EURASIP Journal on Advances in Signal Processing

RESEARCH Open Access

T-S fuzzy systems optimization identification based on FCM and PSO

Yaxue Ren[1], Fucai Liu[1*], Jinfeng Lv[2], Aiwen Meng[1] and Yintang Wen[1]

Abstract

The division of fuzzy space is very important in the identification of premise parameters, and the Gaussian membership function is applied to the premise fuzzy set. However, the two parameters of Gaussian membership function, center and width, are not easy to be determined. In this paper, based on Fuzzy c-means (FCM) and particle swarm optimization (PSO) algorithm, a novel T-S fuzzy model optimal identification method of optimizing two parameters of Gaussian function is presented. Firstly, we use FCM algorithm to determine the Gaussian center for rough adjustment. Then, under the condition that the center of Gaussian function is fixed, the PSO algorithm is used to optimize another adjustable parameter, the width of the Gaussian membership function, to achieve fine-tuning, so as to complete the identification of prerequisite parameters of fuzzy model. In addition, the recursive least squares (RLS) algorithm is used to identify the conclusion parameters. Finally, the effectiveness of this method for T-S fuzzy model identification is verified by simulation examples, and the higher identification accuracy can be obtained by using the novel identification method described compared with other identification methods.

Keywords: T-S fuzzy modeling, System identification, Fuzzy c-means, Gaussian function, PSO algorithm

1 Introduction

In recent years, fuzzy model has been widely studied and has become an effective tool for complex system identification. The identification of fuzzy model consists of structure identification and parameter identification. The structure identification is divided into the identification of precursor structure and conclusion structure. Parameter identification is also divided into premise parameter identification and conclusion parameter identification. Takagi and Sugeno [1] has demonstrated that systems based on fuzzy rules can approximate highly nonlinear systems. T-S fuzzy model is widely used in nonlinear system modeling and model-based control [2–4]. There are many methods to realize premise parameter identification and fuzzy space partition, such as fuzzy c-means (FCM) [5–7], fuzzy c-regression model (FCRM) [8–12], Gath-Geva clustering algorithm [13], and Gustafson-Kessel clustering algorithm [14].

In order to improve the identification accuracy of the model and prepare for further control, this article starts with changing the method of dividing the fuzzy space to improve the identification accuracy of the fuzzy model. Various studies and related data show that FCM is very suitable and widely used to identify the premise parameters of T-S fuzzy model. The core problem of FCM clustering algorithm is to establish a reasonable clustering index to optimize the division of fuzzy input space. The FCM algorithm is mainly based on the Euclidean norm of data clustering center to form spherical clustering algorithm, which is obtained by optimizing the objective function of each sample points for all class center membership degree, and the category of the sample points in order to achieve the goal of automatic classifying sample data. Wang et al. [15] used the nearest neighbor clustering method to preset the initial parameters of fuzzy clustering to achieve the purpose of improving accuracy. The method proposed in the article is helpful to the accuracy

*Correspondence: lfc@ysu.edu.cn
[1]Engineering Research Center of the Ministry of Education for Intelligent Control System and Intelligent Equipment, Yanshan University, Hebei Street, Qinhuangdao, 066004, China
Full list of author information is available at the end of the article

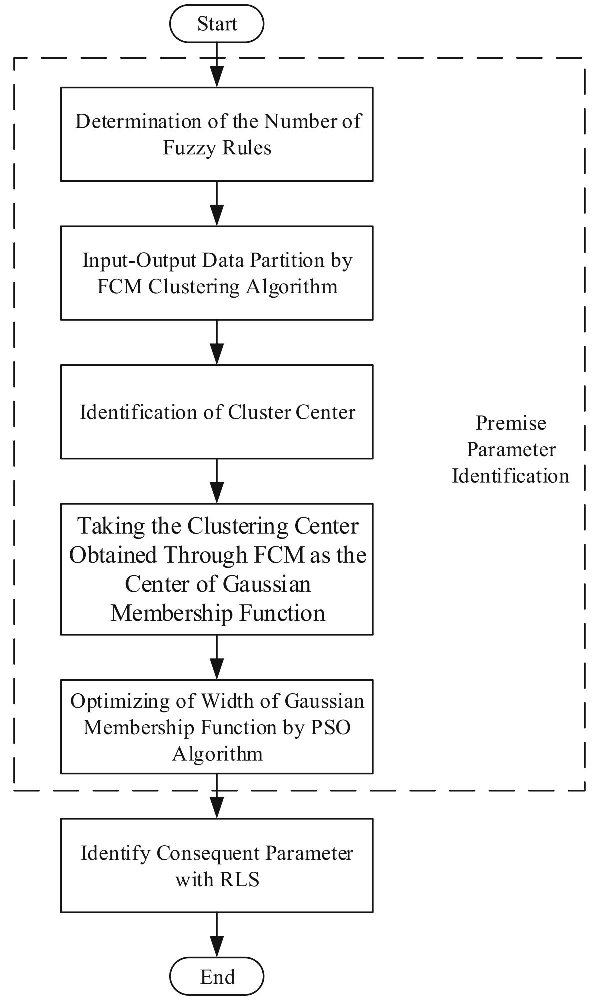
Start
Determination of the Number of Fuzzy Rules
Input-Output Data Partition by FCM Clustering Algorithm
Identification of Cluster Center
Taking the Clustering Center Obtained Through FCM as the Center of Gaussian Membership Function
Optimizing of Width of Gaussian Membership Function by PSO Algorithm
Premise Parameter Identification
Identify Consequent Parameter with RLS
End

Fig. 1 Flowchart of our fuzzy modeling algorithm

improvement, but because there was no method to optimize the fuzzy space division, the accuracy improvement is not very large. Liu et al. [16] combined the improved PSO algorithm with FCM to improve the accuracy of model identification, but the article did not propose an improved method for the membership function. The literature [17] integrated the adaptive parameters of distance measurement into the FCM algorithm, which can flexibly control the depth of the relevant information between different attributes in medical data and improve the accuracy of the algorithm. However, the calculation method of the membership function in the article was not very accurate. High-order neural fuzzy c-means clustering algorithm was used to classify massive heterogeneous data in [18]. FCM clustering algorithm was used to quickly find out the central functions of two classes of clustering in the image domain in [19]. Mahalanobis and Minkowski metrics are used to replace the usual Euclidean distance in order to enhance the cluster detection capacity of FCM by allowing more accurate detection of arbitrary shapes of clusters for high dimensional datasets in [20]. A clustering method of kernel fuzzy c-regression model based on fuzzy correlation was proposed in [21], which solved the problem of identifying the presupposition parameters of T-S fuzzy model.

The division of fuzzy space has a great impact on improving the accuracy of fuzzy model identification. There are many methods for dividing fuzzy space, such as triangle function and bell-shaped Gaussian function. As a matter of fact, the bell-shaped Gaussian fits FCM because distance is also measured in a point-to-point fashion. How to combine the Gaussian membership function with the traditional FCM algorithm to improve the accuracy of model identification will be an interesting problem.

In order to identify the premise parameters more accurately and obtain higher identification accuracy, it is necessary to optimize the membership function parameters. FCRM method was used to optimize the center and width of Gaussian function to obtain higher modeling accuracy in [9]. Chandrakumar and Senthil [22] introduced a new fuzzy c-means objective function called kernel induced fuzzy c-means based on Gaussian function for the purpose of segmentation of medical images. The probability in the algorithm that indicates the spatial influence of the neighboring pixels on the center pixel plays a key role in this algorithm, and it obtains efficient method for calculating membership and updating prototypes by minimizing the new objective function of Gaussian based fuzzy c-means. An intuitionistic fuzzy neural network (IFNN) with Gaussian membership function and Yager-generating function is proposed in [23], and the incorporation of the concept of IFL into a fuzzy neural network (FNN) can enhance the performance of an FNN. It can be seen that Gaussian function plays an important role in identification of modeling. This paper adopts a method to determine and adjust two key parameters of Gaussian function (center and width). The method adopted in this paper is to determine the center of Gaussian function by using the FCM algorithm, and to optimize its width by using PSO algorithm when the center has been determined and remains unchanged, so as to complete the fuzzy division of the premise parameters of the fuzzy model. In addition, the corresponding parameters are determined by RLS method. The novelty of our paper are expressed in the following aspects:

(1) For the first time, FCM clustering algorithm is combined with Gaussian function for fuzzy model identification.
(2) The creative introduction of PSO algorithm achieves fine-tuning, making the fuzzy model identification more accurate.

The rest part of this paper is organized as follows. Section 2 gives a brief and basic introduction to the T-S fuzzy model. In Section 3, we come up with a new fuzzy system identification method and describe the fuzzy modeling method. In Section 4, the validity of the proposed method is verified by three experiments, and its superiority is proved by comparing with other methods. Section 5 is the conclusion.

