

# Particle Swarm Optimization of the Fuzzy Integrators for Time Series Prediction Using Ensemble of IT2FNN Architectures

Jesus Soto, Patricia Melin and Oscar Castillo

**Abstract** This paper describes the construction of intelligent hybrid architectures and the optimization of the fuzzy integrators for time series prediction; interval type-2 fuzzy neural networks (IT2FNN). IT2FNN used hybrid learning algorithm techniques (gradient descent backpropagation and gradient descent with adaptive learning rate backpropagation). The IT2FNN is represented by Takagi–Sugeno–Kang reasoning. Therefore this TSK IT2FNN is represented as an adaptive neural network with hybrid learning in order to automatically generate an interval type-2 fuzzy logic system (TSK IT2FLS). We use interval type-2 and type-1 fuzzy systems to integrate the output (forecast) of each Ensemble of ANFIS models. Particle Swarm Optimization (PSO) was used for the optimization of membership functions (MFs) parameters of the fuzzy integrators. The Mackey-Glass time series is used to test of performance of the proposed architecture. Simulation results show the effectiveness of the proposed approach.

**Keywords** Time series · IT2FNN · Particle swarm optimization · Fuzzy integrators

## 1 Introduction

The analysis of the time series consists of a (usually mathematical) description of the movements that compose it, then building models using movements to explain the structure and predict the evolution of a variable over time [3, 4]. The fundamental procedure for the analysis of a time series is described below

---

J. Soto · P. Melin (✉) · O. Castillo  
Tijuana Institute of Technology, Tijuana, Mexico  
e-mail: pmelin@tectijuana.mx

J. Soto  
e-mail: jesvega83@gmail.com

O. Castillo  
e-mail: ocastillo@tectijuana.mx

1. Collecting data of the time series, trying to ensure that these data are reliable.
2. Representing the time series qualitatively noting the presence of long-term trends, cyclical variations, and seasonal variations.
3. Plot a graph or trend line length and obtain the appropriate trend values using the method of least squares.
4. When seasonal variations are present, obtain these and adjust the data rate to these seasonal variations (i.e., data seasonally).
5. Adjust the seasonally adjusted trend.
6. Represent the cyclical variations obtained in step 5.
7. Combining the results of steps 1–6 and any other useful information to make a prediction (if desired) and if possible discuss the sources of error and their magnitude.

Therefore the above ideas can assist in the important problem of prediction in the time series. Along with common sense, experience, skill and judgment of the researcher, such mathematical analysis can, however, be of value for predicting the short, medium, and long term.

As related work we can mention: Type-1 Fuzzy Neural Network (T1FNN) [15, 18, 19, 29] and Interval Type-2 Fuzzy Neural Network (IT2FNN) [13, 23–25, 44]; type-1 [1, 8, 16, 33, 45] and type-2 [11, 35, 41, 47] fuzzy evolutionary systems are typical hybrid systems in soft computing. These systems combine T1FLS generalized reasoning methods [18, 28, 34, 42, 43, 48, 51] and IT2FLS [21, 30, 46] with neural networks learning capabilities [12, 14, 18, 37] and evolutionary algorithms [2, 5, 9–11, 29, 35–37, 41] respectively.

This paper reports the results of the simulations of three main architectures of IT2FNN (IT2FNN-1, IT2FNN-2 and IT2FNN-3) for integrating a first-order TSK IT2FIS, with real consequents (A2C0) and interval consequents (A2C1), are used. Integration strategies to process elements of TSK IT2FIS are analyzed for each architecture (fuzzification, knowledge base, type reduction, and defuzzification). Ensemble architectures have three choices IT2FNN-1, IT2FNN-2, and IT2FNN-3. Therefore the output of the Ensemble architectures are integrated with a fuzzy system and the MFs of the fuzzy systems are optimized with PSO. The Mackey-Glass time series is used to test the performance of the proposed architecture. Prediction errors are evaluated by the following metrics: root mean square error (RMSE), mean square error (MSE), and mean absolute error (MAE).

In the next section, we describe the background and basic concepts of the Mackey-Glass time series, Interval type-2 fuzzy systems, Interval Type-2 Fuzzy Neural-Networks, and Particle Swarm Optimization. Section 3 presents the general proposed architecture. Section 4 presents the simulations and the results. Section 5 offers the conclusions.

## 2 Background and Basic Concepts

This section presents the basic concepts that describe the background in time series prediction and basic concepts of the Mackey-Glass time series, Interval type-2 fuzzy systems, Interval Type-2 Fuzzy Neural-Networks, and Particle Swarm Optimization.

### 2.1 Mackey-Glass Time Series

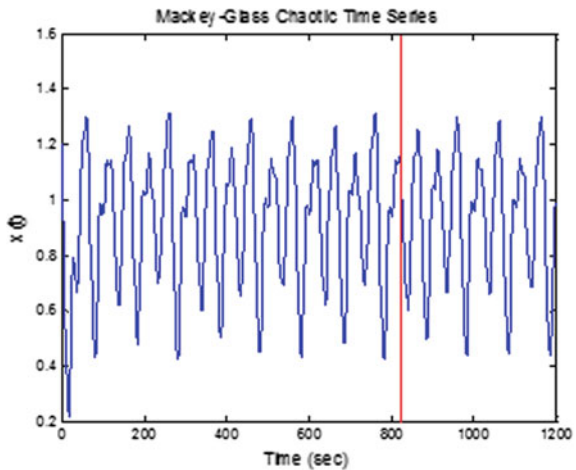
The problem of predicting future values of a time series has been a point of reference for many researchers. The aim is to use the values of the time series known at a point  $x = t$  to predict the value of the series at some future point  $x = t + P$ . The standard method for this type of prediction is to create a mapping from  $D$  points of a  $\Delta$  spaced time series, is  $(x(t - (D - 1)\Delta) \dots x(t - \Delta), x(t))$ , to a predicted future value  $x(t + P)$ . To allow a comparison with previous results in this work [11, 19, 29, 41] the values  $D = 4$  and  $\Delta = P = 6$  were used.

Chaotic time series data used is defined by the Mackey-Glass [26, 27] time series, whose differential equation is given by Eq. (1)

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

For obtaining the values of the time series at each point, we can apply the Runge–Kutta method [17] for the solution of Eq. (1). The integration step was set at 0.1, with initial condition  $x(0) = 1.2$ ,  $\tau = 17$ ,  $x(t)$  is then obtained for  $0 \leq t \leq 1200$ , (Fig. 1) (we assume  $x(t) = 0$  for  $t < 0$  in the integration).

**Fig. 1** The Mackey-Glass time series



## 2.2 Interval Type-2 Fuzzy Systems

Type-2 fuzzy sets are used to model uncertainty and imprecision; originally they were proposed by Zadeh [49, 50] and they are essentially “fuzzy–fuzzy” sets in which the membership degrees are type-1 fuzzy sets (Fig. 2).

