Proceedings of the 33rd Chinese Control Conference
July 28-30, 2014, Nanjing, China

T-S Model Identification Based on Silhouette Index and Improved Gravitational Search Algorithm

Xueming Ding[1], Zhenkai Xu[1], Meimei Liu[1], Juan Wu[1]

1. Department of Control Science and Engineering, University of Shanghai for Science and Technology, Shanghai 200093, China
E-mail: xuzhenkaizj@163.com

Abstract: In this paper, an approach based on silhouette index (SI) and improved gravitational search algorithm (IGSA) is presented to deal with T-S model identification problem. Clustering algorithm employing SI and IGSA is introduced for structure identification. The SI considers both the intra-cluster cohesion and the inter-cluster separation which can highly assess the accuracy of clustering. One cluster represents a fuzzy rule. Cluster center is regarded as the gauss membership function center parameter, which is identified by IGSA. The improved algorithm IGSA is also used for parameter identification of T-S model. It introduces the mutation of genes to standard GSA and considers the best solution, which enhances the search space and improves the ability of sharing global information. The simulation results produced by a two order nonlinear system and Box-Jenkins gas stove illustrate the effectiveness of the proposed method.

Key Words: T-S model, silhouette index(SI),gravitational search algorithm(GSA),clustering algorithm

1 Introduction

The systems of natural world are basically nonlinear. It is very hard, even impossible to build accurate models of these complex nonlinear systems. As an effective technique to system modeling, system identification is important and complicated, in which 'black-box' identification based on input-output data is becoming a hot area of research. Takagi-Sugeno(T-S) fuzzy model [16], which is one of the most outstanding model, has been widely studied and applied in many literatures [2, 14, 18]. A T-S fuzzy model consists of several If-Then rules, where the rule consequents are usually constant values or linear functions of the inputs. T-S model identification consists of structure identification and parameter identification. Structure identification is to obtain fuzzy rules based on the fuzzy partition of the input-output data space, which determines the structure of T-S model. While parameter identification includes identifying gauss membership function basis width parameter and rule consequent parameters.

Gravitational search algorithm(GSA) [3] based on the law of gravity was proposed by Rashedi etc. in 2009. The algorithm is simple, easy to be realized and powerful in local searching ability, which has been successfully used in nonlinear function optimization [3] and system parameter identification [9]. One major weakness of the standard GSA is its global searching ability [3]. Once the best position is close to local optimization, any other masses will converge to local optimal value.

In this paper, a new clustering algorithm is presented based on silhouette index (SI) [4] and the improved algorithm IGSA. SI has been used as clustering index to assess the accuracy of clustering, which considers both the intra-cluster cohesion and the inter-cluster separation to minimize the comparability among clusters and maximize the comparability inside each cluster. IGSA considering the mutation of genes and the best solution found so far is used to identify parameters which include gauss membership function basis width parameter and rule consequent parameters. A specified rule number is given by experiences. One cluster represents a fuzzy rule. Cluster center is used as the gauss membership function center parameter.

The rest of this paper is organized as follows. In Section 2, we briefly review the T-S fuzzy model. Section 3 gives the proposed T-S model identification approach. The section consists of introduction of the GSA, the improvements and analysis in the IGSA, description of SI, structure identification and parameter identification. Section 4 shows the experiments on the proposed method. Some comparison works with other fuzzy approaches is conducted to test the performance of the method. The conclusion is presented in Section 5.

2 T-S model

Takagi and Sugeno [16] proposed the well-known T-S model in 1985 to describe complicated nonlinear system by decomposing input space into several subspaces, each of that is represented with a simple linear regression model system.

The typical T-S fuzzy rule is described as follow:

$R^{(i)}$: IF x_1 is A_1^i , x_2 is A_2^i , $\cdots$, x_n is A_n^i Then

$$y_i = \mathbf{w}_i^T \mathbf{x} + b_i \quad i = 1, 2, \cdots, m \tag{1}$$

where $i = 1, 2, \cdots, m$ is the number of fuzzy rules. $\mathbf{x} = [x_1, x_2, \cdots, x_n]$ is the input variable. n is the dimension of input variable$\mathbf{x}$. $\mathbf{w} = [w_1, w_2, \cdots, w_n]$ are the consequent parameters. y_i is the output of i_{th} fuzzy rule and A_j^i is a fuzzy set.

