Chapter 3
Problem Statement and Development

The first goal of this book is the construction of the Ensembles of IT2FNN models and their optimization of the fuzzy integrators with GAs and PSO algorithms for time series prediction. The second goal is the design of interval type-2 and type-1 fuzzy systems to integrate the outputs (forecasts) of the IT2FNN models forming the Ensemble. The Genetic Algorithm (GAs) and Particle Swarm Optimization (PSO) were used for the optimization the parameters of the MFs of fuzzy integrators. The Mackey-Glass, Mexican Stock Exchange, Dow Jones, NASDAQ time series are used to test of performance of the proposed architectures (Fig. 3.1). When more than one forecasting technique seems reasonable for a particular application, then the forecast accuracy measures can also be used to discriminate between competing models. One can subtract the forecast value from the observed value of the data at that time point and obtain a measure of error. Therefore to evaluate the prediction error, we can apply the metrics to calculate the Mean Absolute Error (MAE) by Eq. (3.1), Mean Square Error (MSE) by Eq. (3.2), Root Mean Square Error (RMSE) by Eq. (3.3), Mean Percentage Error (MPE) by Eq. (3.4) and Mean Absolute Percentage Error (MAPE) by Eq. (3.5), respectively.

$$\text{MAE} = \frac{1}{2}\sum_{t=1}^{n} |(A_t - P_t)| \tag{3.1}$$

$$\text{MSE} = \frac{1}{2}\sum_{t=1}^{n} (A_t - P_t)^2 \tag{3.2}$$

$$\text{RMSE} = \sqrt{\frac{1}{2}\sum_{t=1}^{n} (A_t - P_t)^2} \tag{3.3}$$

$$\text{MPE} = \frac{100\%}{n} = \sum_{t=1}^{n} \frac{(A_t - P_t)}{A_t} \tag{3.4}$$

J. Soto et al., *Ensembles of Type 2 Fuzzy Neural Models and Their Optimization with Bio-Inspired Algorithms for Time Series Prediction*, SpringerBriefs in Computational Intelligence, https://doi.org/10.1007/978-3-319-71264-2_3

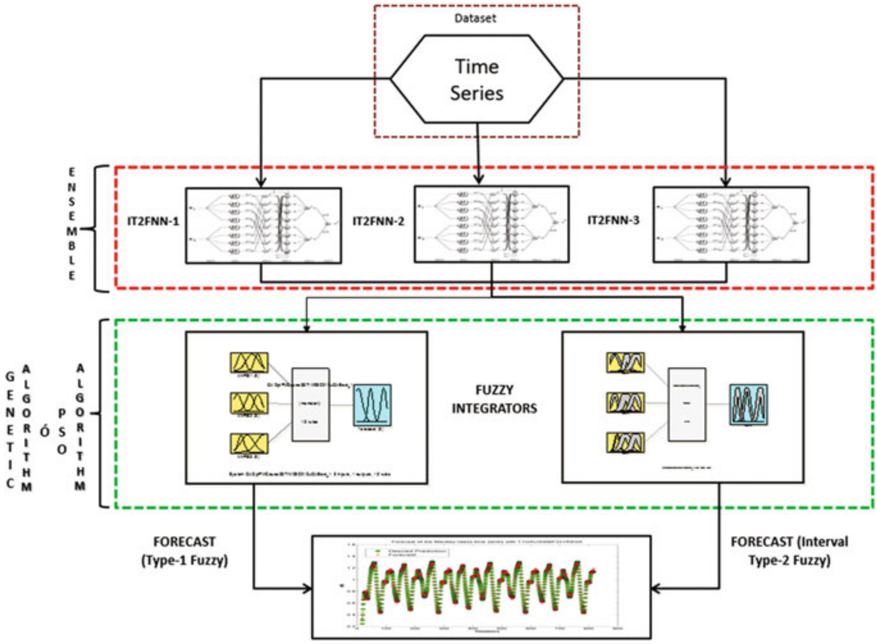

Fig. 3.1 The general proposed architecture

$$\text{MAPE} = \frac{100\%}{n} = \sum_{t=1}^{n} \left| \frac{(A_t - P_t)}{A_t} \right| \tag{3.5}$$

where *A*, corresponds to the real data of the time series, *P* corresponds to the forecast of the NNs or the aggregation models, *t* is the time variable, and *n* is the number of data points of the time series.

The general proposed architecture combines the ensemble of IT2FNN models and the use of fuzzy response integrators optimized with GA and PSO algorithms for time series prediction (Fig. 3.1).

