International Journal of Soft Computing and Engineering (IJSCE)
ISSN: 2231-2307, Volume-4 Issue-4, September 2014

Chaotic Time Series Prediction using Improved ANFIS with Imperialist Competitive Learning Algorithm

Maysam Behmanesh, Majid Mohammadi, Vahid Sattari Naeini

Abstract*—This paper presents an improved adaptive Neuro-fuzzy inference system (ANFIS) for predicting chaotic time series. The previous learning algorithms of ANFIS emphasized on gradient based methods or least squares (LS) based methods, but gradient computations are very computationally and difficult in each stage, also gradient based algorithms may be trapped into local optimum. This paper introduces a new hybrid learning algorithm based on imperialist competitive algorithm (ICA) for training the antecedent part and least square estimation (LSE) method for optimizing the conclusion part of ANFIS. This hybrid method is free of derivation and solves the trouble of falling in a local optimum in the gradient based algorithm for training the antecedent part. The proposed approach is used in order to modeling and prediction of three benchmark chaotic time series. Analysis of the prediction results and comparisons with recent and old studies demonstrates the promising performance of the proposed approach for modeling and prediction of nonlinear and chaotic time series.

***Index Terms*—chaotic time series, Gradient based, imperialist competitive algorithm, Fuzzy systems, ANFIS, least square estimation.**

I. INTRODUCTION

Time series prediction is one of the most important prediction that collect past observations of a variable and analysis it to obtain the underlying relationships between observations and finding out a descriptive example. Then this model will be applied to extrapolate future time series. Prediction of time series has widespread in the areas of science, technology, medicine and Econometrics, among others. In general, time series can have some properties such as nonlinearity, chaotic, non-stationary and cyclic. Among the various types of time series chaotic time series can be typically found in natural phenomena [1], [2]. Anticipation of the behavior of chaotic time series as a nonlinear dynamic system is the case of a comparatively new research that has drawn the attention and efforts of many scientists. In the expanse of time series has been given many classical and statistical approaches, but those methods were often complex and not accurate and efficient in the encounter of chaotic and very large data [3], [4]. In recent years, many new prediction approaches, such as the wavelet networks [5], [6], neural networks [7], [8], fuzzy [9], [10] and Neuro-fuzzy systems [11]-[15] and evolutionary algorithms [16]-[19] have emerged.

Manuscript Received on September 2014.

Maysam Behmanesh, Department of Computer Engineering, Shahid Bahonar University of Kerman, Iran.

Majid Mohammadi, Department of Computer Engineering, Shahid Bahonar University of Kerman, Iran.

Vahid Sattari Naeini, Department of Computer Engineering, Shahid Bahonar University of Kerman, Iran.

One technique for modeling nonlinear systems is the integration of fuzzy logic and neural networks known as Neuro-fuzzy. In Neuro-fuzzy approach the capability of fuzzy-ruled based systems in handling uncertain and noisy data and the learning capability of neural networks is aggregated to form better estimators. Many different structures for fuzzy neural networks (FNNs) have been suggested. A special form of Neuro-fuzzy systems is ANFIS, which has demonstrated significant effects in modeling nonlinear functions. ANFIS is an adaptive neural network based fuzzy inference system that learning processes are performed by interleaving the optimization of the antecedent and conclusion parts parameters. Many types of Neuro-fuzzy systems have been proposed along with their learning algorithms, such as gradient based learning algorithms [20], genetic algorithms (GAs) and evolutionary based [21]. But, most current studies are not hybrid and there are some troubles during training. Training this network in antecedent part is more difficult than consequent part, because most methods for training antecedent part are based on gradient that gradient computation is really difficult in each stage and also may be falling in a local optimum. In this paper a new hybrid learning algorithm based on an imperialist competitive algorithm (ICA) and least square estimation (LSE) has been employed to train ANFIS and adjust its antecedent and consequent parameters. This algorithm is free of derivation which is really difficult to calculate for training antecedent part parameters, also complexity of this algorithm is less than other training algorithms. This new learning algorithm has been used in order to improve ANFIS performance and increasing its accuracy in prediction of chaotic time series. The simulation results of the new method for three types of chaotic time-series and models such as Mackey-Glass, Lorenz and Rossler models are compared with previous methods based on the Hyken criteria [48] to prove the effectiveness of the new method in prediction accuracy. The remainder of the article is organized as follows. In section II imperialist competitive algorithm (ICA) is discussed. Section III describes the ANFIS and its learning algorithms. Section IV describes the determination of ANFIS parameters using new learning method and uses this proposed model for time series prediction. The prediction results of three benchmark time series, including the Mackey–Glass time series, Lorenz and Rossler model are provided in section V. Finally, conclusions are provided in the final section.

II. IMPERIALIST COMPETITIVE ALGORITHM

Imperialist Competitive Algorithm (ICA) is a new social-politically motivated global search strategy that has recently been introduced for dealing with different

Published By:
Blue Eyes Intelligence Engineering
& Sciences Publication Pvt. Ltd.

optimization tasks (Atashpaz and Lucas, 2007) [22], [23]. Fig.1 shows the flowchart of the ICA. Similar to other evolutionary algorithms, this algorithm starts with an initial population and objective function is that calculate for all. Each individual of the population is called a country. Some of the best countries (in optimization terminology, countries with the least cost) are chosen to be the imperialist and the rest form the colonies of these imperialists. Then competition takes place between the colonists for taking colonies. All the colonies of initial countries are divided among the mentioned imperialists based on their power so that strongest imperialist has a greater chance for further colonies. Then each imperialist with their colonies forms an empire.

