last dataset, chaotic Mackey-Glass time series [64] (MG time series for short) can be generated from the first-order nonlinear differential-delay equation defined by (15). This time series has been a benchmark to validate the effectiveness of prediction models [24], [29], [47], [65]–[67]. In this experiment, the MG time series dataset with 1000 points from t = 124 to t = 1123 was obtained by solving (15) using the fourth order Runge-Kutta algorithm. Moreover, to be consistent with the literature, initial condition $x(0)$ is set to 1.2 and τ is set to 17,

$$\dot{x}(t) = \frac{0.2x(t-\tau)}{1+x^{10}(t-\tau)} - 0.1x(t) \tag{15}$$

B. Experimental Setup

The type of transfer function of HFCM controls the range of the output value. Since wavelet coefficients can be negative, the $\tanh$ function is adopted as the transfer function, which makes the predicted value be in [-1, 1]. Thus, it is necessary to normalize each time series into the range of [-1, 1] before feeding them into Wavelet-HFCM for modeling and prediction. Furthermore, normalization avoids numerical difficulties during the calculation and makes the learning process of HFCM more trivial. Here, the max-min normalization method is adopted. Suppose the maximal and minimal value of original time series are x_{max} and x_{min}, respectively, the highest and lowest value of normalized time series are x_{high} and x_{low}, respectively, then the time series are normalized into the range $[x_{low}\ ,\ x_{high}]$ from the range $[x_{min}\ ,\ x_{max}]$ using (16).

$$\bar{x} = \frac{x - x_{min}}{x_{max} - x_{min}}(x_{high} - x_{low}) + x_{low} \tag{16}$$

where x and $\bar{x}$ is the original time series and normalized time series, repectively.

The classic measure metric —— RMSE is used to evaluate the forecasting performance of Wavelet-HFCM, which calculates the difference between the original and the predicted time series.

$$\text{RMSE} = \frac{\sqrt{\sum_{i=1}^{L}(x_i - \bar{x}_i)^2}}{L^2} \tag{17}$$

where L is the length of time series, and x_i and $\bar{x}_i$ represent the predicted time series and the normalized original time series, respectively.

Wavelet-HFCM's forecasting performance is influenced by three hyper-parameters, namely, the order k, the number of nodes N_c and the regularization factor α. Although default values of first two parameters work well in most cases (k = 2, N_c = 5), it is still unknown beforehand which hyper-parameters are the best for the given time series. Thus, the grid search approach is used as the validation procedure to perform the model selection [67]. The normalized time series data are split into three subsets in sequence: training dataset, validation dataset and test dataset. The training dataset is used to learn the weight matrix of HFCM prediction model, the validation dataset is used to select the best model and the test dataset is used to evaluate the forecasting accuracy. The validation procedure can reduce the risk of overfitting and make the prediction model capable of better generalization.

To select the best HFCM model to represent time series, the following experimental scheme is designed to obtain optimal hyper-parameters: (1) let the order k be 1; (2) increase the value of N_c from 2 to 7 and find the best α which has the highest prediction accuracy on the validation dataset and record RMSE on the test dataset; (3) increase order k by 1, then go to Step (2) until k is 6. Since the learning algorithm of Wavelet-HFCM is deterministic, it is not nessessary to repeat the experiments. The robustness of learning algorithm makes Wavelet-HFCM more efficient. The source code of Wavelet-HFCM can be accessed at [68] for non-commercial used only. Table I summarizes the size of each subset of each time series.

TABLE I
SIZE OF EACH SUBSET

	Training	Validation	Test
Sunspot time series	177	44	67
MG time series	400	100	500
S&P 500 stock index	120	30	101
Monthly milk production per cow	108	26	34
Monthly closings of the Dow-Jones industrial index	175	43	73
Monthly critical radio frequencies	184	36	20
CO_2 (ppm) at Mauna Loa	131	32	29
Monthly Lake Erie levels	368	92	140

C. A Case Study of Wavelet-HFCM on Sunspot Time Series

This section demonstrates a step-by-step design of how Wavelet-HFCM is actually applied to predict time series, taking the sunspot time series as an example.

First, the original sunspot time series are normalized into the range of [-1, 1] via the max-min normalization method defined by (16). Second, to select the most robust HFCM to represent time series, the normalized sunspot time series are split into training dataset, validation dataset and test dataset according to a given ratio. The ratio may vary with the characteristics of different prediction tasks, and can be set to 60% : 15% : 25% as the default value. Third, wavelet coefficients of sunspot time series, obtained via (6)-(8), serve as historical data to learn HFCM. The number of nodes of HFCM is equal to the wavelet decomposition level plus one. With the structure of HFCM identified, parameters of HFCM are optimized by using the proposed learning method based on ridge regression. Then, the learned HFCM can be used for forecasting each scale of

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TFUZZ.2018.2831640, IEEE Transactions on Fuzzy Systems

wavelet coefficients. Finally, summing the predicted wavelet coefficients up in each scale at each time point determines the final prediction value of sunspot time series.

To illustrate the outcome of wavelet transform, the wavelet coefficients of sunspot time series are visualized in Fig. 5. Fig. 5 shows that wavelet coefficients in each scale, inheriting a unique characteristic from normalized time series, are more smooth than normalized time series, which makes the prediction process more trival and HFCM more capable to achieve higher accuracy. The learned 2-order HFCM of sunspot time series is illustrated in the form of multilayer networks [69], as visualized in Fig. 6.