Given the input$\mathbf{x} = [x_1, x_2, \cdots, x_n]$, the output of the fuzzy model can be calculated by a weighted mean defuzzification as follow:

$$y = \frac{\sum_{i=1}^{m} \mu_i y_i}{\sum_{i=1}^{m} \mu_i} \tag{2}$$

where the weight strength μ of the i_{th} rule is computed as:

$$\mu_i(\mathbf{x}) = \prod_{j=1}^{n} h_i(\mathbf{x}) \tag{3}$$

This paper was partially supported by The National Natural Science Foundation (61374040), Key Discipline of Shanghai (S30501), Scientific Innovation program(13ZZ115), Graduate Innovation program of Shanghai(54-13-302-102).

$\mu_i(\mathbf{x})$is the grade of membership function(MF), and it is defined by Gaussian function as Eq. (3):

$$h_i(\mathbf{x}) = \exp(-\frac{(x_j - c_{ij})^2}{\sigma_{ij}^2}) \tag{4}$$

where c_{ij} and σ_{ij} denote the center and width of the fuzzy set, respectively.

3 T-S model structure and parameter identification

In this section, firstly we analyze the disadvantage of GSA. Then, IGSA is proposed and used in T-S model identification. Thirdly, the new clustering algorithm based on SI and IGSA with a specified rule number is presented for structure identification. Lastly, gauss membership function basis width parameter and rule consequent parameters are identified based on IGSA.

3.1 Gravitational search algorithm (GSA)

In GSA , agents are considered as objects and their performance is measured by their masses. All these objects attract each other by the gravity force, and this force causes a global movement of all objects towards the objects with heavier masses. Hence, masses cooperate using a direct form of communication, through gravitational force. The heavy masses, which correspond to good solutions, move more slowly than lighter ones. This guarantees the exploitation step of the algorithm.

In GSA, each mass (agent) has four specifications: position, inertial mass, active gravitational mass, and passive gravitational mass. The position of the mass corresponds to a solution of the problem, and its gravitational and inertial masses are determined using a fitness function.

In other words, each mass presents a solution, and the algorithm is navigated by properly adjusting the gravitational and inertia masses. By lapse of time, we expect that masses be attracted by the heaviest mass. This mass will present an optimum solution in the search space.

Now, consider a system with N agents (masses). We define the position of the i_{th} agent by:

$$\mathbf{x}_i = \left(x_i^1, x_i^2, \cdots, x_i^d, \cdots x_i^n\right), i = 1, 2, \cdots, N, \tag{5}$$

where x_i^d presents the position of i_{th} agent in the d_{th} dimension.

At a specific time t , we define the force acting on mass i from mass j as following:

$$F_{ij}^d(t) = G(t)\frac{M_{pi}(t) \times M_{aj}(t)}{R_{ij}(t) + \varepsilon}(x_j^d(t) - x_i^d(t)) \tag{6}$$

where M_{aj} is the active gravitational mass related to agent j , M_{pj} is the passive gravitational mass related to agent i , $G(t)$ is gravitational constant at time t , ε is a small constant, and $R_{ij}(t)$ is the Euclidian distance between two agents i and j :

$$R_{ij}(t) = \|\mathbf{x}_i(t), \mathbf{x}_j(t)\|_2 \tag{7}$$

To give a stochastic characteristic to our algorithm, we suppose that the total force that acts on agent i in a dimension d be a randomly weighted sum of d_{th} components of the forces exerted from other agents:

$$F_i^d(t) = \sum_{j \in kbest, j \neq i}^{N} rand_j F_{ij}^d(t) \tag{8}$$

where rand j is a random number in the interval $[0, 1]$.

Hence, by the law of motion, the acceleration of the agent i at time t , and in direction d_{th} , $a_i^d(t)$, is given as follows:

$$a_i^d(t) = \frac{F_i^d(t)}{M_{ii}(t)} \tag{9}$$

where M_{ii} is the inertial mass of i_{th} agent.