2 T-S fuzzy model

T-S model is a rule-based model in which the preconditions of rules are fuzzy variables and the conclusion is a linear function of input and output. It is based on local linearity and achieves global nonlinearity through fuzzy reasoning. T-S model is generally defined as:

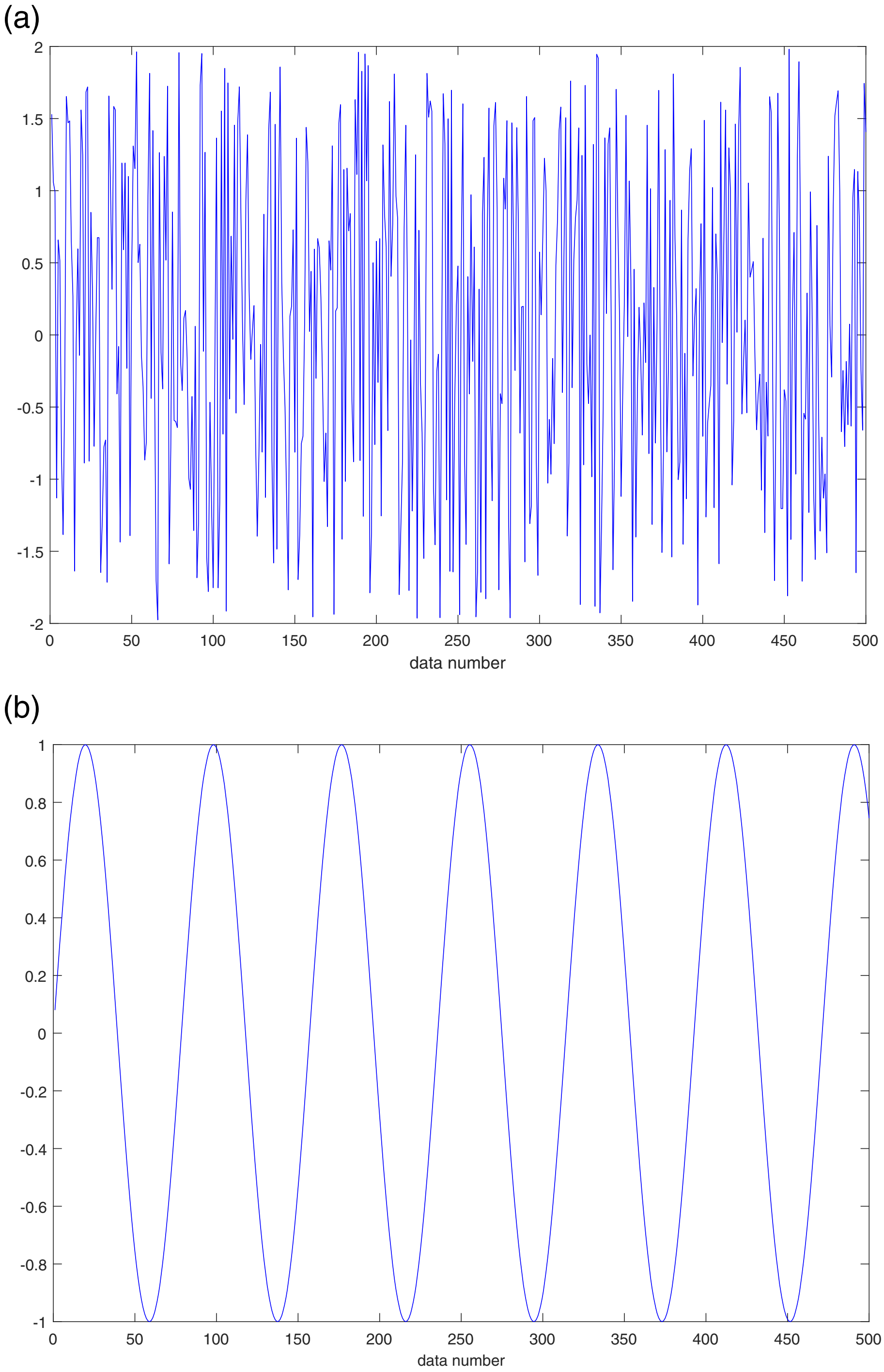

Fig. 2 Training and testing inputs for a nonlinear differential equation

Table 1 Center and width of Gaussian membership functions before and after optimization for the nonlinear difference equation example

	Before optimization			After optimization		
Center (v_{C*r})	−0.1430	0.7568	0.7568	−1.5403	−1.6843	−1.6839
	0.6456	2.2100	2.2100	−0.4553	−0.4047	−0.4027
	0.9755	2.9367	2.9367	0.6287	1.0216	1.0139
	1.1735	3.3726	3.3726	1.6045	2.5294	2.5254
Width (b_{C*r})	0.4000	0.4000	0.4000	2.4092	1.0312	1.0561
	0.4000	0.4000	0.4000	3.0067	3.4380	3.2153
	0.4000	0.4000	0.4000	4.6322	0.7850	0.6624
	0.4000	0.4000	0.4000	5.9032	1.8199	1.5576

$$\begin{aligned}&\mathbf{R}_i: \text{ If } (x_1 \text{is } A_{i1}) \text{ and} \ldots \text{ and } (x_n \text{ is} A_{in}) \\ &\text{then } \left(y_i = p_0^i + p_1^i x_1 + p_2^i x_2 \ldots + p_n^i x_n\right);\end{aligned}$$

where $\mathbf{R}_i$ is the ith fuzzy rule, $i = 1, 2, \ldots, c$; c is the number of fuzzy rule; A_{ij} is the ith fuzzy subset of variable x_j; $\mathbf{x}$ is the input variable, $\mathbf{x} = [1, x_1, x_2, \ldots, x_n]^T$; y_i is the output variable of ith fuzzy rule; and p_i^l is the consequent parameters, $l = 0, 1, \ldots, n$.

Each fuzzy rule has a matching degree, which represents the contribution of ith rule to the total T-S fuzzy model:

$$\begin{aligned}\tau_i &= \mu_{i1}(x_1) \times \mu_{i2}(x_2) \times \ldots \times \mu_{in}(x_n) \\ &= \bigcap_{j=1}^{n} \mu_{ij}(x_j)\end{aligned} \tag{1}$$

Some forms of membership functions (triangle, trapezoid, and bell) can be applied to the presupposition fuzzy set. The bell-shaped fuzzy set A_{ik} is used in this paper:

$$\mu_{ij}(x_j) = \exp\left\{-\left(x_j - c_{ij}\right)^2 / b_{ij}^2\right\} \tag{2}$$

c_{ij} and b_{ij} are the parameters of Gaussian membership function. The output of T-S model is a weighted average of individual rules:

$$y = \sum_{i=1}^{c} \omega_i y_i = \sum_{i=1}^{c} \omega_i \mathbf{x}^T \pi_i \tag{3}$$

where $\omega_i = \tau_i / \sum_{j=1}^{c} \tau_j$ is the validity function of ith rule, y_i is the output of ith submodel, and $\pi_i = \left[p_i^0, p_i^1, p_i^2, \ldots, p_i^n\right]^T$, $i = 1, 2, \ldots, c$ is the conclusion parameter of ith rule.

3 The proposed T-S fuzzy model identification approach

In present section, we will introduce a novel prerequisite parameter identification method of T-S fuzzy model in detail. Firstly, FCM algorithm is used to initialize input-output space, decompose input space into c fuzzy subspace, and determine the clustering center of fuzzy subspace. After that, the center of the fuzzy subspace which is gotten in the first step is substituted into the Gaussian membership function. In the third step, the PSO algorithm is utilized to optimize the width of the Gaussian function and determine the membership function while keeping the center of the Gaussian function unchanged. The center and width of the Gaussian function are not easy to be determine. Finally, RLS method is used to identify the conclusion parameters. Then, the identification model is obtained and the specific flow diagram of this method is shown in Fig. 1.

The key problem of the new modeling method proposed in this paper lies in the application of Gaussian membership function and how to quickly optimize its two parameters, center and width. These are discussed in detail in this section.

3.1 A novel premise parameter identification method is based on FCM and PSO

In this part, we will elaborate the method of using traditional FCM clustering algorithm and PSO algorithm

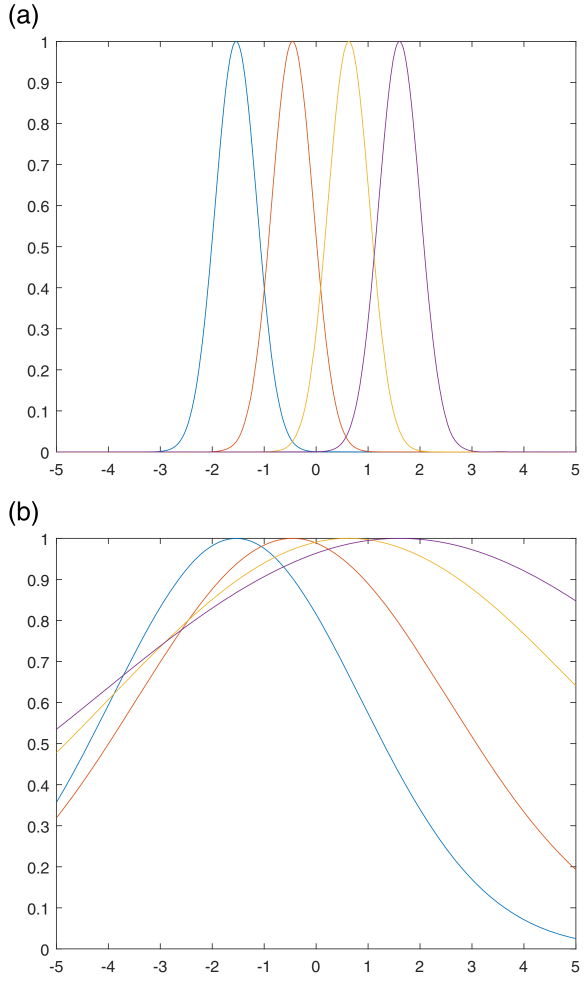

Fig. 3 Gaussian function center and width optimization in regard to variable $u(k)$ for the nonlinear difference equation example