The basic structure of a type-2 fuzzy system implements a nonlinear mapping of input to output space. This mapping is achieved through a set of type-2 if-then fuzzy rules, each of which describes the local behavior of the mapping.

The uncertainty is represented by a region called footprint of uncertainty (FOU). When  $\mu_{\tilde{A}}(x, u) = 1, \forall u \in I_x \subseteq [0, 1]$ ; we have an interval type-2 membership function [5, 7, 20, 31] (Fig. 3).

The uniform shading for the FOU represents the entire interval type-2 fuzzy set and it can be described in terms of an upper membership function  $\bar{\mu}_{\tilde{A}}(x)$  and a lower membership function  $\underline{\mu}_{\tilde{A}}(x)$ .

A fuzzy logic systems (FLS) described using at least one type-2 fuzzy set is called a type-2 FLS. Type-1 FLSs are unable to directly handle rule uncertainties, because they use type-1 fuzzy sets that are certain [6, 7, 46]. On the other hand, type-2 FLSs are very useful in circumstances where it is difficult to determine an exact certainty value, and there are measurement uncertainties.

## 2.3 Interval Type-2 Fuzzy Neural Networks (IT2FNN)

One way to build interval type-2 fuzzy neural networks (IT2FNN) is by fuzzifying a conventional neural network. Each part of a neural network (the activation function, the weights, and the inputs and outputs) can be fuzzified. A fuzzy neuron is

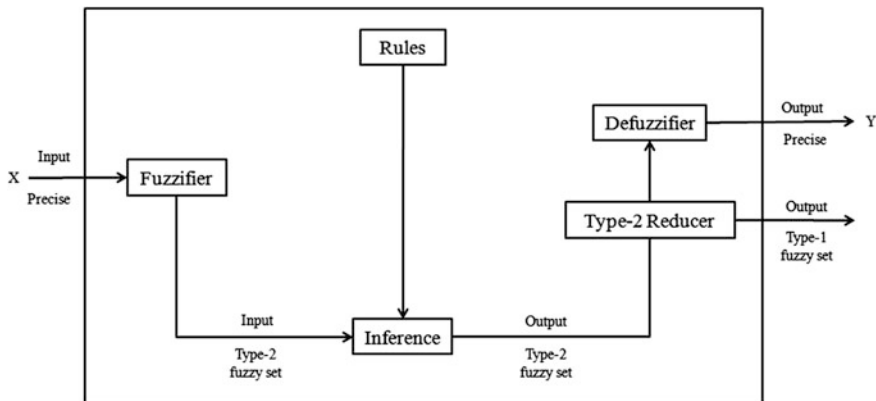
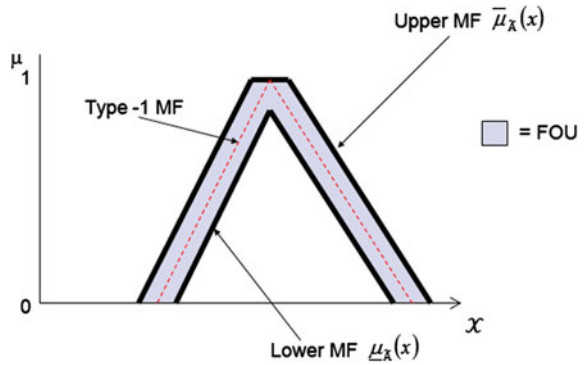


Fig. 2 Basic structure of the interval type-2 fuzzy logic system

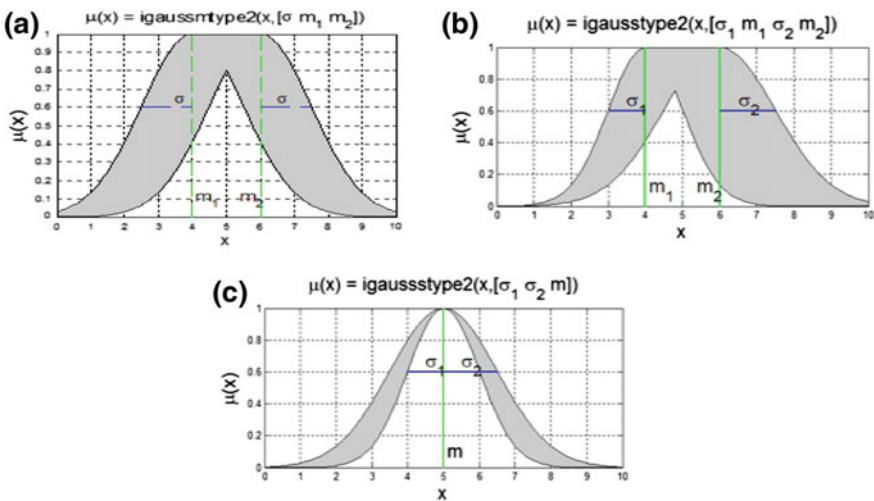
**Fig. 3** Interval type-2 membership function



basically similar to an artificial neuron, except that it has the ability to process fuzzy information.

The interval type-2 fuzzy neural network (IT2FNN) system is one kind of interval Takagi–Sugeno–Kang fuzzy inference system (IT2-TSK-FIS) inside neural network structure. An IT2FNN is proposed by Castro [6], with TSK reasoning and processing elements called interval type-2 fuzzy neurons (IT2FN) for defining antecedents, and interval type-1 fuzzy neurons (IT1FN) for defining the consequents of rules.

An IT2FN is composed by two adaptive nodes represented by squares, and two non-adaptive nodes represented by circles. Adaptive nodes have outputs that depend on their inputs, modifiable parameters, and transference function while non-adaptive, on the contrary, depend solely on their inputs, and their outputs represent lower  $\underline{\mu}_A(x)$  and upper  $\bar{\mu}_A(x)$  membership functions (Fig. 4).