Furthermore, the next velocity of an agent is considered as a fraction of its current velocity added to its acceleration. Therefore, its position and its velocity could be calculated as follows:

$$v_i^d(t+1) = rand_i \times v_i^d(t) + a_i^d(t) \tag{10}$$

$$x_i^d(t+1) = x_i^d(t) + v_i^d(t+1) \tag{11}$$

where $rand_i$ is a uniform random variable in the interval $[0, 1]$. We use this random number to give a randomized characteristic to the search.

The gravitational constant G ,is initialized at the beginning and will be reduced with time to control the search accuracy. In other words, G is a function of the initial value G_0 and time t:

$$G(t) = G_0 e^{-\alpha \frac{t}{T}} \tag{12}$$

Gravitational and inertia masses are simply calculated by the fitness evaluation. A heavier mass means a more efficient agent. This means that better agents have higher attractions and walk more slowly. Assuming the equality of the gravitational and inertia mass, the values of masses are calculated using the map of fitness. We update the gravitational and inertial masses by the following equations:

$$M_{ai} = M_{pi} = M_{ii} = M_i \quad i = 1, 2, \cdots, N \tag{13}$$

$$m_i(t) = \frac{fit_i(t) - worst(t)}{best(t) - worst(t)} \tag{14}$$

$$M_i(t) = \frac{m_i(t)}{\sum_{j=1}^{N} m_j(t)} \tag{15}$$

where $fit_i(t)$ represent the fitness value of the agent i at time t , and, $worst(t)$ and $best(t)$ are defined as follows (for a minimization problem):

$$best(t) = \min_{j \in \{1,2,\cdots,N\}} fit_j(t) \tag{16}$$

$$worst(t) = \max_{j \in \{1,2,\cdots,N\}} fit_j(t) \tag{17}$$

It is to be noted that for a maximization problem, Eqs. (16) and (17) are changed to Eqs. (18) and (19), respectively:

$$best(t) = \max_{j \in \{1,2,\cdots,N\}} fit_j(t) \tag{18}$$

$$worst(t) = \min_{j \in \{1,2,\cdots,N\}} fit_j(t) \tag{19}$$

3.2 Improved gravitational search algorithm (IGSA)

To overcome the GSA's weakness of its poor global searching ability, two methods are proposed as follows which make up IGSA.

Method 1: Considering the best solution found so far, the next velocity of an agent is calculated as follows:

$$V_i^d(t+1) = r1 \times V_i^d(t) + c \times r2 \times (gbest - X_i(t)) + a_i^d(t) \tag{20}$$

where $V_i^d(t)$ is the velocity of agent i at iteration t, c is a weighting factor, r_j is a random number between 0 and 1, $a_i^d(t)$ is the acceleration of agent i at iteration t, and $gbest$ is the best solution so far.

In each iteration, the positions of particles are updated as follow:

$$X_i^d(t+1) = X_i^d(t) + V_i^d(t+1) \tag{21}$$

In method 1, the quality of solutions (fitness) is considered in the updating procedure. The agents near good solutions try to attract the other agents which are exploring the search space. When all agents are near a good solution, they move very slowly. In this case, the $gbest$ help them to exploit the global best solution. Method 1 use a memory $(gbest)$ to save the best solution found so far, so it is accessible anytime. Each agent can observe the best solution so far and tend toward it.

Method 2:The mutation of genes is introduced in GSA. The specific steps are as follows:

1) Let p_m is the mutation probability to creat new agent (mass) while p_r is a random number in interval $[0, 1]$ .If p_r is great than p_m , new agent is created .Else position is unchanged.

$$r' = rand(x_j^d(t) - x_k^d(t)) \tag{22}$$

$$x_i^d(t+1) = \begin{cases} x_i^d(t) + r' & p_r > p_m \\ x_i^d(t) & else \end{cases} \tag{23}$$

In Eq. (22), r' is the step size to generate new agent(mass), $x_j^d(t)$ and $x_k^d(t)$ are the positions selected from the whole population randomly. In Eq. (23), $x_i^d(t)$ is the old position.

2) Calculate the new fitness value fit_{new} of created agent. If fit_{new} is better than old fitness fit_{old} , then the new position x_{new} replaces the old position $x_i^d(t)$. Else position is unchanged.

$$x_{new} = \begin{cases} x_i^d(t+1) & fit_{new} > fit_{old} \\ x_i^d(t) & else \end{cases} \tag{24}$$

Method 2 that introduces in the mutation of genes makes the population evolve in an open environment and its diversification get enhanced.