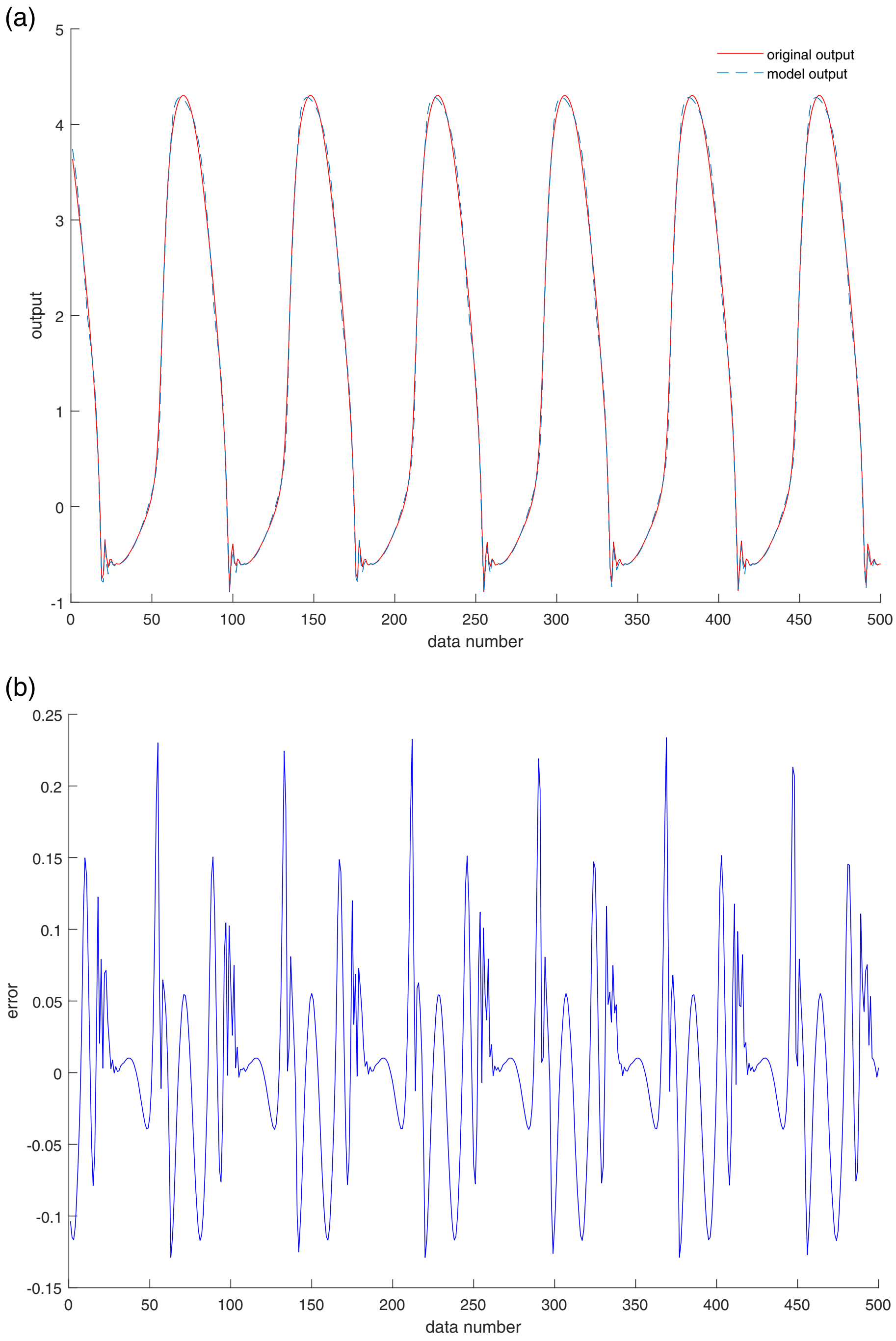

Fig. 4 The nonlinear differential equation example fuzzy model performance

to determine the parameters of Gaussian function. FCM algorithm is used to obtain rough tuning results, and then, PSO algorithm is used to achieve fine-tuning. The used methods of minute forming fuzzy set are all conventional algorithms, which are characterized by simple structure and helpful to make the identification of premise parameters more concise and effective.

3.1.1 Determination of center of Gaussian membership function by FCM

The FCM algorithm [10] can be expressed as minimizing the following objective function:

$$J_m(U,v)=\sum_{j=1}^{n}\sum_{i=1}^{c}\left(\mu_{ij}\right)^m\left(d_{ij}\right)^2 \tag{4}$$

Table 2 Comparison of model evaluation indexes of different models for the nonlinear differential equation example

Model	No. of rules	*MSE*	
		Training	Test
Farag et al. [26]	75	0.0374	0.0403
Wang and Lee [27]	8	0.6184	0.2037
Evsukoff et al. [28]	100	0.1577	0.0185
Bagis [29]	4	0.0341	0.0378
Li et al. [12]	4	0.0149	0.0115
Our model	4	0.0069	0.0046

satisfying

$$\sum_{i=1}^{c} \mu_{ij} = 1, 1 \le j \le n, \mu_{ij} \ge 0, 1 \le i \le c \qquad (5)$$

where n is the input variable dimension and c is the cluster center number. $m > 1$ is weight index of membership function. If m is too small, the membership of the input variable is around 1, which will affect the identification accuracy; if m is too large, the number of crossover among membership functions is too much, which will also affect the identification accuracy. In practice, $m = 2$ is often taken. U is a fuzzy partition matrix containing the membership of each feature vector for each cluster. z is the center of clustering, $z = \{z_1, z_2, \ldots, z_c\}$, $z_i \in R^n$. The clustering center can be calculated according to formula (6):

$$z_i = \sum_{j=1}^{n} \left(\mu_{ij}\right)^m x_j / \sum_{j=1}^{n} \left(\mu_{ij}\right)^m, \forall i \qquad (6)$$

The fuzzy membership function matrix U can be obtained by the following formulas:

$$\mu_{ij} = 1 / \sum_{k=1}^{c} \left(\frac{d_{ij}}{d_{kj}}\right)^{2/(m-1)} \qquad (7)$$

$$d_{ij} = \|x_j - z_i\| > 0, \forall i, j \qquad (8)$$

if $d_{ij} = 0$, then $\mu_{ij} = 1$, $\mu_{kj} = 0$, for all $k \neq i$

The initial value of the FCM center matrix z is given at random; after that, the fuzzy partition matrix U is calculated by using formula (7) for all the eigenvectors. The initialization of z is obtained by randomly selecting the eigenvalues of each cluster center$\left(z_{ij}\right)$, which should be within the set of the listed eigendata. The stop condition is achieved by setting ε. Set it according to users' needs.

Offline calculation method is as follows:

(1) Random number generator is used to give the initial value to the clustering center matrix z, and the clustering center was recorded, and set $k = 0$;
(2) The initial value of the fuzzy partition matrix $U^{(k=0)}$ is calculated by using Eqs. (7) and (8);
(3) Increase k so that $k = k + 1$, and use Eq. (6) to update cluster center z;
(4) Equations (7) and (8) are used to renew the fuzzy partition matrix $U^{(k)}$;
(5) If $\|U^{(k)} - U^{(k-1)}\| < \varepsilon$ is satisfied, the calculation stops; otherwise, repeat steps 3∼5.

The center of Gaussian function (the clustering center) can be obtained from the above steps.

3.1.2 *Optimization of the width of Gaussian membership function by PSO*

In 1995, Kennedy et al. proposed PSO algorithm [24], which has the advantages of evolutionary computation and swarm intelligence, and it is a heuristic global optimization algorithm. In this paper, the purpose of using PSO is to optimize the width of Gaussian function and realize the fine-tuning of fuzzy division of premise parameters to get higher modeling accuracy. In addition, when optimizing the width parameter, the minimum mean square error (MSE) (formula (18)) is used as the objective function of PSO algorithm for global search to find the best particle location.