**Fig. 4** The MFs used for training the IT2FNN architecture

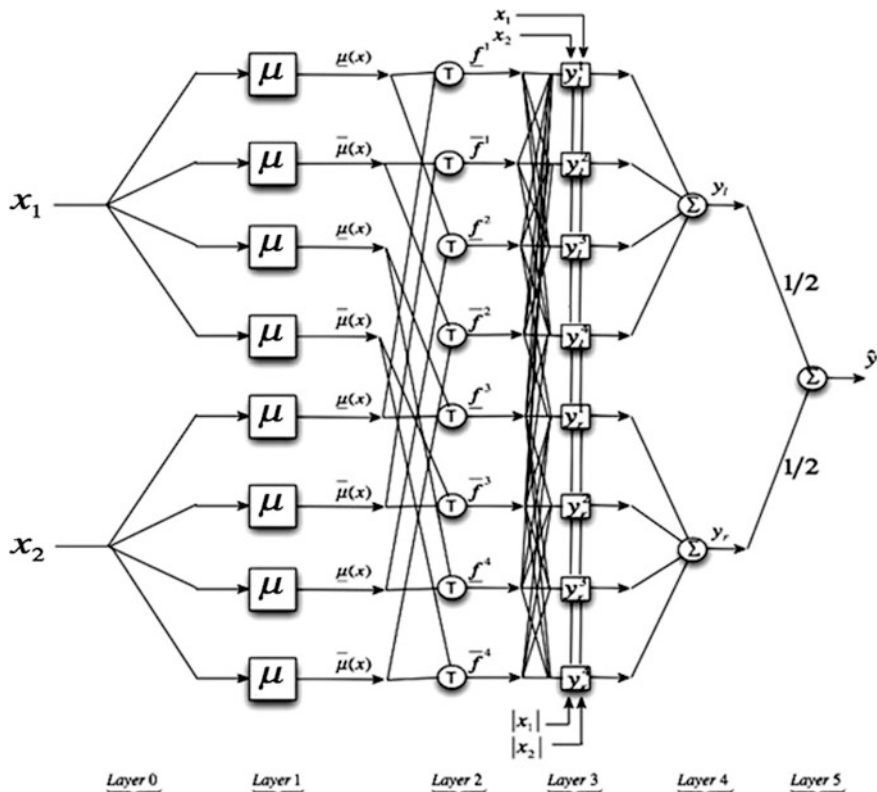


Fig. 5 IT2FNN1 architecture

The IT2FNN-1 architecture has five layers (Fig. 5), consists of adaptive nodes with an equivalent function to lower-upper membership in fuzzification layer (layer 1). Non-adaptive nodes in the rules layer (layer 2) interconnect with fuzzification layer (layer 1) in order to generate TSK IT2FIS rules antecedents. The adaptive nodes in consequent layer (layer 3) are connected to input layer (layer 0) to generate rules consequents. The non-adaptive nodes in type-reduction layer (layer 4) evaluate left-right values with KM algorithm [19–21]. The non-adaptive node in defuzzification layer (layer 5) average left-right values.

The IT2FNN-2 architecture has six layers (Fig. 6 and uses IT2FN for fuzzifying inputs (layers 1–2). The non-adaptive nodes in the rules layer (layer 2) interconnect with lower-upper linguistic values layer (layer 1) to generate TSK IT2FIS rules antecedents. The non-adaptive nodes in the consequents layer (layer 4) are connected to the input layer (layer 0) to generate rule consequents. The non-adaptive nodes in type-reduction layer (layer 5) evaluate left-right values with KM algorithm. The non-adaptive node in defuzzification layer (layer 6) averages left-right values.

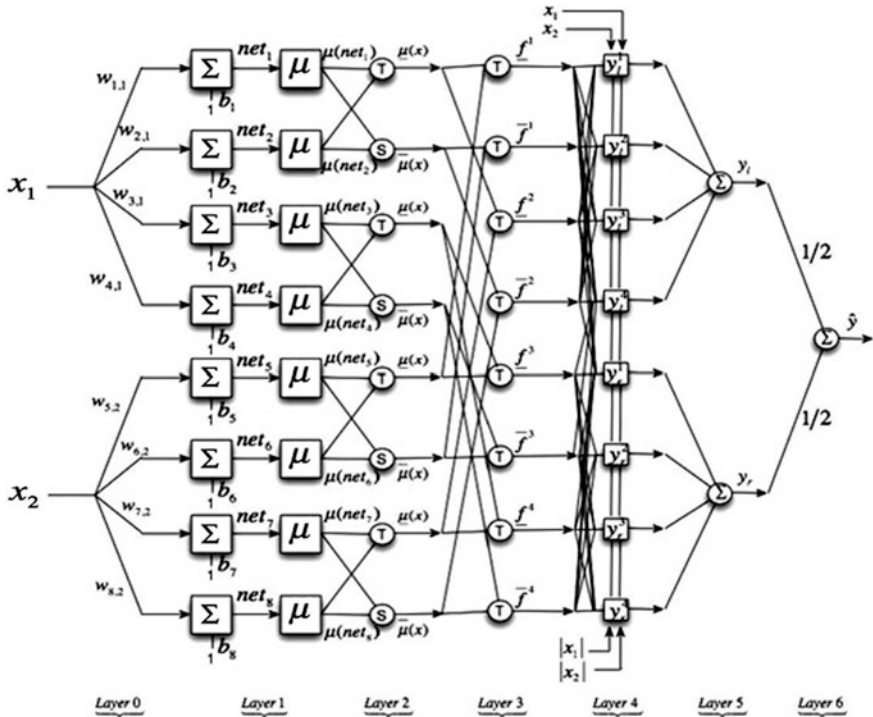


Fig. 6 ITFNN2 architecture

IT2FNN-3 architecture has seven layers (Fig. 7). Layer 1 has adaptive nodes for fuzzifying inputs; layer 2 has non-adaptive nodes with the interval fuzzy values. Layer 3 (rules) has non-adaptive nodes for generating firing strength of TSK IT2FIS rules. Layer 4, lower and upper values the rules firing strength are normalized. The adaptive nodes in layer 5 (consequent) are connected to layer 0 for generating the rules consequents. The non-adaptive nodes in layer 6 evaluate values from left-right interval. The non-adaptive node in layer 7 (defuzzification) evaluates average of interval left-right values.

### 2.4 Particle Swarm Optimization

Particle Swarm Optimization (PSO) is a metaheuristic search technique based on a population of particles (Fig. 8). The main idea of PSO comes from the social behavior of schools of fish and flocks of birds [22, 32]. In PSO, each particle moves