3.3 Silhouette index criterion

The silhouette statistic coefficient that measures both the intra-cluster cohesion and the inter-cluster separation, which minimize the comparability among clusters and maximize the comparability inside each cluster, and has been used as clustering index to assess the accuracy of clustering. It is defined as follows.

Let i be an instance belonging to cluster A. Let $a(i)$ represent the average dissimilarity of instance i to all other instances of A . Now, let us take into account a cluster C , $C \neq A$.The average dissimilarity of instance i to all instances of C will be denoted by $d(i, C)$. After computing $d(i, C)$ for all clusters C, the smallest one is selected, i.e.$b(i) = \min d(i, C)$, $C \neq A$. This value represents the dissimilarity of instance i to its nearest neighbor cluster, and the silhouette coefficient $S(i)$ [7] is given by:

$$S(i) = \frac{b(i) - a(i)}{\max\{b(i), a(i)\}} \tag{25}$$

The sample size is L , then the average value S_{av} of silhouette coefficient is calculated as follows:

$$S_{av} = \frac{\sum_{i=1}^{L} S(i)}{L} \tag{26}$$

It is easy to verify that $0 \leq S_{av} \leq 1$. The higher S_{av} , the better the assignment of instance i to the given cluster is. Here are the rules as follows:

- If $S_{av} > 0.5$, then each cluster is separated clearly.
- If $S_{av} < 0.5$, then some clusters are overlapped.
- If $S_{av} < 0.2$, clustering sucks.

This paper employs the silhouette statistic S_{av} as evaluation criterion in structure identification.

3.4 Structure identification

Let n be the dimension of cluster centers. m is designated cluster number. The cluster centers are encoded in Fig. 1.The implementation of structure identification is listed by the following pseudo-code:

Algorithm 1. The structure identification algorithm

```
begin
    Encode cluster centers as Fig. 1.
    Randomly initialize the cluster centers.
    for t = 1, ..., T_max do
        Each instance belonging to its nearest neighbor cluster.
        Calculate silhouette coefficient S_av based on Eq. (26).
        Calculate fitness of each individual.
        Calculate the values of masses M_i(t) , i = 1, ..., N.
        Update G(t) ,best(t) and worst(t).
        Calculate the total force F_i^d(t) and acceleration a_i^d(t).
        Update velocity v_i^d(t+1) and position x_i^d(t+1) based on
        Eq. (20) , (21), (22) and (23).
        Get current best cluster centers.
    end
end
```

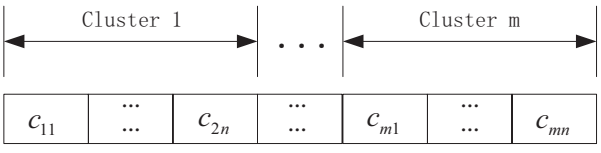

Fig. 1: Cluster center coding method

3.5 Parameter identification

Each individual represents a complete solution of parameter σ_i , w_i , b_i . Individuals are encoded in Fig. 2.

The mean square error (MSE) which is defined as Eq. (27) is used as the fitness function to evaluate various particles

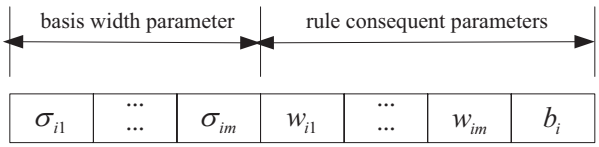

Fig. 2: Parameter coding method

within the swarm of potential solutions.

$$MSE = \frac{1}{n}\sum_{i=1}^{n}(y_i - \hat{y}_i)^2 \tag{27}$$

where n is the number of training samples, y_i and $\hat{y}_i$ are the desired reference output and the fuzzy model output of the i_{th} input sample.

