Neural Networks 23 (2010) 1264–1275

Contents lists available at ScienceDirect

Neural Networks

journal homepage: www.elsevier.com/locate/neunet

Design of fuzzy cognitive maps using neural networks for predicting chaotic time series

H.J. Song [a,b,*], C.Y. Miao [a], Z.Q. Shen [a], W. Roel [b], D.H. Maja [b,c], C. Francky [b]

[a] Nanyang Technological University, Singapore
[b] IMEC, Kapeldreef 75, Leuven, Belgium
[c] Vrije Universiteit Brussel, Belgium

ARTICLE INFO

Article history:
Received 17 October 2008
Received in revised form 25 July 2010
Accepted 3 August 2010

Keywords:
Fuzzy cognitive maps
Causality
Membership function
Mutual subsethood

ABSTRACT

As a powerful paradigm for knowledge representation and a simulation mechanism applicable to numerous research and application fields, Fuzzy Cognitive Maps (FCMs) have attracted a great deal of attention from various research communities. However, the traditional FCMs do not provide efficient methods to determine the states of the investigated system and to quantify causalities which are the very foundation of the FCM theory. Therefore in many cases, constructing FCMs for complex causal systems greatly depends on expert knowledge. The manually developed models have a substantial shortcoming due to model subjectivity and difficulties with accessing its reliability. In this paper, we propose a fuzzy neural network to enhance the learning ability of FCMs so that the automatic determination of membership functions and quantification of causalities can be incorporated with the inference mechanism of conventional FCMs. In this manner, FCM models of the investigated systems can be automatically constructed from data, and therefore are independent of the experts. Furthermore, we employ mutual subsethood to define and describe the causalities in FCMs. It provides more explicit interpretation for causalities in FCMs and makes the inference process easier to understand. To validate the performance, the proposed approach is tested in predicting chaotic time series. The simulation studies show the effectiveness of the proposed approach.

© 2010 Elsevier Ltd. All rights reserved.

1. Introduction

Since the pioneering work of Kosko (1986), fuzzy cognitive maps (FCMs) have attracted a great deal of attention from various research communities. As a modeling methodology for complex systems, FCMs model the investigated causal system as a collection of concepts and causal relations among concepts, which originate from the combination of fuzzy logic and neural networks. Intuitively, a FCM is a signed directed graph with feedback, which consists of a collection of nodes and directed weighted arcs interconnecting nodes. Fig. 1 gives the graphical representation of a FCM and its neural network structure.

In FCMs, nodes represent the concepts with semantic meaning that are abstracted from the investigated systems. The state of a concept is characterized by a number x_i in the interval [0, 1] or in the interval [−1, 1].

Accordingly, the behavior of an investigated system at time point t is expressed by a vector $X(t) = (x_1(t), \ldots, x_N(t))$ where N denotes the number of concepts. In addition, the directed arcs interconnecting nodes represent the causal–effect relationships (causalities) among the different concepts. The weight w_{ij} $(i, j \in N)$ that associates with the arc pointing from x_i to x_j, describes the type and the strength of the causality from x_i to x_j, ranging in the interval [−1, 1]. Usually, w_{ij} has three possible types: positive causality ($w_{ij} > 0$), negative causality ($w_{ij} < 0$) or no causality ($w_{ij} = 0$). The absolute value of w_{ij} quantifies the strength of the causality; the sign of w_{ij} indicates that the causality from x_i to x_j is direct or inverse. For the sake of simplicity, weights in a FCM can be represented by a matrix $W \in \mathbb{R}^N$.

Note that in FCMs, all the elements of the principal diagonal w_{ii} are equal to zero because a concept cannot cause itself and there is no causal relationship between a concept and itself. On the basis of the above definitions, the inference mechanism of FCMs can be described by Eq. (1),

$$X(t+1) = f(X(t) \cdot W) \tag{1}$$

where $X(t+1)$ and $X(t)$, respectively describe the dynamical behavior of the investigated system at discrete times $(t+1)$ and t; f is a sigmoid function, which squashes the result in the interval

* Corresponding author at: Nanyang Technological University, Singapore.
E-mail addresses: song@imec.be, SONG0027@ntu.edu.sg (H.J. Song), ASCYMIAO@ntu.edu.sg (C.Y. Miao), ZQSHEN@ntu.edu.sg (Z.Q. Shen), wuyts@imec.be (W. Roel), dhondtm@imec.be (D.H. Maja), catthoor@imec.be (C. Francky).

0893-6080/$ – see front matter © 2010 Elsevier Ltd. All rights reserved.
doi:10.1016/j.neunet.2010.08.003

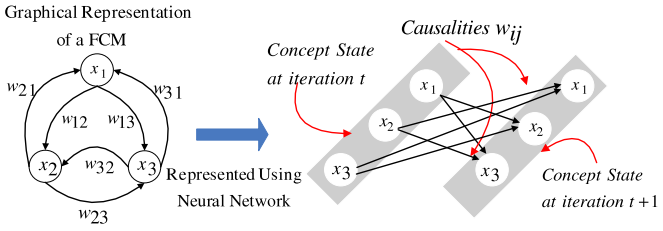

Fig. 1. Example of a FCM.

[0, 1]. Therefore, the inference mechanism could be regarded as an iterative process that applies the scalar product and sigmoid function to generate the discrete time series of the system until the state vector ends in a fixed point, limit cycle, or more complex ones may end in an aperiodic or chaotic attractor (Boutails, Kottas, & Christodoulou, 2009; Kottas & Boutalis, 2004).

By storing the existing knowledge on the behaviors of the investigated system in the structure of nodes and interconnections of the map, FCMs provide a more flexible and natural mechanism for knowledge representation and reasoning which are essential to intelligent systems (Miao & Liu, 2001). The advantages of applying FCMs formalism are summarized as follows.

a. FCMs' graphical nature allows one to visualize the structure of the investigated system and makes the models relatively simple and legible.
b. FCMs' mathematical foundation allows one to express the dynamic behavior of the investigated system in algebraic forms.

Just because FCMs have several desirable properties (Liu & Satur, 1999; Stylios & Groumpos, 2004), such as abstraction, flexibility, adaptability and fuzzy reasoning, the application examples can be found in decision-making methods, geographical information systems, and prediction of time series (Kottas & Boutalis, 2006; Liu & Zhang, 2003; Papageorgiou, Stylios, & Groumpos, 2003; Pelaez & Bowles, 1995; Stach, Kurgan, & Pedrycz, 2008; Stylios & Groumpos, 1999).

In the past decades, several works on FCMs have been proposed in order to investigate the extensions of FCMs. Liu and Satur (1999), Miao and Liu (2000, 2001) have made some significant and extensive investigations on the inference properties of FCMs. Moreover, Miao and Liu (2001) proposed dynamic causal networks (DCN) to quantify the description of concepts with required precision as well as the strength of causality between concepts. Based on the further study of the inference mechanism and convergent features of FCMs, Zhou, Liu, and Zhang (2006) proposed fuzzy causal networks (FCNs). An integrated two-level hierarchical system based fuzzy cognitive map has been proposed and applied into decision making in radiation therapy (Papageorgiou et al., 2003). In addition, Stach and Lukasz employed FCMs to predict time series in Stach et al. (2008). Besides these applied researches, the decomposition theory of FCMs has been studied in Zhang, Liu, and Zhou (2003) and Zhang, Liu, and Zhou (2006), which provides an effective framework for calculating and simplifying causal patterns in complicated real world applications.

Although some research works have been done on FCMs, very few researches have investigated how to automatically identify the membership functions and quantify the causalities in FCMs. As a kind of inference model based on fuzzy theory, the membership functions are crucial to fuzzification and defuzzification in FCMs. However FCMs lack effective methods to identify membership functions which are absolutely necessary to describe system states in FCMs. Therefore, the inference results generated by FCMs are always limited into the fuzzy domain so that it is difficult to directly compare the inference results against real data. As a result, in most existing literature about FCMs, the interpretation of inference result greatly depends on expert knowledge. The subjective interpretation is not more or less convincing. From this perspective, the automatic identification of membership functions for different concepts in the concerned systems has become a very urgent and realistic issue.

Secondly in many research works on FCMs, the causalities are still quantified based on expert knowledge. But for various practical problems, it is a hard task for human experts to accurately pre-specify the causalities among concepts which constitute a specific causal system. To deal with this problem, the learning algorithms applicable to FCMs have been proposed in Koulouriotis (2002), Koulouriotis and Diakoulakis (2001) and Song, Shen, and Miao (2007) that are respectively based on evolution strategies and particle swarm optimization. However in these approaches, the definition about causality is not exactly consistent with that of conventional FCMs. Accordingly, the quantified causalities lack transparent mathematical interpretation and make the inference process ambiguous. More recently, Stach et al. (2008) presented an application framework which employed FCMs to implement numerical and linguistic prediction of time series. However in Stach et al. (2008), many procedures are introduced for pre-processing the raw dataset. It makes the prediction more complex. Additionally, in all documents about FCMs (Koulouriotis, 2002; Koulouriotis & Diakoulakis, 2001; Song et al., 2007; Stach et al., 2008), the membership functions are still pre-specified based on the analysis of the dataset or expert knowledge.

To solve these two problems, this paper proposes a novel fuzzy neural network in which the inference mechanism of traditional FCMs is integrated with automatic identification of membership functions and quantification of causalities. By identifying the membership functions and causalities from real data, the proposed approach is able to construct the FCMs for the investigated system with less human intervention and prior knowledge. Furthermore, the employed fuzzy neural network defines and describes the causalities in the investigated systems using mutual subsethood. It provides more transparent interpretation on causalities and makes the inference process easier to understand.

The rest of this paper is organized as follows. Section 2 describes the structure, the functionalities and the corresponding operations of our approach in detail. Section 3 presents the supervised learning algorithm which is employed to tune the related parameters in the proposed fuzzy neural network. In Section 4, the performance of our approach is tested in predicting three benchmarking chaotic time series. The comparisons of the experimental results of the proposed approach with other models are also given in Section 4. Finally, Section 5 presents the conclusions.

2. The proposed approach

From the above introduction, we can see that the manually developed FCMs have a substantial shortcoming due to model subjectivity and difficulties with accessing its reliability. To deal with this problem, we propose a four-layer fuzzy neural network in terms of the definition and description of conventional FCMs. The proposed approach integrates the inference process of FCMs with the identification of membership functions, as well as the quantification of causalities. In this section, we will describe the basic structure and functionalities which are necessary for the understanding of our approach in detail.

Fig. 2 depicts the basic structure of the proposed four-layer fuzzy neural network. In the fuzzy neural network, the inputs and outputs are represented by non-fuzzy vectors $X^T = (x_1, \ldots, x_i, \ldots, x_N)$ and $Y^T = (y_1, \ldots, y_i, \ldots, y_N)$, respectively, where N denotes the numbers of the input and output variables. *We stress that, in the proposed approach, x_i and y_i represent the same variable i.* Therefore in the inference process, x_i and y_i, respectively represent the state of the variable i at iteration t and iteration $t+1$.