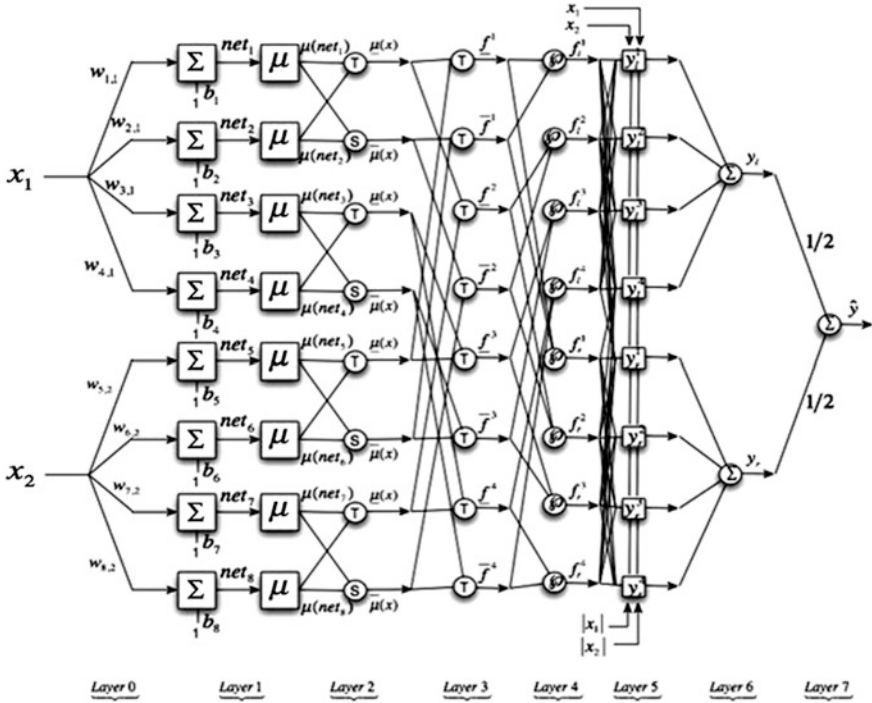


Fig. 7 IT2FNN3 architecture

in a D-dimensional space based on its own past experience and those of other particles. Each particle has a position and a velocity represented by the vectors  $x_i = (x_{i1}, x_{i2}, \dots, x_{iD})$  and  $V_i = (v_{i1}, v_{i2}, \dots, v_{iD})$  for the  $i$ -th particle. At each iteration, particles are compared with each other to find the best particle [32, 38]. Each particle records its best position as  $P_i = (p_{i1}, p_{i2}, \dots, p_{iD})$ . The best position of all particles in the swarm is called the global best, and is represented as  $G = (G_1, G_2, \dots, G_D)$ . The velocity of each particle is given by Eq. (2).

$$V_{id} = wv_{id} + C_1 \cdot \text{rand}_1() \cdot (pbest_{id} - x_{id}) + C_2 \cdot \text{rand}_2() \dots \cdot (gbest - x_{id}) \quad (2)$$

In this equation,  $i = 1, 2, \dots, M, d = 1, 2, \dots, D, C_1$  and  $C_2$  are positive constants (known as acceleration constants),  $\text{rand}_1()$  and  $\text{rand}_2()$  are random numbers in  $[0,1]$ , and  $w$ , introduced by Shi and Eberhart [39] is the inertia weight. The new position of the particle is determined by Eq. (3)

$$x_{id} = x_{id} + v_{id} \quad (3)$$



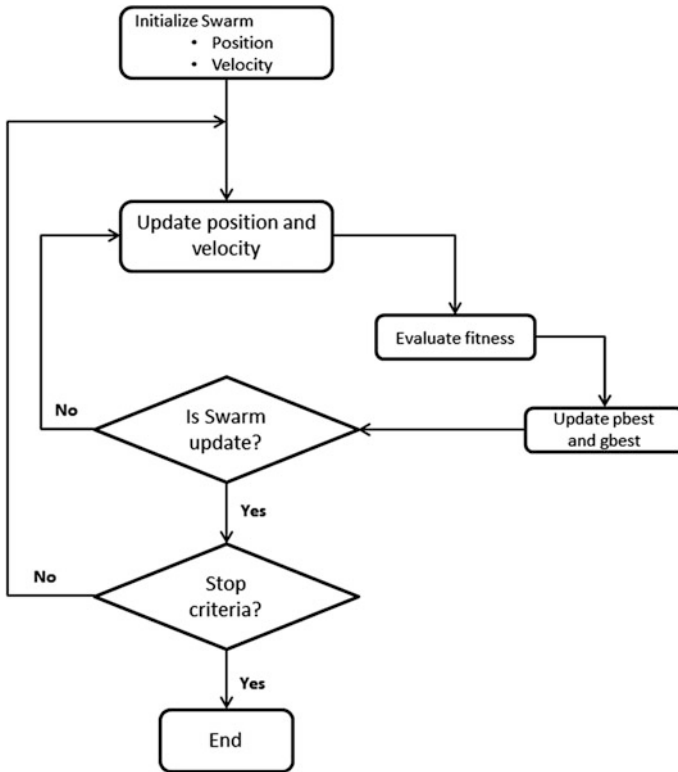


Fig. 8 Flowchart of the PSO algorithm

### 3 Problem Statement and Proposed Architecture

The general proposed architecture combines the ensemble of IT2FNN models and the use of fuzzy systems as response integrators using PSO for time series prediction (Fig. 9).

This architecture is divided into four sections, where the first phase represents the database to simulate in the Ensemble [40] of IT2FNN, which in this case is the historical data of the Mackey-Glass [26, 27] time series. From the Mackey-Glass time series we used 800 pairs of data points (Fig. 1), similar to [35, 36].

We predict  $x(t)$  from three past (delays) values of the time series, that is,  $x(t - 18)$ ,  $x(t - 12)$ , and  $x(t - 6)$ . Therefore the format of the training and checking data is

$$[x(t - 18), x(t - 12), x(t - 6); x(t)] \tag{4}$$

where  $t = 19-818$  and  $x(t)$  is the desired prediction of the time series.

In the second phase, training (the first 400 pairs of data are used to train the IT2FNN architecture) and validation (the second 400 pairs of data are used to

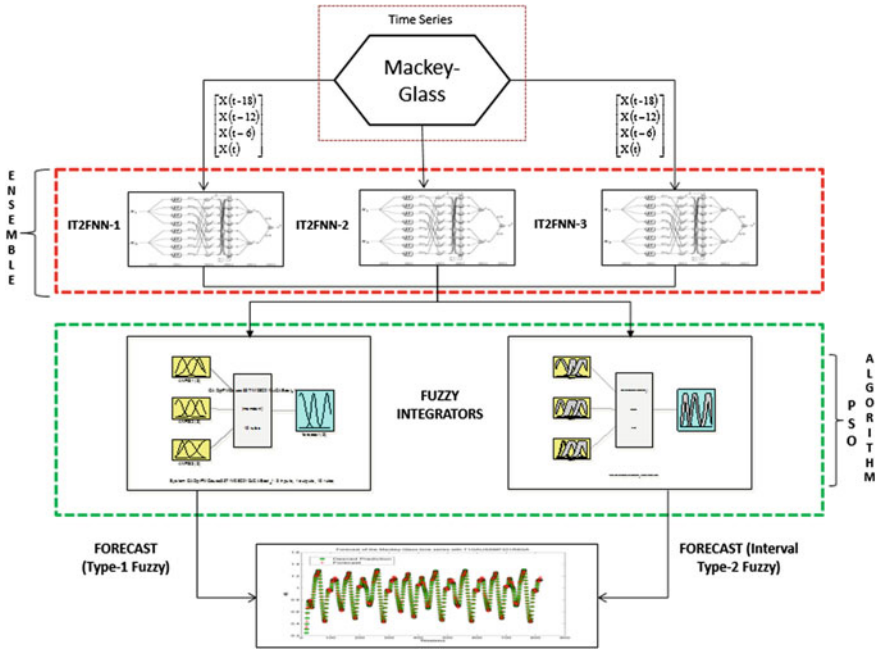


Fig. 9 The general proposed architecture

validate the ITFNN architecture) is performed sequentially in each IT2FNN, in this case we are dealing with a set of 3 IT2FNN (IT2FNN-1, IT2FNN-2, and IT2FNN-3) in the Ensemble. Therefore each IT2FNN architecture has three input variables ( $x(t - 18)$ ,  $x(t - 12)$ ,  $x(t - 6)$ ) and one output variable ( $x(t)$ ) is the desired prediction.