The PSO algorithm is briefly described as follows: let particles search in D-dimensional space, and the number of particles is N. Where the position of kth particle is $B_k = (b_{k1}, b_{k2}, \ldots, b_{kD})$, the velocity of the particle is $V_k = (v_{k1}, v_{k2}, \ldots, v_{kD})$, each particle is a solution to

Table 3 Center of Gaussian membership functions before and after optimization for the Box and Jenkins example (case 1)

	Center (v_{c*r})					
Before optimization	0.0590	0.0590	0.0590	53.0500	53.0500	53.0500
	0.9840	0.9840	0.9840	55.5333	55.5333	55.5333
	1.4465	1.4465	1.4465	56.7750	56.7750	56.7750
	1.7240	1.7240	1.7240	57.5200	57.5200	57.5200
After optimization	−1.5724	−1.57394	−1.5724	48.9316	48.9559	48.9836
	−0.3519	−0.3570	−0.3575	52.2294	52.2705	52.3161
	0.6625	0.6535	0.6506	55.2552	55.2838	55.3178
	1.6878	1.6819	1.6806	58.1957	58.2380	58.2777

Table 4 Width of Gaussian membership functions before and after optimization for the Box and Jenkins example (case 1)

	Width (b_{c*r})					
Before optimization	0.4000	0.4000	0.4000	0.4000	0.4000	0.4000
	0.4000	0.4000	0.4000	0.4000	0.4000	0.4000
	0.4000	0.4000	0.4000	0.4000	0.4000	0.4000
	0.4000	0.4000	0.4000	0.4000	0.4000	0.4000
After optimization	0.5915	0.2889	1.4532	2.4482	0.7629	4.0486
	0.4509	1.0336	0.5369	1.1616	3.2833	1.1024
	0.7113	3.0976	4.0372	0.3325	0.3561	0.4420
	0.4605	0.5408	1.7274	0.9292	1.7803	1.9757

the optimization problem, and the particle finds a new solution by constantly changing its position and speed. The optimal solution of the kth particle searched so far is $P_k = (p_{k1}, p_{k2}, \ldots, p_{kD})$, and the optimal position experienced by the whole group is $P_g = (p_{g1}, p_{g2}, \ldots, p_{gD})$. The velocity and position of each particle vary in line with Eqs. (9) and (10):

$$\begin{aligned} v_{kd}(t+1) =& \omega v_{kd}(t) + c_1 r_1 \left(p_{kd}(t) - b_{kd}(t)\right) \\ &+ c_2 r_2 \left(p_{gd}(t) - b_{kd}(t)\right) \end{aligned} \tag{9}$$

$$b_{kd}(t+1) = b_{kd}(t) + v_{kd}(t+1) \tag{10}$$

where r_1 and r_2 are the random numbers between [0,1]; c_1 and c_2 are the normal numbers, which are called accelerators; and w is the inertia weight. The range of velocity and position variation in d-dimension of each particle is $\left[-v_{d,\max}, v_{d,\max}\right]$ and $\left[-x_{d,\max}, x_{d,\max}\right]$. If the maximum velocity of the particle, $v_{d,\max}$, is too high, it might cause the particle to fly through the best solution; if the maximum velocity is too small to make the search speed too slow, it may lead to fall into local optimal solution. Inertia weight w can well control the search range of particles. When w is large, particles are searched in a wide range. When w is small, particles are excavated in a small range. When PSO algorithm is used to optimize the width of Gaussian function, the learning factors c_1, c_2 are both set as 2 and the inertia weight ω is updated by the following formula:

$$\omega = \omega_{\min} + DT \cdot \frac{\omega_{\max} - \omega_{min}}{\max DT} \tag{11}$$

where DT is the number of iterations. Let $\max DT = 100$ be the maximum number of iterations, and $\omega_{\min} = 0.4$, $\omega_{\max} = 0.9$.

According to the above methods, the optimal widths of Gaussian membership function are obtained. The new premise parameter identification method can be specifically described as:

1) Determine the number of input variables r, and make a fuzzy division of each input space (determine c). Initialize the center and width of the Gaussian.
2) FCM algorithm is used to optimize the centers of Gaussian function and determine the centers of Gaussian function. The center of Gaussian function is determined by FCM algorithm. Firstly, the FCM algorithm is used to automatically obtain the initial

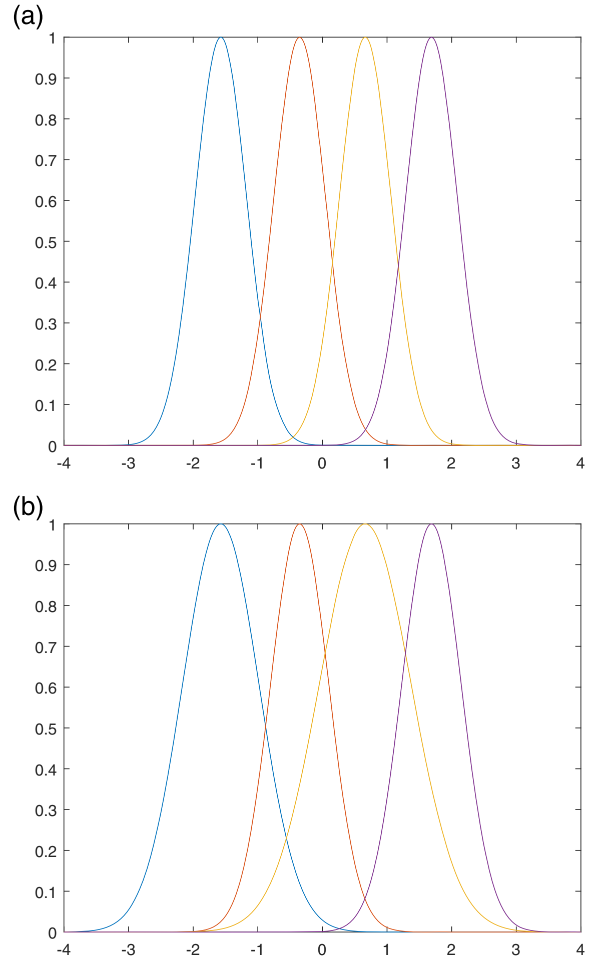

Fig. 5 Gaussian function center and width optimization in regard to variable $u(k)$ for the Box and Jenkins example (case 1)

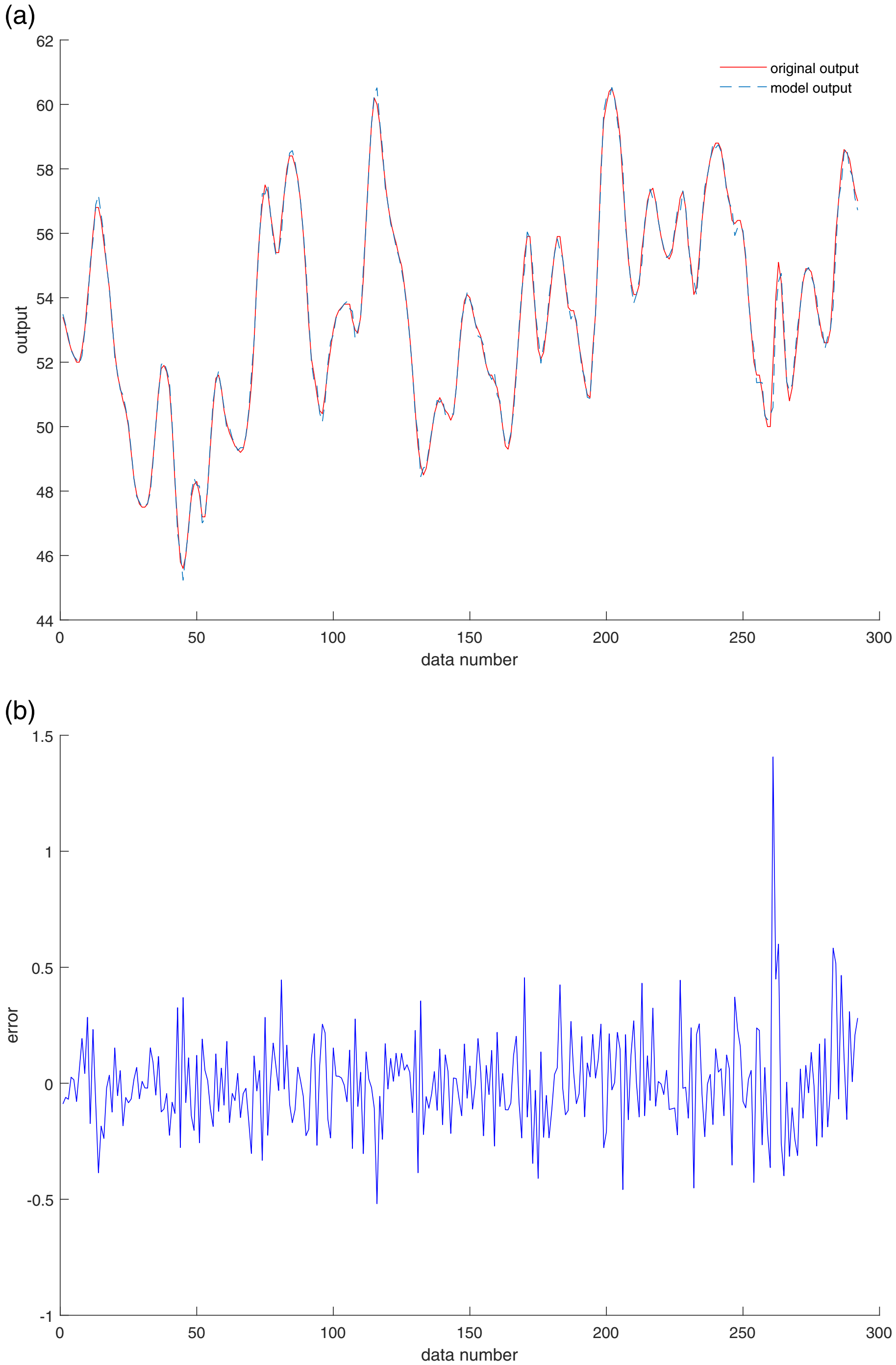

Fig. 6 Box and Jenkins example (case 1) fuzzy model performance

cluster centers of the dataset. Then, it is optimized step by step. Finally, the determined clustering centers are treated as the centers of Gaussian function. This algorithm is not sensitive to the initial value. On the basis of the above results, the width of Gaussian membership function is determined by PSO algorithm. First, the initial value is 0.4 according to the experience, and then, it is optimized gradually to determine the width of Gaussian function.