3.1 Historical Data

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 Δ spaced time series, i.e. (x(t − (D − 1) Δ), …, x(t − Δ), x(t)), to a predicted future value x(t + P), for example the values D = 4 and Δ = P = 6 [1, 2] or 250 were used in this work. The data used in this book are the Mackey-Glass for τ = 13, 17, 30, 34, 68, 100, 136; the Mexican Stock Exchange, the Dow Jones and the NASDAQ time series.

3.1.1 Mackey-Glass Time Series

The chaotic time series data used is defined by the Mackey-Glass [3, 4] time series, whose differential equation is given by Eq. (3.6):

$$x(t) = \frac{0.2x(t-\tau)}{1 - x^{10}(t-\tau)} - 0.1x(t-\tau) \tag{3.6}$$

For obtaining the values of the time series at each point, we can apply the Runge-Kutta method [1, 2] for the solution of Eq. (3.6). The integration step was set at 0.1, with initial condition x(0) = 1.2, τ = 17, x(t) is then obtained for 0 ≤ t ≤ 1200, (We assume x(t) = 0 for t < 0 in the integration). From the Mackey-Glass time series we used 800 pairs of data points (Fig. 3.2), similar to [5–9]. The first 400 pairs of points are used for training (50%) and the other 400 pairs of points are used to validate the IT2FNN models.

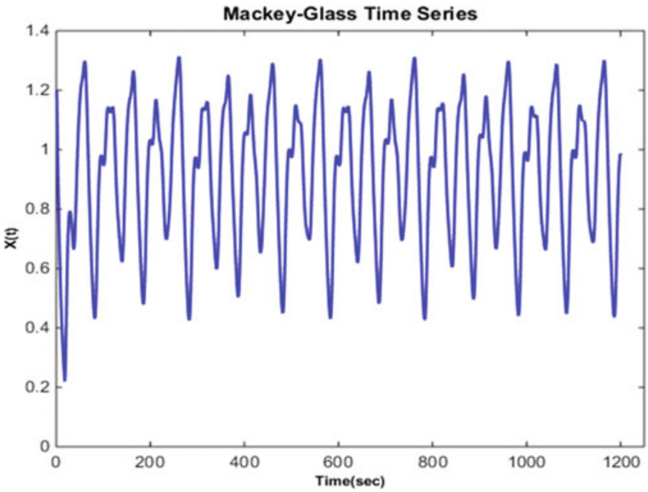

Fig. 3.2 Mackey-Glass time series

3.1.2 Mexican Stock Exchange

The Mexican Stock Exchange (BMV) is a financial institution that operates by a grant from the Department of Finance and Public Credit, with the goal of following closely the Securities Market of Values in Mexico [10, 11]with the initial public offering taking place on June 13 of 2008 with its shares representing its capital [12]. From the BMV time series we extracted 1250 pairs of data that correspond to a period from 01/03/2011 to 12/31/2015 (Fig. 3.3) and can be downloaded from daily live Yahoo database [13], where the first 625 pairs of points are used for training (50%) and the other 625 pairs of points are used to validate the IT2FNN models.

3.1.3 Dow Jones Time Series

The better represent the movements of the stock market at the time, the Dow Jones & Company designed a barometer of economic activity meter with twelve companies creating the Dow Jones stock index [14, 15]. Like the New York Times and Washington Post newspapers the company is open to the market but is controlled the by the private sector. So far, the company is controlled by the Bancroft family, which controls 64% of the shares entitled to vote [16]. From the Dow Jones Industries Average time series we are using 1250 pairs of data that correspond from 01/03/2011 to 12/31/2015 (Fig. 3.4) and can be downloaded from daily live Yahoo database [17], where the first 625 pairs of points are used for training (50%) and the other 625 pairs of points are used to validate the IT2FNN models.