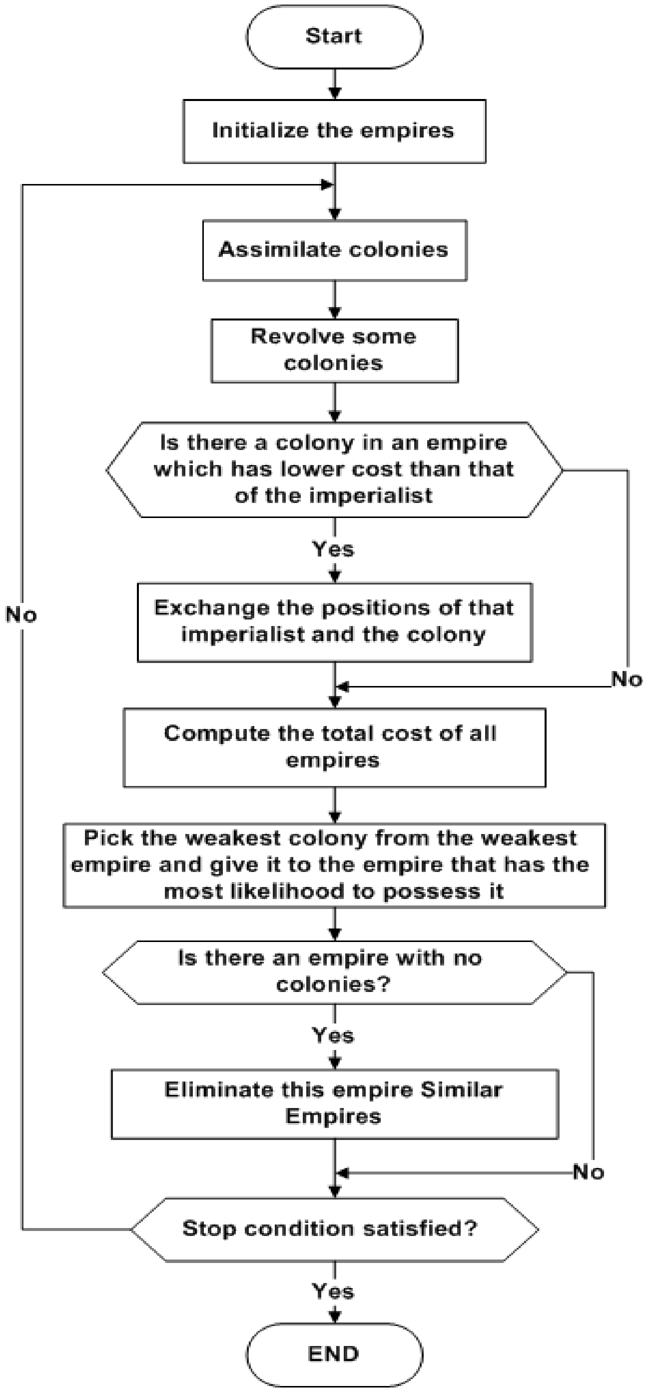

Fig. 1 Flowchart of the Imperialist Competitive Algorithm

Fig. 2 shows the initial empires. In this figure imperialist 1 has formed the most powerful empire and consequently has the greatest number of colonies.

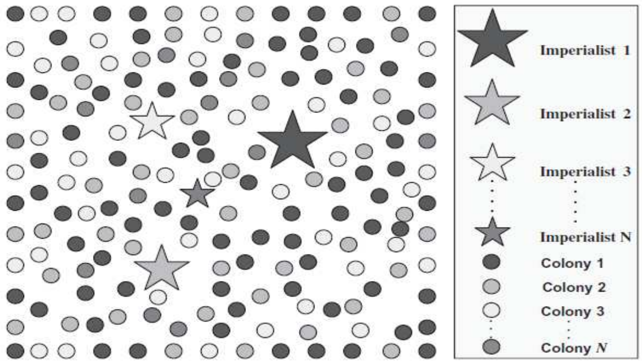

Fig. 2 Generating the Initial Empires: The More Colonies an Imperialist Possess, the Bigger is its Relevant ★ Mark

After dividing colonies between imperialists, these colonies approach their related imperialist countries. Fig. 3 represents this movement. Based on this concept, each colony moves toward the imperialist by ***a*** unit and reaches its new position. Where ***a*** is a random variable with uniform (or any proper) distribution, β a number greater than 1, causes colonies move toward their imperialists from different direction and ***d*** is the distance between colony and imperialist. An appropriate selection is $\beta = 2$ that $\beta > 1$ causes the colonies to get closer to the imperialist state from both sides.

$$a \sim U(0, \beta \times d) \tag{1}$$

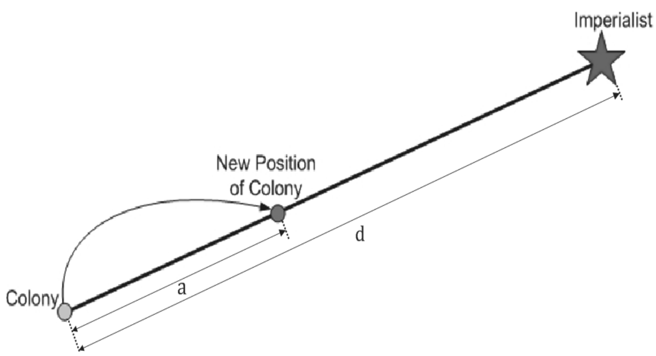

Fig. 3 Movement of Colonies Toward their Relevant Imperialist

In this paper, in order to increase the convergence speed and accuracy ICA, we used a modified ICA algorithm. To change this algorithm and increase diversity in population and increase the power of searching more area around the imperialist, a random amount of deviation is added to the direction of movement to modify the introduced algorithm. Fig. 4 shows the new direction. Namely instead of moves toward the imperialist by θ units with the same amount moves toward the imperialist but has θ deviation in the direction of movement. θ is a parameter with uniform distribution.

$$\theta \sim U(-\gamma, \gamma) \tag{2}$$

Increasing γ causes search enhanced in the around of imperialist and with reducing, colonies moves near the vector between colony and imperialist. Premature convergence may occur in different conditions: population will converge to a local optimum or algorithm go ahead slowly. In order to avoid from a fall in a local optimum and premature convergence and increase diversity of the population we use the revolution operator in ICA. In order to implementation of revolution operator and increasing the exploration, we allow each colony

Published By:
Blue Eyes Intelligence Engineering
& Sciences Publication Pvt. Ltd.

to change arbitrarily in some features. With apply this operator colonies obtains completely new position which increases the exploration of the search space. In this paper revolution operator apply randomly for some colonies after assimilation operator, in addition, this operator will also apply randomly to imperialists.