2.1. Input layer

Each node x_i ($i \in N$) in the 'Input layer' represents a domain variable of the investigated system, which is connected with the 'Linguistic layer' using the fixed weight 1. Therefore, x_i directly transmits the crisp (non-fuzzy) input value to the next layer. Accordingly, the net input $f_i^{(1)}$ and net output $x_i^{(1)}$ of the ith node are given in Eq. (2) (the number in parentheses in superscript represents the level number of the proposed neural network).

$$f_i^{(1)} = x_i; \qquad x_i^{(1)} = f_i^{(1)}. \tag{2}$$

2.2. Linguistic layer

The node $IL_i^{n_i}(m_i = 1, \ldots, M_i)$ in the linguistic layer expresses a semantic symbol of the input x_i, such as 'Small', 'Medium' or 'Large', etc. Hence, each $IL_i^{n_i}$ represents a fuzzy subset on x_i. In our approach, the fuzzy set $IL_i^{n_i}$ is modeled by membership function $\mu_{IL_i^{n_i}}(x_i)$ that is a symmetric Gaussian function and described by,

$$\mu_{IL_i^{n_i}}(x_i) = \exp\left(-\left(x_i - C_{IL_i^{n_i}}\right)^2 / \sigma^2_{IL_i^{n_i}}\right) \tag{3}$$

where $C_{IL_i^{n_i}}$ and $\sigma_{IL_i^{n_i}}$ represent center and width, respectively. The symmetric Gaussian membership function instead of triangular or trapezoidal function ensures the differentiability that is a necessary property for the backpropagation algorithm employed in the learning process.

Accordingly, the net input $f^{(2)}_{IL_i^{n_i}}$ and net output $x^{(2)}_{IL_i^{n_i}}$ of the $IL_i^{n_i}$th linguistic node are given in Eq. (4),

$$\begin{aligned} f^{(2)}_{IL_i^{n_i}} &= -\left(x_i^{(1)} - C_{IL_i^{n_i}}\right)^2 / \sigma^2_{IL_i^{n_i}} \\ x^{(2)}_{IL_i^{n_i}} &= \mathrm{e}^{f^{(2)}_{IL_i^{n_i}}} = \mathrm{e}^{-\left(x_i^{(1)} - C_{IL_i^{n_i}}\right)^2 / \sigma^2_{IL_i^{n_i}}}. \end{aligned} \tag{4}$$

It is obvious that the fuzzification of the system is accomplished in the linguistic layer. The net output $x^{(2)}_{IL_i^{n_i}}$ is the membership grade of the input variable x_i belonging to the fuzzy subset $IL_i^{n_i}$.

2.3. Mapping layer

As the semantic mapping of the linguistic layer, the node in the mapping layer denotes the current state of the linguistic term $IL_i^{n_i}$ under the influences that are imposed by the previous states of its own and other variables.

As mentioned previously, a given input variable x_i and the corresponding output variable y_i represent the same concept. Therefore, the linguistic term $OL_i^{n_i}$ of the output variable y_i indicates the same semantic symbol expressed by $IL_i^{n_i}$. So, the linguistic term $OL_i^{n_i}$ is also described by the symmetric Gaussian function with center $C_{OL_i^{n_i}}$ and spread $\sigma_{OL_i^{n_i}}$, denoted by $OL_i^{n_i} = (C_{OL_i^{n_i}}, \sigma_{OL_i^{n_i}})$. According to the above introduction, we can draw $C_{IL_i^{n_i}} = C_{OL_i^{n_i}}$ and $\sigma_{IL_i^{n_i}} = \sigma_{OL_i^{n_i}}$.

Note that in traditional FCMs, a concept cannot cause itself and there is no causal relationship between a concept and itself. Due to this limitation, the mapping layer is connected with the linguistic layer along with fuzzy weight $1 - \varepsilon(IL_i^{n_i}, OL_j^{m_j})$, where $\varepsilon(IL_i^{n_i}, OL_j^{m_j})$ is a mutual subsethood (Kosko, 1997; Song, Miao, & Shen, 2009) between $IL_i^{n_i}$ and $OL_j^{m_j}$. According to the definition of mutual subsethood, $\varepsilon(IL_i^{n_i}, OL_j^{m_j})$ measures the similarity between fuzzy subset $IL_i^{n_i}$ and fuzzy subset $OL_j^{m_j}$. For the given fuzzy sets $IL_i^{n_i}$ and $OL_j^{m_j}$ that are respectively described by Gaussian membership functions $\exp(-((x-C_{IL_i^{n_i}})/\sigma_{IL_i^{n_i}})^2)$ and $\exp(-((x-C_{OL_j^{m_j}})/\sigma_{OL_j^{m_j}})^2)$, the mutual subsethood $\varepsilon(IL_i^{n_i}, OL_j^{m_j})$ is formulated,

$$\begin{aligned} \varepsilon(IL_i^{n_i}, OL_j^{m_j}) &= \frac{C(IL_i^{n_i} \cap OL_j^{m_j})}{C(IL_i^{n_i} \cup OL_j^{m_j})} \\ &= \frac{C(IL_i^{n_i} \cap OL_j^{m_j})}{C(IL_i^{n_i}) + C(OL_j^{m_j}) - C(IL_i^{n_i} \cap OL_j^{m_j})}; \\ \varepsilon(IL_i^{n_i}, OL_j^{m_j}) &\in [0, 1]. \end{aligned} \tag{5}$$

In addition, the cardinality $C(IL_i^{n_i})$ of fuzzy set $IL_i^{n_i}$ and the cardinality $C(OL_j^{m_j})$ of fuzzy set $OL_j^{m_j}$ can be defined by,

$$C\left(IL_i^{n_i}\right) = \int_{-\infty}^{+\infty} \exp\left(-\left(\left(x - C_{IL_i^{n_i}}\right)/\sigma_{IL_i^{n_i}}\right)^2\right) \mathrm{d}x \tag{6}$$

$$C\left(OL_j^{m_j}\right) = \int_{-\infty}^{+\infty} \exp\left(-\left(\left(x - C_{OL_j^{m_j}}\right)/\sigma_{OL_j^{m_j}}\right)^2\right) \mathrm{d}x. \tag{7}$$

It is worth noting that, in Eq. (5), the denominator represents the area of union set of fuzzy sets $IL_i^{n_i}$ and $OL_j^{m_j}$, while numerator represents the area of the intersection set of fuzzy sets $IL_i^{n_i}$ and $OL_j^{m_j}$. Therefore, we can derive $0 \le C(IL_i^{n_i} \cap OL_j^{m_j}) \le C(IL_i^{n_i} \cap OL_j^{m_j})$. Based on Eq. (5), we can see that the fuzzy weight $0 \le 1 - \varepsilon(IL_i^{n_i}, OL_j^{m_j}) \le 1$. For example, if $IL_i^{n_i}$ and $OL_j^{m_j}$ represent the same concept ($i = j$) in a given FCM, then $\varepsilon(IL_i^{n_i}, OL_j^{m_j})$ takes the maximum value 1. As a result, the fuzzy weight between $IL_i^{n_i}$ and $OL_j^{m_j}$ is $1 - \varepsilon(IL_i^{n_i}, OL_j^{m_j}) = 0$. This particular situation complies with the definition of causalities in traditional FCMs in which $w_{ii} = 0$ because a concept cannot cause itself and there is no causal relationship between a concept and itself.

Therefore, we can see that the fuzzy weight $1 - \varepsilon(IL_i^{n_i}, OL_j^{m_j})$ effectively describes the causal–effect relationship from the input linguistic term $IL_i^{n_i}$ to the output linguistic term $OL_j^{m_j}$.

Furthermore, the mapping layer realizes the defuzzification of output variables. The defuzzification is performed using standard volume based centroid defuzzification (Kosko, 1997; Song et al., 2009). As a result, the input $f^{(3)}_{OL_j^{m_j}, IL_i^{n_i}}$ and output $x^{(3)}_{OL_j^{m_j}}$ of $OL_j^{m_j}$ are expressed by Eq. (8)

$$\begin{aligned} f^{(3)}_{OL_j^{m_j}, IL_i^{n_i}} &= x^{(2)}_{IL_i^{n_i}}\left(1 - \varepsilon\left(IL_i^{n_i}, OL_j^{m_j}\right)\right) \\ x^{(3)}_{OL_j^{m_j}} &= \sum_{\substack{i=1 \text{ and } i \neq j \\ n_i = 1,2,\ldots,N_i}}^{N} \left(f^{(3)}_{OL_j^{m_j}, IL_i^{n_i}} \cdot C_{IL_i^{n_i}} \cdot \sigma_{IL_i^{n_i}}\right) \Big/ \\ &\qquad \sum_{\substack{i=1 \text{ and } i \neq j \\ n_i = 1,2,\ldots,N_i}}^{N} \left(f^{(3)}_{OL_j^{m_j}, IL_i^{n_i}} \cdot \sigma_{IL_i^{n_i}}\right). \end{aligned} \tag{8}$$

In Eq. (8), the calculation of mutual subsethood $\varepsilon(IL_i^{n_i}, OL_j^{m_j})$ depends on the nature of the overlap of the $IL_i^{n_i}$ and $OL_j^{m_j}$, i.e., upon the values of $C_{IL_i^{n_i}}$, $C_{OL_j^{m_j}}$, $\sigma_{IL_i^{n_i}}$ and $\sigma_{OL_j^{m_j}}$. Case-wise expressions of mutual subsethood $\varepsilon(IL_i^{n_i}, OL_j^{m_j})$ therefore need to be derived and the expressions are summarized in Table 1.

In Table 1, erf(x) denotes the standard error function that is described as,

$$\mathrm{erf}(x) = \frac{1}{\sqrt{2\pi}} \int_0^x \mathrm{e}^{-\frac{t^2}{2}} \mathrm{d}x \tag{9}$$

Table 1
Calculation of mutual subsethood $\varepsilon\left(IL_i^{n_i}, IL_j^{m_j}\right)$.