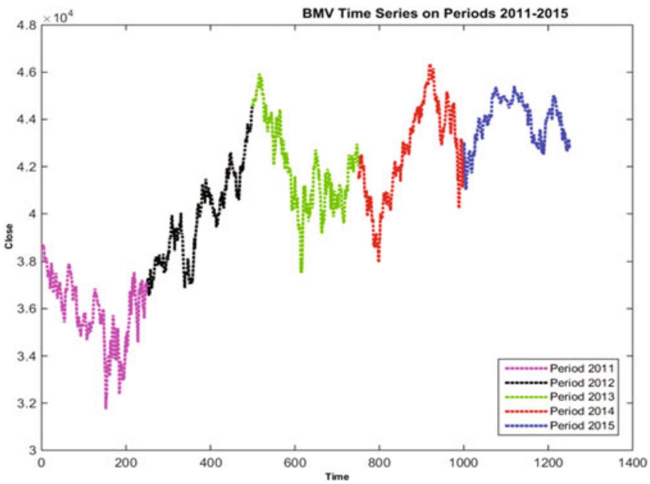

Fig. 3.3 Mexican Stock Exchange time series

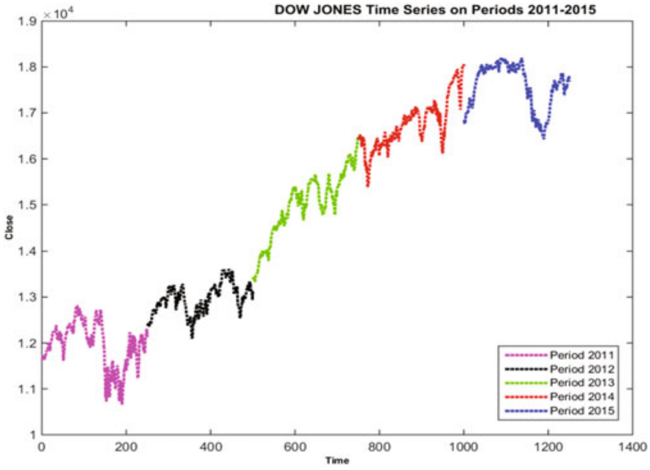

Fig. 3.4 Dow Jones time series

3.1.4 NASDAQ Time Series

NASDAQ is the largest U.S. electronic stock market. It has listed around 3300 companies; it may probably list most of the companies and, on average, trades more shares per day than other U.S. markets [18] price. The price is able to represent the tendency of variety of NASDAQ market in some sense. Therefore, the forecast of the price can benefit of analyzing the whole market [19, 20]. From the NASDAQ time series we are using 1250 pairs of data that correspond from 01/03/2011 to 12/31/2015 (Fig. 3.5) and can be downloaded from daily live Yahoo database [21], where the first 625 pairs of points are used for training (50%) and the other 625 pairs of points are used to validate the IT2FNN models.

3.2 Ensembles of IT2FNN Architectures

The ensembles of IT2FNN architectures imply a significant learning improvement comparatively to a single IT2NN and especially to the learning algorithms. Each IT2FNN works independently in its own domain. Each of the IT2FNN is build and trained for a specific task for each module. Three modules are used in each experiment of the ensemble of IT2FNN. In module 1 we have the IT2FNN-1, in module 2 we have the IT2FNN-2 and in module 3 we have the IT2FNN-3. Therefore each IT2FNN architecture has three/four input variables and one output variable that are described as follows:

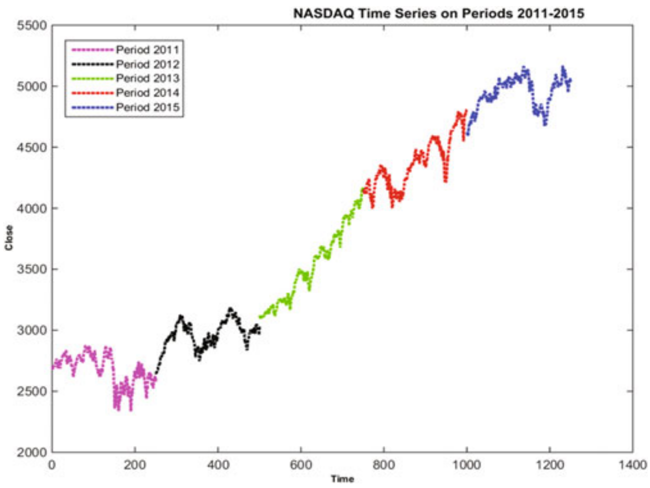

Fig. 3.5 NASDAQ time series

Mackey-Glass time series, in this case 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 data is:

$$[x(t-18), x(t-12), x(t-6); x(t)] \quad (3.7)$$

where $t = 19$ to 818 and $x(t)$ is the desired prediction of the Mackey-Glass time series (Fig. 3.2).

The BMV, Dow Jones and NASDAQ time series, we predict $x(t)$ (corresponds to the 2015 period) from four past (delays) values of the time series, that is, $x(t-1000)$ corresponds to the 2011 period, $x(t-750)$ corresponds to the 2012 period, $x(t-500)$ corresponds to the 2013 period, and $x(t-250)$ corresponds to the 2014 period. Therefore the format of the training data is:

$$[x(t-1000), x(t-750), x(t-500), x(t-250); x(t)] \quad (3.8)$$

where $t = 1001$ to 1250 and $x(t)$ is the desired prediction of the time series (see Figs. 3.3, 3.4 and 3.5).