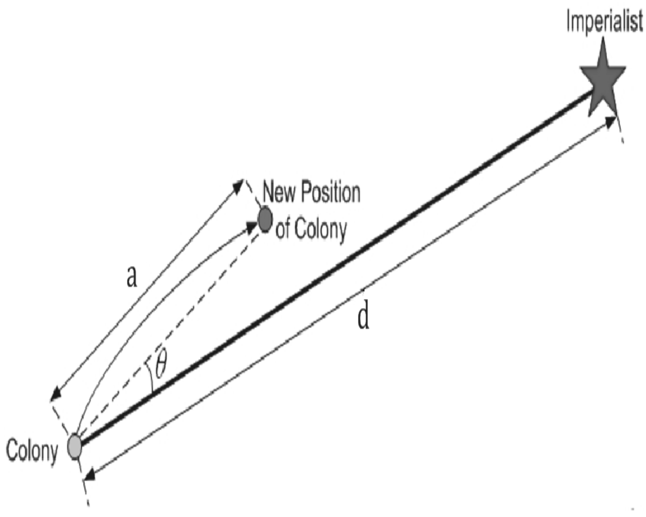

Fig. 4 Movement of Colonies Toward their Relevant Imperialist in a Randomly Deviated Direction

After moving toward the imperialist, a colony might reach to a place with better position than the imperialist (when cost in the location of the colony is lower than the cost in the location of imperialists). In this instance, the imperialist and the colony change their positions, then the algorithm will continue by the imperialist in the new location (intergroup competition). After this step total power of empire is calculated. The total power of an empire is mainly affected by the power of imperialist country. However the power of the colonies of an empire has an effect, albeit negligible, on the total power of that empire. Total power of empire ***n*** is defined by equation 3.

$$Cost\,(Emp_n) = Cost\,(Imperialist_n) + \xi \; mean\{Cost\,(Colonies \;\; of \;\; empire_n)\} \tag{3}$$

That ξ is a positive small number and generally considered to be between zero and one and near zero. In the following imperialistic are competing with based on the criteria of total power of empire. In the imperialistic competition process, all empires try to take the possession of colonies of other empires and control them. The imperialistic competition gradually brings about a reduction in the power of weaker empires and an increase in the power of more powerful ones, which is modeled by just picking some (usually one) of the weakest colonies of the weakest empires and making a competition among all empires to possess these (this) colonies. Also when an empire loses all of its colonies, it is assumed to be collapsed. In this method for selecting an empire for getting colony we use a roulette wheel method. Also in this paper, the collapsed empire also competed with other empires. After a while, all the empires except the most powerful one will collapse and all the colonies will be under the ascendancy of this unique empire. Ideally, all the colonies have the same positions and the same costs and they are controlled by an imperialist with the same position and costs as themselves, which means the algorithm converges to the best solution.

III. The Concept of ANFIS

A. *ANFIS Structure*

In this section, type III ANFIS topology and the learning method that used for this Neuro-fuzzy networks are presented. Both neural network and fuzzy logic [36] are model-free estimators and share the mutual ability to deal with uncertainties and noise. The ANFIS combines two approaches: neural networks and fuzzy systems. If both these two intelligent approaches are combined, good reasoning will be achieved in quality and quantity. In other words, both fuzzy reasoning and network calculation will be available simultaneously. The ANFIS is composed of two parts. The first is the antecedent part and the second is the conclusion part, which are connected to each other with the fuzzy rules base in network form. The type III ANFIS with tow inputs structure shown in Fig.5. As shown in this figure, it is a five layer network that can be described as a multi-layered neural network [24].

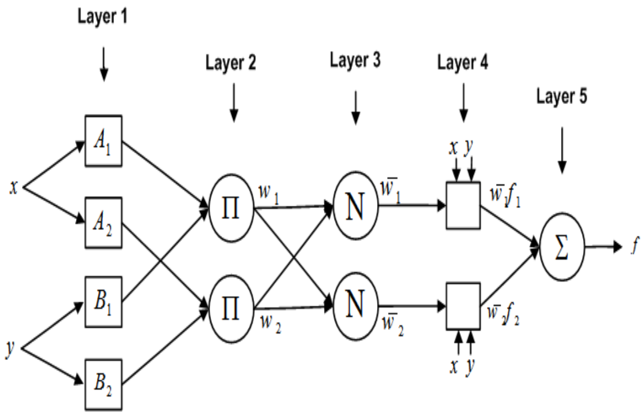

Fig. 5 The Equivalent Structure of ANFIS (Type III ANFIS) with Two Inputs and One Output

The first layer executes a fuzzification process, the second layer executes the fuzzy AND of the antecedent part of the fuzzy rules, the third layer normalizes the MFs, the fourth layer executes the conclusion part of the fuzzy rules, and the last layer computes the output of the fuzzy system by summing up the outputs of layer four. The feedforward equations of the ANFIS structure with two inputs and two labels for each input shown in Fig. 5 according to type III rules are as follows:

$$w_i = \mu_{A_i}(x) \times \mu_{B_i}(y), i = 1,2 \tag{4}$$

$$\bar{w}_i = \frac{w_i}{w_1 + w_2} \quad i = 1,2 \tag{5}$$

$$\left.\begin{aligned} f_1 &= a_1 x + b_1 y + c_1 \\ f_2 &= a_2 x + b_2 y + c_2 \end{aligned}\right\} \Rightarrow f = \frac{w_1 f_1 + w_2 f_2}{w_1 + w_2} = \bar{w}_1 f_1 + \bar{w}_2 f_2 \tag{6}$$

ANFIS has high ability of approximation that will depend on the resolution of the input space partitioning, which is determined by the number of MFs in the antecedent part for each input. In this paper, the MFs are used as Gaussian MF that *m* represents the center and σ determines the width of the MF respectively. Such as:

$$\mu_{A_i}(x) = e^{-\frac{1}{2}\left(\frac{x - m_i}{\sigma_i}\right)^2} \tag{7}$$

Published By:
Blue Eyes Intelligence Engineering
& Sciences Publication Pvt. Ltd.

B. Learning Algorithms

The learning algorithm of an ANFIS is to determine the parameters $(a_i\ ,b_i$, and $c_i)$ such that the error between the ANFIS output and the actual data can be minimized. Subsequent to the development of ANFIS approach, a number of methods have been proposed for training its parameters. For example, Mascioli et al. [25] have proposed to merge Min–Max and ANFIS model to obtain Neuro-fuzzy network and determine the optimal set of fuzzy rules. Jang and Mizutani [26] have presented the application of Levenberg–Marquardt method, which is essentially a nonlinear least squares technique, for training the ANFIS network structure. In another paper, Jang [27] has presented a system for input selection and Kumar and Garg [28] have used the Kohonen's map for training. Jang in his famous article [24] introduced four methods to update the parameters of the ANFIS structure, as listed below according to their computation complexities:

1. GD only: all parameters are updated by the GD.
2. GD only and one pass of LSE: the LSE is applied only once at the very beginning to get the initial values of the conclusion parameters and then the GD takes over to update all parameters.
3. GD only and LSE: this is the hybrid learning.
4. Sequential LSE: using extended Kalman filter to update all parameters.