		Mutual subsethood $\varepsilon(IL_i^{n_i}, IL_j^{m_j})$ $\lambda_1 = (\sigma_{IL_j^{m_j}} C_{IL_i^{n_i}} - \sigma_{IL_i^{n_i}} C_{IL_j^{m_j}})/(\sigma_{IL_j^{m_j}} - \sigma_{IL_i^{n_i}})$; $\lambda_2 = (\sigma_{IL_j^{m_j}} C_{IL_i^{n_i}} + \sigma_{IL_i^{n_i}} C_{IL_j^{m_j}})/(\sigma_{IL_j^{m_j}} + \sigma_{IL_i^{n_i}})$ are crossing points between $IL_i^{n_i}$ and $IL_j^{m_j}$
Case 1 $C_{IL_j^{m_j}} = C_{IL_i^{n_i}}$	A. $\sigma_{IL_j^{m_j}} = \sigma_{IL_i^{n_i}}$	1
	B. $\sigma_{IL_j^{m_j}} > \sigma_{IL_i^{n_i}}$	$\sigma_{IL_i^{n_i}}/\sigma_{IL_j^{m_j}}$
	C. $\sigma_{IL_j^{m_j}} < \sigma_{IL_i^{n_i}}$	$\sigma_{IL_j^{m_j}}/\sigma_{IL_i^{n_i}}$
Case 2 $C_{IL_j^{m_j}} > C_{IL_i^{n_i}}$	A. $\sigma_{IL_j^{m_j}} = \sigma_{IL_i^{n_i}}$	$\dfrac{\sigma_{IL_i^{n_i}}\left[\frac{1}{2}-\mathrm{erf}\left(\sqrt{2}\left(\lambda_2-C_{IL_i^{n_i}}\right)/\sigma_{IL_i^{n_i}}\right)\right]+\sigma_{IL_j^{m_j}}\left[\frac{1}{2}+\mathrm{erf}\left(\sqrt{2}\left(\lambda_2-C_{IL_j^{m_j}}\right)/\sigma_{IL_j^{m_j}}\right)\right]}{\sigma_{IL_i^{n_i}}\left[\frac{1}{2}+\mathrm{erf}\left(\sqrt{2}\left(\lambda_2-C_{IL_i^{n_i}}\right)/\sigma_{IL_i^{n_i}}\right)\right]+\sigma_{IL_j^{m_j}}\left[\frac{1}{2}-\mathrm{erf}\left(\sqrt{2}\left(\lambda_2-C_{IL_j^{m_j}}\right)/\sigma_{IL_j^{m_j}}\right)\right]}$
	B. $\sigma_{IL_j^{m_j}} > \sigma_{IL_i^{n_i}}$	$\dfrac{\sigma_{IL_j^{m_j}}\left[\mathrm{erf}\left(\sqrt{2}\left(\lambda_2-C_{IL_j^{m_j}}\right)/\sigma_{IL_j^{m_j}}\right)-\mathrm{erf}\left(\sqrt{2}\left(\lambda_1-C_{IL_j^{m_j}}\right)/\sigma_{IL_j^{m_j}}\right)\right]+\sigma_{IL_i^{n_i}}\left[1+\mathrm{erf}\left(\sqrt{2}\left(\lambda_1-C_{IL_i^{n_i}}\right)/\sigma_{IL_i^{n_i}}\right)-\mathrm{erf}\left(\sqrt{2}\left(\lambda_2-C_{IL_i^{n_i}}\right)/\sigma_{IL_i^{n_i}}\right)\right]}{\sigma_{IL_j^{m_j}}\left[1+\mathrm{erf}\left(\sqrt{2}\left(\lambda_1-C_{IL_j^{m_j}}\right)/\sigma_{IL_j^{m_j}}\right)-\mathrm{erf}\left(\sqrt{2}\left(\lambda_2-C_{IL_j^{m_j}}\right)/\sigma_{IL_j^{m_j}}\right)\right]+\sigma_{IL_i^{n_i}}\left[\mathrm{erf}\left(\sqrt{2}\left(\lambda_2-C_{IL_i^{n_i}}\right)/\sigma_{IL_i^{n_i}}\right)-\mathrm{erf}\left(\sqrt{2}\left(\lambda_1-C_{IL_i^{n_i}}\right)/\sigma_{IL_i^{n_i}}\right)\right]}$
	C. $\sigma_{IL_j^{m_j}} < \sigma_{IL_i^{n_i}}$	$\dfrac{\sigma_{IL_j^{m_j}}\left[1+\mathrm{erf}\left(\sqrt{2}\left(\lambda_2-C_{IL_j^{m_j}}\right)/\sigma_{IL_j^{m_j}}\right)-\mathrm{erf}\left(\sqrt{2}\left(\lambda_1-C_{IL_j^{m_j}}\right)/\sigma_{IL_j^{m_j}}\right)\right]+\sigma_{IL_i^{n_i}}\left[\mathrm{erf}\left(\sqrt{2}\left(\lambda_1-C_{IL_i^{n_i}}\right)/\sigma_{IL_i^{n_i}}\right)-\mathrm{erf}\left(\sqrt{2}\left(\lambda_2-C_{IL_i^{n_i}}\right)/\sigma_{IL_i^{n_i}}\right)\right]}{\sigma_{IL_j^{m_j}}\left[\mathrm{erf}\left(\sqrt{2}\left(\lambda_1-C_{IL_j^{m_j}}\right)/\sigma_{IL_j^{m_j}}\right)-\mathrm{erf}\left(\sqrt{2}\left(\lambda_2-C_{IL_j^{m_j}}\right)/\sigma_{IL_j^{m_j}}\right)\right]+\sigma_{IL_i^{n_i}}\left[1-\mathrm{erf}\left(\sqrt{2}\left(\lambda_1-C_{IL_i^{n_i}}\right)/\sigma_{IL_i^{n_i}}\right)+\mathrm{erf}\left(\sqrt{2}\left(\lambda_2-C_{IL_i^{n_i}}\right)/\sigma_{IL_i^{n_i}}\right)\right]}$
Case 3 $C_{IL_j^{m_j}} < C_{IL_i^{n_i}}$	A. $\sigma_{IL_j^{m_j}} = \sigma_{IL_i^{n_i}}$	$\dfrac{\sigma_{IL_j^{m_j}}\left[\frac{1}{2}-\mathrm{erf}\left(\sqrt{2}\left(\lambda_2-C_{IL_j^{m_j}}\right)/\sigma_{IL_j^{m_j}}\right)\right]+\sigma_{IL_i^{n_i}}\left[\frac{1}{2}+\mathrm{erf}\left(\sqrt{2}\left(\lambda_2-C_{IL_i^{n_i}}\right)/\sigma_{IL_i^{n_i}}\right)\right]}{\sigma_{IL_j^{m_j}}\left[\frac{1}{2}+\mathrm{erf}\left(\sqrt{2}\left(\lambda_2-C_{IL_j^{m_j}}\right)/\sigma_{IL_j^{m_j}}\right)\right]+\sigma_{IL_i^{n_i}}\left[\frac{1}{2}-\mathrm{erf}\left(\sqrt{2}\left(\lambda_2-C_{IL_i^{n_i}}\right)/\sigma_{IL_i^{n_i}}\right)\right]}$
	B. $\sigma_{IL_j^{m_j}} > \sigma_{IL_i^{n_i}}$	$\dfrac{\sigma_{IL_j^{m_j}}\left[\mathrm{erf}\left(\sqrt{2}\left(\lambda_2-C_{IL_j^{m_j}}\right)/\sigma_{IL_j^{m_j}}\right)-\mathrm{erf}\left(\sqrt{2}\left(\lambda_1-C_{IL_j^{m_j}}\right)/\sigma_{IL_j^{m_j}}\right)\right]+\sigma_{IL_i^{n_i}}\left[1+\mathrm{erf}\left(\sqrt{2}\left(\lambda_2-C_{IL_i^{n_i}}\right)/\sigma_{IL_i^{n_i}}\right)-\mathrm{erf}\left(\sqrt{2}\left(\lambda_1-C_{IL_i^{n_i}}\right)/\sigma_{IL_i^{n_i}}\right)\right]}{\sigma_{IL_j^{m_j}}\left[1+\mathrm{erf}\left(\sqrt{2}\left(\lambda_2-C_{IL_j^{m_j}}\right)/\sigma_{IL_j^{m_j}}\right)-\mathrm{erf}\left(\sqrt{2}\left(\lambda_1-C_{IL_j^{m_j}}\right)/\sigma_{IL_j^{m_j}}\right)\right]+\sigma_{IL_i^{n_i}}\left[\mathrm{erf}\left(\sqrt{2}\left(\lambda_1-C_{IL_i^{n_i}}\right)/\sigma_{IL_i^{n_i}}\right)-\mathrm{erf}\left(\sqrt{2}\left(\lambda_2-C_{IL_i^{n_i}}\right)/\sigma_{IL_i^{n_i}}\right)\right]}$
	C. $\sigma_{IL_j^{m_j}} < \sigma_{IL_i^{n_i}}$	$\dfrac{\sigma_{IL_j^{m_j}}\left[1+\mathrm{erf}\left(\sqrt{2}\left(\lambda_1-C_{IL_j^{m_j}}\right)/\sigma_{IL_j^{m_j}}\right)-\mathrm{erf}\left(\sqrt{2}\left(\lambda_2-C_{IL_j^{m_j}}\right)/\sigma_{IL_j^{m_j}}\right)\right]+\sigma_{IL_i^{n_i}}\left[\mathrm{erf}\left(\sqrt{2}\left(\lambda_2-C_{IL_i^{n_i}}\right)/\sigma_{IL_i^{n_i}}\right)-\mathrm{erf}\left(\sqrt{2}\left(\lambda_1-C_{IL_i^{n_i}}\right)/\sigma_{IL_i^{n_i}}\right)\right]}{\sigma_{IL_j^{m_j}}\left[\mathrm{erf}\left(\sqrt{2}\left(\lambda_2-C_{IL_j^{m_j}}\right)/\sigma_{IL_j^{m_j}}\right)-\mathrm{erf}\left(\sqrt{2}\left(\lambda_1-C_{IL_j^{m_j}}\right)/\sigma_{IL_j^{m_j}}\right)\right]+\sigma_{IL_i^{n_i}}\left[1-\mathrm{erf}\left(\sqrt{2}\left(\lambda_2-C_{IL_i^{n_i}}\right)/\sigma_{IL_i^{n_i}}\right)+\mathrm{erf}\left(\sqrt{2}\left(\lambda_1-C_{IL_i^{n_i}}\right)/\sigma_{IL_i^{n_i}}\right)\right]}$

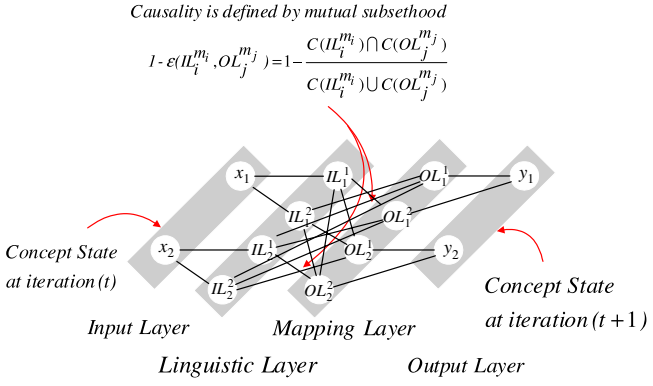

Fig. 2. The structure of the proposed fuzzy neural network.

which has limited values $\mathrm{erf}(-\infty) = -\frac{1}{2}$, $\mathrm{erf}(\infty) = \frac{1}{2}$ and $\mathrm{erf}(0) = 0$.

2.4. Output layer

In the output layer, each node y_j represents the current state of concept j, which is only connected with the linguistic term nodes $OL_j^{m_j}$ $(m_j = 1, \ldots, M_j)$ in the mapping layer along with the crisp weight ξ_{j,m_j}. The input $f_j^{(4)}$ and output $x_j^{(4)}$ of the jth node are calculated by,

$$f_j^{(4)} = \sum_{m_j=1}^{M_j} \xi_{j,m_j} \cdot x_{OL_j^{m_j}}^{(3)}; \qquad x_j^{(4)} = f_j^{(4)}. \tag{10}$$

Eq. (10) indicates that the current state of concept j is a linear function of the defuzzified linguistic terms $OL_j^{m_j}$ $(m_j = 1, \ldots, M_j)$. By doing so, our approach makes better use of the mapping capability offered by the defuzzified linguistic terms.

3. Supervised learning algorithm

In the proposed fuzzy neural network, the supervised learning algorithm based on backpropagation is used to tune the related parameters. According to the architecture and the operations of the proposed approach that have been discussed previously, the parameters that need to be trained can be represented by a vector $P = \left(C_{IL_i^{n_i}}, \sigma_{IL_i^{n_i}}, \xi_{j,m_j}\right)$.

The training process involves repeated presentation of input patterns drawn from the training set and comparing the output of the network with the desired value to obtain the error. The instantaneous squared error $E(\tau) = 1/2 \sum_{i=1}^{N} (d_i(\tau) - x_i(\tau))^2$ is computed as a training performance parameter, where $d_i(\tau)$ and $x_i(\tau)$, respectively denote the desired and computed state value of concept x_i at time step τ. Then the iterative gradient descent update equation is written as,

$$\xi_{j,m_j} = \xi_{j,m_j}(\tau) - \eta(\partial E(\tau)/\partial \xi_{j,m_j}(\tau)) \tag{11}$$

$$C_{IL_i^{n_i}} = C_{IL_i^{n_i}}(\tau) - \eta(\partial E(\tau)/\partial C_{IL_i^{n_i}}(\tau)) \tag{12}$$

$$\sigma_{IL_i^{n_i}} = \sigma_{IL_i^{n_i}}(\tau) - \eta(\partial E(\tau)/\partial \sigma_{IL_i^{n_i}}(\tau)) \tag{13}$$

where η is the learning rate. Then the standard iterative pattern based gradient descent method and the chain rule are utilized,

Table 2
Calculation of $\partial\varepsilon(IL_i^{n_i}, OL_l^{m_l})/\partial C_{IL_i^{n_i}}$.