3) Under the condition that the center is determined and unchanged, PSO intelligent optimization algorithm is used to optimize the width of Gaussian

Table 5 Comparison of model evaluation indexes of different models for the Box and Jenkins example (case 1)

Model	No. of inputs	No. of rules	*MSE*
Box and Jenkins [30]	6	6	0.202
Xu and Lu [31]	2	25	0.328
Kung and Su [11]	6	2	0.0518
Kim et al. [9]	6	2	0.055
Kim et al. [10]	6	2	0.048
Li et al. [12]	6	4	0.0498
Liu et al. [16]	2	6	0.1238
Our model	6	4	0.0431

function, and a relatively ideal membership function is finally obtained.

3.2 Consequent parameter identification

The identification of the premise parameters is determined, followed by the identification of the consequent parameters.

The output of the system can be expressed as:

$$y = \sum_{i=1}^{c} \omega_i y_i / \sum_{i=1}^{c} \omega_i \tag{12}$$

$$\omega_i = \prod_{k \in I} \mu_{A_{kj}}(x_k) \qquad I = \{1, 2, \ldots, n\}, i = 1, 2, \ldots, c \tag{13}$$

where x_k is the kth input variable of the fuzzy model; $\mu_{A_{jk}}$ is the membership of the jth fuzzy subset of variable x_k, which is obtained by the previous fuzzy partition; y_i is the output of rule i; and $\prod$ is a fuzzy operator, usually using small operation.

Define

$$\overline{\omega_i} = \omega_i / \sum_{m=1}^{c} \omega_m \tag{14}$$

so the output of the fuzzy system is:

$$\begin{aligned} y &= \sum_{i=1}^{c} \overline{\omega_i} y_i \\ &= \sum_{i=1}^{c} \overline{\omega_i} \left(p_0^i + p_1^i x_1 + p_2^i x_2 + \ldots + p_n^i x_n\right) \\ &= \left[\, \overline{\omega_1}\ \overline{\omega_1} x_1\ \ldots\ \overline{\omega_1} x_n\ \overline{\omega_c}\ \overline{\omega_c} x_1\ \ldots\ \overline{\omega_c} x_n \,\right] \\ &\quad \times \left[\, p_0^1\ p_1^1\ \ldots\ p_n^1\ \ldots\ p_0^c\ p_1^c\ \ldots\ p_n^c \,\right]^T \end{aligned} \tag{15}$$

substitute N pairs of input and output data into (14) to get a matrix equation.

$$Y = XP \tag{16}$$

where P is the $L = (r + 1)c$-dimensional consequent parameter vector and Y and X are the matrices of $N \times 1$ and $N \times L$. r is the number of input variables, and c is the fuzzy rule number. $P^* = \left(X^T X\right)^{-1} X^T Y$ is the least square estimation of P. In order to iteratively optimize the consequent parameter matrix P and avoid matrix inverse, the recursive least squares algorithm is adopted here. If the ith row vector of X is x_i and the ith component of Y is y_i, then the recursive algorithm is:

$$P_{i+1} = P_i + \frac{S_{i+1} \cdot X_{i+1}^T \cdot \left(y_{i+1} - X_{i+1}^T \cdot P_i\right)}{1 + X_{i+1} \cdot S_i \cdot X_{i+1}^T} \tag{17}$$

$$S_{i+1} = S_i - \frac{S_{i+1} \cdot X_{i+1}^T \cdot X_{i+1} \cdot S_i)}{1 + X_{i+1} \cdot S_i \cdot X_{i+1}^T} \qquad i = 0, 1, \ldots, N - 1 \tag{18}$$

Initial condition is $P_0 = 0, S_0 = \alpha I$. α is always going to be more than 10,000. I is the identity matrix of $L \times L$. Formula (16) is used to calculate the optimal conclusion parameters in the sense of error square, and output the conclusion parameters and the minimum mean square error MSE after the recursive termination.

$$MSE = \sum_{i=1}^{N} \left(y_i - \widehat{y_i}\right)^2 / N \tag{19}$$

The complete fuzzy identification algorithm proposed in this paper is as follows:

Table 6 Center of Gaussian membership functions before and after optimization for the Box and Jenkins example (case 2)

	Center (v_{c*r})					
Before optimization	0.0590	0.0590	0.0590	52.9000	52.9000	52.9000
	0.9840	0.9840	0.9840	55.5333	55.5333	55.5333
	1.4465	1.4465	1.4465	56.5500	56.5500	56.5500
After optimization	−1.3863	−1.3802	−1.3900	48.7186	48.7561	48.7456
	0.1611	0.1847	0.1854	52.4433	52.5454	52.5965
	1.4585	1.4662	1.4627	56.9984	57.0210	57.0316

Table 7 Width of Gaussian membership functions before and after optimization for the Box and Jenkins example (case 2)

	Width (b_{c*r})					
Before optimization	0.4000	0.4000	0.4000	0.4000	0.4000	0.4000
	0.4000	0.4000	0.4000	0.4000	0.4000	0.4000
	0.4000	0.4000	0.4000	0.4000	0.4000	0.4000
After optimization	3.6963	2.9739	2.8264	2.1233	0.5232	0.3797
	3.7974	3.9682	0.9303	2.1257	0.5094	0.1820
	1.1199	3.3114	0.0278	3.7457	1.8712	0.9683

(1) Determine the number of input variables *r*, and conduct fuzzy division of each input space (determine *c*);
(2) Calculate the premise parameters $\mu_{A_{ij}}(x_j)$ according to Eq. (2) of this paper;
(3) Get X from Eq. (14);
(4) P is obtained by using Eqs. (16) and (17);
(5) Calculate the performance indicator *MSE*. If the value is less than the threshold or two adjacent times are unchanged, then go to step 6. Otherwise, go to step 4;
(6) If *MSE* satisfies the required recognition accuracy, the identification is terminated; if not, add *c* and go to step 2.

4 Simulation experiment and application

In this paper, two well-known simulation examples and a practical application system are cited to confirm that the performance of the proposed identification method is superior to some previous methods, which mainly includes the prediction performance and generalization of the model. In the simulation example, through comparison with other methods in the literature, such as the traditional FCM algorithm [9], FCRM algorithm [10], and some other improved FCM algorithms [12], the prediction performance of the model in simulation examples is verified. In order to verify the generalization of the model, the data sample set is divided into two parts: training and testing. The training data is used to build the fuzzy model, and the testing data is used to check the generalization of the model.

4.1 A nonlinear difference equation

In this section, the nonlinear difference equation proposed by Narendra and Parthasarathy [25] is taken as the simulation object, whose expression is formula (20):

$$y(k) = \frac{y(k-1)y(k-2)\left(y(k-1)+2.5\right)}{1+y^2(k-1)+y^2(k-2)} + u(k) \tag{20}$$

This experiment uses cross-validation to test the predictive performance of the proposed method. The random number between $[-2, 2]$ is taken as the input signal $u(k)$ of the training data and is substituted into formula (20) to obtain 500 training data. Then, we change the input signal to $u(k) = \sin(2k/25)$ and plug it into the formula to get 500 sets of test data. The training data and test data are demonstrated in Fig. 2.

In this model, $u(k), y(k-1), y(k-2)$ are selected as input data and $y(k)$ as output data for modeling. The number of fuzzy rules is set as 4. After the first phase of modeling is completed, the test data-driven model is used to test the

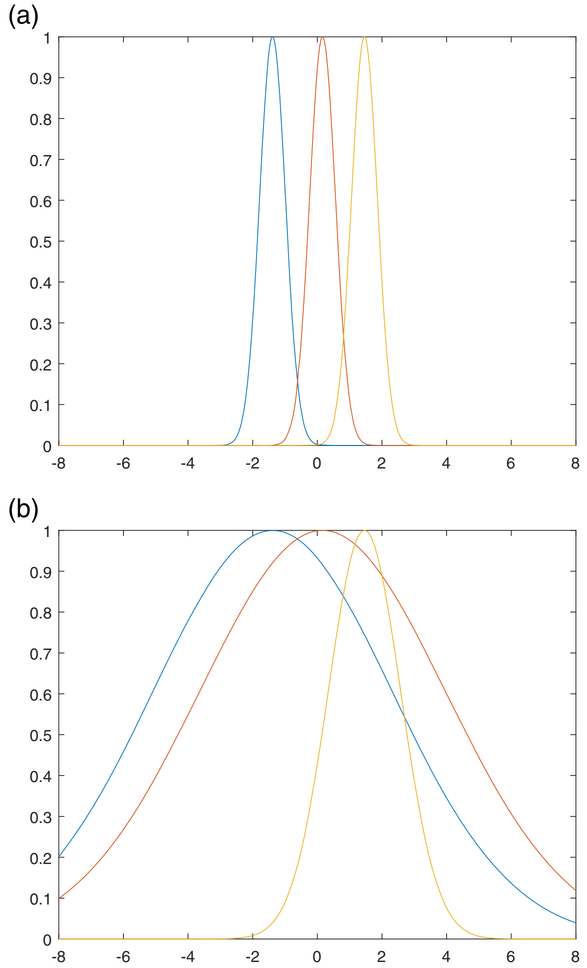

Fig. 7 Gaussian function center and width optimization in regard to variable $u(k)$ for the Box and Jenkins example (case 2)