The implementation of parameter identification is listed by the following pseudo-code:

Algorithm 2. The parameter identification algorithm

```
begin
    Encode individuals as Fig. 2.
    Randomly initialize individuals.
    for t = 1, ..., T_max do
        Calculate fitness based on Eq. 27.
        Calculate the values of masses M_i(t) , i = 1, ..., N.
        Update G(t) ,best(t) and worst(t).
        Calculate the total force F_i^d(t) and acceleration a_i^d(t).
        Update velocity v_i^d(t+1) and position x_i^d(t+1) based on
        Eq. (20) , (21), (22) and (23).
        Get current best individuals.
    end
end
```

4 Numerical simulation

This section shows two simulation experiments to illustrate the identification accuracy and the generalization ability of the proposed T-S model identification algorithm based on SI and IGSA. The first experiment considers the problems of function approximation and the last one is a representative Box-Jenkins problem.

4.1 Static nonlinear system

In this section, a static nonlinear system with two inputs and single output that is taken from [2] is used for modeling test, which is represented as following:

$$y = (1 + u_2^{-2} + u_1^{-1.5})^2, 1 \le u_1, u_2 \le 5 \tag{28}$$

For the purpose of comparison, we get 50 input-output data of Eq. 28 from [2] on testing the prediction performance, assuming the random input signals u_1 , u_2 uniformly distributed in the interval $[1, 5]$. The result is obtained and compared with results in the literature. Parameters of the IGSA are set as $G_0 = 10$,$\beta = 6$, the maximum iteration number $\max\_it = 500$,and the number of agents $Popsize = 30$. The number of fuzzy rules is 6 and the total parameters are 42. The values of the premise parameter are listed in Table 1, together with the corresponding identified consequence parameters. The performance of T-S model is exhibited by Fig. 3, which shows the comparison of the outputs of the identified T-S model and the real system as well as the error between them. The T-S model possesses an MSE of 0.005, showing the performance of the proposed approach in achieving model accuracy. The detailed performance comparisons are shown in Table 2, which shows our model owns a higher identification accuracy.

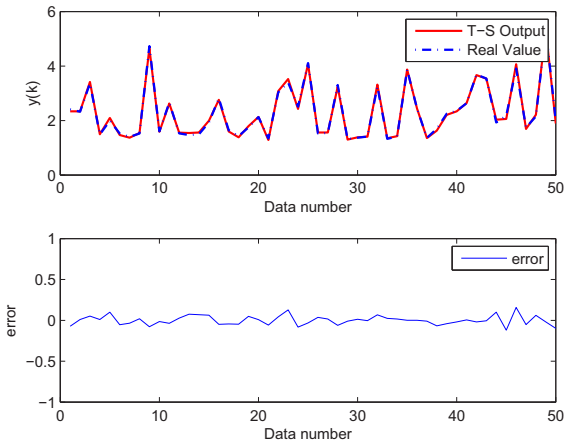

Fig. 3: Comparison of our model and the real system for the static nonlinear function

Table 1: List of the parameters of our model for the static function

R^i	R^1	R^2	R^3	R^4	R^5	R^6
c_{i1}	1.1675	3.9951	2.0022	1.7549	3.5534	1.5718
c_{i2}	4.4150	3.5105	4.6558	3.8413	1.0723	3.9749
σ_{i1}	0.9323	1.5666	2.8323	3.2685	1.8529	1.0382
σ_{i2}	0.9607	31.4007	23.0732	2.1873	0.7710	1.4585
w_{i1}	-6.0769	0.5113	5.8718	-6.8498	-1.0437	-6.8302
w_{i2}	-0.1781	0.7190	-8.2437	7.1286	-7.3283	4.2380
b_i	4.0514	-9.9700	16.6556	1.5050	15.7019	-2.5056

Table 2: Comparison of different models for the static function.

Model	Rule No.	TPP ‡	MSE
Sugeno and Yasukawa [15]	6	65	0.0790
Kim et al.[5]	3	21	0.0197
Kim et al.[6]	3	21	0.0090
Kung and Su[7]	3	21	0.0196
Li et al.[11]	6	42	0.0085
Nozaki et al.[13]	25	125	0.0085
Lee et al.[8]	10	−−	0.0148
Our model	6	42	0.0034

‡: total number of parameters.