	$\partial\varepsilon(IL_i^{n_i}, OL_l^{m_l})/\partial C_{IL_i^{n_i}}$
$C_{IL_j^{m_j}} = C_{IL_i^{n_i}}$	0
$C_{IL_j^{m_j}} > C_{IL_i^{n_i}}$ or $C_{IL_j^{m_j}} < C_{IL_i^{n_i}}$	$\dfrac{\frac{\partial\left(C\left(IL_i^{n_i}\cap OL_j^{m_j}\right)\right)}{\partial C_{IL_i^{n_i}}}\left[\sqrt{\pi}\left(\sigma_{OL_j^{m_j}}+\sigma_{IL_i^{n_i}}\right)-C\left(IL_i^{n_i}\cap OL_j^{m_j}\right)\right]-\frac{\partial\left[\sqrt{\pi}\left(\sigma_{OL_j^{m_j}}+\sigma_{IL_i^{n_i}}\right)-C\left(IL_i^{n_i}\cap OL_j^{m_j}\right)\right]}{\partial C_{IL_i^{n_i}}}C\left(IL_i^{n_i}\cap OL_j^{m_j}\right)}{\left[\sqrt{\pi}\left(\sigma_{OL_j^{m_j}}+\sigma_{IL_i^{n_i}}\right)-C\left(IL_i^{n_i}\cap OL_j^{m_j}\right)\right]^2}$

Table 3
Calculation of $\partial\varepsilon(IL_i^{n_i}, OL_l^{m_l})/\partial\sigma_{IL_i^{n_i}}$.

	$\partial\varepsilon(IL_i^{n_i}, OL_l^{m_l})/\partial\sigma_{IL_i^{n_i}}$		
$C_{IL_j^{m_j}} = C_{IL_i^{n_i}}$	$\sigma_{IL_i^{n_i}} = \sigma_{OL_j^{m_j}}$	$\sigma_{OL_j^{m_j}} > \sigma_{IL_i^{n_i}}$	$\sigma_{OL_j^{m_j}} < \sigma_{IL_i^{n_i}}$
	0	$1/\sigma_{OL_j^{m_j}}$	$-\sigma_{OL_j^{m_j}}/\sigma^2_{IL_i^{n_i}}$
$C_{IL_j^{m_j}} > C_{IL_i^{n_i}}$ or $C_{IL_j^{m_j}} < C_{IL_i^{n_i}}$	$\dfrac{\frac{\partial\left(C\left(IL_i^{n_i}\cap OL_j^{m_j}\right)\right)}{\partial\sigma_{IL_i^{n_i}}}\left[\sqrt{\pi}\left(\sigma_{OL_j^{m_j}}+\sigma_{IL_i^{n_i}}\right)-C\left(IL_i^{n_i}\cap OL_j^{m_j}\right)\right]-\frac{\partial\left[\sqrt{\pi}\left(\sigma_{OL_j^{m_j}}+\sigma_{IL_i^{n_i}}\right)-C\left(IL_i^{n_i}\cap OL_j^{m_j}\right)\right]}{\partial\sigma_{IL_i^{n_i}}}C\left(IL_i^{n_i}\cap OL_j^{m_j}\right)}{\left[\sqrt{\pi}\left(\sigma_{OL_j^{m_j}}+\sigma_{IL_i^{n_i}}\right)-C\left(IL_i^{n_i}\cap OL_j^{m_j}\right)\right]^2}$		

layer by layer, starting from the output layer. Once the network is trained to the desired level of error, it is tested by unseen test set patterns.

Particularly, we need to stress the learning process in the mapping layer. If $i = j$ and $n_i = m_j$, then, the node $IL_i^{n_i}$ in the linguistic term layer and the node $OL_j^{m_j}$ in the mapping layer should be described by the same Gaussian function so that they have the same semantic meaning. Therefore, according to the updated values of $C_{IL_i^{n_i}}$ and $\sigma_{IL_i^{n_i}}$ in the linguistic layer, the parameters $C_{OL_j^{m_j}}$ and $\sigma_{OL_j^{m_j}}$ in the mapping layer are tuned simultaneously.

Therefore, the expressions of partial derivatives in the above update equations are derived as follows.

3.1. Update in output layer

In this layer, the weight ξ_{j,m_j} which connects the output linguistic term $OL_j^{m_j}$ with the jth crisp output is adjusted according to Eq. (11). Since the output y_l ($l \neq j$) is independent of the given ξ_{j,m_j}, the error derivative with respect to ξ_{j,m_j} is computed by,

$$\partial E(\tau)/\partial\xi_{j,m_j}(\tau) = \frac{\partial E}{\partial y_l}\cdot\frac{\partial y_l}{\partial\xi_{j,m_j}} = -(d_j - y_j)x^{(3)}_{OL_j^{m_j}} \tag{14}$$

where d_j and y_j, respectively denote the desired and the computed state value of concept j.

3.2. Update in mapping layer

As Eq. (8) shows, the operation of the neurons in the mapping layer completely depends on the output of the linguistic layer, center $C_{OL_i^{n_i}}$ and width $\sigma_{OL_i^{n_i}}$. As mentioned in Section 2.3, the center $C_{OL_i^{n_i}}$ and width $\sigma_{OL_i^{n_i}}$ of the linguistic term $OL_i^{n_i}$ are the same as center $C_{IL_i^{n_i}}$ and width $\sigma_{IL_i^{n_i}}$ of the linguistic term $IL_i^{n_i}$ which have been defined in the linguistic layer. Therefore, the updates of parameters $C_{OL_i^{n_i}}$ and $\sigma_{OL_i^{n_i}}$ are simultaneously performed as in the linguistic layer.

3.3. Update in linguistic layer

In the linguistic layer, the center $C_{IL_i^{n_i}}$ and width $\sigma_{OL_i^{n_i}}$ associating with the linguistic term $IL_i^{n_i}$ are tuned according to Eqs. (12) and (13). For the error derivative with respect to $C_{IL_i^{n_i}}$ and $\sigma_{IL_i^{n_i}}$,

$$\frac{\partial E}{\partial C_{IL_i^{n_i}}} = \sum_{\substack{l=1\\ l\neq j}}^{N}\frac{\partial E}{\partial y_l}\sum_{m_l=1}^{N}\frac{\partial y_l}{\partial x^{(3)}_{OL_l^{m_l}}}\cdot\frac{\partial x^{(3)}_{OL_l^{m_l}}}{\partial f^{(3)}_{OL_l^{m_l},IL_i^{n_i}}}\cdot\frac{\partial f^{(3)}_{OL_l^{m_l},IL_i^{n_i}}}{\partial C_{IL_i^{n_i}}} \tag{15}$$

$$\frac{\partial E}{\partial \sigma_{IL_i^{n_i}}} = \sum_{\substack{l=1\\ l\neq j}}^{N}\frac{\partial E}{\partial y_l}\sum_{m_l=1}^{N}\frac{\partial y_l}{\partial x^{(3)}_{OL_l^{m_l}}}\cdot\frac{\partial x^{(3)}_{OL_l^{m_l}}}{\partial f^{(3)}_{OL_l^{m_l},IL_i^{n_i}}}\cdot\frac{\partial f^{(3)}_{OL_l^{m_l},IL_i^{n_i}}}{\partial \sigma_{IL_i^{n_i}}}. \tag{16}$$

So we can rewrite Eqs. (15) and (16) by substituting Eqs. (4), (8) and (10), and obtain the expressions in Box I.

In Eqs. (17) and (18), $\partial\varepsilon(IL_i^{n_i}, OL_l^{m_l})/\partial C_{IL_i^{n_i}}$ and $\partial\varepsilon(IL_i^{n_i}, OL_l^{m_l})/\partial\sigma_{IL_i^{n_i}}$ are essential to compute $\partial E/\partial C_{IL_i^{n_i}}$ and $\partial E/\partial\sigma_{IL_i^{n_i}}$. Corresponding to the case-wise expressions of mutual subsethood $\varepsilon(IL_i^{n_i}, OL_j^{m_j})$ that is described in Table 1, the $\partial\varepsilon(IL_i^{n_i}, OL_l^{m_l})/\partial C_{IL_i^{n_i}}$ and $\partial\varepsilon(IL_i^{n_i}, OL_l^{m_l})/\partial\sigma_{IL_i^{n_i}}$ are calculated and summarized in Tables 2 and 3 respectively.

For the $\partial(C(IL_i^{n_i}\cap OL_j^{m_j}))/\partial C_{IL_i^{n_i}}$ and $\partial(C(IL_i^{n_i}\cap OL_j^{m_j}))/\partial\sigma_{IL_i^{n_i}}$ in Tables 2 and 3, we can respectively derive and describe them in Tables 4 and 5.

In terms of the calculated $\partial(C(IL_i^{n_i}\cap OL_j^{m_j}))/\partial C_{IL_i^{n_i}}$, $\partial(C(IL_i^{n_i}\cap OL_j^{m_j}))/\partial\sigma_{IL_i^{n_i}}$, $\partial\mathcal{E}(IL_i^{n_i}, OL_l^{m_l})/\partial C_{IL_i^{n_i}}$ and $\partial\mathcal{E}(IL_i^{n_i}, OL_l^{m_l})/\partial\sigma_{IL_i^{n_i}}$, we can update the center $C_{IL_i^{n_i}}$ and width $\sigma_{OL_i^{n_i}}$ in the linguistic layer according to Eqs. (17) and (18).

4. Simulations

This section describes three simulation examples of the proposed approach. These examples include chaotic Duffing forced-oscillation system (Example 1), Mackey–Glass dataset (Example 2)

$$\frac{\partial E}{\partial C_{IL_i^{n_i}}} = \sum_{\substack{l=1 \\ l\neq j}}^{N}(y_l - d_l)\sum_{m_l=1}^{N_l}\frac{\partial y_l}{\partial x_{OL_l^{m_l}}^{(3)}}\cdot\frac{\partial x_{OL_l^{m_l}}^{(3)}}{\partial f_{OL_l^{m_l},IL_i^{n_i}}^{(3)}}\left\{\left[\frac{\partial x_{IL_i^{n_i}}^{(2)}}{\partial C_{IL_i^{n_i}}}\left(1-\varepsilon\left(IL_i^{n_i},OL_l^{m_l}\right)\right)\right]-\left[x_{IL_i^{n_i}}^{(2)}\cdot\frac{\partial\varepsilon\left(IL_i^{n_i},OL_l^{m_l}\right)}{\partial C_{IL_i^{n_i}}}\right]\right\}$$

$$= \sum_{\substack{l=1 \\ l\neq j}}^{N}(y_l - d_l)\sum_{m_l=1}^{N_l}\xi_{l,m_l}\cdot\frac{\sum_{i=1,i\neq l}^{N}\left(C_{IL_i^{n_i}}\cdot\sigma_{IL_i^{n_i}}\right)\cdot\left(\sum_{i=1,i\neq l}^{N}f_{OL_l^{m_l},IL_i^{n_i}}^{(3)}\cdot\sigma_{IL_i^{n_i}}\right)-\sum_{i=1,i\neq l}^{N}\left(f_{OL_l^{m_l},IL_i^{n_i}}^{(3)}\cdot C_{IL_i^{n_i}}\cdot\sigma_{IL_i^{n_i}}^2\right)}{\left(\sum_{i=1,i\neq l}^{N}f_{OL_l^{m_l},IL_i^{n_i}}^{(3)}\cdot\sigma_{IL_i^{n_i}}\right)^2}$$