3.2.1 IT2FNN-1 Model

The IT2FNN-1 model has 5 layers (Fig. 3.6), consists of adaptive nodes with an equivalent function for the lower-upper membership in the fuzzification layer (layer 1). Non-adaptive nodes in the rules layer (layer 2) interconnect with the fuzzification layer (layer 1) in order to generate TSK IT2FIS rules antecedents. The

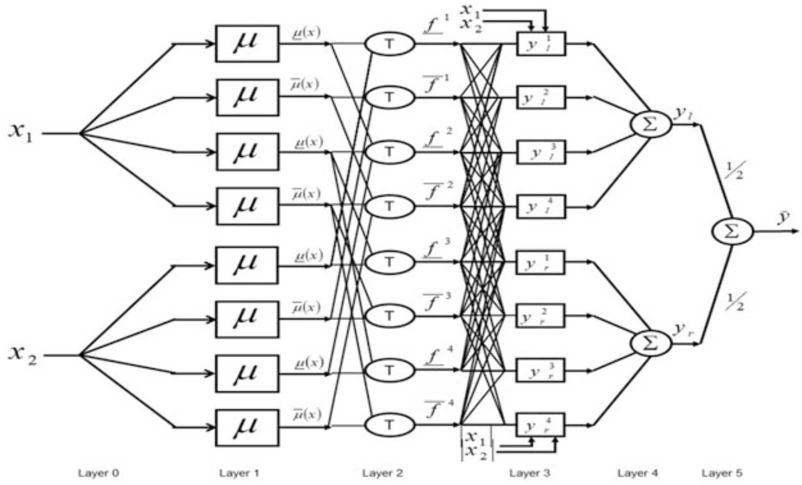

Fig. 3.6 IT2FNN-1 architecture

adaptive nodes in the 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 [22]. The non-adaptive node in the defuzzification layer (layer 5) average left-right values.

For simplicity, we assume the IT2FNN-1 under consideration has $\boldsymbol{n}$ inputs and one output. The forward-propagation procedure is described as follows:

Layer 0: Inputs

$$x = (x_1, \ldots, x_i, \ldots, x_n,)^t$$

Layer 1: IT2 MFs $\tilde{\mu}_{k,i}(x_i) = \left\{ \underline{\mu}_{k,i}(x_i), \overline{\mu}_{k,i}(x_i) \right\}$

for example (Fig. 3.9a)

$$\begin{aligned}
&\mu_{k,i}(x_i) = \left\{ \underline{\mu}_{k,i}(x_i), \overline{\mu}_{k,i}(x_i) \right\} = i\,gaussmtype2\left(x_i\left[\sigma_{k,i}, {}^1m_{k,i}, {}^2m_{k,i}\right]\right); k = 1, 2, \ldots, M; i = 1, 2, \ldots, n, \\
&{}^1\mu_{k,i}\left(x_i\left[\sigma_{k,i}, {}^1m_{k,i}\right]\right), e^{-\frac{1}{2}\left(\frac{x_i - {}^1m_{k,i}}{\sigma_{k,i}}\right)^2}; {}^2\mu_{k,i}\left(x_i\left[\sigma_{k,i}, {}^1m_{k,i}\right]\right), e^{-\frac{1}{2}\left(\frac{x_i - {}^2m_{k,i}}{\sigma_{k,i}}\right)^2}, \\
&\overline{\mu}_{k,i}(x_i) = \begin{cases} {}^1\mu\left(x_i\left[\sigma_{k,i}, {}^1m_{k,i}\right]\right), & x_i < {}^1m_{k,i}, \\ 1, & {}^1m_{k,i} \leq x_i \geq {}^2m_{k,i} \\ {}^2\mu\left(x_i\left[\sigma_{k,i}, {}^2m_{k,i}\right]\right), & x_i > {}^2m_{k,i}, \end{cases} \\
&\underline{\mu}_{k,i}(x_i) = \begin{cases} {}^2\mu\left(x_i\left[\sigma_{k,i}, {}^2m_{k,i}\right]\right), & x_i \leq \frac{{}^1m_{k,i} - {}^2m_{k,i}}{2}, \\ {}^1\mu\left(x_i\left[\sigma_{k,i}, {}^1m_{k,i}\right]\right), & x_i > \frac{{}^1m_{k,i} - {}^2m_{k,i}}{2}. \end{cases}
\end{aligned} \tag{3.9}$$

Layer 2: Rules

$$\underline{f}^k = \overset{n}{\underset{i=1}{\tilde{*}}} \left(\underline{\mu}_{k,i}\right); \overline{f}^k = \overset{n}{\underset{i=1}{\tilde{*}}} \left(\overline{\mu}_{k,i}\right) \tag{3.10}$$