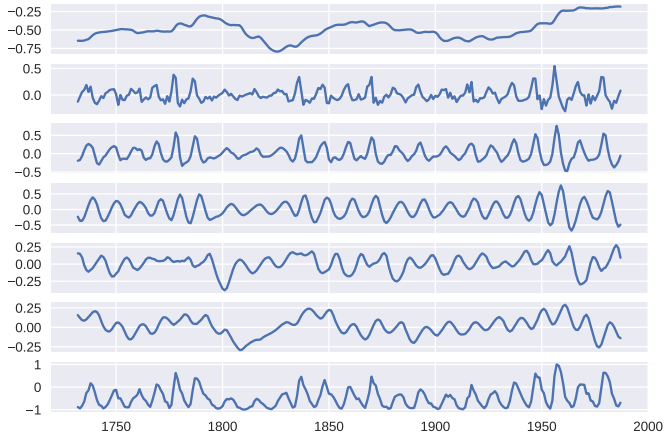


Fig. 5. Wavelet coefficients of sunspot time series. From top to bottom: d_1, d_2, d_3, d_4, d_5, $c5$ (smoothed residual) and c_0 (original time series), respectively.

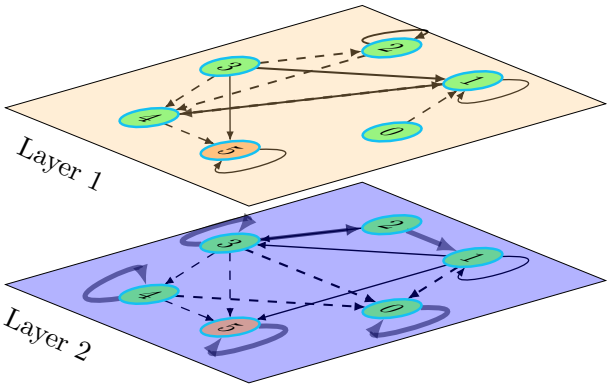


Fig. 6. The learned 2-order HFCM of sunspot time series with six nodes, visualized in the form of two-layer networks.

D. Effect of Hyper-Parameters on Wavelet-HFCM

This section studies the effect of three hyper-parameters on Wavelet-HFCM. To make the prediction error of different datasets comparable, normalized RMSE is adopted here to evaluate the performance of Wavelet-HFCM.

1) Effect of N_c on Wavelet-HFCM: The first set of experiments study the influence of N_c on the prediction accuracy, taking sunspot time series and S&P 500 stock index as examples. With different wavelet decomposition levels of time series stemming from different N_c, time series can be observed and analyzed at different resolution levels. Fig. 7 exemplifies RMSE versus N_c throughout these two datasets. From Fig. 7(a) to (f), we can see that the prediction accuracy is highly sensitive to the choice of the number of nodes. For the sunspot time series, the prediction error gets substantially lower when increasing N_c until N_c goes beyond a certain value, say 5 or 6. This phenomenon means the prediction error does not decrease continuously with the increasing N_c. The significant decrease in the prediction error happens when N_c moves from very low values to some higher values, while the prediction error slightly increases when the value of N_c is greater than 5 or 6. The same situation can be also observed in the S&P 500 stock index time series: when increasing N_c by one from 2 to 6, the prediction error goes lower gradually until RMSE reaches its minimum value when N_c = 6. Then RMSE becomes higher when increasing N_c again. We can draw conclusions that N_c affects the performance of Wavelet-HFCM and can be set to 5 or 6 as the default value before the further fine-tuning using the validation procedure.

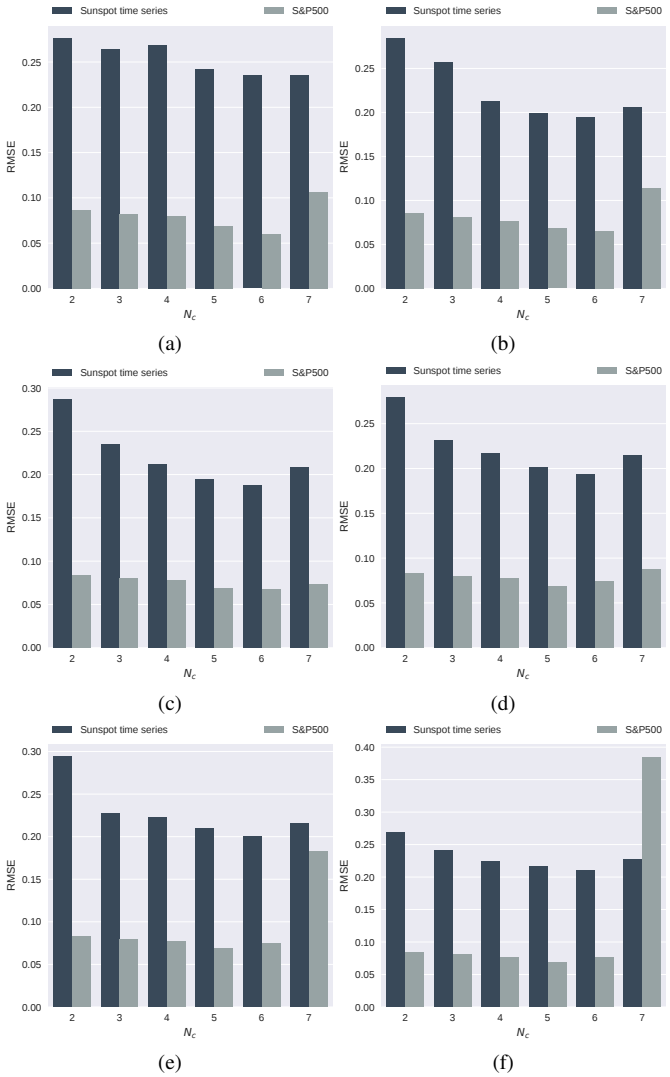


Fig. 7. RMSE versus varying number of nodes of Wavelet-HFCM with (a) k =1, (b) k =2, (c) k =3, (d) k =4, (e) k =5, and (f) k =6.