These methods update antecedent parameters by using GD or Kalman filtering and due to the good performance still applies. Methods that have been proposed were gradient based and due to the proper performance also methods based on least squares as effective ways to optimize the parameters of the consequent part still applies. Training of antecedent parameters is very important. But this technique due to the high complexity in the gradient computation and the nonlinear presence of antecedent parameters in the output is not suitable. To resolve this problem population-based approaches such as genetic algorithm, ACO and other related methods can be raised [29], [30]. Many works proposed train Neuro-fuzzy networks by combining evolutionary algorithms and gradient descent methods, least squares and Kalman filter. For example, use of hybrid optimization method like PSO for the antecedent part and GD for the conclusion part in [31], [32]. In [33] training of antecedent and conclusion part done by PSO and LSE respectively. This paper proposed new method for determining the parameters of ANFIS that has less complexity and more accuracy and training of antecedent and consequent parameters are done by ICA and LSE algorithms respectively in a repetitive process.

IV. PROPOSED APPROACH

The algorithm that was used for the proposed method is described in this section step-by-step.

First Step**:** Create a matrix with time series data. A chaotic time series generally exhibits stochastic characteristics in time or frequency domain. However, based on Taken theory in phase-space reconstruction in time series and by using the coordinates with appropriate embedding dimension D and time delay τ , a quasiperiodic attractor in the phase space can be derived [34]. Consider a time series $\{x(1), x(2),..., x(N)\}$. An embedded phase vector $u(i)$ is:

$$u(i) = [x(i)\ x(i-\tau)\dots\ x(i-(D-1)\tau)] \tag{8}$$

Where $\forall i \in [1+(D-1)\tau, N]$, D is the embedding dimension, τ is the time delay and $u(i)$ is a vector in the D-dimensional phase space R^D . A trajectory in R^D is defined as:

$$U = [u^T(i)\quad u^T(i+1)\ \dots\ u^T(i+m)] \tag{9}$$

In order to extract the behaviors of the time series in an efficient way, optimal values of D and τ have to be determined. In this paper, τ and D are determined by using the average mutual information in [35], and the global false nearest neighbors in [1], respectively.

Second Step**:** Get the time series matrix to the entrance of the ANFIS structure. The inputs of ANFIS is a matrix in equation (9) with the embedding phase vector as the columns and the chaotic time series as the rows. Its dimension is $D \times (m+1)$, where i is an integer, $i \in [1+(D-1)\tau, N-m-k]$ and k is the number of prediction steps.

Third Step**:** Training ANFIS with data obtained in both time domain and phase space in an iterative and hybrid procedure by ICA and LSE algorithms. This combined training process is used to tune the antecedent and consequent parameters in order to achieve accurate prediction. Whenever the number of steps performed or desired training error is reached the process stops. After defining the training data, checking data, number of training epochs and type of membership functions, the optimal values for the antecedent parameters are determined by ICA in each step and after this, consequent parameters of improved ANFIS are estimated by LSE. This algorithm is performed in seven steps.

Step 1: Creating the initial empires.

1. Initializing the membership functions of the input parameters $(m_i\ ,\sigma_i)$ randomly and then based on those, initial estimates of the consequent parameters $(a_i\ ,b_i$ and $c_i)$ by LSE.
2. Calculate the fitness countries with MSE criterion in equation (10).
3. Select the best countries as imperialists and the remainder as the colonies of these imperialists. In this algorithm, SUS method used to select the colonies of imperialists.

Step 2: Apply assimilation and revolution operators. On each iteration, one of the parameters of membership function is being updated. In other words, in the first iteration, for example, m_i is updated, then in the second iteration, σ_i is updated. Then after updating all parameters again, the first parameter update is considered and so on. These parameters are grouped in a vector that is being updated iteration to iteration. With any change in the antecedent parameters, consequent parameters $(a_i\ ,b_i\ ,\ c_i)$ are estimated by LSE.

Step 3: Intergroup competition. If after applying operators, prediction error in ANFIS with parameters of one of the colonies is lower than the ANFIS with parameters of imperialist, then the imperialist and the colony change their

Published By:
Blue Eyes Intelligence Engineering
& Sciences Publication Pvt. Ltd.

positions and this algorithm will continue by the imperialist in the new location.

Step 4: Calculate the total power of the empires. The total power of the empire is calculated based on equation (3). In this paper, our purpose is minimizing the difference between ANFIS output and the actual output, thus the cost function is specified for each empire as the objective function with an equation (10).

$$Cost(colony_i) = \frac{1}{2.N}\sum_{i=1}^{N}(y-\hat{y})^2 \tag{10}$$

Step 5: Imperialist competition. Empires are competing based on total power of each empire. This step begins by picking some of the weakest colonies of the weakest empires and making a competition among all empires to possess these colonies. With this method the collapsed empires also compete with other empires.

Step 6: Choice best imperialist of empires as solution of antecedent parameter in ANFIS and estimate consequent parameters with LSE. In this method, in order to avoid overfitting is used validation data that is independent of the training data.

Step 7: The algorithm terminates. If the termination condition is satisfied or the prediction error is obtained, then the algorithm will terminate.

V. Experimental Results

In order to assess the proposed improver ANFIS-based ICA and LSE hybrid learning algorithm, comprehensive experiments and simulations based on three nonlinear and chaotic benchmark time series were conducted. It must be noticed that, in all case studies, amount the last 30% of the training data is utilized as the validation data to select best model structure. The validation data are not utilized during the training, and the model with lowest validation error is selected as the best model, which is applied to predict test data. For numerical assessment of the prediction accuracy, the following error criteria are applied.

1. Root-mean-square error (RMSE)
2. Normalized mean-square error (NMSE)

$$RMSE = \sqrt{\frac{1}{N}\sum_{n=1}^{N}(y(n)-\hat{y}(n))^2} \tag{11}$$

$$NMSE = \sum_{n=1}^{N}\frac{(y(n)-\hat{y}(n))^2}{(y(n)-\bar{y})^2},\quad \bar{y} = \frac{1}{N}\sum_{n=1}^{N}y(n) \tag{12}$$

Where $y(n)$ and $\hat{y}(n)$ are the actual and forecasted outputs at sample n, respectively, and N is the number of forecasted samples. In all cases, the 50% of data are used as the training set and 20% of data are used as the validating set and the 30% are employed as test series.

A. Prediction of Mackey–Glass Time Series

Mackey–Glass time series is generated using the following time-delay differential equation, introduced as a model for white blood cell production:

$$\frac{dx}{dt} = \frac{0.2x(t-\lambda)}{1+x^{10}(t-\lambda)} - 0.1x(t) \tag{13}$$

The time series is very sensitive to the initial conditions and this time series is chaotic for $\lambda \geq 16.8$ and exhibits no clearly defined period. The standard input variables in this case for phase-space reconstruction are $\tau = 6$ and $D = 4$ are determined by using the average mutual information [35], and the global false nearest neighbors [1], respectively. Therefore, the phase space is reconstructed as $x(t-18), x(t-12), x(t-6)$ and $x(t)$ for predicting $x(t+6)$ (i.e., this is a case of six-step-ahead prediction). The proposed model for prediction this time series trained with the parameters listed in Table I. For this case study, convergence diagram shown in Fig. 6. The actual and predicted test series is shown in Fig. 7 and prediction error in test time series is shown in Fig. 8.