$$\times\left\{\left[\left(x_{IL_i^{n_i}}^{(2)}\cdot\frac{2\left(x_i^{(1)}-C_{IL_i^{n_i}}\right)}{\sigma_{IL_i^{n_i}}^2}\right)\left(1-\varepsilon\left(IL_i^{n_i},OL_l^{m_l}\right)\right)\right]-\left[x_{IL_i^{n_i}}^{(2)}\cdot\frac{\partial\varepsilon\left(IL_i^{n_i},OL_l^{m_l}\right)}{\partial C_{IL_i^{n_i}}}\right]\right\} \quad (17)$$

$$\frac{\partial E}{\partial \sigma_{IL_i^{n_i}}} = \sum_{\substack{l=1 \\ l\neq i}}^{N}(y_l - d_l)\sum_{m_l=1}^{N_l}\xi_{l,m_l}\cdot\frac{\sum_{i=1,i\neq l}^{N}\left(C_{IL_i^{n_i}}\cdot\sigma_{IL_i^{n_i}}\right)\cdot\left(\sum_{i=1,i\neq l}^{N}f_{OL_l^{m_l},IL_i^{n_i}}^{(3)}\cdot\sigma_{IL_i^{n_i}}\right)-\sum_{i=1,i\neq l}^{N}\left(f_{OL_l^{m_l},IL_i^{n_i}}^{(3)}\cdot C_{IL_i^{n_i}}\cdot\sigma_{IL_i^{n_i}}^2\right)}{\left(\sum_{i=1,i\neq l}^{N}f_{OL_l^{m_l},IL_i^{n_i}}^{(3)}\cdot\sigma_{IL_i^{n_i}}\right)^2}$$

$$\times\left\{\left[\left(x_{IL_i^{n_i}}^{(2)}\cdot\frac{2\left(x_i^{(1)}-C_{IL_i^{n_i}}\right)^2}{\sigma_{IL_i^{n_i}}^3}\right)\left(1-\varepsilon\left(IL_i^{n_i},OL_l^{m_l}\right)\right)\right]-\left[x_{IL_i^{n_i}}^{(2)}\cdot\frac{\partial\varepsilon\left(IL_i^{n_i},OL_l^{m_l}\right)}{\partial \sigma_{IL_i^{n_i}}}\right]\right\} \quad (18)$$

Box I.

Table 4
Calculation of $\partial(C(IL_i^{n_i}\cap OL_j^{m_j}))/\partial C_{IL_i^{n_i}}$.

		$\lambda_1=(\sigma_{IL_j^{m_j}}C_{IL_i^{n_i}}-\sigma_{IL_i^{n_i}}C_{IL_j^{m_j}})/(\sigma_{IL_j^{m_j}}-\sigma_{IL_i^{n_i}})$; $\lambda_2=(\sigma_{IL_j^{m_j}}C_{IL_i^{n_i}}+\sigma_{IL_i^{n_i}}C_{IL_j^{m_j}})/(\sigma_{IL_j^{m_j}}+\sigma_{IL_i^{n_i}})$ are crossing points between $IL_i^{n_i}$ and $IL_j^{m_j}$
Case 2 $C_{IL_j^{m_j}} > C_{IL_i^{n_i}}$	A. $\sigma_{IL_j^{m_j}}=\sigma_{IL_i^{n_i}}$	$\exp\left(-\left(\left(\lambda_2-C_{IL_i^{n_i}}\right)/\sigma_{IL_i^{n_i}}\right)^2\right)$
	B. $\sigma_{IL_j^{m_j}}>\sigma_{IL_i^{n_i}}$	$\exp\left(-\left(\left(\lambda_2-C_{IL_i^{n_i}}\right)/\sigma_{IL_i^{n_i}}\right)^2\right)-\exp\left(-\left(\left(\lambda_1-C_{IL_i^{n_i}}\right)/\sigma_{IL_i^{n_i}}\right)^2\right)$
	C. $\sigma_{IL_j^{m_j}}<\sigma_{IL_i^{n_i}}$	$\exp\left(-\left(\left(\lambda_1-C_{IL_i^{n_i}}\right)/\sigma_{IL_i^{n_i}}\right)^2\right)-\exp\left(-\left(\left(\lambda_2-C_{IL_i^{n_i}}\right)/\sigma_{IL_i^{n_i}}\right)^2\right)$
Case 3 $C_{IL_j^{m_j}} < C_{IL_i^{n_i}}$	A. $\sigma_{IL_j^{m_j}}=\sigma_{IL_i^{n_i}}$	$-\exp\left(-\left(\left(\lambda_2-C_{IL_i^{n_i}}\right)/\sigma_{IL_i^{n_i}}\right)^2\right)$
	B. $\sigma_{IL_j^{m_j}}>\sigma_{IL_i^{n_i}}$	$\exp\left(-\left(\left(\lambda_1-C_{OL_j^{m_j}}\right)/\sigma_{OL_j^{m_j}}\right)^2\right)-\exp\left(-\left(\left(\lambda_2-C_{OL_j^{m_j}}\right)/\sigma_{OL_j^{m_j}}\right)^2\right)$
	C. $\sigma_{IL_j^{m_j}}<\sigma_{IL_i^{n_i}}$	$\exp\left(-\left(\left(\lambda_1-C_{OL_j^{m_j}}\right)/\sigma_{OL_j^{m_j}}\right)^2\right)-\exp\left(-\left(\left(\lambda_2-C_{OL_j^{m_j}}\right)/\sigma_{OL_j^{m_j}}\right)^2\right)$

and chaotic Lorenz attractor system (Example 3). The performance of the proposed approach is compared with the recently developed fuzzy systems and fuzzy neural networks (Alpaydn & Dundar, 2002, Chun, 2007, Jun, Henry, & Lo, 2008, Song et al., 2009, Zhang et al., 2003) in these examples.

In all applications, the centers of fuzzy sets are randomly initialized in the range of the minimum and maximum values of the input variables. The spreads of all fuzzy sets involved in the proposed method are randomly initialized in the range (0, 1). Once the proposed approach is trained to the desired level of error, it is tested by presenting unseen test set patterns. As discussed in Section 2, the FCM model corresponding to each application can be reconstructed on the basis of the membership functions and the relevant parameters that are automatically identified by the training process.

4.1. Duffing forced-oscillation system

As a second-order chaotic system, the Duffing forced-oscillation system describes a special nonlinear circuit or a pendulum moving in a viscous medium, which is formulated,

$$\ddot{x} = f(x) + \mu \quad (19)$$

where $f(x) = -P\dot{x} - P_1x - P_2x^3 + q\cos(wt)$ is the system dynamics; t is the time variable, w is the frequency; μ is the control effort; P, P_1, P_2 and q are real constants. Depending on the choice of these constrains, it is known that the solutions of Eq. (19) may exhibit periodic, almost periodic, and chaotic behavior. To test the capability and applicability of the proposed approach, the complex phase space of the Duffing forced-oscillation system is selected as an example. For observing the complex phase space of the Duffing forced-oscillation system, the system behavior with $\mu = 0$ is simulated with $P = 0.4, P_1 = -1.1, P_2 = 1.0, w = 1.8$ and $q = 1.95$. The phase plane plotting from an initial condition point $[x(t=0) = -1.2501, \dot{x}(t=0) = 1.5602]$ is shown in Fig. 3. Note that, in Fig. 3, the magnitudes of $x(t)$ and $\dot{x}(t)$ are one-tenth of their actual values.

To facilitate comparisons with other approaches (Alpaydn & Dundar, 2002; Chun, 2007; Jun et al., 2008; Song et al., 2009), the

Table 5
Calculation of $\partial(C(IL_i^{n_i} \cap OL_j^{m_j}))/\partial\sigma_{IL_i^{n_i}}$.

		$\lambda_1 = (\sigma_{IL_j^{m_j}} C_{IL_i^{n_i}} - \sigma_{IL_i^{n_i}} C_{IL_j^{m_j}})/(\sigma_{IL_j^{m_j}} - \sigma_{IL_i^{n_i}})$; $\lambda_2 = (\sigma_{IL_j^{m_j}} C_{IL_i^{n_i}} + \sigma_{IL_i^{n_i}} C_{IL_j^{m_j}})/(\sigma_{IL_j^{m_j}} + \sigma_{IL_i^{n_i}})$ are crossing points between $IL_i^{n_i}$ and $IL_j^{m_j}$
Case 2 $C_{IL_j^{m_j}} > C_{IL_i^{n_i}}$	A. $\sigma_{IL_j^{m_j}} = \sigma_{IL_i^{n_i}}$	$\frac{\lambda_2 - C_{IL_i^{n_i}}}{\sigma_{IL_i^{n_i}}} \exp\left(-\left(\left(\lambda_2 - C_{IL_i^{n_i}}\right)/\sigma_{IL_i^{n_i}}\right)^2\right) + \sqrt{\pi}\left[\frac{1}{2} - \operatorname{erf}\left(\frac{\sqrt{2}\left(\lambda_2 - C_{IL_i^{n_i}}\right)}{\sigma_{IL_i^{n_i}}}\right)\right]$
	B. $\sigma_{IL_j^{m_j}} > \sigma_{IL_i^{n_i}}$	$\frac{\lambda_2 - C_{IL_i^{n_i}}}{\sigma_{IL_i^{n_i}}} \exp\left(-\left(\left(\lambda_2 - C_{IL_i^{n_i}}\right)/\sigma_{IL_i^{n_i}}\right)^2\right) - \frac{\lambda_1 - C_{IL_i^{n_i}}}{\sigma_{IL_i^{n_i}}} \exp\left(-\left(\left(\lambda_1 - C_{IL_i^{n_i}}\right)/\sigma_{IL_i^{n_i}}\right)^2\right) + \sqrt{\pi}\left[\operatorname{erf}\left(\frac{\sqrt{2}\left(\lambda_1 - C_{IL_i^{n_i}}\right)}{\sigma_{IL_i^{n_i}}}\right) - \operatorname{erf}\left(\frac{\sqrt{2}\left(\lambda_2 - C_{IL_i^{n_i}}\right)}{\sigma_{IL_i^{n_i}}}\right)\right]$
	C. $\sigma_{IL_j^{m_j}} < \sigma_{IL_i^{n_i}}$	$\frac{\lambda_2 - C_{IL_i^{n_i}}}{\sigma_{IL_i^{n_i}}} \exp\left(-\left(\left(\lambda_2 - C_{IL_i^{n_i}}\right)/\sigma_{IL_i^{n_i}}\right)^2\right) - \frac{\lambda_1 - C_{IL_i^{n_i}}}{\sigma_{IL_i^{n_i}}} \exp\left(-\left(\left(\lambda_1 - C_{IL_i^{n_i}}\right)/\sigma_{IL_i^{n_i}}\right)^2\right) + \sqrt{\pi}\left[1 + \operatorname{erf}\left(\frac{\sqrt{2}\left(\lambda_1 - C_{IL_i^{n_i}}\right)}{\sigma_{IL_i^{n_i}}}\right) - \operatorname{erf}\left(\frac{\sqrt{2}\left(\lambda_2 - C_{IL_i^{n_i}}\right)}{\sigma_{IL_i^{n_i}}}\right)\right]$
Case 3 $C_{IL_j^{m_j}} < C_{IL_i^{n_i}}$	A. $\sigma_{IL_j^{m_j}} = \sigma_{IL_i^{n_i}}$	$\sqrt{\pi}\left[\frac{1}{2} + \operatorname{erf}\left(\frac{\sqrt{2}\left(\lambda_2 - C_{IL_i^{n_i}}\right)}{\sigma_{IL_i^{n_i}}}\right)\right] - \frac{\left(\lambda_2 - C_{IL_i^{n_i}}\right)}{\sigma_{IL_i^{n_i}}} \exp\left(-\left(\frac{\lambda_2 - C_{IL_i^{n_i}}}{\sigma_{IL_i^{n_i}}}\right)^2\right)$
	B. $\sigma_{IL_j^{m_j}} > \sigma_{IL_i^{n_i}}$	$\frac{\lambda_1 - C_{IL_i^{n_i}}}{\sigma_{IL_i^{n_i}}} \exp\left(-\left(\frac{\lambda_1 - C_{IL_i^{n_i}}}{\sigma_{IL_i^{n_i}}}\right)^2\right) - \frac{\lambda_2 - C_{IL_i^{n_i}}}{\sigma_{IL_i^{n_i}}} \exp\left(-\left(\frac{\lambda_2 - C_{IL_i^{n_i}}}{\sigma_{IL_i^{n_i}}}\right)^2\right) + \sqrt{\pi}\left[1 + \operatorname{erf}\left(\frac{\sqrt{2}\left(\lambda_2 - C_{IL_i^{n_i}}\right)}{\sigma_{IL_i^{n_i}}}\right) - \operatorname{erf}\left(\frac{\sqrt{2}\left(\lambda_1 - C_{IL_i^{n_i}}\right)}{\sigma_{IL_i^{n_i}}}\right)\right]$
	C. $\sigma_{IL_j^{m_j}} < \sigma_{IL_i^{n_i}}$	$\frac{\lambda_1 - C_{IL_i^{n_i}}}{\sigma_{IL_i^{n_i}}} \exp\left(-\left(\frac{\lambda_1 - C_{IL_i^{n_i}}}{\sigma_{IL_i^{n_i}}}\right)^2\right) - \frac{\lambda_2 - C_{IL_i^{n_i}}}{\sigma_{IL_i^{n_i}}} \exp\left(-\left(\frac{\lambda_2 - C_{IL_i^{n_i}}}{\sigma_{IL_i^{n_i}}}\right)^2\right) + \sqrt{\pi}\left[\operatorname{erf}\left(\frac{\sqrt{2}\left(\lambda_2 - C_{IL_i^{n_i}}\right)}{\sigma_{IL_i^{n_i}}}\right) - \operatorname{erf}\left(\frac{\sqrt{2}\left(\lambda_1 - C_{IL_i^{n_i}}\right)}{\sigma_{IL_i^{n_i}}}\right)\right]$