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TFUZZ.2018.2831640, IEEE Transactions on Fuzzy Systems

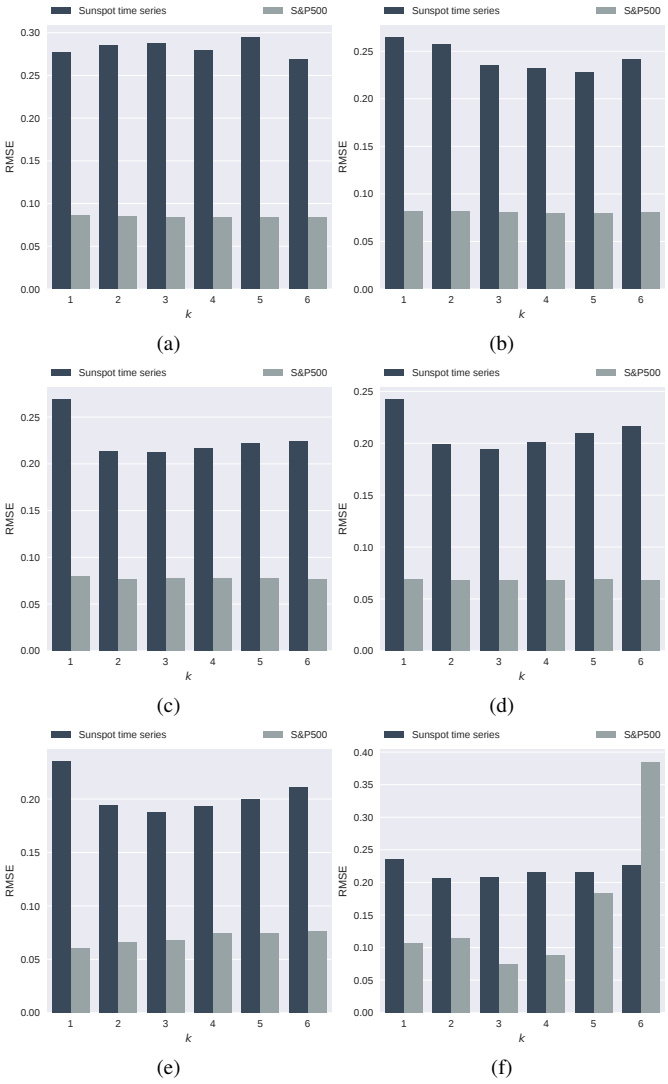


Fig. 8. RMSE versus varying order k of Wavelet-HFCM with (a) N_c = 2, (b) N_c = 3, (c) N_c = 4, (d) N_c = 5, (e) N_c = 6, and (f) N_c = 7.

2) Effect of k on Wavelet-HFCM: Fig. 8 shows RMSE versus different order k of Wavelet-HFCM when N_c varies from 2 to 7. We can see that for both time series, the order k of HFCM slightly influences the forecasting performance of Wavelet-HFCM, except for the outlier of S&P 500 index in Fig. 8(f). For the sunspot time series, when N_c is 2, the prediction error fluctuates randomly with the increased k and does not show a fixed pattern. When N_c is greater than 2, k = 2 and k = 3 are appropriate points which correspond to lower RMSE. RMSE does not become lower all along with the increasing order of HFCM. This situation can be also found in the experiment on the S&P 500 stock index time series. Thus, we can draw conclusions that the prediction model is insensitive to the value of order of Wavelet-HFCM, and can be set to 2 or 3 as the default value before the further fine-tuning using the validation procedure.

To facilitate the analysis of the experiment on the MG time series, Fig. 9 shows its RMSE versus varying k and N_c of Wavelet-HFCM. We can clearly see that the prediction error

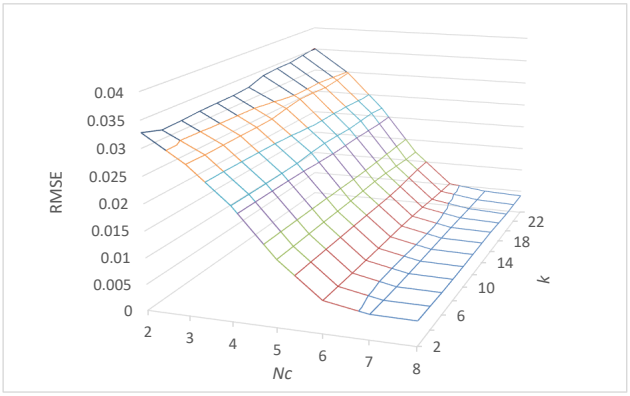


Fig. 9. RMSE versus varying order k and N_c of HFCM on MG chaos time series.

TABLE II
OPTIMAL α OF WAVELET-HFCM USED FOR FORECASTING TIME SERIES

	α
Sunspot time series	1e-20
MG time series	1e-20
S&P 500 stock index	1e-20
Monthly milk production per cow	1e-20
Monthly closings of the Dow-Jones industrial index	1e-20
Monthly critical radio frequencies	1e-20
CO_2 (ppm) at Mauna Loa	1e-20
Monthly Lake Erie levels	1e-14

just slightly changes when increasing order k with fixed N_c, and gradually gets lower when increasing N_c with fixed order k. These two outcomes agree with our previous results on the two other datasets. A possible explanation for this is that for large-scale time series as MG chaos time series, with larger N_c giving more details via the wavelet decomposition, Wavelet-HFCM can fully learn the pattern of time series. We also found when N_c is fixed, RMSE is slightly lower with the increased order k until k reaches 23. This is caused by the characteristic of the generation equation, and agrees with the fact that many researchers used only four past observations $[x(t-24), x(t-18), x(t-12), x(t-6)]$ to predict $x(t)$ [24], [65].

3) Effect of α on Wavelet-HFCM: The regularization factor α balances the prediction error on training dataset and the generalization ability on test dataset, and larger values specify stronger regularization. Table II records the optimal α for each dataset obtained via the grid search method. For monthly Lake Erie levels time series, the optimal α is 1e-14, while the optimal α is 1e-20 for seven other time series. The results come with no surprise and agree with the characteristics of each time series. We can draw the conclusions that α near zero (1e-20) is optimal for near stationary time series and time series with a linear trend, and that α needs to be greater (1e-14) for the time series with a quadratic upward or downward trend as monthly Lake Erie levels time series. These conclusions scan help decision-makers choose the optimal α easily.