In the fourth phase, we integrate the overall results of each Ensemble of IT2FNN which are (IT2FNN-1, IT2FNN-2, and IT2FNN-3) architecture, and such integration will be done by the fuzzy inference system (type-1 and interval type-2 fuzzy system) of Mamdani type; but each fuzzy integrators will be optimized with PSO of the MFs parameters. Finally the forecast output determined by the proposed architecture is obtained and it is compared with desired prediction.

### 3.1 Design of the Fuzzy Integrators

The design of the type-1 and interval type-2 fuzzy inference systems integrators are of Mamdani type and have three inputs (IT2FNN1, IT2FNN2, and IT2FNN3) and one output (Forecast), so each input is assigned two MFs with linguistic labels “Small and Large” and the output will be assigned three MFs with linguistic labels “OutIT2FNN1, Out IT2FNN2 and Out IT2FNN3” (Fig. 10) and have eight if-then

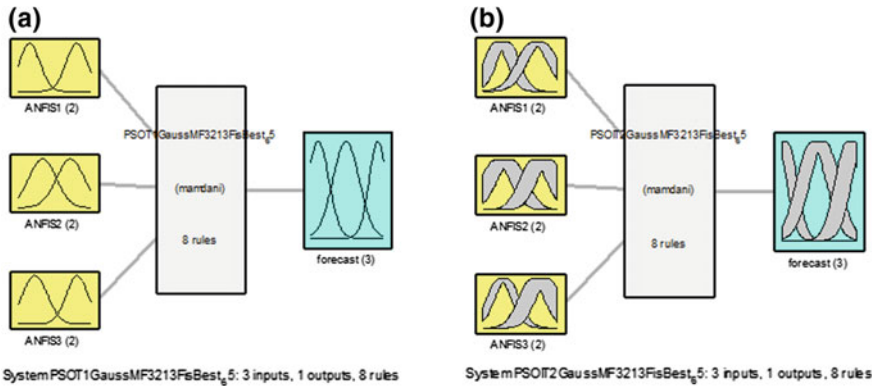


Fig. 10 Structure of the type-1 FIS (a) and interval type-2 FIS (b) integrators

rules. The design of the if-then rules for the fuzzy inference system depends on the number of membership functions used in each input variable using the system [e.g., our fuzzy inference system uses three input variables which each entry contains two membership functions, therefore the total number of possible combinations for the fuzzy rules is 8 (e.g.,  $2*2*2 = 8$ )], therefore we used eight fuzzy rules for the experiments (Fig. 11) because the performance is better and minimized the prediction error of the Mackey-Glass time series.

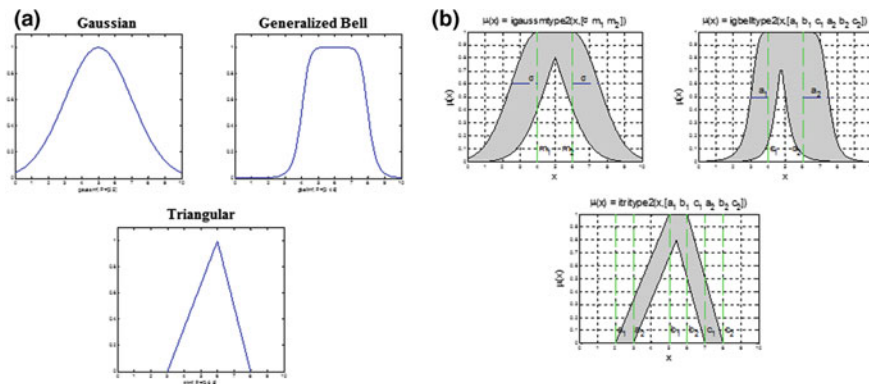
In the type-1 FIS integrators, we used different MFs (Gaussian, Generalized Bell, and Triangular) Fig. 12a and for the interval type-2 FIS integrators we used different MFs (igaussmtype2, igbelltype2, and itritype2) Fig. 12b [7] to observe the behavior of each of them and determine which one provides better forecast of the time series.

### 3.2 Design of the Representation for the Particle Swarm Optimization

The PSO is used to optimize the parameters values of the MFs in each of the type-1 and interval type-2 fuzzy integrators. The representation in PSO is of Real-Values

1. If (ANFIS1 is small) and (ANFIS2 is small) and (ANFIS3 is small) then (forecast is OutANFIS1) (1)
2. If (ANFIS1 is small) and (ANFIS2 is small) and (ANFIS3 is large) then (forecast is OutANFIS1) (1)
3. If (ANFIS1 is small) and (ANFIS2 is large) and (ANFIS3 is small) then (forecast is OutANFIS2) (1)
4. If (ANFIS1 is small) and (ANFIS2 is large) and (ANFIS3 is large) then (forecast is OutANFIS2) (1)
5. If (ANFIS1 is large) and (ANFIS2 is small) and (ANFIS3 is small) then (forecast is OutANFIS2) (1)
6. If (ANFIS1 is large) and (ANFIS2 is small) and (ANFIS3 is large) then (forecast is OutANFIS2) (1)
7. If (ANFIS1 is large) and (ANFIS2 is large) and (ANFIS3 is small) then (forecast is OutANFIS3) (1)
8. If (ANFIS1 is large) and (ANFIS2 is large) and (ANFIS3 is large) then (forecast is OutANFIS3) (1)

Fig. 11 If-then rules for the fuzzy integrators



**Fig. 12** Type-1 MFs (a) and interval type-2 MFs (b) for the fuzzy integrators

and the particle size will depend on the number of MFs that are used in each design of the fuzzy integrators.