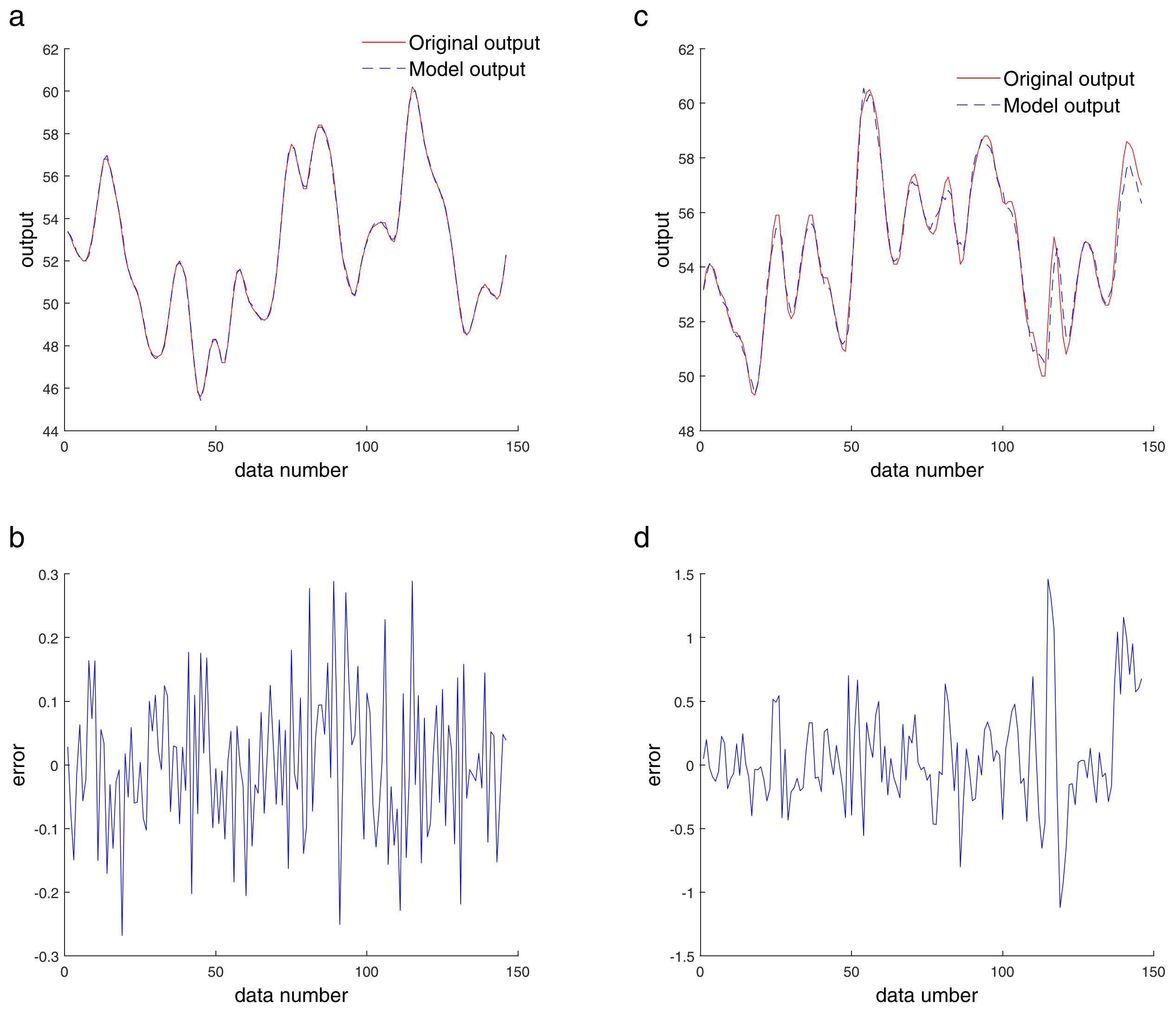

Fig. 8 Box and Jenkins example (case 2) fuzzy model performance

predictive performance. Table 1 indicates the results of the center and width of the Gaussian function before and after optimization, and Fig. 3 shows the change of membership function from initial optimization center to further optimization width under the 4 rules represented by variable $u(k)$ in this case. The comparison diagram of model output and error obtained after the simulation experiment is shown in Fig. 4. At the same time, the mean square error of modeling and testing of the fuzzy model and the real model is also obtained, and the detailed values and comparison are shown in Table 2.

4.2 Box-Jenkins system

The famous gas furnace data such as the Box-Jenkins dataset (Box and Jenkins [30]) have been used by many scholars as standard experimental data to test identification methods. The input $u(k)$ is the flow to the gas stove, and the output $y(k)$ is the concentration of carbon dioxide at the outlet. The Box-Jenkins system is a SISO dynamic system, which has 296 pairs of input-output measurements. Here, $u(k)$, $u(k-1)$, $u(k-2)$, $y(k-1)$, $y(k-2)$, $y(k-3)$ are chosen as input variables and $y(k)$ is chosen as output, which are conducted simulation experiment.

In order to verify the effectiveness of the algorithm, this experiment is set as two cases. In case 1, all 296 sets of data are used to build the model; in case 2, the data is divided into two groups, one of which is used as training data to establish a fuzzy model, and the other set of data is used as test data to test the prediction performance. When all the data is used for modeling, the fuzzy rule number c is set as 4, and the fuzzy rule number c is set as 3 in case 2.

Table 8 Comparison of model evaluation indexes of different models for the Box and Jenkins example (case 2)

Model	No. of inputs	No. of rules	*MSE*	
			Training	Test
Kim et al. [10]	6	2	0.034	0.244
Tsekouras [32]	6	7	0.022	0.236
Li et al. [12]	6	3	0.0159	0.1255
Our model	6	3	0.0124	0.1699

Tables 3 and 4 respectively exhibit the centers and widths before and after the membership function optimization of this experiment case 1, and the change of membership function from optimization center to further optimization width under the 4 rules represented by variable $u(k)$ is shown in Fig. 5. Figure 6 shows the performance of the fuzzy model identified in case 1, where Fig. 6a visually exhibits the original output and model output, and Fig. 6b demonstrates the error between model output and predicted output of each data point. The model performance evaluation index MSE of case 1 is 0.0428, and the comparison results with other methods are exhibited in Table 5. It can be seen from the performance comparisons shown in Table 3 that the method we proposed has great advantages in modeling. The fuzzy rules of the fuzzy system obtained in this case are shown as follows:

$\mathbf{R}_1$: If $u(k)$ is A_{11} and $u(k-1)$ is A_{12} and $u(k-2)$ is A_{13} and $y(k-1)$ is A_{14} and $y(k-2)$ is A_{15} and $y(k-3)$ is A_{16}

Then $y_1 = 9.8901 + 6.8226u(k) + 7.7769u(k-1) + 5.0171u(k-2) + 1.8158y(k-1) - 0.1480y(k-2) - 0.3980y(k-3)$;

$\mathbf{R}_2$: If $u(k)$ is A_{21} and $u(k-1)$ is A_{22} and $u(k-2)$ is A_{23} and $y(k-1)$ is A_{24} and $y(k-2)$ is A_{25} and $y(k-3)$ is A_{26}

Then $y_2 = 0.1320 - 4.0212u(k) + 0.5494u(k-1) + 1.4731u(k-2) - 0.3340y(k-1) + 1.0539y(k-2) - 0.8154y(k-3)$;

$\mathbf{R}_3$: If $u(k)$ is A_{31} and $u(k-1)$ is A_{32} and $u(k-2)$ is A_{33} and $y(k-1)$ is A_{34} and $y(k-2)$ is A_{35} and $y(k-3)$ is A_{36}

Then $y_3 = -1.7639 + 0.0195u(k) - 0.0027u(k-1) + 1.7228u(k-2) + 0.8379y(k-1) + 1.8002y(k-2) + 1.5261y(k-3)$;