Layer 3: Consequents left-right firing points

$$\begin{aligned} y_l^k &= \sum_{i=1}^{n} C_{k,i}x_i + C_{k,0} - \sum_{i=1}^{n} S_{k,i}|x_i| - S_{k,0}, \\ y_r^k &= \sum_{i=1}^{n} C_{k,i}x_i + C_{k,0} + \sum_{i=1}^{n} S_{k,i}|x_i| + S_{k,0}. \end{aligned} \tag{3.11}$$

Layer 4: Left-right points (type-reduction using KM algorithm)

$$\begin{aligned} \widehat{y}_l &= \widehat{y}_l\left(\overline{f}^1, \ldots, \overline{f}^L, \underline{f}^{L+1}, \ldots, \underline{f}^M, \quad y_l^1, \ldots, y_l^M\right) = \frac{\sum_k^M f_l^k \cdot y_l^k}{\sum_{k=1}^M f_l^k} = \frac{\sum_{k=1}^L \overline{f}^k \cdot y_l^k + \sum_{k=L+1}^L \underline{f}^k \cdot y_l^k}{\sum_{k=1}^L \overline{f}^k + \sum_{k=L+1}^L \underline{f}^k} \\ \widehat{y}_r &= \widehat{y}_r\left(\overline{f}^1, \ldots, \overline{f}^R, \underline{f}^{R+1}, \ldots, \underline{f}^M, \quad y_r^1, \ldots, y_r^M\right) = \frac{\sum_k^M f_r^k \cdot y_r^k}{\sum_{k=1}^M f_r^k} = \frac{\sum_{k=1}^R \overline{f}^k \cdot y_r^k + \sum_{k=R+1}^M \underline{f}^k \cdot y_l^k}{\sum_{k=1}^R \overline{f}^k + \sum_{k=R+1}^M \underline{f}^k} \end{aligned} \tag{3.12}$$

Layer 5: Defuzzification

$$\widehat{y} = \frac{\widehat{y}_l + \widehat{y}_r}{2}. \tag{3.13}$$

IT2FNN uses backpropagation method with heuristic techniques (variable learning rate backpropagation, and resilient backpropagation), which were developed from a performance analysis of the standard steepest descent algorithm and numerical optimization techniques for IT2FNN training: conjugate gradient, quasi-Newton, and Levenberg-Marquardt for learning how to determine premise parameters (to find the parameters related to interval type-2 MFs) and consequent parameters. The learning procedure has two parts: in the first part the input patterns are propagated, the consequent parameters and the premise parameters are assumed to be fixed for the current cycle through the training set. In the second part the pattern are propagated again, and at this moment, backpropagation is used to modify the premise parameters, and consequent parameters. These two parts are considered an epoch.

Give an input-output training pair $\left\{\left(x_p : t_p\right)\right\} \forall p = 1, \ldots, q$, in order to get the design of the IT2FNN, the error function (E) must minimized.

$$e_p = t_p - \hat{y}_p \tag{3.14}$$

$$E_p = \frac{1}{2} e_p^2 = \frac{1}{2} \left(t_p - \hat{y}_p \right)^2 \tag{3.15}$$

$$E = \sum_{p=1}^{q} E_p \tag{3.16}$$

3.2.2 IT2FNN-2 Model

The IT2FNN-2 model has 6 layers (Fig. 3.7), if uses NN for fuzzifying the inputs (layers 1 to 2). The non-adaptive nodes in the rules layer (layer 3) interconnect with the lower-upper linguistic values layer (layer 2) to generate TSK IT2FIS rules antecedents. The non-adaptive nodes in the consequents layer (layer 4) are connected with 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.

The forward-propagation procedure is described as follows:

Layer 0: Inputs

$$x = (x_1, \ldots, x_i, \ldots, x_n)^t$$

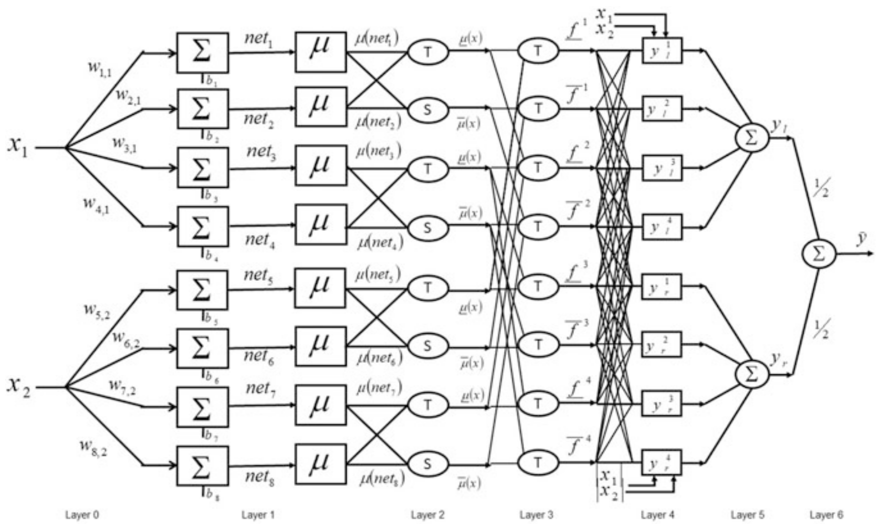

Fig. 3.7 IT2FNN-2 architecture

Layer 1: Every node ℓ in this layer is a square (Fig. 3.7) with node function

for $k = 1$ to M

for $i = 1$ to n

$$\begin{aligned} {}^{1}net_{k,i} &= {}^{1}w_{k,i}x_i + {}^{1}b_{k,i};\, {}^{1}\mu_{k,i}(x_i) = \mu\left({}^{1}net_{k,i}\right), \\ {}^{2}net_{k,i} &= {}^{2}w_{k,i}x_i + {}^{2}b_{k,i};\, {}^{2}\mu_{k,i}(x_i) = \mu\left({}^{2}net_{k,i}\right) \end{aligned} \tag{3.17}$$

end

end

where μ is the transfer function which can be Gaussian, GBell or logistic (e.g. Gaussian with uncertain mean "igaussmtype2" and transfer function GBell with uncertain mean "igbellmtype2") (see Fig. 3.9b).

Layer 2: Every node ℓ in this layer is a circle labeled with *T*-norm and *S*-norm alternated.

$$\begin{aligned} &\underline{\mu}_{k,i}(x_i) = {}^{1}\mu_{k,i}(x_i) \cdot {}^{2}\mu_{k,i}(x_i) \\ &\overline{\mu}_{k,i}(x_i) = {}^{1}\mu_{k,i}(x_i) + {}^{2}\mu_{k,i}(x_i) - \underline{\mu}_{k,i}(x_i) \\ &k = 1, 2, \ldots, M;\, i = 1, 2, \ldots, n \end{aligned} \tag{3.18}$$

Layer 3 to 6: This layer are equivalent to layers 2 to 5 on the IT2FNN-1 architecture.

3.2.3 IT2FNN-3 Model

The IT2FNN-3 model has 7 layers (Fig. 3.8). Layer 1 has adaptive nodes for fuzzifying the inputs; layer 2 has non-adaptive nodes with the interval fuzzy values (Fig. 3.9c). 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 the left-right interval. The non-adaptive node in layer 7 (defuzzification) evaluates average of interval left-right values.

The forward-propagation procedure is described as follows: the first 3 layers (0 to 3) are identical to the corresponding layers on the IT2FNN-2 architecture.