The objective function is defined to minimize the prediction error as follows in Eq. (5)

$$f(t) = \sqrt{\frac{\sum_{t=1}^n (a_t - p_t)^2}{n}} \tag{5}$$

where  $a_t$ , corresponds to the real data of the time series,  $p_t$  corresponds to the output of each fuzzy integrators,  $t$  is de sequence time series, and  $n$  is the number of data points of time series.

The general representation of the particles represents the utilized membership functions. The number of parameters varies according to the kind of membership function of the type-1 fuzzy system (e.g., two parameter are needed to represent a Gaussian MF's are "sigma and mean") Fig. 13a and interval type-2 fuzzy system (e.g., three parameter are needed to represent "igaussmtype2" MF's are "sigma, mean1 and mean2") Fig. 13b. Therefore the number of parameters that each fuzzy inference system integrator has depends of the MFs type assigned to each input and output variables.

The parameters of particle swarm optimization used for optimizing the type-1 and interval type-2 fuzzy inference systems integrators are shown on Table 1.

We performed experiments in time series prediction, specifically for the Mackey-Glass time series in ensembles of IT2FNN architectures using fuzzy integrators optimized with PSO.

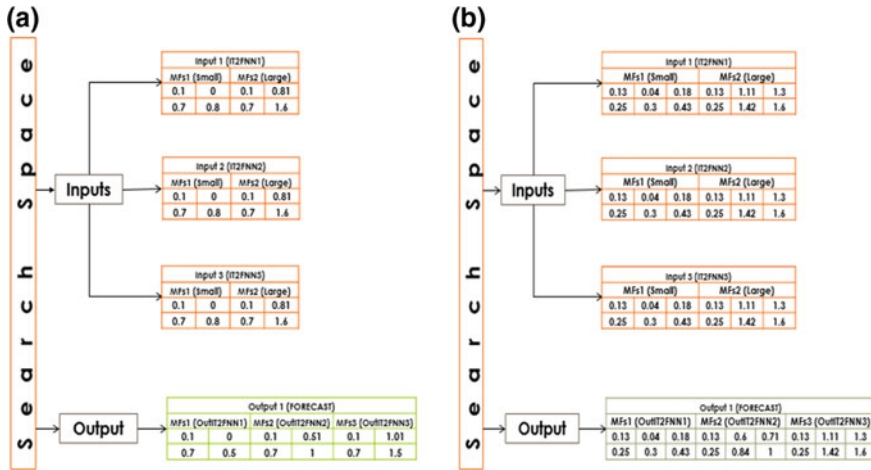


Fig. 13 Representation of the particles structure of the type-1 (a) and interval type (b) fuzzy integrators

Table 1 Parameters of PSO

Parameters	Value
Particles	100
Iterations	65
Inertia Weight “ $\omega$ ”	Linear decrement [0.88–0]
Constriction “ $C$ ”	Linear increment [0.01–0.9]
$r1, r2$	Random
$c1$	Linear decrement [2–0.5]
$c2$	Linear increment [0.5–2]

### 4 Simulations Results

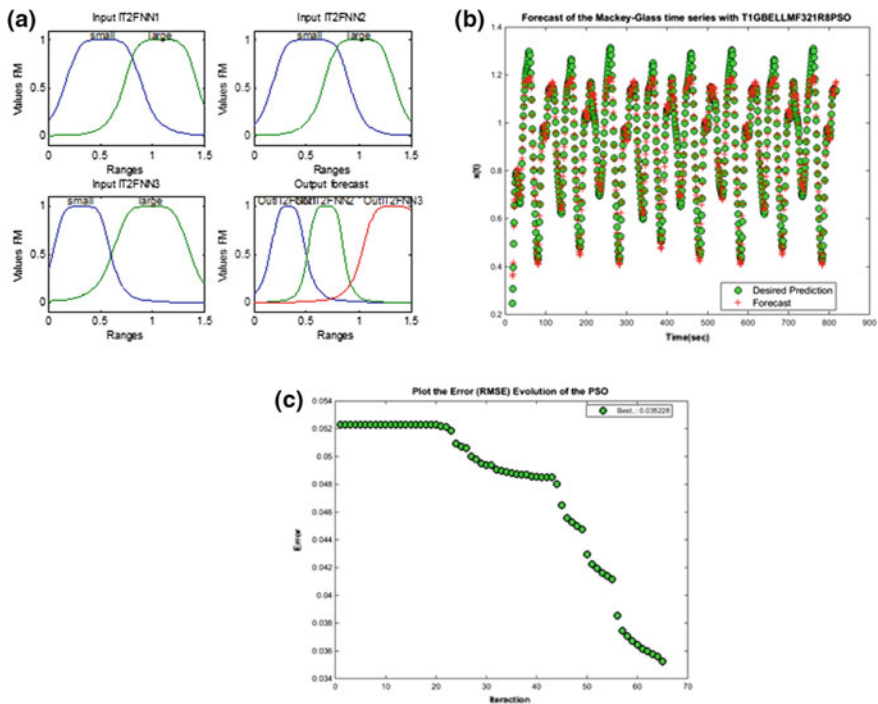
This section presents the results obtained through experiments on the architecture for the optimization of the fuzzy integrators in ensembles of IT2FNN architectures for time series prediction, which show the performance that was obtained from each experiment to simulate the Mackey-Glass time series.

The best errors were produced by the type-1 fuzzy integrator (using Generalized Bell MFs) with PSO are shown on Table 2. The RMSE is 0.035228102 and the average RMSE is 0.047356657, the MSE is 0.005989357 and the MAE is 0.056713089, respectively. The MFs optimized with PSO are presented in Fig. 14a, the forecasts in Fig. 14b, and the evolution errors in Fig. 14c are obtained for the proposed architecture.

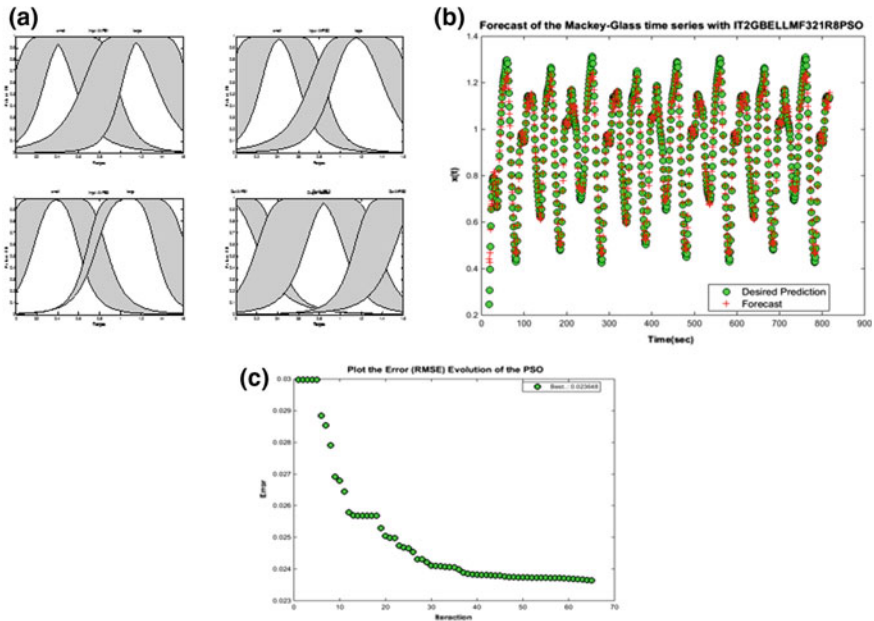
The best errors were produced by the interval type-2 fuzzy integrator (using igbelltype2 MFs) with PSO are shown on Table 2. The RMSE is 0.023648414 and the average RMSE is 0.024988012, the MSE is 0.00163873 and the MAE is