4.2 Box-Jenkins gas furnace data

In this experiment, Box and Jenkins gas furnace data[1] is introduced as an example of system identification. This data set originally consists of 296 input-output observations $\{y(t), u(t)\}$ from $t = 1$ to 296 of a gas furnace process. At each sampling time t , $u(t)$ is the input gas flow rate and $y(t)$

is the output CO_2 concentration. The simulation is conducted to predict $y(t)$ based on $y(t-1)$,$y(t-2)$,$u(t-1)$,$u(t-2)$. To validate the robustness of the proposed approach, the first 148 input-output data were utilized as training data, and the last 148 as testing data. The result is obtained and compared with results in the literature. Parameters of the IGSA are set as $G_0 = 5$, $\beta = 6$, the maximum iteration number $\max\_it = 600$, and the number of agents $Popsize = 30$. The rule number of the fuzzy model is 3 and the total parameters are 39.

Table 3. shows the identified parameters. Fig. 4. expresses the prediction of T-S model and the real outputs as well as the errors between them for the training data. Fig. 5. depicts testing result. Their corresponding MSE equal to 0.036 and 0.101, respectively. The comparison with other models is listed in Table 4which shows the generalization ability of our method is better than that occurred in the literature.

Table 3: List of the parameters of our model for the box and jeckins system

R^i		Parameters vectors			
R^1	c_1	57.8591	57.7844	-0.6725	-0.7129
	σ_1	0.2293	0.1702	0.4231	0.1682
	w_1	0.8135	0.5671	0.0987	3.7561
	b_1	5.0000			
R^2	c_2	53.5580	53.5517	-0.3742	-0.3133
	σ_2	0.3804	0.1650	0.1512	0.1666
	w_2	4.2040	1.9463	-2.9789	-0.3970
	b_2	4.9997			
R^3	c_3	49.7387	49.8509	-0.7420	-0.7918
	σ_3	0.2546	0.1599	0.2178	0.1510
	w_3	-0.2072	-4.6052	3.8356	-4.9669
	b_3	4.9980			

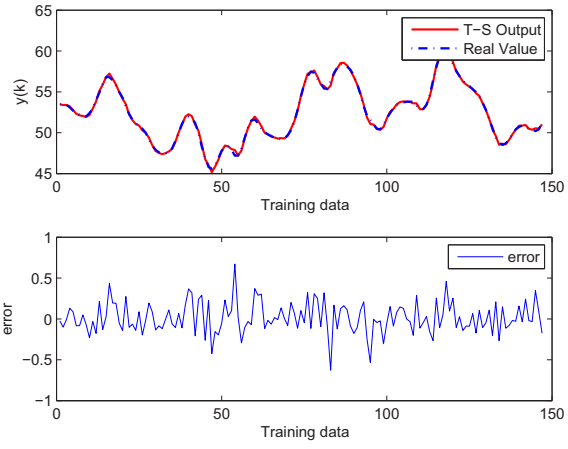

Fig. 4: Comparison of our model and the real system for the Box and Jenkins system using training data

5 Conclusions

T-S model that can approach arbitrary nonlinear system is sum of linear subsystems and its weight product. In the paper, an improved algorithm IGSA that introduces the mutation of genes to GSA and considers the best solution found so far is proposed and used in T-S model identification. The

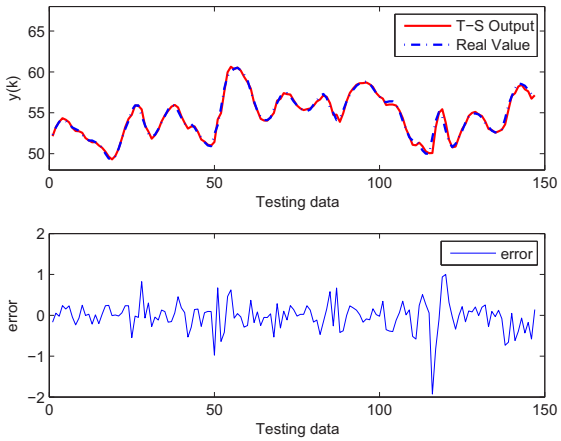

Fig. 5: Comparison of our model and the real system for the Box and Jenkins system using testing data

Table 4: Comparison of different models for the box and jeckins system

		MSE	
Model	Rule No.	Training data	Testing data
Lin et al.[12]	5	0.071	0.261
Kim et al.[6]	6	0.034	0.244
Tsekouras[17]	6	0.022	0.236
Li et al.[10]	3	0.015	0.147
Our model	3	0.036	0.101

clustering algorithm based on SI and IGSA with a specified rule number is presented for structure identification. Based on empirical risk minimization principle, gauss membership function basis width parameters and rule consequent parameters are identified by IGSA. The simulation results of two experiments demonstrated that the proposed method has better performance comparing with some existing methods.