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TFUZZ.2018.2831640, IEEE Transactions on Fuzzy Systems

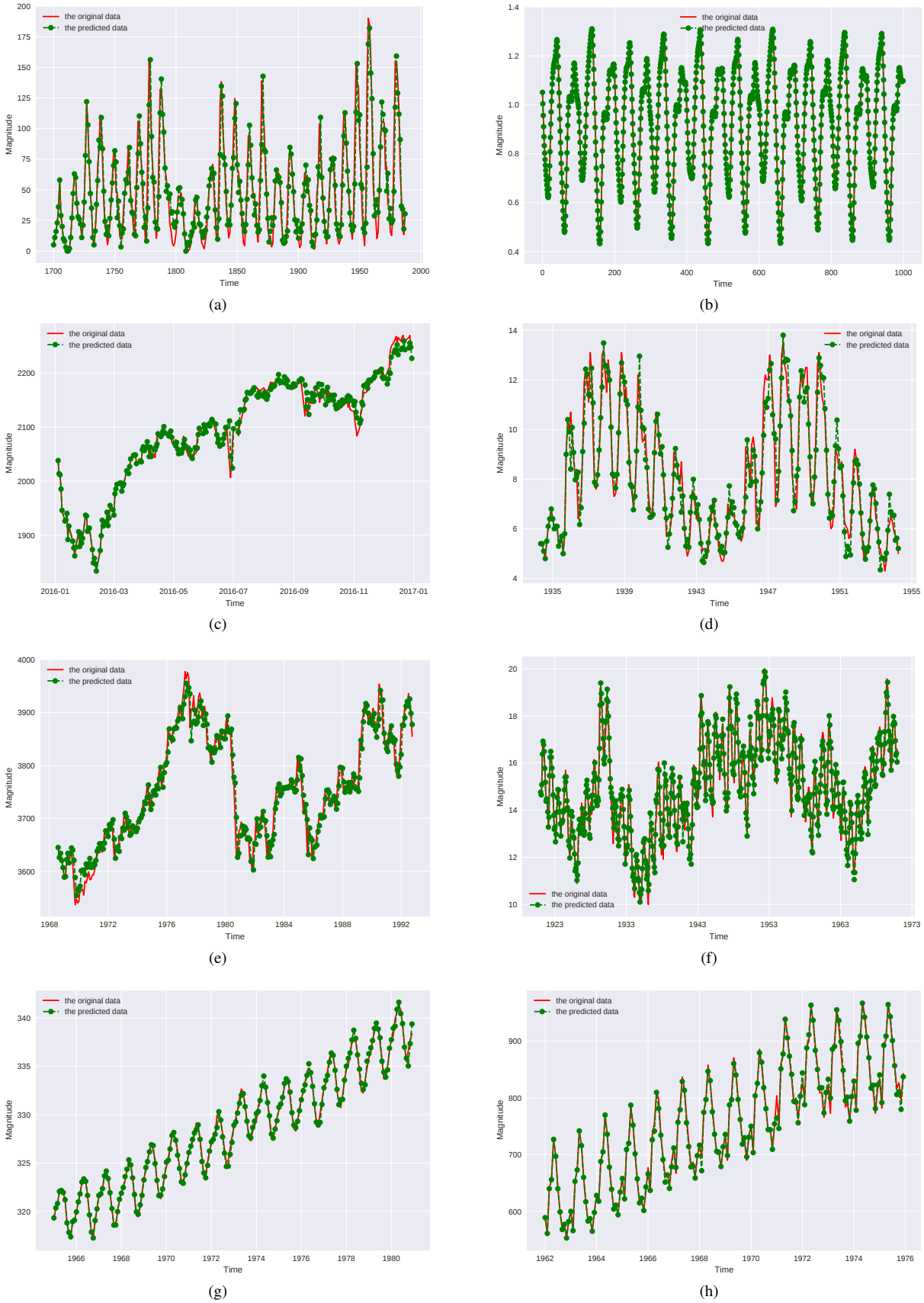


Fig. 10. Original time series and predicted time series of (a) sunspot time series, (b) MG chaos time series, (c) S&P 500 time series, (d) monthly critical radio frequencies, (e) monthly closings of the Dow-Jones industrial index, (f) monthly Lake Erie levels, (g) CO_2 (ppm) at Mauna Loa, and (h) monthly milk production per cow.

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TFUZZ.2018.2831640, IEEE Transactions on Fuzzy Systems

TABLE III
COMPARISON WITH OTHER CLASSICAL PREDICTION MODELS IN TERMS OF RMSE

	Wavelet-HFCM	ANFIS [6]	AR* model per scale [37]	Multiresolution AR model [38]	ANN [70]
Sunspot time series	**18.916**	22.753	35.262	19.186	19.901
MG time series	0.004	**0.001**	0.035	0.002	0.005
S&P 500 stock index	16.105	**14.935**	17.897	16.041	17.696
Milk production in pounds	**8.258**	9.578	57.717	37.838	27.113
Monthly closings of the Dow-Jones industrial index	**23.159**	27.526	29.822	26.733	28.532
Monthly critical radio frequencies	**0.547**	0.651	0.902	0.662	0.652
CO_2 (ppm) at Mauna Loa	**0.560**	0.910	1.350	0.812	1.695
Monthly Lake Erie levels	**0.377**	0.458	0.638	0.390	0.402

* AR is short for autoregressive.

TABLE IV
RMSE OF WAVELET-HFCM ON DIFFERENT SUBSETS.