**Table 2** PSO results for the optimization of the fuzzy integrators

Metrics	Type-1 MFs			Interval type-2 MFs		
	Gaussian	Generalized Bell	Triangular	Igaussmtype2	Igbelltype2	Itritype2
RMSE (best)	0.035946912	<b>0.035228102</b>	0.0797536	0.024183221	<b>0.023648414</b>	0.0251151
RMSE (average)	0.044289015	<b>0.047356657</b>	0.0928408	0.026416896	<b>0.024988012</b>	0.0286363
MSE	0.008587152	<b>0.005989357</b>	0.0147635	0.003288946	<b>0.00163873</b>	0.0022031
MAE	0.065247859	<b>0.056713089</b>	0.0968261	0.039278738	<b>0.028366955</b>	0.0326414
Time (HH:MM:SS)	00:19:06	00:18:10	00:17:33	02:23:06	02:12:10	02:59:21



**Fig. 14** Variables inputs/output MFs (a), forecast (b), and evolution errors of “65 iterations” (c), are generated for the optimized of the type-1 FIS integrator with PSO



**Fig. 15** Variables inputs/output MFs (a), forecast (b), and evolution errors of “65 iterations” (c), are generated for the optimized of the interval type-2 FIS integrator with PSO

0.028366955 respectively. The MFs optimized with PSO are presented in Fig. 15a, the forecasts in Fig. 15b, and the evolution errors in Fig. 15c are obtained for the proposed architecture.

## 5 Conclusion

Particle swarm optimization of the fuzzy integrators for time series prediction using ensembles of IT2FNN architecture was proposed in this paper.

The best result generated for the optimization the interval type-2 FIS (using igbelltype2 MFs) integrator is with a prediction error of 0.023648414 (98 %).

The best result generated for the optimization of type-1 FIS (using Generalized Bell MFs) integrator with a prediction error of 0.035228102 (97 %).

These results showed efficient results in the prediction error of the time series Mackey-Glass generated by proposed architecture.

## References

1. Ascia, G., Catania, V., Panno, D.: An Integrated Fuzzy-GA Approach for Buffer Management. *IEEE Trans. Fuzzy Syst.* 14(4), pp. 528–541. (2006).
2. Bonissone, P.P., Subbu, R., Eklund, N., Kiehl, T.R.: Evolutionary Algorithms + Domain Knowledge = Real-World Evolutionary Computation. *IEEE Trans. Evol. Comput.* 10(3), pp. 256–280. (2006).
3. Brocklebank J. C., Dickey, D.A.: SAS for Forecasting Series. SAS Institute Inc. Cary, NC, USA, pp. 6-140. (2003).
4. Brockwell, P. D., Richard, A.D.: Introduction to Time Series and Forecasting. Springer-Verlag New York, pp 1-219. (2002).
5. Castillo, O., Melin, P.: Optimization of type-2 fuzzy systems based on bio-inspired methods: A concise review, *Information Sciences*, Volume 205, pp. 1-19. (2012).
6. Castro J.R., Castillo O., Melin P., Rodriguez A.: A Hybrid Learning Algorithm for Interval Type-2 Fuzzy Neural Networks: The Case of Time Series Prediction. Springer-Verlag Berlin Heidelberg, Vol. 15a, pp. 363-386. (2008).
7. Castro, J.R., Castillo, O., Martínez, L.G.: Interval type-2 fuzzy logic toolbox. *Engineering Letters*, 15(1), pp. 89–98. (2007).
8. Chiou, Y.-C., Lan, L.W.: Genetic fuzzy logic controller: an iterative evolution algorithm with new encoding method. *Fuzzy Sets Syst.* 152(3), pp. 617–635. (2005).
9. Deb, K.: A population-based algorithm-generator for real-parameter optimization. Springer, Heidelberg. (2005).
10. Engelbrecht, A.P.: Fundamentals of computational swarm intelligence. John Wiley & Sons, Ltd., Chichester. (2005).
11. Gaxiola, F., Melin, P., Valdez, F., Castillo, O.: Optimization of type-2 fuzzy weight for neural network using genetic algorithm and particle swarm optimization. *Nature and Biologically Inspired Computing (NaBIC)*. World Congress on, vol., no., pp. 22-28. (2013).
12. Hagan, M.T., Demuth, H.B., Beale, M.H.: *Neural Network Design*. PWS Publishing, Boston. (1996).
13. Hagrass, H.: Comments on Dynamical Optimal Training for Interval Type-2 Fuzzy Neural Network (T2FNN). *IEEE Transactions on Systems Man And Cybernetics Part B* 36(5), pp. 1206–1209. (2006).
14. Haykin, S.: *Adaptive Filter Theory*. Prentice Hall, Englewood Cliffs. (2002) ISBN 0-13-048434-2.
15. Horikawa, S., Furuhashi, T., Uchikawa, Y.: On fuzzy modeling using fuzzy neural networks with the backpropagation algorithm. *IEEE Transactions on Neural Networks* 3, (1992).
16. Ishibuchi, H., Nozaki, K., Yamamoto, N., Tanaka, H.: Selecting fuzzy if-then rules for classification problems using genetic algorithms. *IEEE Trans. Fuzzy Syst.* 3, pp. 260–270. (1995).
17. Jang J.S.R.: Fuzzy modeling using generalized neural networks and Kalman filter algorithm. *Proc. of the Ninth National Conference on Artificial Intelligence*. (AAAI-91), pp. 762-767. (1991).
18. Jang, J.S.R., Sun, C.T., Mizutani, E.: *Neuro-fuzzy and Soft Computing*. Prentice-Hall, New York. (1997).
19. Jang, J.S.R.: ANFIS: Adaptive-network-based fuzzy inference systems. *IEEE Trans. on Systems, Man and Cybernetics*. Vol. 23, pp. 665-685 (1992).
20. Karnik, N.N., Mendel, J.M., Qilian L.: Type-2 fuzzy logic systems. *Fuzzy Systems, IEEE Transactions on*. vol.7, no.6, pp. 643,658. (1999).
21. Karnik, N.N., Mendel, J.M.: Applications of type-2 fuzzy logic systems to forecasting of time-series. *Inform. Sci.* 120, pp. 89–111. (1999).
22. Kennedy, J., Eberhart, R.: Particle swarm optimization. *Neural Networks. Proceedings., IEEE International Conference on*. vol. 4. pp. 1942-1948. (1995).