References

[1] G.E.P. Box and G.M. Jenkins,*Time Series AnalysisForecasting and Control*,CA:Holden Day, San Francisco, 1970.

[2] C.F.Juang and C.D.Hsieh,"A locally recurrent fuzzy neural network with support vector regression for dynamic-system modeling," *Fuzzy Systems, IEEE Transactions on*,vol. 18, no. 2, pp. 261-273, Apr 2010.

[3] R.Esmat, N.P. Hossein, and S.Saeid,"SA: A gravitational search algorithm," *Information Sciences*,vol. 179, no. 13, pp. 2232-2248, 2009.

[4] K.J.Wang, J.Y.Zhang, X.N.Zhang D.Li, and T.Guo,"Adaptive affinity propagation clustering," *Acta Automatica Sinica*,vol. 33,no.12,pp.1242-1246, 2007.

[5] E.Kim, M.Park, and S.Ji,"A new approach to fuzzy modeling," *Fuzzy Systems, IEEE Transactions on*,vol. 5,no.3, pp. 328-337, Aug 1997.

[6] E.Kim, M.Park, and S.Kim,"A transformed input-domain approach to fuzzy modeling," *Fuzzy Systems, IEEE Transactions on*,vol.6,no.4,pp.596-604, Nov 1998.

[7] C.C. Kung and 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.*, vol.1,no.5,pp.1255-1265, 2007.

[8] S.J. Lee and G.S. Ouyang,"A neuro-fuzzy system model-

ing with selfconstructing rule generation and hybrid svd-based learning," *Fuzzy Systems, IEEE Transactions on*,vol. 11,no.3,pp.341-353, Jun 2003.

[9] C.Li and J.Zhou,"Parameters identification of hydraulic turbine governing system using improved gravitational search algorithm," *Energy Convers. Manage*,vol. 52,no.1,pp.374-381, 2011.

[10] C.Li, J.Zhou, B.Fu, P.Kou, and J.Xiao,"T-S fuzzy model identification with a gravitational search-based hyperplane clustering algorithm," *Fuzzy Systems, IEEE Transactions on*, vol.20,no.2,pp.305-317, Apr 2012.

[11] C.Li, J.Zhou, X.Xiang, Q.Li, and X.An,"T-S fuzzy model identification based on a novel fuzzy c-regression model clustering algorithm," *J. Eng.Appl. Artif. Intell.*, vol.22,pp.646-653, 2009.

[12] Y.Lin and G.A. I.Cunningham," A new approach to fuzzy-neural system modeling," *Fuzzy Systems, IEEE Transactions on*, vol.3,pp.190-198, May 1995.

[13] K.Nozaki, H.Ishibuchi, and H.Tanaka,"A simple but powerful method for generating fuzzy rules from numerical data,"*Fuzzy Sets and Systems*, vol.86,pp.251-270, 1997.

[14] P.Pulkkinen and H.Koivisto,"A dynamically constrained multiobjective genetic fuzzy system for regression problems,"*Fuzzy Systems, IEEE Transactions on*, vol.18,no.1,pp.161-177, Feb 2010.

[15] M.Sugeno and T.Yasukawa,"A fuzzy-logic-based approach to qualitative modeling,"*Acta Automatica Sinica*, vol.1,no.1,pp.7-31, Feb 1993.

[16] T.Takagi and M.Sugeno,"Fuzzy identification of systems and its applications to modeling and control,"*System, Man, and Cybernetics, IEEE Transaction on*,vol.15,no.1,pp.116-132, Feb 1985.

[17] G.E. Tsekouras,"On the use of the weighted fuzzy c-means in fuzzy modeling,"*Advances in Engineering Software*, vol.36,no.5,pp.287-300, 2005.

[18] Z.Xi, G.Feng, and T.Hesketh,"Piecewise integral sliding-mode control for t-s fuzzy systems,"*Fuzzy Systems, IEEE Transactions on*, vol.19,no.1,pp.65-74, Feb 2011.