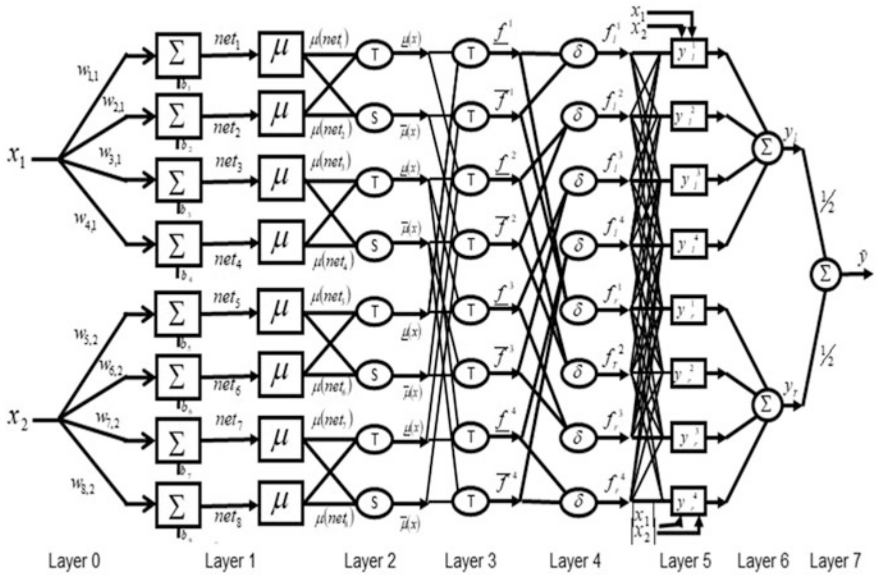

Fig. 3.8 IT2FNN-3 architecture

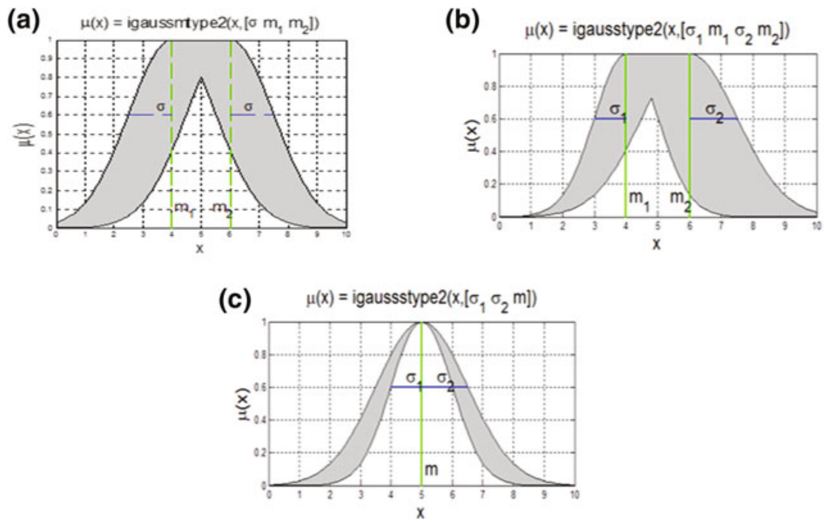

Fig. 3.9 The MFs used for training the IT2FNN architectures

Layer 4: Every node ℓ in this layer is a circle labeled $\wp$ which evaluates the left-most and right-most firing points denoted by:

$$f_l^k = \frac{\overline{w}_l^k \overline{f}^k + \underline{w}_l^k \underline{f}^k}{\overline{w}_l^k + \underline{w}_l^k}; f_r^k = \frac{\underline{w}_r^k \underline{f}^k + \overline{w}_r^k \overline{f}^k}{\underline{w}_r^k + \overline{w}_r^k} \tag{3.19}$$

where w values are adjustable weights.

Layer 5 is equivalent to layer 3 on the IT2FNN-1 architecture

Layer 6: The two nodes in this layer are circles labeled with "Σ" that evaluates the two end-points, y_l and y_r:

$$\hat{y}_l = \frac{\sum_{k=1}^{M} f_l^k \cdot y_l^k}{\sum_{k=1}^{M} f_l^k}; \hat{y}_r = \frac{\sum_{k=1}^{M} f_r^k \cdot y_r^k}{\sum_{k=1}^{M} f_r^k} \tag{3.20}$$

Layer 7: The single node in this layer is a circle labeled "Σ" that computes the output.

$$\widehat{y} = \frac{\hat{y}_l + \hat{y}_r}{2} \tag{3.21}$$

3.3 Fuzzy Integrators

The design of the type-1 (Fig. 3.10) and interval type-2 (Fig. 3.11) fuzzy inference systems integrators are of Mamdani type and have 3 inputs (IT2FNN1, IT2FNN2 and IT2FNN3) and 1 output (Forecast), so each Input-Output variable is assigned two MFs with linguistic labels "Small and Large" and have 8 if-then 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. the fuzzy inference system uses 3 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 8 fuzzy-rules for the experiments (Fig. 3.12) because the performance is better and minimized the prediction error of the Mackey-Glass, BMV, Dow Jones and NASDAQ time series.

In the type-1 fuzzy integrators we used different MFs [Gaussian, Generalized Bell, and Triangular (Fig. 3.13a)] and for the interval type-2 fuzzy integrators we used different MFs [igaussmtype2, igbelltype2 and itritype2 (Fig. 3.13b)] [23] to observe the behavior of each of them and determine which one provides better forecast of the time series.