23. Lee, C.H., Hong, J.L., Lin, Y.C., Lai, W.Y.: Type-2 Fuzzy Neural Network Systems and Learning. *International Journal of Computational Cognition* 1(4), pp. 79–90. (2003).
24. Lee, C.-H., Lin, Y.-C.: Type-2 Fuzzy Neuro System Via Input-to-State-Stability Approach. In: Liu, D., Fei, S., Hou, Z., Zhang, H., Sun, C. (eds.) *ISSN 2007. LNCS*, vol. 4492, pp. 317–327. Springer, Heidelberg (2007).
25. Lin, Y.-C., Lee, C.-H.: System Identification and Adaptive Filter Using a Novel Fuzzy Neuro System. *International Journal of Computational Cognition* 5(1) (2007).
26. Mackey, M.C., Glass, L.: Oscillation and chaos in physiological control systems. *Science*, Vol. 197, pp. 287-289. (1997).
27. Mackey, M.C.: Mackey-Glass. McGill University, Canada, [http://www.sholarpedia.org/-article/Mackey-Glass\\_equation](http://www.sholarpedia.org/-article/Mackey-Glass_equation), September 5th, (2009).
28. Mamdani, E.H., Assilian, S.: An experiment in linguistic synthesis with a fuzzy logic controller. *Int. J. Man-Mach. Stud.* 7, pp. 1–13. (1975).
29. Melin, P., Soto, J., Castillo, O., Soria, J.: A New Approach for Time Series Prediction Using Ensembles of ANFIS Models. *Experts Systems with Applications*. Elsevier, Vol. 39, Issue 3, pp 3494-3506. (2012).
30. Mendel, J.M.: *Uncertain rule-based fuzzy logic systems: Introduction and new directions*. Ed. USA: Prentice Hall, pp 25-200. (2000).
31. Mendel, J.M.: Why we need type-2 fuzzy logic systems. Article is provided courtesy of Prentice Hall, By Jerry Mendel. (2001).
32. Parsopoulos, K.E., Vrahatis, M.N.: *Particle Swarm Optimization Intelligence: Advances and Applications*. Information Science Reference. USA. pp. 18-40. (2010).
33. Pedrycz, W.: *Fuzzy Evolutionary Computation*. Kluwer Academic Publishers, Dordrecht. (1997).
34. Pedrycz, W.: *Fuzzy Modelling: Paradigms and Practice*. Kluwer Academic Press, Dordrecht. (1996).
35. Pulido M., Melin P., Castillo O.: Particle swarm optimization of ensemble neural networks with fuzzy aggregation for time series prediction of the Mexican Stock Exchange. *Information Sciences*, Volume 280., pp. 188-204. (2014).
36. Pulido, M., Mancilla, A., Melin, P.: An Ensemble Neural Network Architecture with Fuzzy Response Integration for Complex Time Series Prediction. *Evolutionary Design of Intelligent Systems in Modeling, Simulation and Control*, pp. 85-110. (2009).
37. Russell, S., Norvig, P.: *Artificial Intelligence: A Modern Approach*. Prentice-Hall, NJ. (2003).
38. Shi, Y., Eberhart, R.: A modified particle swarm optimizer. In: *Proceedings of the IEEE congress on evolutionary computation*, pp. 69-73. (1998).
39. Shi, Y., Eberhart, R.: Empirical study of particle swarm optimization. In: *Proceedings of the IEEE congress on evolutionary computation*, pp. 1945-1950. (1999).
40. Sollich, P., Krogh, A.: Learning with ensembles: how over-fitting can be useful. in: D.S. Touretzky M.C. Mozer, M.E. Hasselmo (Eds.). *Advances in Neural Information Processing Systems* 8, Denver, CO, MIT Press, Cambridge, MA, pp. 190-196. (1996).
41. Soto, J., Melin, P., Castillo, O.: Time series prediction using ensembles of ANFIS models with genetic optimization of interval type-2 and type-1 fuzzy integrators. *International Journal Hybrid Intelligent Systems* Vol. 11(3): pp. 211-226. (2014).
42. Takagi T., Sugeno M.: Derivation of fuzzy control rules from human operation control actions. *Proc. of the IFAC Symp. on Fuzzy Information, Knowledge Representation and Decision Analysis*, pp. 55-60. (1983).
43. Takagi, T., Sugeno, M.: Fuzzy identification of systems and its applications to modeling and control. *IEEE Trans. Syst., Man, Cybern.* 15, pp. 116–132. (1985).
44. Wang, C.H., Cheng, C.S., Lee, T.-T.: Dynamical optimal training for interval type-2 fuzzy neural network (T2FNN). *IEEE Trans. on Systems, Man, and Cybernetics Part B: Cybernetics* 34(3), pp. 1462–1477. (2004).
45. Wang, C.H., Liu, H.L., Lin, C.T.: Dynamic optimal Learning rate of A Certain Class of Fuzzy Neural Networks and Its Applications with Genetic Algorithm. *IEEE Trans. Syst. Man, Cybern.* 31(3), pp. 467–475. (2001).

46. Wu, D., Mendel, J.M.: A Vector Similarity Measure for Interval Type-2 Fuzzy Sets and Type-1 Fuzzy Sets. *Information Sciences* 178, pp. 381–402. (2008).
47. Wu, D., Wan Tan, W.: Genetic learning and performance evaluation of interval type-2 fuzzy logic controllers. *Engineering Applications of Artificial Intelligence* 19(8), pp. 829–841. (2006).
48. Xiaoyu L., Bing W., Simon Y.: Time Series Prediction Based on Fuzzy Principles. Department of Electrical & Computer Engineering FAMU-FSU College of Engineering, Florida State University Tallahassee, FL 32310, (2002).
49. Zadeh L. A.: Fuzzy Logic = Computing with Words. *IEEE Transactions on Fuzzy Systems*, 4 (2), 103, (1996).
50. Zadeh L. A.: Fuzzy Logic. *Computer*, Vol. 1, No. 4, pp. 83-93. (1988).
51. Zadeh, L.A.: Fuzzy Logic, Neural Networks and Soft Computing. *Communications of the ACM* 37(3), pp. 77–84. (1994).