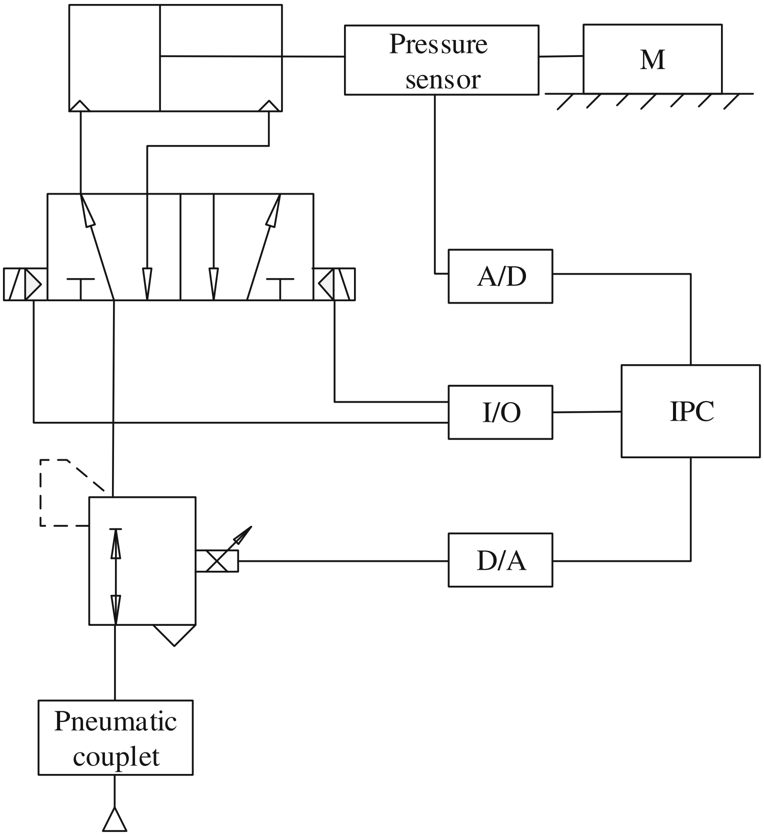

Fig. 9 Structural diagram of the variable load pneumatic loading system

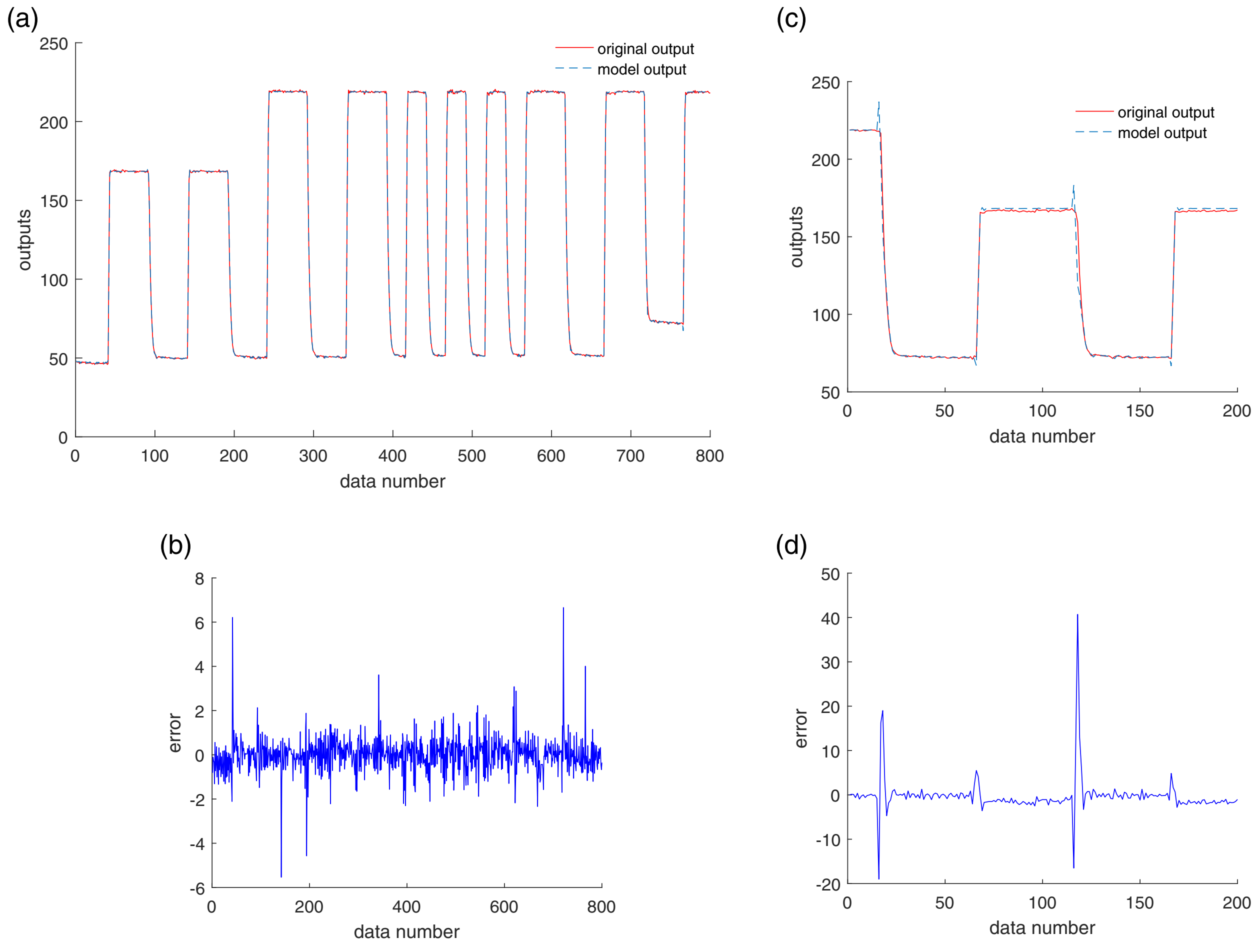

Fig. 10 Square wave loading test results for the variable load pneumatic loading system

$\mathbf{R}_4$: If $u(k)$ is A_{41} and $u(k-1)$ is A_{42} and $u(k-2)$ is A_{43} and $y(k-1)$ is A_{44} and $y(k-2)$ is A_{45} and $y(k-3)$ is A_{46}

Then $y_4 = -1.1831 + 0.3165u(k) - 0.9875u(k-1) - 0.7267u(k-2) + 0.3323y(k-1) - 0.3059y(k-2) + 0.0961y(k-3)$;

Tables 6 and 7 respectively demonstrate the center and width before and after the membership function optimization of this experiment case 2. The change of membership function is shown in Fig. 7. Figure 8 exhibits the fuzzy model performance of case 2, where a and b are the fuzzy modeling output of the training data and the modeling error of each data point, and c and d are the fuzzy model performance reflected in the prediction data. The modeling evaluation index of case 2 is 0.0123, and the prediction evaluation index is 0.168. The detailed comparison is shown in Table 8. In this case, although the prediction accuracy of the model is improved, it is not obvious. However, to some extent, it also proves the effectiveness of the algorithm in prediction.

4.3 The variable load pneumatic loading system

The variable load pneumatic loading system has the advantages of low cost, high output/mass ratio, no pollution, convenient maintenance, and so on, which is widely used in the field of industrial automation [33, 34]. Because of the complexity of gas flow, the compressibility of gas, the nonlinearity of valve, the friction characteristics of cylinder, and the vulnerability of system parameters to environment, the modeling and control of pneumatic loading system have become a very challenging work.

Generally speaking, there are two ways to establish the system model: one is that the operation law of the system is completely known and the model is built according to the physical law; the another one is to identify the system model from the operation and experimental data of the system. In this paper, data-driven fuzzy modeling method is used to build the model of the variable load pneumatic loading system.

Figure 9 is the structure diagram of the pneumatic loading system for test. The system includes stabilized pressure air source, pneumatic couplet, SMC ITV2050 pilot electric proportional pressure valve, SMC CDQ2A50 single rod double acting cylinder with cylinder diameter of 40 mm and stroke of 50 mm, and other pneumatic components. The measurement and control system includes MCL-L pull pressure sensor for real-time pressure measurement, Advantech PCI1710 data acquisition card for analog input, and Advantech PCI17 20 for control output. The system controller is IPC-610H industrial computer.

In this paper, in the dynamic range of the system, the pseudo-random sequence is used as the excitation signal, which continuously acts on the system in the open-loop state and collects the input and output data of the system. The sampling period is 0.1 s, the sampling time is 100 s, and 1000 sample points $\left[u(k), y(k)\right]$ are obtained, of which the first 800 data were used as training data and the rest were predicted data. The following variables $u(k)$, $u(k-1)$, $u(k-2)$, $y(k-1)$, $y(k-2)$, and $y(k-3)$ are selected as the candidate input variables of the model, and $y(k)$ as the output variable. The number of fuzzy rules is set as 3.

Figure 10 shows the offline modeling process curve of the variable load pneumatic loading system based on the method proposed in this paper, where Fig. 10a shows the outputs of the fuzzy model comparing with that of the real system, and Fig. 10b shows the errors between the system outputs and the model outputs. Figure 10c and d show the output and error of the predicted data. If 6 variables above are all selected as inputs, the training MSE of the our model is 0.6982 and the testing MSE is 18.1004. Table 9 shows the comparison between the traditional identification method (Gaussian function based on bisection method and optimizing the center of Gaussian function using FCM) and the method presented in this paper.

The experimental results show that the algorithm proposed in this paper can effectively reduce the influence of time delay on the system, more effectively control the variable load pneumatic loading system, and achieve the rapid response and accurate tracking of the system, and has good adaptive ability.