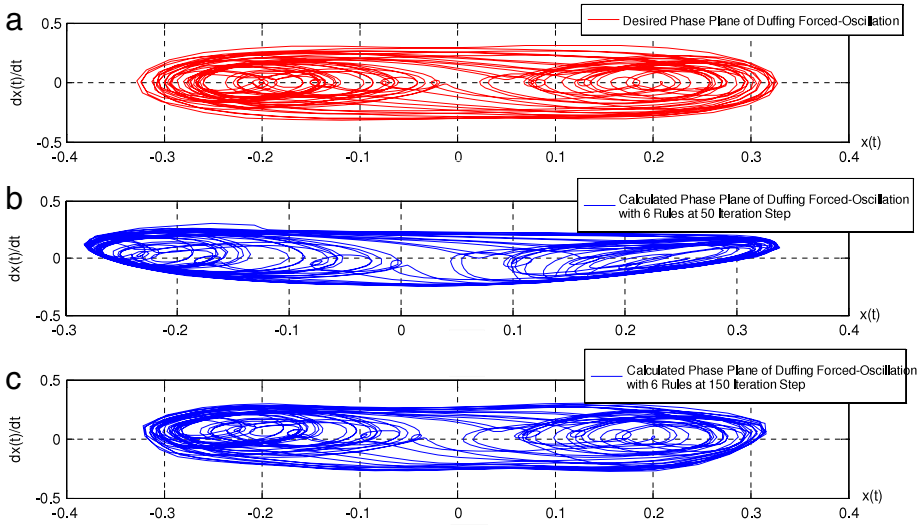

Fig. 3. The Duffing forced oscillation and reasoning results.

Table 6
Summarized fuzzy rules for the forced-oscillation system.

Rule index	Summarization of fuzzy rules		
	Antecedent parts	Causalities	Output
R1	IF '$x(t)$ IS S' ON '$x(t)$ IS L'	0.50945	$x(t) = 0.52798 \propto (IL_1^1)$
R2	IF '$x(t)$ IS S' ON '$\dot{x}(t)$ IS S'	0.64268	$+0.34232 \propto (IL_1^2)$
R3	IF '$x(t)$ IS S' ON '$\dot{x}(t)$ IS L'	0.85885	
R4	IF '$x(t)$ IS L' ON '$\dot{x}(t)$ IS S'	0.67420	$\dot{x}(t) = -0.71172 \propto (IL_2^1)$
R5	IF '$x(t)$ IS L' ON '$\dot{x}(t)$ IS L'	0.00111	$+0.46623 \propto (IL_2^2)$
R6	IF '$\dot{x}(t)$ IS L' ON '$\dot{x}(t)$ IS L'	0.24360	

input–output pair is described as $(x(t), \dot{x}(t); x(t+\Delta\tau), \dot{x}(t+\Delta\tau))$ where $\Delta\tau = 6$. Furthermore, 500 data points and 1100 data points are randomly selected as training set and test data. Each input and each output are described by two linguistic terms (Large and Small). All testing results given in this example are the average performance after 10 runs. The intermediate and final testing results are also respectively depicted in Fig. 3. In addition, the membership functions corresponding to the linguistic terms of each input in antecedent parts have been demonstrated in Fig. 4. Since $\varepsilon(IL_i^{m_i}, IL_j^{m_j}) = \varepsilon(IL_j^{m_j}, IL_i^{m_i})$ as described by Eq. (5) and each input is described by two linguistic terms, Table 6 summarizes the derived six fuzzy rules where $\propto (IL_i^{m_i})$ $(i = 1, 2; m_i = 1, 2)$ denotes the

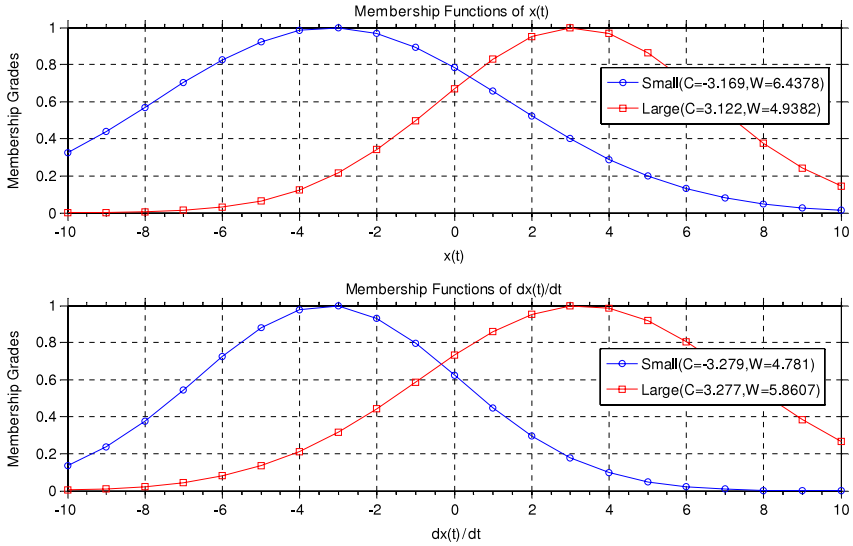

Fig. 4. Membership functions of inputs.

Table 7
Comparisons with other methods for the Duffing forced-oscillation system based on NMSE and MSE.

Model	Training data (no. of data)	Training epoch	No. of rules	NMSE	MSE
MANFIS[b]	500	500	4	$0.02878^{(1)}$	NA
SAFNC[c]	NA	NA	5	$0.0266^{(1)}$	NA
EXML[a]	500	1	36	0.1742	NA
MSBFNN[d]	500	50	9	$0.0247^{(1)}$	NA
		150	9	$0.0192^{(2)}$	NA
SVD-QR[e]	NA	NA	5	NA	0.0038
SBFN[e]	NA	NA	5	NA	0.0015
Our approach	500	50	6	$0.0257^{(1)}$	0.00273
		150	6	$0.0165^{(2)}$	0.0013

[a] 36 is the number of hidden neurons in EXML. Source code is available in http://www3.ntu.edu.sg/home/egbhuang.
[b] Results adapted from Jun et al. (2008).
[c] Results adapted from Chun (2007).
[d] Results adapted from Song et al. (2009).
[e] Results adapted from Alpaydn and Dundar (2002).

defuzzified value of $x(t)$ and $\dot{x}(t)$. Note that the fuzzy rules essentially mean the different combinations of hidden neurons between the linguistic layer and the mapping layer.

The prediction error is studied by calculating the mean squared error (MSE) and the normalized MSE (NMSE), which are respectively described as,

$$\text{MSE} = 1/(N)\sum_{k\in\varkappa}\left(x_k - \hat{x}_k\right)^2;$$

$$\text{NMSE} = 1/\left(N\cdot\widehat{\sigma}_{\varkappa}^2\right)\sum_{k\in\varkappa}\left(x_k - \hat{x}_k\right)^2;$$

where x_k and $\hat{x}_k$ are the actual and the predicted kth point of the series of length $\varkappa$. $\widehat{\sigma}_{\varkappa}$ denotes the sample variance of the actual values (targets) in $\varkappa$.

On the basis of NMSE and MSE, Table 7 compares the structure and results generated by our approach, multi-input–multi-output-ANFIS (MANFIS) (Jun et al., 2008), self-organizing adaptive fuzzy neural control (SAFNC) (Chun, 2007), Extreme Learning Machine (EXLM), MSBFNN (Song et al., 2009), Evolution-Based fuzzy networks (SBFN) (Alpaydn & Dundar, 2002) and Singular-Value-QR (SVD-QR). To facilitate comparison, 'no. of rules' in Table 7 are explained as number of the different combinations of hidden neurons between the linguistic layer and the mapping layer. In the above comparison, it is worth noting that EXML significantly reduces the training epoch compared to other models due to its special learning algorithm. However, this also makes it hard to further improve the performance of EXML which greatly depends on the number of hidden neurons.

Compared with other models, our approach generates the smallest MSE and NMSE after 150 training epochs. For the models with similar $\text{NMSE}^{(1),(2)}$, our approach needs fewer rules or training epochs, which indicates that the rule set generated by our approach is more efficient and compact to replicate the dynamic behavior of Duffing forced-oscillation system than other models.

4.2. Mackey–Glass chaotic time series

Here, we evaluate the performance of our approach by applying it to the forecasting of a Mackey–Glass chaotic time series (MG). The dataset of MG is generated from the following delay differential equation (Mackey & Glass, 1997),

$$\frac{dx(t)}{dt} = \frac{0.2x(x-\tau)}{1+x^{10}(t-\tau)} - 0.1x(t) \tag{20}$$

when $\tau > 17$, Eq. (20) exhibits chaotic behavior.

The task of the time series prediction is to predict future values $x(t+\Delta t)$ (Δt being the prediction time step) based on a set of values of $x(t)$ at certain times. To facilitate comparison with other models, including dynamic evolving neural-fuzzy inference system (DENFIS) (Kasabov & Song, 2002), data-driven linguistic model (DDLM) (Gaweda & Zurada, 2003) and support vector echo state machine (SVESM) (Shi & Han, 2007), the time series dataset is generated using the Runge–Kutta procedure with initial condition $x(0) = 1.2$ and $\Delta t = 6$. Furthermore, the input–output pair is described by $[x(t-24), x(t-18), x(t-12), x(t-6); x(t), x(t+6), x(t+12), x(t+18)]$.