TABLE I. The parameters of the proposed method

Parameters	*Type or Value*
Clusters	20
Countries	50
Empires	10
Epochs	200
α	2
β	2
Revolution Probability	0.2
Revolution Rate (μ)	0.05
ξ	0.01

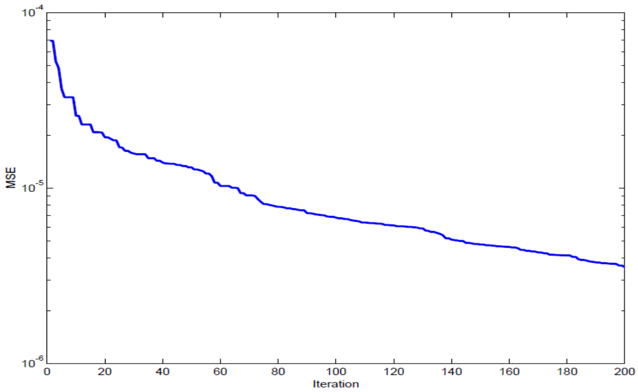

Fig. 6 Convergence Diagram for the Mackey-Glass Time Series

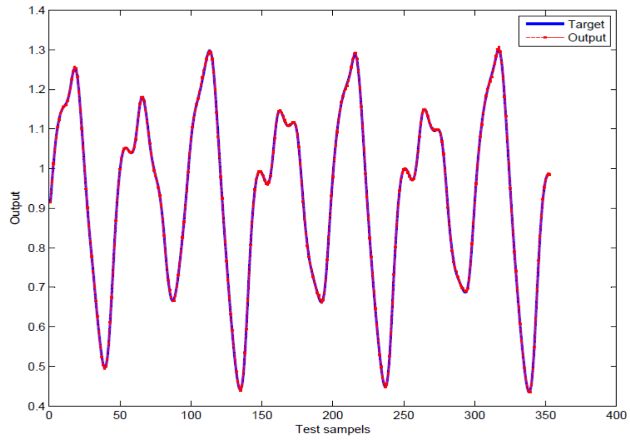

Fig. 7 Target and Predicted Test Series for the Mackey–Glass Time Series

In order to numerical evaluation and for the purpose of comparison, the training and test RMSEs of the proposed approach and a number of methods available in the literature are presented in Table II.

Published By:
Blue Eyes Intelligence Engineering
& Sciences Publication Pvt. Ltd.

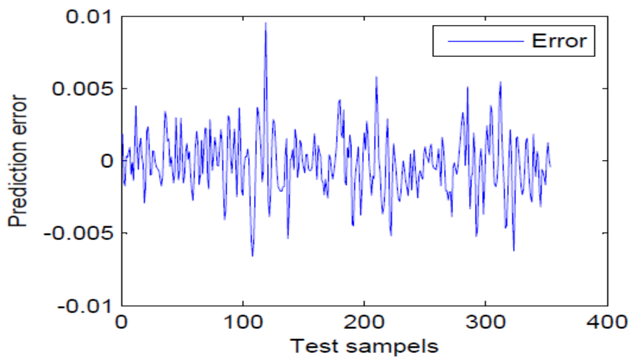

Fig. 8 Prediction Error for the Mackey–Glass Time Series

Based on the presented results in Table II, the proposed approach has much better performance over all other compared methods for both training and test data.

TABLE II. PERFORMANCE COMPARISON FOR MACKEY–GLASS TIME SERIES PREDICTION

Method	*RMSE Train*	*RMSE Test*
Auto-regressive model [36]	-	0.19
Cascade correlation NN [36]	-	0.06
Back propagation NN [36]	-	0.02
Linear prediction method [36]	-	0.55
Product T-norm [37]	-	0.09
Classical RBF (with 23 neurons) [38]	-	0.0114
PG-RBF network [39]	-	0.0028
Genetic algorithm and fuzzy system [40]	-	0.049
Neural tree model [41]	-	0.0069
WNN [37]+gradient	0.0067	0.0071
LLWNN [36]+gradient	0.0038	0.0041
LLWNN [36]+hybrid	0.0033	0.0036
Recurrent ANFIS [42]	-	0.0013
ANFIS [24]	-	0.00156
RBF network [43]	-	0.0015
Ensembles of ANFIS [44]	-	0.0025
Fuzzy prediction based on SVD [10]	-	0.012
LS-SVM based on PSO [45]	-	0.0028
LLNF[46]	0.0013	0.0020
Proposed Approach	0.0011	0.0011

A. Prediction of Lorenz Model

The Lorenz model is an example of a nonlinear dynamic system corresponding to the long-term behavior of the Lorenz oscillator. The Lorenz model is expressed by the following mathematical equation:

$$\begin{cases} \dot{x} = a(y - x) \\ \dot{y} = \beta x - zx - y \\ \dot{z} = xy - yz \end{cases} \quad (14)$$

Where x, y and z are real functions of time and $\alpha = 10, \gamma = 8/3, \beta > 24.74$. The standard input variables in this case for phase-space reconstruction are $\tau = 3$ and $D = 3$. Thus, the phase space is reconstructed as $x(t-6), x(t-3), x(t)$ for predicting $x(t-3)$ (i.e., this is a case of tree-step-ahead prediction). The proposed model for prediction this time series trained with the parameters listed in Table I. Convergence diagram shown in Fig. 9. The actual and predicted test series are shown in Fig. 10 and prediction error in test time series are shown in Fig. 11.

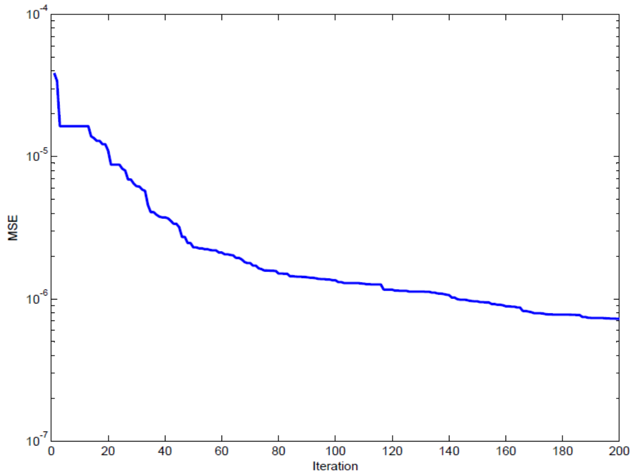

Fig. 9 Convergence Diagram for the Lorenz Time Series

In order of numerical evaluation and for the purpose of comparison, the training and test NMSE of the proposed approach and a number of methods available in the literature are presented in Table III. Based on the presented results in Table III, the proposed approach has much better performance over all other compared methods for both training and test data.