	Training	Validation	Test
Sunspot time series	14.113	12.839	18.916
MG time series	0.003	0.006	0.004
S&P 500 stock index	5.682	20.600	16.105
Monthly milk production per cow	6.718	12.098	8.258
Monthly closings of the Dow-Jones industrial index	23.216	23.171	23.159
Monthly critical radio frequencies	0.598	0.897	0.547
CO_2 (ppm) at Mauna Loa	0.296	0.407	0.560
Monthly Lake Erie levels	0.434	0.421	0.377

TABLE V
COMBINATIONS OF OPTIMAL N_c, α AND ORDER OF WAVELET-HFCM USED FOR FORECASTING EACH TIME SERIES

	N_c	α	order
Sunspot time series	6	1e-20	2
MG time series	8	1e-20	23
S&P 500 stock index	7	1e-20	3
Monthly milk production per cow	6	1e-20	6
Monthly closings of the Dow-Jones industrial index	4	1e-20	3
Monthly critical radio frequencies	5	1e-20	4
CO_2 (ppm) at Mauna Loa	6	1e-20	6
Monthly Lake Erie levels	6	1e-14	6

E. Comparison Against Other Methods

Fig. 10 shows the original and the predicted time series by using Wavelet-HFCM. Then comparing our proposal with other classical prediction models [6], [37], [38], [70], the results shown in Table III indicate that the proposed modeling method can produce the highest accuracy for most time series and can still produce satisfactory predictions for the two other time series. Furthermore, to make the predicted results easy to be reproduced, Table IV records the RMSE of Wavelet-HFCM on the training dataset, the validation dataset and the test dataset of each time series, and Table V summarizes the combinations of optimal N_c, α and order of HFCM used to forecast each time series.

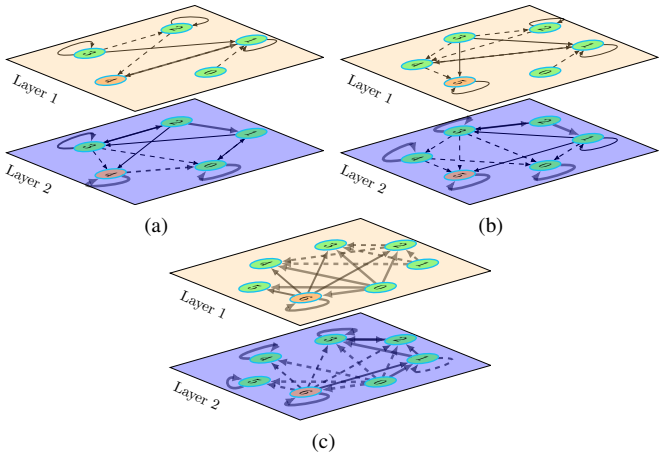


Fig. 11. Visualization of the learned 2-order HFCMs of (a) monthly critical radio frequencies, (b) sunspot time series and (c) S&P 500 stock index, respectively, in the form of two-layer networks.

Another advantage of Wavelet-HFCM is to provide high interpretability in the form of directed maps. For example, Fig. 11 visualizes the learned HFCM models in the form of two-layer networks for monthly critical radio frequencies, sunspot time series and S&P 500 time series, respectively. The cognitive map on the first layer represents the wavelet coefficients at time step t, and the one on the second layer represents the wavelet coefficients at time step $t-1$. Each node of an FCM corresponds to each scale of wavelet coefficients. With the opacity of the edge proportional to the weight of the edge, the maps interpret vividly which node influences mostly on a certain node, thus allowing decision-makers analyze the inter-connected relationships among different scales of wavelet coefficients easily. Comparing these three learned HFCM models, we can draw conclusions that the structure of learned HFCM varies with the characteristic of time series

and the self-loop property can not be ignored. Furthermore, by using techniques in graph theory such as node centrality [71], decision-makers can determine which nodes are more important and pay more attention to these nodes when performing the prediction.

V. Conclusion

In this paper, we have developed a time series prediction model based on the hybrid combination of HFCM with wavelet transform. Wavelet transform can decompose one time series into different resolution levels, and the resulting multivariate time series can be modeled and predicted by HFCM, achieving high prediction accuracy. In the process of learning HFCM to represent multivariate time series, a fast HFCM learning algorithm based on ridge regression is specifically designed, developing HFCMs from historical time series with less time cost. Finally, summing multivariate time series up at each level yields the predicted time series at each time step.

This paper describes the proposed prediction method and performs comprehensive experiments to verify its performance. The experiments show that both the number of nodes and regularization factor α have important impacts on the performance of Wavelet-HFCM, while the order of HFCM impacts slightly on Wavelet-HFCM. To achieve higher accuracy for forecasting stationary time series or time series with a linear trend, the number of nodes, the factor α, and the order of HFCM could be set to 5, 1e-20 and 2, respectively. These three parameters need to be fine-tuned using the validation procedure when dealing with non-stationary time series. Small α near zero works well for stationary time series and time series with a linear trend, while time series with a quadratic upward (or downward) trend will require a large value of α. Not only does Wavelet-HFCM outperform several existing methods, but also offers an ability to provide high interpretability in the form of directed maps.

The wavelet transform plays an important role in modeling time series by using Wavelet-HFCM, and thus developing more forms of wavelet transformation will be studied in the future as a useful approach to improve Wavelet-HFCM. Furthermore, Wavelet-HFCM has great potential in assisting decision-makers to analyze large-scale time series. For example, another interesting and practical follow-up to this work is to apply Wavelet-HFCM to detect anomaly in time series and provide comprehensive comparisons with other detection algorithms.

References

[1] M. Sugeno, *Industrial Applications of Fuzzy Control*. Elsevier Science Inc., 1985.

[2] S. Vrkalovic, T.-A. Teban, and I.-D. Borlea, "Stable Takagi-Sugeno fuzzy control designed by optimization," *International Journal of Artificial Intelligence*, vol. 15, pp. 17–29, 2017.