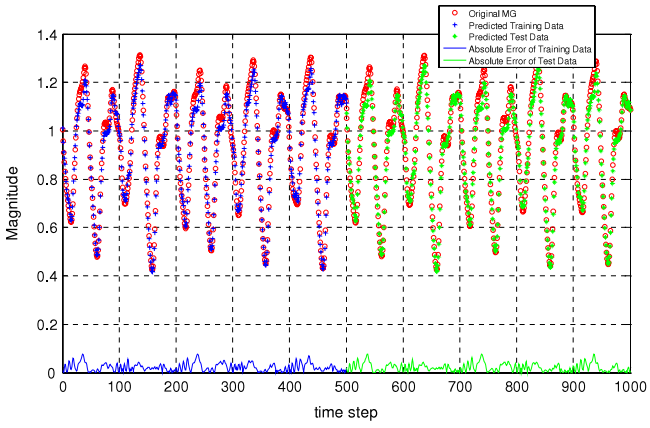

Fig. 5. The original time series for training and test data sets.

In this case, the following experiment was conducted: 1000 data patterns are generated, from $t = 124$ to $t = 1123$. The first 500 patterns are taken as the training data and the other 500 patterns are used for testing. Each input variable and the single output are represented by two linguistic terms ('Large', 'Medium' and 'Small'). To avoid a possible distortion of test results due to function regularities (periodicity, etc.), this experiment is repeated 6 times for eliminating possible outliers.

Fig. 5 shows the predicted and the original time series for training and test data sets. By using the learning algorithms that are described in Section 3, the membership functions of each input are automatically identified and depicted by Fig. 6.

On the basis of root mean squared error (RMSE) and the non-dimensional error index (NDEI, defined as the RMSE divided by the standard deviation of the target series), Tables 8 and 9 compare the average performance of the proposed approach with other methods in Gaweda and Zurada (2003), Kasabov and Song (2002), Shi and Han (2007) and Song et al. (2009).

In contrast with DENFIS, ANFIS and SVESM, the proposed approach provides a better prediction accuracy with fewer training epochs and training parameters. This indicates that our approach results in better compression of linguistic information to describe the dynamics underlying the MG dataset. In terms of the RMSE and NDEI, we can see that the prediction accuracy generated by our approach is comparable with that of MSBFNN (Song et al.,

Table 8
Comparison of MSBFNN with other methods for MG DATA based on NDEI.

Model	Training epochs	Testing NDEI
MLP-BP[b]	500	0.022
DENFIS I[b]	2	0.019
DENFIS II[b]	100	0.016
ANFIS[b]	50	0.036
ANFIS[b]	200	0.029
SVESM[c]	NA	0.0159
DDLM[d]	NA	0.009
MSBFNN[a]	150	0.0107
Our approach	50	0.0212
	100	0.012

[a] Results adapted from Song et al. (2009).
[b] Results adapted from Kasabov and Song (2002).
[c] Results adapted from Shi and Han (2007).
[d] Results adapted from Gaweda and Zurada (2003).

Table 9
Comparison of ProFNN with other methods for MG DATA based on RMSE.

Model	No. of parameters	RMSE
NEFPROX[b]	NA	0.0533
G-FNN[b]	NA	0.0056
SEIT2FNN[b]	NA	0.0034
D-FNN[b]	100	0.008
ILA[c]	37	0.0066
ANFIS[c]	104	0.003
OLS[c]	211	0.0089
RBF-AFS[c]	210	0.0128
MSBFNN[a]	178	0.0024
Our approach	36	0.0027

[a] Results adapted from Song et al. (2009).
[b] Results adapted from Juang and Tsao (2008).
[c] Results adapted from Kasabov and Song (2002).

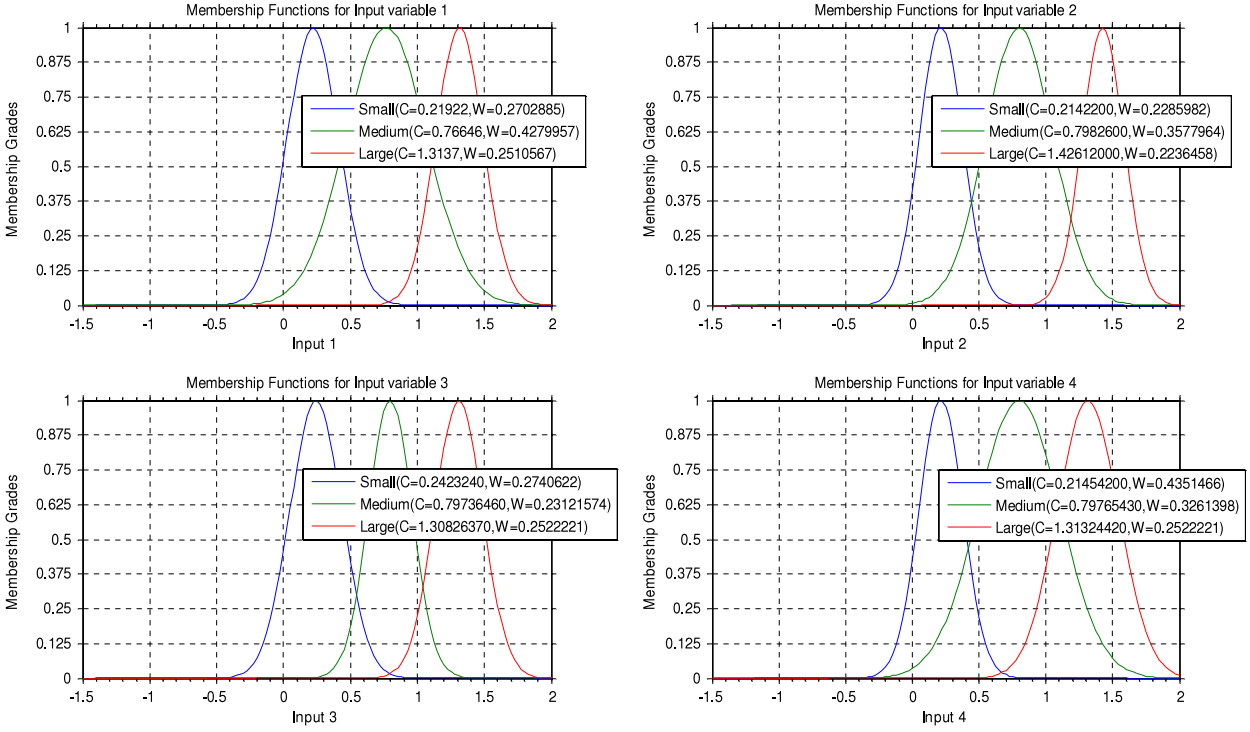

Fig. 6. Membership functions of input variables for Mackey–Class dataset.

Table 10
Comparisons with other methods for the Lorenz attractor system based on RMSE and MSE.

Model	Training/Testing (no. of data)	Training epoch	No. of hidden neurons	No. of parameters	RMSE
NN-based AR model[a]	1000/400	1000	17	81	0.0108
NN-based ARMA model[a]	1000/400	400	18	89	0.018
SVM[b]	200/300	900	NA	NA	0.1410
ES-SVM[b]	200/300	900	NA	NA	0.0352
KKF[c]	NA	NA	NA	NA	0.098
Our approach	1000/1128	150	18	27	0.0273
	1000/1128	300	18	27	0.01647

[a] Results adapted from Dudul (2005).
[b] Results adapted from Hou and Li (2009).
[c] Results adapted from Ralaivola and Alche-Buc (2003).

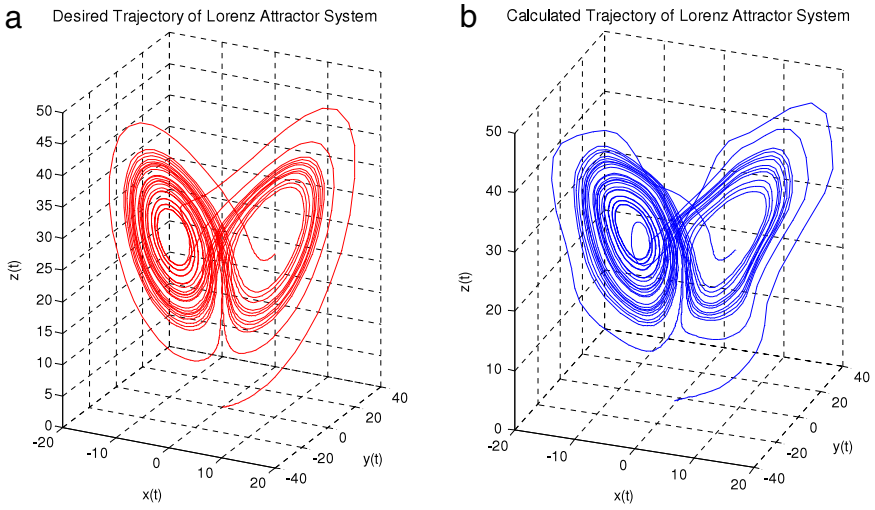

Fig. 7. The desired and the calculated trajectory of the Lorenz attractor.

2009). However, MSBFNN has the drawback that it needs more training epochs and parameters than the proposed approach. This improvement on structure indicates that the proposed approach performs better in architectural economy than other models do.

4.3. Lorenz attractor system

To further test the effectiveness of the proposed approach, it is also applied in a multiple input/output system–Lorenz attractor system.

The Lorenz attractor is a fractal structure corresponding to the long-term behavior of the Lorenz oscillator. The Lorenz oscillator is a three-dimensional dynamical system that exhibits chaotic flow, noted for its butterfly shape. The map shows how the state of a dynamical system (the three variables of a three-dimensional system) evolves over time in a complex, non-repeating pattern. The differential equations that govern the Lorenz oscillator are:

$$\frac{dx(t)}{dt} = -ax + ay; \qquad \frac{dy(t)}{dt} = -xz + rx - y;$$
$$\frac{dz(t)}{dt} = xy - bz;$$

when setting $a = 10$, $r = 28$ and $b = 8/3$, the dynamic trajectory is illustrated in Fig. 7(a).

To facilitate comparisons with the support vector machine using evolution strategy (ES-SVM) (Hou & Li, 2009), the kernel Kalman filter (KKF) (Ralaivola & Alche-Buc, 2003), NN-based autoregressive model (NN-based AR) and Auto-Regressive Moving Average model (NN-based ARMA) (Dudul, 2005), the input–output pair is described as $(x(t-\Delta\tau), y(t-\Delta\tau), z(t-\Delta\tau); x(t), y(t); z(t))$, where $\Delta\tau = 7$. By respectively describing each input and corresponding output neuron as three linguistic terms 'Large', 'Medium' and 'Small', the identified membership functions are illustrated by Fig. 8. Furthermore, Fig. 7(b) shows the reasoning results of the Lorenz attractor system.

On the basis of RMSE, Table 10 summarizes and compares the structures and performances of our approach, SVM, ES-SVM, KKF, NN-based AR and NN-based ARMA. The analysis of the results leads to the first conclusion that, for the Lorenz attractor system, the proposed approach provides better reasoning results than most models listed in Table 10. Despite the fact that the RMSE of the proposed approach is larger than that of the NN-based AR model, our approach significantly reduces the training epoch and number of parameters with acceptable reasoning accuracy.