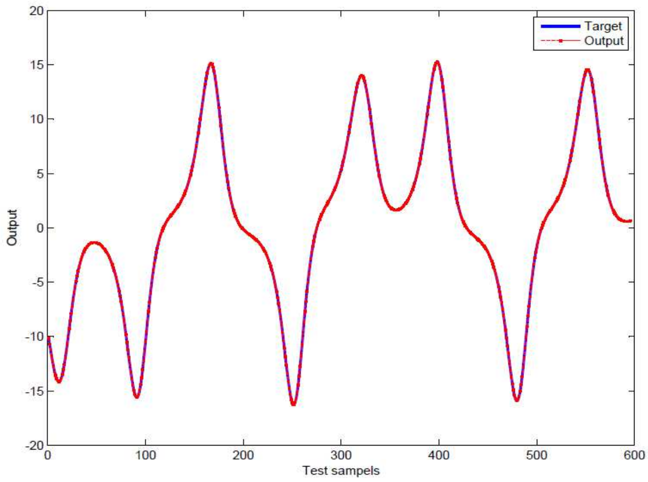

Fig. 10 Target and Predicted Test Series for the Lorenz Time Series

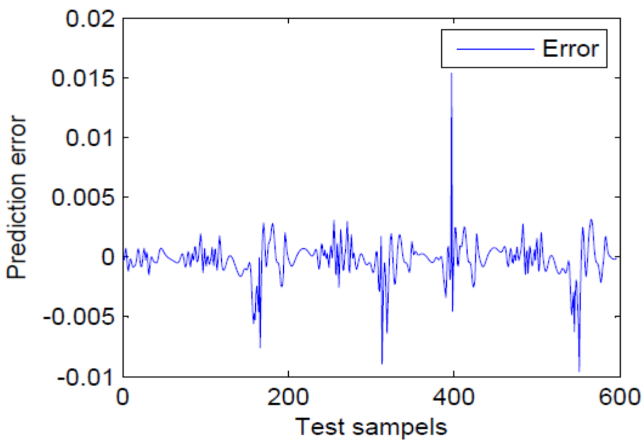

Fig. 11 Prediction Error for the Mackey–Glass Time Series

Published By:
Blue Eyes Intelligence Engineering & Sciences Publication Pvt. Ltd.

TABLE III. PERFORMANCE COMPARISON FOR LORENZ MODEL TIME SERIES PREDICTION

Method	*NMSE Train*	*NMSE Test*
MLP-EKF [47]	0.00023	0.00162
MLP-BLM [47]	0.00033	0.00096
RNN-BPTT [47]	0.00056	0.00185
RNN-RTRL [47]	0.00057	0.00172
RNN-EKF [47]	0.00036	0.00121
RBLM-RNN [47]	0.00036	0.00090
ANFIS[24]	0.0026	0.0021
Fuzzy prediction based on SVD [10]	-	0.0106
LS-SVM based on PSO [45]	-	0.00018
LLNF[46]	0.00013	0.00029
Proposed Approach	0.00007	0.00012

B. Prediction of Rossler Model

Rossler model is a system of three ordinary differential equations which define a continuous dynamical system that exhibits chaotic dynamics associated with the fractal properties of the Rossler attractor. This map is depicted by the following differential equations:

$$\begin{cases} \dot{x} = -y - z \\ \dot{y} = x - ay \\ \dot{z} = b + z(x - c) \end{cases} \tag{15}$$

Where x, y and z real functions of time and the attractor are shows chaotic behavior for $a = 0.2$, $b = 0.2$ and c=4.6. The standard input variables in this case for phase-space reconstruction are $\tau = 2$ and $D = 3$. Thus, the phase space is reconstructed as $x(t-4), x(t-2), x(t)$ for predicting $x(t+2)$ (i.e., this is a case of tow-step-ahead prediction). The proposed model for prediction this time series trained with the parameters listed in Table I. Convergence diagram shown in Fig. 12. The target and predicted test data of Rossler model are shown in Fig. 13, and prediction error in test time series are shown in Fig. 14. In order to numerical evaluation and for the purpose of comparison, the training and test NMSE of the proposed approach and a number of methods available in the literature are presented in Table IV. Results that presented in this table demonstrating successful performance of the proposed hybrid approach in capturing and modeling the dynamical behavior of this nonlinear chaotic system.

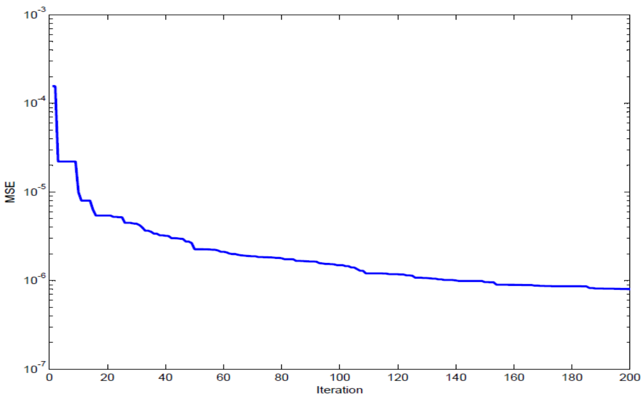

Fig. 12 Convergence Diagram for the Rossler Time Series

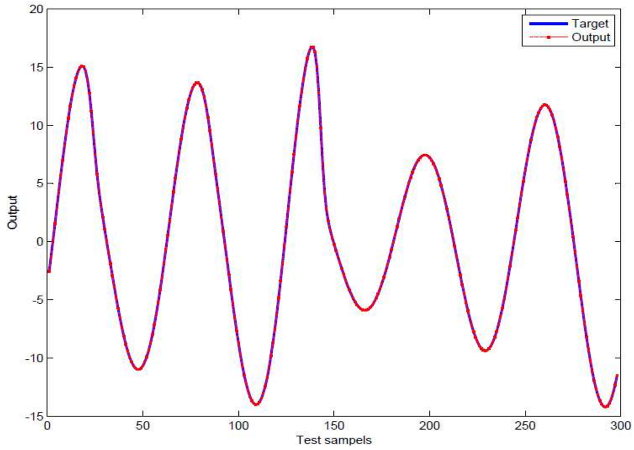

Fig. 13 Target and Predicted Test Series for the Rossler Time Series

TABLE IV. PERFORMANCE COMPARISON FOR ROSSLER MODEL TIME SERIES PREDICTION

Method	*NMSE Train*	*NMSE Test*
MLP-EKF [47]	0.00025	0.00193
MLP-BLM [47]	0.00047	0.00101
RNN-BPTT [47]	0.00070	0.00311
RNN-RTRL [47]	0.00071	0.00312
RNN-EKF [47]	0.00060	0.00191
RBLM-RNN [47]	0.00057	0.00092
ANFIS[24]	0.0147	0.0118
LLNF[24]	0.000048	0.000071
Proposed Approach	0.000083	0.00010