[3] Y. Chen and J. Z. Wang, "A region-based fuzzy feature matching approach to content-based image retrieval," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 24, no. 9, pp. 1252–1267, 2002.

[4] J. Nowaková, M. Prílepok, and V. Snášel, "Medical image retrieval using vector quantization and fuzzy S-tree," *Journal of Medical Systems*, vol. 41, no. 2, p. 18, 2017.

[5] C. Pozna, N. Minculete, R.-E. Precup, L. T. Kóczy, and Á. Ballagi, "Signatures: Definitions, operators and applications to fuzzy modelling," *Fuzzy Sets and Systems*, vol. 201, pp. 86–104, 2012.

[6] J.-S. Jang, "ANFIS: adaptive-network-based fuzzy inference system," *IEEE Transactions on Systems, Man, and Cybernetics*, vol. 23, no. 3, pp. 665–685, 1993.

[7] Q. Song and B. S. Chissom, "Fuzzy time series and its models," *Fuzzy Sets and Systems*, vol. 54, no. 3, pp. 269–277, 1993.

[8] O. Castillo, P. Melin, and W. Pedrycz, "Design of interval type-2 fuzzy models through optimal granularity allocation," *Applied Soft Computing*, vol. 11, no. 8, pp. 5590–5601, 2011.

[9] P. Melin, J. Soto, O. Castillo, and J. Soria, "A new approach for time series prediction using ensembles of ANFIS models," *Expert Systems with Applications*, vol. 39, no. 3, pp. 3494–3506, 2012.

[10] J. Soto, P. Melin, and O. Castillo, "Time series prediction using ensembles of ANFIS models with genetic optimization of interval type-2 and type-1 fuzzy integrators," *International Journal of Hybrid Intelligent Systems*, vol. 11, no. 3, pp. 211–226, 2014.

[11] J. Soto and P. Melin, "Optimization of the fuzzy integrators in ensembles of ANFIS model for time series prediction: The case of Mackey-Glass," in *Proceedings of the International Fuzzy Systems Association and the European Society for Fuzzy Logic and Technology Conference*, 2015, pp. 218–25.

[12] J. Soto, P. Melin, and O. Castillo, "Particle swarm optimization of the fuzzy integrators for time series prediction using ensemble of IT2FNN architectures," in *Nature-Inspired Design of Hybrid Intelligent Systems*. Springer, 2017, pp. 141–158.

[13] B. Kosko, "Fuzzy cognitive maps," *International Journal of Man-Machine Studies*, vol. 24, no. 1, pp. 65–75, 1986.

[14] G. Pajares, M. Guijarro, P. J. Herrera, J. J. Ruz, and J. M. de la Cruz, *Fuzzy Cognitive Maps Applied to Computer Vision Tasks*. Springer Berlin Heidelberg, 2010, pp. 259–289.

[15] A. K. Tsadiras, "Using fuzzy cognitive maps for e-commerce strategic planning," in *Proceedings of the 9th Panhellenic Conference on Informatics*, 2003.

[16] J. L. Salmeron, "Fuzzy cognitive maps for artificial emotions forecasting," *Applied Soft Computing*, vol. 12, no. 12, pp. 3704–3710, 2012.

[17] E. Papageorgiou, P. Oikonomou, and A. Kannappan, "Bagged nonlinear hebbian learning algorithm for fuzzy cognitive maps working on classification tasks," *Artificial Intelligence: Theories and Applications*, pp. 157–164, 2012.

[18] G. A. Papakostas, Y. S. Boutalis, D. E. Koulouriotis, and B. G. Mertzios, "Fuzzy cognitive maps for pattern recognition applications," *International Journal of Pattern Recognition and Artificial Intelligence*, vol. 22, no. 08, pp. 1461–1486, 2008.

[19] K. Wu, J. Liu, and Y. Chi, "Wavelet fuzzy cognitive maps," *Neurocomputing*, vol. 232, pp. 94–103, 2017.

[20] J. Liu, Y. Chi, and C. Zhu, "A dynamic multiagent genetic algorithm for gene regulatory network reconstruction based on fuzzy cognitive maps," *IEEE Transactions on Fuzzy Systems*, vol. 24, no. 2, pp. 419–431, 2016.

[21] G. Acampora and A. Vitiello, "Learning of fuzzy cognitive maps for modelling gene regulatory networks through big bang-big crunch algorithm," in *Proceedings of IEEE International Conference on Fuzzy Systems*, 2015, pp. 1–6.

[22] Y. Chi and J. Liu, "Reconstructing gene regulatory networks with a memetic-neural hybrid based on fuzzy cognitive maps," *Natural Computing*, Online, 2018.

[23] J. L. Salmeron and W. Froelich, "Dynamic optimization of fuzzy cognitive maps for time series forecasting," *Knowledge-Based Systems*, vol. 105, pp. 29–37, 2016.

[24] H. Song, C. Miao, Z. Shen, W. Roel, D. Maja, and C. Francky, "Design of fuzzy cognitive maps using neural networks for predicting chaotic time series," *Neural Networks*, vol. 23, no. 10, pp. 1264–1275, 2010.

[25] W. Stach, L. Kurgan, and W. Pedrycz, "Linguistic signal prediction with the use of fuzzy cognitive maps," in *Proceedings of Symposium on Human-Centric Computing Conference*, 2005, pp. 64–71.

[26] W. Stach, L. A. Kurgan, and W. Pedrycz, "Numerical and linguistic prediction of time series with the use of fuzzy cognitive maps," *IEEE Transactions on Fuzzy Systems*, vol. 16, no. 1, pp. 61–72, 2008.

[27] W. Pedrycz, "The design of cognitive maps: A study in synergy of granular computing and evolutionary optimization," *Expert Systems with Applications*, vol. 37, no. 10, pp. 7288–7294, 2010.