A second observation relates to the improvement of efficiency. Unlike other models in which the interrelationships among neuron units are described by simple weights, the proposed approach quantifies the interactions among neuron units using mutual subsethood. As Eq. (8) shows, these interactions are determined by relevant parameters (centers/widths) of the membership functions. By avoiding tuning the weights that interconnect different neurons, our approach significantly reduces the training epoch and number of tunable parameters. From this perspective, the improvement on structure indicates that the proposed approach performs better in architectural economy than other models.

5. Conclusion

In this paper, a novel four-layer fuzzy neural network is proposed to automatically construct FCMs. By using the mutual subsethood to define and quantify the causalities in FCMs, our approach distinctly upholds the basic definition of FCMs and makes the inference process easier to understand. Additionally, the proposed approach is able to automatically identify the membership

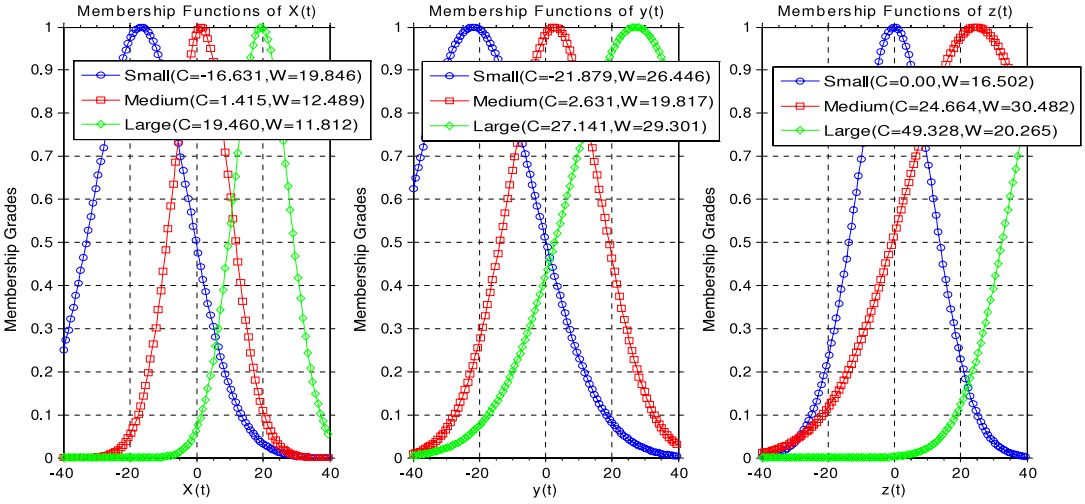

Fig. 8. Identified membership functions for input neurons.

functions and the causalities underlying the concerned systems. In this manner, our approach reduces the excessive dependence of traditional FCMs on expert knowledge.

The performance of the proposed approach is demonstrated by testing it on three benchmarking chaotic time series: Duffing forced-oscillation system, Mackey–Glass dataset and Lorenz attractor system. Simulation results show that the proposed approach performs better prediction accuracy and architectural economy in most cases.

References

Alpaydn, G., & Dundar, G. (2002). Evolution-based design of neural fuzzy networks using self-adaptive genetic parameters. *IEEE Transactions on Fuzzy Systems*, *10*, 211–221.

Boutails, Y., Kottas, L. T., & Christodoulou, M. (2009). Adaptive estimation of fuzzy cognitive maps with proven stability and parameter convergence. *IEEE Transactions on Fuzzy Systems*, *17*, 874–889.

Chun, F. H. (2007). Self-organizing adaptive fuzzy neural control for a class of nonlinear systems. *IEEE Transactions on Fuzzy Systems*, *18*, 1232–1241.

Dudul, S. V. (2005). Prediction of Lorenz chaotic attractor using two-layer perceptron neural network. *Applied Soft Computing*, *5*, 333–355.

Gaweda, A. E., & Zurada, J. M. (2003). Data-driven linguistic modeling using relational fuzzy rules. *IEEE Transactions on Fuzzy Systems*, *11*, 121–134.

Hou, S. M., & Li, Y. R. (2009). Short-term fault prediction based on support vector machines with parameter optimization by evolution strategy. *Expert Systems with Applications*, *36*, 12383–12391.

Juang, C., & Tsao, Y. (2008). A self-evolving interval type-2 fuzzy neural network with on-line structure and parameter learning. *IEEE Transactions on Fuzzy Systems*, *16*, 1411–1424.

Jun, Z., Henry, S. H. C., & Lo, W. L. (2008). Chaotic time series prediction using a neuro-fuzzy system with time-delay coordinates. *IEEE Transactions on Knowledge and Data Engineering*, *20*, 956–964.

Kasabov, N. K., & Song, Q. (2002). DENFIS: dynamic evolving neural-fuzzy inference system and its application for time-series prediction. *IEEE Transactions on Fuzzy Systems*, *10*, 144–154.

Kosko, B. (1986). Fuzzy cognitive maps. *International Journal of Man–Machine Studies*, *24*, 65–75.

Kosko, B. (1997). *Fuzzy engineering*. Englewood Cliffs, NJ: Prentice-Hall.

Kottas, F. L., & Boutalis, Y. S. (2004). A new method for reaching equilibrium points in fuzzy cognitive maps. In *Proc. 2nd international IEEE conference on intelligent sys. Vol. 1* (pp. 53–60).

Kottas, T. L., & Boutalis, Y. S. (2006). New maximum power point tracker for PV arrays using fuzzy controller in close cooperation with fuzzy cognitive networks. *IEEE Transactions on Energy Conversion*, *21*, 793–803.

Koulouriotis, D. E. (2002). Anamorphosis of fuzzy cognitive maps for operation in ambiguous and multi-stimulus real world environments. In *IEEE international conference on fuzzy systems. Vol. 3* (pp. 1156–1159).

Koulouriotis, D. E., & Diakoulakis, I. E. (2001). Learning fuzzy cognitive maps using evolution strategies: a novel schema for modeling and simulating high-level behavior. In *Proc. IEEE conference on evolutionary computation* (pp. 364–371).

Liu, Z. Q., & Satur, R. (1999). Contextual fuzzy cognitive map for decision support in geographic information systems. *IEEE Transactions on Fuzzy Systems*, *7*, 495–507.

Liu, Z. Q., & Zhang, J. Y. (2003). Interrogating the structure of fuzzy cognitive maps. *Soft Computing*, *7*, 148–153.

Mackey, M. C., & Glass, L. (1997). Oscillation and chaos in physiological control systems. *Science*, *197*, 287–289.

Miao, Y., & Liu, Z. Q. (2000). On causal inference in fuzzy cognitive maps. *IEEE Transactions on Fuzzy Systems*, *8*, 107–119.

Miao, Y., & Liu, Z. Q. (2001). Dynamic cognitive network—an extension of fuzzy cognitive map. *IEEE Transactions on Fuzzy Systems*, *9*, 760–770.

Papageorgiou, E. I., Stylios, C. D., & Groumpos, P. P. (2003). An integrated two-level hierarchical system for decision making in radiation therapy based on fuzzy cognitive maps. *IEEE Transactions on Biomedical Engineering*, *50*, 1326–1339.

Pelaez, C. E., & Bowles, J. B. (1995). Applying fuzzy cognitive maps knowledge-representation to failure modes effects analysis. In *Proc. IEEE annu. reliability maintainability symp.* (pp. 450–456).

Ralaivola, L., & Alche-Buc, F. D. (2003). Dynamical modeling with kernels for nonlinear time series prediction. In *Processing seventeenth annual conference on neural information processing systems*.

Shi, Z. W., & Han, M. (2007). Support vector echo-state machine for chaotic time-series prediction. *IEEE Transactions on Neural Networks*, *18*, 359–371.

Song, H. J., Miao, C. Y., & Shen, Z. Q. (2009). A fuzzy neural network with fuzzy impact grades. *Neurocomputing*, *72*, 3098–3122.

Song, H. J., Shen, Z. Q., & Miao, C. Y. M. (2007). Fuzzy cognitive map learning based on multi-objective particle swarm optimization. In *The 9th annual conference on genetic and evolutionary computation* (p. 69).

Stach, W. S., Kurgan, L. A., & Pedrycz, W. (2008). Numerical and linguistic prediction of time series with the use of fuzzy cognitive maps. *IEEE Transactions on Fuzzy Systems*, *16*, 61–72.

Stylios, C. D., & Groumpos, P. P. (1999). Fuzzy cognitive maps: a model for intelligent supervisory control systems. *Computers in Industry*, *39*, 229–238.

Stylios, C. D., & Groumpos, P. P. (2004). Modeling complex systems using fuzzy cognitive maps. *IEEE Transactions on Systems, Man, and Cybernetics—Part A*, *34*, 155–162.

Zhang, J. Y., Liu, Z. Q., & Zhou, S. (2003). Quotient FCMs—a decomposition theory for fuzzy cognitive maps. *IEEE Transactions on Fuzzy Systems*, *11*, 593–604.

Zhang, J. Y., Liu, Z. Q., & Zhou, S. (2006). Dynamic domination in fuzzy causal networks. *IEEE Transactions on Fuzzy Systems*, *14*, 42–56.

Zhou, S., Liu, Z. Q., & Zhang, J. Y. (2006). Fuzzy causal networks: general model, inference, and convergence. *IEEE Transactions on Fuzzy Systems*, *14*(Jun), 412–420.

H.J. Song received the Ph.D. degree from Nanyang Technological University (NTU), Singapore, in 2010. He is currently a research fellow in the Emerging Research Lab (ER Lab), Computer Engineering, NTU. From 2008 to 2010, he was researcher at IMEC, Belgium. His research interests include neural networks, fuzzy systems, fuzzy rule-based systems, fuzzy cognitive maps, architecture design methods and system-level exploration for power and memory footprint within real-time constraints.

C.Y. Miao is an Assistant Professor in the School of Computer Engineering, Nanyang Technological University (NTU). Her research areas include Agent, Multi-Agent Systems (MAS), Agent Oriented Software Engineering (AOSE), AgentWeb/Grid/Cloud and agents for wireless communications. Her current research works focus on infusing intelligent agents into interactive new media (virtual, mixed, mobile and pervasive media) to create novel experiences and dimensions in virtual world, game design, interactive narrative and other real world agent systems.

Z.Q. Shen is currently a Visiting Assistant Professor at Information Engineering Division, School of EEE, NTU. He enjoyed working for industry (Siemens, Singapore and MDSI, Canada), research institute (I2R/NUS) and university (NTU). His research interests include intelligent agents, computational intelligence, e-learning, mixed reality, semantic web/grid and interactive storytelling/games. He has extensive experience in transferring the research results into commercialized technologies. He successfully led a number of well know research projects in Singapore and Canada and transferred the research outcome into commercial applications.

W. Roel is a Principal Scientist at IMEC Belgium and a Part-time Professor in the Katholieke Universiteit Leuven Computer Science Department. His research interests include embedded software engineering, specifically componentized runtime resource managers and distributed resource managers for networked embedded devices, and programming language design. He has a Ph.D. in Computer Science from Vrije Universiteit Brussel.

D.H. Maja is a Senior Researcher at IMEC Belgium and a Part-time Professor in Vrije Universiteit Brussel, Computer Science Department. Her current research activities include architecture design methods and software engineering.

C. Francky received the Ph.D. degree from the Katholieke Universiteit Leuven, Belgium, in 1987. He is a fellow at the Interuniversity Microelectronics Center (IMEC), Heverlee, Belgium. His current research activities include architecture design methods and system-level exploration for power and memory footprint within real-time constraints, oriented toward data storage management, global data transfer optimization, and concurrency exploitation, also including deep submicron technology issues. He was elected an IEEE fellow in 2005. He is also a member of the IEEE Computer Society.