5 Results and discussion

In order to improve the accuracy and efficiency of model recognition, a novel method of prerequisite structure recognition is proposed in this paper. On the condition of not using complex structure and algorithm, FCM algorithm, which is commonly used in fuzzy space partition, is selected to complete the coarse-tuning of the algorithm. In order to further complete fine-tuning, we choose PSO optimization algorithm. After two steps of adjustment, the Gaussian fuzzy set can be obtained, and the identification of the premise parameters can be completed. At the same time, the RLS method is used to identify the conclusion parameters and complete the identification of the fuzzy model.

In this paper, the robustness and predictive performance of the proposed algorithm are verified by two international standard examples and an actual system application. In order to highlight the advantages of this algorithm, the modeling accuracy is compared with other methods in literatures, which fully verifies that this method has obvious advantages in improving the modeling accuracy. With the continuous development and maturity of intelligent optimization algorithms, more and more excellent optimization algorithms have emerged, such as hybrid frog leaping algorithm, firefly algorithm, and cockroach algorithm. For a specific fuzzy identification problem, it is a more practical research direction to the appropriate intelligent optimization algorithm which is used for parameter identification, to explore a fuzzy identification method with faster convergence speed and higher accuracy, and to better and more successfully apply it to the actual fuzzy system identification.

Table 9 Comparison of model evaluation indexes of different models for the variable load pneumatic loading system

Model	No. of inputs	No. of rules	*MSE*	
			Training	Test
Gaussian	6	3	30.9576	21.8597
Gauss+FCM	6	3	14.4222	32.2377
Our model	6	3	0.6982	18.1004

Abbreviations

FCM: Fuzzy c-means; PSO: Particle swarm optimization; RLS: Recursive least squares; FCRM: Fuzzy c-regression model

Acknowledgements

Not applicable.

Authors' contributions

LFC proposed the research idea of the paper and collected the experimental data. RYX conducted the data collation and simulation experiments on research ideas, and was a major contributor in writing the manuscript. LJF, MAW, and WYT further examined the manuscript and corrected it. All authors read and approved the final manuscript.

Funding

This work was supported in part by the Natural Science Foundation of Hebei Province under Project Number F2019203505.

Availability of data and materials

The datasets used and/or analyzed during the current study are available from the corresponding author on reasonable request.

Competing interests

The authors declare that they have no competing interests.

Author details

[1]Engineering Research Center of the Ministry of Education for Intelligent Control System and Intelligent Equipment, Yanshan University, Hebei Street, Qinhuangdao, 066004, China. [2]Hebei Normal University of Science and Technology, Hebei Street, Qinhuangdao, 066004, China.

Received: 26 June 2020 Accepted: 2 November 2020
Published online: 24 November 2020

References

1. T. Takagi, M. Sugeno, Fuzzy identification of systems and its applications to modeling and control. IEEE Trans. Syst. Man Cybern. Syst. **SMC-15**(1), 116–132 (1985)
2. C. Li, J. Zhou, L. Chang, Z. Huang, Y. Zhang, T-s fuzzy model identification based on a novel hyperplane-shaped membership function. IEEE Trans. Fuzzy Syst. **25**(5), 1364–1370 (2017)
3. L. Xiaoshen, Y. Xuehai, J. Mingzuo, Z. Chunling, Z. Xiao, K. Li, Fuzzy inference modeling method based on t-s fuzzy system. J. Intell. Fuzzy Syst. **31**(2), 727–736 (2016)
4. S. Feng, C. L. P. Chen, Nonlinear system identification using a simplified fuzzy broad learning system: stability analysis and a comparative study. Neurocomputing. **337**, 274–286 (2019)
5. M. Sugeno, T. Yasukawa, A fuzzy-logic-based approach to qualitative modeling. IEEE Trans. Fuzzy Syst. **1**(1), 7–31 (1993)
6. J. Q. Chen, Y. G. Xi, Z. J. Zhang, A clustering algorithm for fuzzy model identification. Fuzzy Sets Syst. **98**(3), 319–329 (1998)
7. U. Qamar, A dissimilarity measure based fuzzy c-means (fcm) clustering algorithm. J. Intell. Fuzzy Syst. **26**(1), 229–238 (2014)
8. R. J. Hathaway, J. C. Bezdek, Switching regression models and fuzzy clustering. IEEE Trans. Fuzzy Syst. **1**(3), 195–204 (1993)
9. E. Kim, M. Park, S. Ji, M. Park, A new approach to fuzzy modeling. IEEE Trans. Fuzzy Syst. **5**(3), 328–337 (1997)
10. E. Kim, M. Park, S. Kim, M. Park, A transformed input-domain approach to fuzzy modeling. IEEE Trans. Fuzzy Syst. **6**(4), 596–604 (1998)
11. C. C. Kung, J. Y. Su, Affine Takagi-Sugeno fuzzy modelling algorithm by fuzzy c-regression models clustering with a novel cluster validity criterion. Iet Control Theory Appl. **1**(5), 1255–1265 (2007)
12. C. Li, J. Zhou, X. Xiang, Q. Li, X. An, T-S fuzzy model identification based on a novel fuzzy c-regression model clustering algorithm. Eng. Appl. Artif. Intell. **22**(4-5), 646–653 (2009)
13. I. Gath, A. B. Geva, Unsupervised optimal fuzzy clustering. IEEE Trans. Pattern Anal. Mach. Intell. **11**(7), 773–781 (1989)
14. D. E. Gustafson, W. C. Kessel, Fuzzy clustering with a fuzzy covariance matrix. Decision and Control including the 17th Symposium on Adaptive Processes, 1978 IEEE Conference on IEEE, San Diego California, 761–766 (1978)
15. N. Wang, C. Hu, T-S fuzzy identification method based on nearest neighbor fuzzy clustering. Control Eng. China. **026**(006), 1068–1073 (2019)
16. N. Liu, F. Liu, A. Meng, Fuzzy identification based on improved pso and fcm. CAAI Trans. Intell. Syst. **014**(002), 378–384 (2019)
17. L. Wang, S. Wang, Takagi-Sugeno fuzzy modeling based on improved fuzzy clustering for health care data. J. Nanjing Univ. Nat. Sc. **56**(2), 186–196 (2020)
18. P. Li, Z. Chen, L. T. Yang, L. Zhao, Q. Zhang, A privacy-preserving high-order neuro-fuzzy c-means algorithm with cloud computing. Neurocomputing. **256**, 82–89 (2017)
19. R. Jin, G. Weng, A robust active contour model driven by fuzzy c-means energy for fast image segmentation. Digit. Sig. Process. **90**, 100–109 (2019)
20. N. Gueorguieva, I. Valova, G. Georgiev, Mmfcm: fuzzy c-means clustering with Mahalanobis and Minkowski distance metrics. Procedia Comput. Sci. **114**, 224–233 (2017)
21. L. Liang Qun, W. Xiao Li, X. Wei Xin, Z. XiangLiu, A novel recursive t-s fuzzy semantic modeling approach for discrete state-space systems. Neurocomputing. **340**, 222–232 (2019)
22. R. D. Chandrakumar, S. Senthil, Efficient kernel induced fuzzy c-means based on Gaussian function for image data analyzing. J. Intell. Fuzzy Syst. Appl. Eng. Technol. **30**(2), 983–990 (2016)
23. R. J. Kuo, W. C. Cheng, An intuitionistic fuzzy neural network with gaussian membership function. J. Intell. Fuzzy Syst. **36**(6), 6731–6741 (2019)
24. J. Kennedy, R. C. Eberhart, in *Proceedings of the IEEE International Conference on Neural Networks*. Particle swarm optimization (IEEE, Australia, 1995), pp. 1942–1948
25. K. S. Narendra, K. Parthasarathy, Identification and control of dynamical systems using neural networks. IEEE Trans. Neural Netw. **1**(1), 4–27 (1990)
26. W. A. Farag, V. H. Quintana, A genetic-based neuro-fuzzy approach for modeling and control of dynamical systems. IEEE Trans. Neural Netw. **9**(5), 756–767 (1998)
27. W. Sheng De, C. H. Lee, Fuzzy system modeling using linear distance rules. Fuzzy Sets Syst. **108**(2), 179–191 (1999)
28. A. Evsukoff, A. C. S. Branco, S. Galichet, Structure identification and parameter optimization for non-linear fuzzy modeling. Fuzzy Sets Syst. **132**(2), 173–188 (2002)
29. A. Bagis, Fuzzy rule base design using tabu search algorithm for nonlinear system modeling. ISA Trans. **47**(1), 32–44 (2008)
30. D. Bartholomew, G. E. P. Box, G. M. Jenkins, Time series analysis forecasting and control. J. Oper. Res. Soc. **22**(2), 199–201 (1971)
31. C. W. Xu, Y. Z. Lu, Fuzzy model identification and self-learning for dynamic systems. IEEE Trans. Syst. Man Cybern. **17**(4), 683–689 (1987)
32. G. E. Tsekouras, On the use of the weighted fuzzy c-means in fuzzy modeling. Adv. Eng. Softw. **36**(5), 287–300 (2005)
33. F. Liu, Fuzzy adaptive inverse control for pneumatic loading system. J. Mech. Eng. **50**(14), 185 (2014)
34. F. Liu, Application of linear/nonlinear active disturbance rejection switching control in variable load pneumatic loading system. J. Mech. Eng. **54**(12), 225 (2018)

Publisher's Note

Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.