[28] H. Song, C. Miao, W. Roel, Z. Shen, and F. Catthoor, "Implementation of fuzzy cognitive maps based on fuzzy neural network and application in prediction of time series," *IEEE Transactions on Fuzzy Systems*, vol. 18, no. 2, pp. 233–250, 2010.

[29] W. Lu, J. Yang, X. Liu, and W. Pedrycz, "The modeling and prediction of time series based on synergy of high-order fuzzy cognitive map and fuzzy c-means clustering," *Knowledge-Based Systems*, vol. 70, pp. 242–255, 2014.

[30] W. Pedrycz, A. Jastrzebska, and W. Homenda, "Design of fuzzy cognitive maps for modeling time series," *IEEE Transactions on Fuzzy Systems*, vol. 24, no. 1, pp. 120–130, 2016.

[31] D. Koulouriotis, I. Diakoulakis, and D. Emiris, "Learning fuzzy cognitive maps using evolution strategies: a novel schema for modeling and simulating high-level behavior," in *Proceedings of the 2001 Congress on Evolutionary Computation*, vol. 1, 2001, pp. 364–371.

[32] K. E. Parsopoulos, E. I. Papageorgiou, P. Groumpos, and M. N. Vrahatis, "A first study of fuzzy cognitive maps learning using particle swarm optimization," in *Proceedings of the 2003 Congress on Evolutionary Computation*, vol. 2, 2003, pp. 1440–1447.

[33] W. Stach, L. Kurgan, W. Pedrycz, and M. Reformat, "Genetic learning of fuzzy cognitive maps," *Fuzzy Sets and Systems*, vol. 153, no. 3, pp. 371–401, 2005.

[34] Y. Chi and J. Liu, "Learning of fuzzy cognitive maps with varying densities using a multiobjective evolutionary algorithm," *IEEE Transactions on Fuzzy Systems*, vol. 24, no. 1, pp. 71–81, 2016.

[35] X. Zou and J. Liu, "A mutual information based two-phase memetic algorithm for large-scale fuzzy cognitive map learning," *IEEE Transactions on Fuzzy Systems*, in press, 2018.

[36] A. Aussem, J. Campbell, and F. Murtagh, "Wavelet-based feature extraction and decomposition strategies for financial forecasting," *Journal of Computational Intelligence in Finance*, vol. 6, no. 2, pp. 5–12, 1998.

[37] G. Zheng, J. Starck, J. Campbell, and F. Murtagh, "Multiscale transforms for filtering financial data streams," *Journal of Computational Intelligence in Finance*, vol. 7, no. 18-35, 1999.

[38] O. Renaud, J.-L. Starck, and F. Murtagh, "Wavelet-based combined signal filtering and prediction," *IEEE Transactions on Systems, Man, and Cybernetics, Part B (Cybernetics)*, vol. 35, no. 6, pp. 1241–1251, 2005.

[39] A. E. Hoerl and R. W. Kennard, "Ridge regression: Biased estimation for nonorthogonal problems," *Technometrics*, vol. 12, no. 1, pp. 55–67, 1970.

[40] W. Stach, L. Kurgan, and W. Pedrycz, "Higher-order fuzzy cognitive maps," in *Proceedings of Annual Meeting of the North American Fuzzy Information Processing Society Conference*. IEEE, 2006, pp. 166–171.

[41] W. Froelich and W. Pedrycz, "Fuzzy cognitive maps in the modeling of granular time series," *Knowledge-Based Systems*, vol. 115, pp. 110–122, 2017.

[42] E. I. Papageorgiou and K. Poczęta, "A two-stage model for time series prediction based on fuzzy cognitive maps and neural networks," *Neurocomputing*, vol. 232, pp. 113–121, 2017.

[43] F. Vanhoenshoven, G. Nápoles, S. Bielen, and K. Vanhoof, "Fuzzy cognitive maps employing ARIMA components for time series forecasting," in *Proceedings of the International Conference on Intelligent Decision Technologies*. Springer, 2017, pp. 255–264.

[44] K. Wu and J. Liu, "Robust learning of large-scale fuzzy cognitive maps via the lasso from noisy time series," *Knowledge-Based Systems*, vol. 113, pp. 23–38, 2016.

[45] ——, "Learning large-scale fuzzy cognitive maps based on compressed sensing and application in reconstructing gene regulatory networks," *IEEE Transactions on Fuzzy Systems*, vol. 25, no. 6, pp. 1546–1560, 2017.

[46] L. Buitinck, G. Louppe, M. Blondel, F. Pedregosa, A. Mueller, O. Grisel, V. Niculae, P. Prettenhofer, A. Gramfort, J. Grobler, R. Layton, J. VanderPlas, A. Joly, B. Holt, and G. Varoquaux, "API design for machine learning software: experiences from the scikit-learn project," in *ECML PKDD Workshop: Languages for Data Mining and Machine Learning*, 2013, pp. 108–122.

[47] S. Soltani, "On the use of the wavelet decomposition for time series prediction," *Neurocomputing*, vol. 48, no. 1, pp. 267–277, 2002.

[48] H. S. Lopes and W. R. Weinert, "EGIPSYS: an enhanced gene expression programming approach for symbolic regression problems," *International Journal of Applied Mathematics and Computer Science*, vol. 14, no. 3, pp. 375–384, 2004.

[49] C. Ferreira, *Gene Expression Programming: Mathematical Modeling by an Artificial Intelligence*. Springer, 2006, vol. 21.

[50] H. Cao, L. Kang, Y. Chen, and J. Yu, "Evolutionary modeling of systems of ordinary differential equations with genetic programming," *Genetic Programming and Evolvable Machines*, vol. 1, no. 4, pp. 309–337, 2000.

[51] Y. Peng, C. Yuan, X. Qin, J. Huang, and Y. Shi, "An improved gene expression programming approach for symbolic regression problems," *Neurocomputing*, vol. 137, pp. 293–301, 2014.