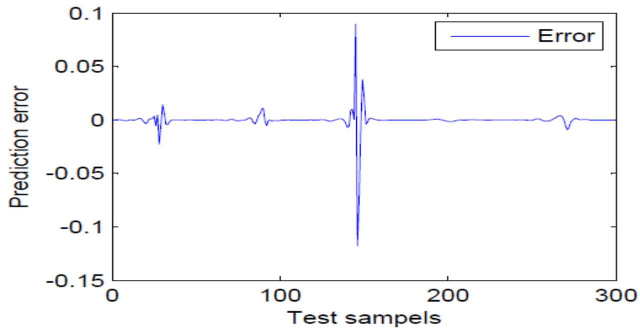

Fig. 14 Prediction Error for the Rossler Time Series

VI. CONCLUSION

This paper proposes an improved version of ANFIS with imperialist competitive learning algorithm for chaotic time series prediction. This version of ANFIS improves the ability of ANFIS in learning parameters with utilize a new hybrid learning algorithm for training its parameters. This hybrid learning algorithm used ICA for updating the antecedent parameters and LSE for estimating consequent parameters in an iterative process. This new learning algorithm has comparable performance with less parameters than gradient based methods which means that the new approach demonstrates high capability in the learning process. Also the complexity of this new algorithm is less than the other training algorithms and free of derivation and with this algorithm the local optimum problem in the gradient based algorithm for training the antecedent part is solved.

Published By:
Blue Eyes Intelligence Engineering
& Sciences Publication Pvt. Ltd.

The proposed approach was used for modeling and predicting a wide range of different nonlinear time series and chaotic systems and processes. The investigated time series included Mackey–Glass time series, Lorenz model, and Rossler model. The first series are recognized as real-world nonlinear time series, while the last two are known to be dynamical systems with chaotic behaviors. The obtained prediction results and the comprehensive comparisons with some recently published studies revealed the outstanding performance of the proposed approach in modeling and prediction of nonlinear, chaotic, and complex time series and processes.

REFERENCES

[1] Abarbanel, H. D. I. Analysis of Observed Chaotic Data. New York: Springer, 1996.

[2] Kantz, Holger, and Thomas Schreiber. Nonlinear Time Series Analysis. Cambridge: Cambridge University Press, 1997. Print.

[3] Box, George E. P., Gwilym M. Jenkins, and Gregory C. Reinsel. Time Series Analysis: Forecasting and Control. Englewood Cliffs, N.J.: Prentice Hall, 1994.

[4] De Gooijer, J.G., and R.J. Hyndman. 2006. "25 Years of Time Series Forecasting". International Journal of Forecasting. 22, no. 3: 443-473.

[5] Bodyanskiy, Y., and O. Vynokurova. 2013. "Hybrid Adaptive Wavelet-Neuro-Fuzzy System for Chaotic Time Series Identification". Information Sciences. 220: 170-179.

[6] Hsu, C.F. 2011. "Adaptive Fuzzy Wavelet Neural Controller Design for Chaos Synchronization". Expert Systems With Applications. 38, no. 8: 10475-10483.

[7] Henry Leung, H.C., H.C. Titus Lo, and H.C. Sichun Wang. 2001. "Prediction of noisy chaotic time series using an optimal radial basis function neural network". IEEE Transactions on Neural Networks. 12, no. 5: 1163-1172.

[8] Han, M., J. Xi, S. Xu, and F.-L. Yin. 2004. "Prediction of Chaotic Time Series Based on the Recurrent Predictor Neural Network". IEEE Transactions on Signal Processing. 52, no. 12: 3409-3416.

[9] Lee, C-H L, A Liu, and W-S Chen. 2006. "Data Mining - Pattern Discovery of Fuzzy Time Series for Financial Prediction". IEEE Transactions on Knowledge and Data Engineering. 18, no. 5: 613.

[10] Gu, H., and H. Wang. 2007. "Fuzzy Prediction of Chaotic Time Series Based on Singular Value Decomposition". Applied Mathematics and Computation. 185, no. 2: 1171-1185.

[11] Ang, K.K., and C. Quek. 2006. "Stock Trading Using RSPOP: A Novel Rough Set-Based Neuro-Fuzzy Approach". IEEE Transactions on Neural Networks. 17, no. 5: 1301-1315.

[12] Zaheeruddin and Garima. 2006. "A Neuro-Fuzzy Approach for Prediction of Human Work Efficiency in Noisy Environment". Applied Soft Computing. 6, no. 3: 283-294.

[13] Aznarte M., J.L., J.M. Benitez Sanchez, D.N. Lugilde, C. de Linares Fernandez, C.D. de la Guardia, and F.A. Sanchez. 2007. "Forecasting Airborne Pollen Concentration Time Series with Neural and Neuro-Fuzzy Models". Expert Systems With Applications. 32, no. 4: 1218-1225.

[14] Zhang J., Chung H.S.-H., and 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, no. 7: 956-964.

[15] Samanta, B. 2011. "Prediction of Chaotic Time Series Using Computational Intelligence". Expert Systems With Applications. 38, no. 9: 11406-11411.

[16] Gromov, Vasilii A., and Artem N. Shulga. 2012. "Chaotic Time Series Prediction with Employment of Ant Colony Optimization". Expert Systems With Applications. 39, no. 9: 8474-8478.

[17] Niu, D., Y. Wang, and D.D. Wu. 2010. "Power Load Forecasting Using Support Vector Machine and Ant Colony Optimization". Expert Systems With Applications. 37, no. 3: 2531-2539.

[18] Wang, Jianzhou, Dezhong Chi, Jie Wu, and Hai-yan Lu. 2011. "Chaotic Time Series Method Combined with Particle Swarm Optimization and Trend Adjustment for Electricity Demand Forecasting". Expert Systems with Applications. 38, no. 7: 8419-8429.

[19] Zhao L., and Yang Y. 2009. "PSO-Based Single Multiplicative Neuron Model for Time Series Prediction". Expert Systems with Applications. 36, no. 2 PART 2: 2805-2812.