[52] S. T. A. Niaki and S. Hoseinzade, "Forecasting S&P 500 index using artificial neural networks and design of experiments," *Journal of Industrial Engineering International*, vol. 9, no. 1, pp. 1–1, 2013.

[53] R. Majhi, G. Panda, G. Sahoo, A. Panda, and A. Choubey, "Prediction of S&P 500 and DJIA stock indices using particle swarm optimization technique," in *Proceedings of the 2008 IEEE International Conference on Fuzzy Systems*, 2008, pp. 1276–1282.

[54] R. Tsaih, Y. Hsu, and C. C. Lai, "Forecasting S&P 500 stock index futures with a hybrid AI system," *Decision Support Systems*, vol. 23, no. 2, pp. 161–174, 1998.

[55] M. Martens, "Measuring and forecasting S&P 500 index-futures volatility using high-frequency data," *Journal of Futures Markets*, vol. 22, no. 6, pp. 497–518, 2002.

[56] E. Hajizadeh, A. Seifi, M. F. Zarandi, and I. Turksen, "A hybrid modeling approach for forecasting the volatility of S&P 500 index return," *Expert Systems with Applications*, vol. 39, no. 1, pp. 431–436, 2012.

[57] S. A. Hamid and Z. Iqbal, "Using neural networks for forecasting volatility of S&P 500 index futures prices," *Journal of Business Research*, vol. 57, no. 10, pp. 1116–1125, 2004.

[58] "^GSPC historical prices | S&P 500 stock, Yahoo!" https://finance.yahoo.com/quote/%5EGSPC/history?p=%5EGSPC, accessed: 2017-06-14.

[59] "Monthly milk production: pounds per cow. Jan 62 - Dec 75," https://datamarket.com/data/set/22ox/monthly-milk-production-pounds-per-cow-jan-62-dec-75#!ds=22ox&display=line, accessed: 2017-12-19.

[60] "Monthly closings of the Dow-Jones industrial index, Aug. 1968 - Aug. 1981," https://datamarket.com/data/set/22v9/monthly-closings-of-the-dow-jones-industrial-index-aug-1968-aug-1981#!ds=22v9&display=line, accessed: 2017-12-19.

[61] "Monthly critical radio frequencies in Washington, D.C., May 1934 - April 1954," https://datamarket.com/data/set/22u2/monthly-critical-radio-frequencies-in-washington-dc-may-1934-april-1954-these-frequencies-reflect-the-highest-radio-frequency-that-can-be-used-for-broadcasting#!ds=22u2&display=line, accessed: 2017-12-19.

[62] "Co2 (ppm) mauna loa, 1965-1980," https://datamarket.com/data/set/22v1/co2-ppm-mauna-loa-1965-1980#!ds=22v1&display=line, accessed: 2017-12-19.

[63] "Monthly Lake Erie levels 1921 - 1970," https://datamarket.com/data/set/22pw/monthly-lake-erie-levels-1921-1970#!ds=22pw&display=line, accessed: 2017-12-19.

[64] M. C. Mackey, L. Glass *et al.*, "Oscillation and chaos in physiological control systems," *Science*, vol. 197, no. 4300, pp. 287–289, 1977.

[65] C.-F. Juang and Y.-W. Tsao, "A self-evolving interval type-2 fuzzy neural network with online structure and parameter learning," *IEEE Transactions on Fuzzy Systems*, vol. 16, no. 6, pp. 1411–1424, 2008.

[66] B.-T. Zhang, P. Ohm, and H. Mühlenbein, "Evolutionary induction of sparse neural trees," *Evolutionary Computation*, vol. 5, no. 2, pp. 213–236, 1997.

[67] T. Hastie, J. Friedman, and R. Tibshirani, "Model assessment and selection," in *The Elements of Statistical Learning*. Springer, 2001, pp. 193–224.

[68] S. Yang and J. Liu, "Wavelet-HFCM," https://github.com/yangysc/Wavelet-HFCM, 2018.

[69] M. KivelÃd', A. Arenas, M. Barthelemy, J. P. Gleeson, Y. Moreno, and M. A. Porter, "Multilayer networks," *Journal of Complex Networks*, vol. 2, no. 3, pp. 203–271, 2014.

[70] A. B. Geva, "Scalenet-multiscale neural-network architecture for time series prediction," *IEEE Transactions on Neural Networks*, vol. 9, no. 6, pp. 1471–1482, 1998.

[71] M. Newman, *Networks: an Introduction*. Oxford University Press, 2010.

Shanchao Yang received the B.S. degree in intelligent science and technology in 2017 from Xidian University, Xi'an, China, where he is currently working toward the Academic Master's degree with the Key Laboratory of Intelligent Perception and Image Understanding of Ministry of Education.

His current research interests vary from evolutionary computation, complex networks, fuzzy cognitive maps to machine learning.

Jing Liu (SM'15) received the B.S. degree in computer science and technology and the Ph.D. degree in circuits and systems from Xidian University in 2000 and 2004, respectively.

In 2005, she joined Xidian University as a lecturer, and was promoted to a full professor in 2009. From Apr. 2007 to Apr. 2008, she worked at The University of Queensland, Australia as a postdoctoral research fellow, and from Jul. 2009 to Jul. 2011, she worked at The University of New South Wales at the Australian Defence Force Academy as a research associate.

Now, she is a full professor in the Key Laboratory of Intelligent Perception and Image Understanding of Ministry of Education, Xidian University. Her research interests include evolutionary computation, complex networks, fuzzy cognitive maps, multiagent systems, and data mining. She is the associate editor of *IEEE Trans. Evolutionary Computation* and the Chair of Emerging Technologies Technical Committee (ETTC) of IEEE Computational Intelligence Society. Please see her homepage (http://see.xidian.edu.cn/faculty/liujing/) for more information.