[20] Mandic, Danilo P., and Jonathon A. Chambers. Recurrent Neural Networks for Prediction Learning Algorithms, Architectures, and Stability. Chichester: John Wiley, 2001

[21] Goldberg, David E. Genetic Algorithms in Search, Optimization, and Machine Learning. Reading, Mass: Addison-Wesley Pub. Co, 1989.

[22] 23-Atashpaz-Gargari, E.,Lucas. 2007. "Designing an Optimal PID Controller using Colonial Competitive Algorithm". Proceedings of the First Iranian Joint Congress on Fuzzy and Intelligent Systems Mashhad,Iran.

[23] Atashpaz-Gargari, E.,Lucas,C. 2007. "Imperialist Competitive Algorithm:An Algorithm for Optimization Inspired by Imperialists Competition". Proceedings of the IEEE Congress on Evolutionary Computation Singapore. 4661–4667.

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

[25] Mascioli, F.M., Varazi, G.M. and Martinelli, G. 1997. "Constructive Algorithm for Neuro-Fuzzy Networks". Proceedings of the Sixth IEEE International Conference on Fuzzy Systems. 459 –464.

[26] Jang, J.-S. R., and Mizutani, E. 1996. "Levenberg-Marquardt Method for ANFIS Learning". Biennial Conference of the North American Fuzzy Information Processing Society. 87 –91.

[27] Jang, J.-S.R. 1996. "Input Selection for ANFIS Learning". Proceedings of the Fifth IEEE International Conference on Fuzzy Systems. 1493 –1499.

[28] Manish Kumar, Devendra P. Garg. 2004. "Intelligent Learning of Fuzzy Logic Controllers via Neural Network and Genetic Algorithm". Proceedings of 2004 JUSFA 2004 Japan – USA Symposium on Flexible Automation Denver Colorado.

[29] C. Juang, Y. Liou. 2004."TSK-Type Recurrent Fuzzy Network Design by the Hybrid of Genetic Algorithm and Particle Swarm Optimization". IEEE International Conference on Systems.

[30] Juang, C. F. 2002. "A TSK-type Recurrent Fuzzy Network for Dynamic Systems Processing by Neural Network and Genetic Algorithm". IEEE Transactions on Fuzzy Systems. 10, no. 2:155-170.

[31] M. Aliyari. Sh, M. Teshnehlab, A. K. Sedigh. 2006. "A Novel Training Algorithm in ANFIS Structure". American Control Conf.

[32] Aliyari Shoorehdeli, Mahdi, Mohammad Teshnehlab, and Ali Sedigh. 2009. "Identification Using ANFIS with Intelligent Hybrid Stable Learning Algorithm Approaches". Neural Computing and Applications. 18, no. 2: 157-174.

[33] Aliyari. Sh, M. Teshnehlab, A. K. Sedigh. 2007. "A Novel Hybrid Learning Algorithm for Tuning ANFIS Parameters using Adaptive Weighted PSO". IEEE Int Fuzzy Sys Conf.

[34] F.Takens. 1981. "Detecting Strange Attractors in Turbulence". in: lecture notes in mathematics 898 :366-400.

[35] Alparslan, A. K., M. Sayar, and A. R. Atilgan. 1998. "State-Space Prediction Model for Chaotic Time Series". Physical Review -Series E-. 58, no. 2/B: 2640-2643.

[36] Chen, Y., B. Yang, and J. Dong. 2006. "Time-Series Prediction Using a Local Linear Wavelet Neural Network". Neurocomputing. 69, no. 4-6: 449-465.

[37] L.X. Wang, J.M. Mendel. 1992. "Generating fuzzy rules by learning from examples". IEEE Transactions on Systems, Man and Cybernetics. 22, no. 6: 1414 - 1427 .

[38] Cho, Kwang Bo, and Bo Hyeun Wang. 1996. "Radial Basis Function Based Adaptive Fuzzy Systems and Their Applications to System Identification and Prediction". Fuzzy Sets and Systems. 83, no. 3: 325-339.

[39] Rojas, I., H. Pomares, J.L. Bernier, J. Ortega, B. Pino, F.J. Pelayo, and A. Prieto. 2002. "Time Series Analysis Using Normalized PG-RBF Network with Regression Weights". Neurocomputing. 42, no. 1-4: 267-285.

[40] Daijin Kim, P.D., and P.D. Chulhyun Kim. 1997. "Forecasting time series with genetic fuzzy predictor ensemble". IEEE Transactions on Fuzzy Systems. 5, no. 4: 523-535.

[41] Y. Chen, B. Yang, and J. Dong. 2004. "Nonlinear system modeling via optimal design of neural trees". International Journal of Neural Systems. 14, no. 02: 125-137.

[42] H. Tamura, K. Tanno, H. Tanaka, C. Vairappan, and Z. Tang. 2008. "Recurrent type ANFIS using local search technique for time series prediction". IEEE Asia Pacific Conf. Circuits Syst. 380–383.

[43] Harpham, C., and C. W. Dawson. 2006. "The Effect of Different Basis Functions on a Radial Basis Function Network for Time Series Prediction: A Comparative Study". Neurocomputing . 69, no. 16-18: 2161-2170.

Published By:
Blue Eyes Intelligence Engineering
& Sciences Publication Pvt. Ltd.

[44] Melin, Patricia, Jesus Soto, Oscar Castillo, and Jose Soria. 2012. "A New Approach for Time Series Prediction Using Ensembles of ANFIS Models". Expert Systems With Applications. 39, no. 3: 3494-3506.
[45] Ping Liu, Jian Yao. 2009. "Application of Least Square Support Vector Machine based on Particle Swarm Optimization to Chaotic Time Series Prediction". Intelligent Computing and Intelligent Systems, ICIS 2009. IEEE International Conference on (4).
[46] M. M. R. Yousefi, M. Mirmomeni, and C. Lucas. 2007. "Input Variables Selection Using Mutual Information for Neuro Fuzzy Modeling with The Application to Time Series Forecasting". in Proc. Int. Joint Conf. neural Network. 1121-1126.
[47] Mirikitani DT, and N Nikolaev. 2010. "Recursive Bayesian Recurrent Neural Networks for Time-Series Modeling". IEEE Transactions on Neural Networks / a Publication of the IEEE Neural Networks Council. 21, no. 2: 262-74.
[48] Haykin, Simon S. Neural Networks: A Comprehensive Foundation. New York: Macmillan, 1994. Print.

Published By:
Blue Eyes Intelligence Engineering
& Sciences Publication Pvt. Ltd.