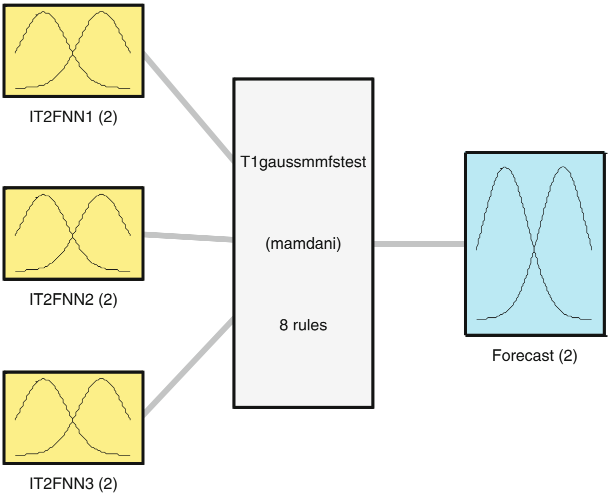

Fig. 3.10 Structure of the type-1 fuzzy integrator

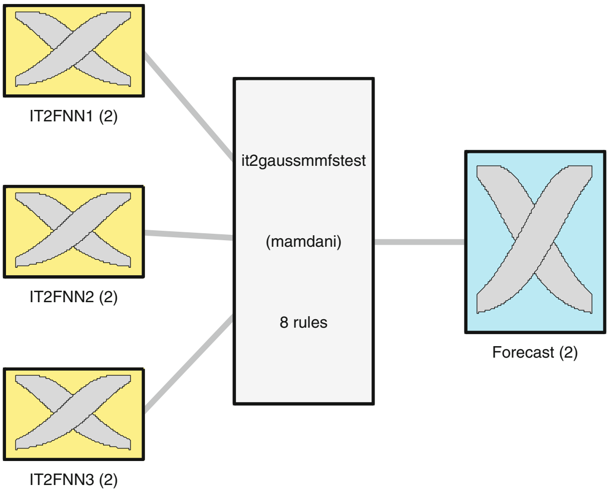

Fig. 3.11 Structure of the interval type-2 fuzzy integrator

1. If (IT2FNN1 is Small) and (IT2FNN2 is Small) and (IT2FNN3 is Small) then (Forecast is Small) (1)
2. If (IT2FNN1 is Small) and (IT2FNN2 is Small) and (IT2FNN3 is Large) then (Forecast is Small) (1)
3. If (IT2FNN1 is Small) and (IT2FNN2 is Large) and (IT2FNN3 is Small) then (Forecast is Small) (1)
4. If (IT2FNN1 is Small) and (IT2FNN2 is Large) and (IT2FNN3 is Large) then (Forecast is Large) (1)
5. If (IT2FNN1 is Large) and (IT2FNN2 is Small) and (IT2FNN3 is Small) then (Forecast is Small) (1)
6. If (IT2FNN1 is Large) and (IT2FNN2 is Small) and (IT2FNN3 is Large) then (Forecast is Large) (1)
7. If (IT2FNN1 is Large) and (IT2FNN2 is Large) and (IT2FNN3 is Small) then (Forecast is Large) (1)
8. If (IT2FNN1 is Large) and (IT2FNN2 is Large) and (IT2FNN3 is Large) then (Forecast is Large) (1)

Fig. 3.12 If-then rules for the fuzzy integrators

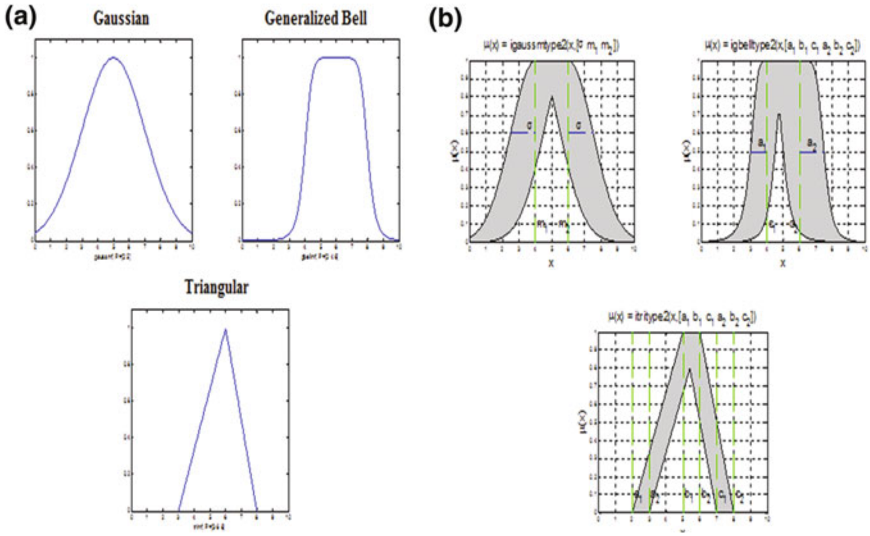

Fig. 3.13 Type-1 MFs (**a**) and interval type-2 MFs (**b**) for the fuzzy integrators

3.4 Optimization of the Fuzzy Integration with the Genetic Algorithm

We optimize the parameters values of the MFs in each type-1 and interval type-2 fuzzy integrators with GAs. The representation of GAs is of Real-Values and the chromosome size will depend of the number of MFs that are used in each design of the type-1 and interval type-2 fuzzy inference system integrators.

The objective function is defined to minimize the prediction error as follows:

$$f(t) = \sqrt{\frac{\sum_{t=1}^{n} (a_t - p_t)^2}{n}} \tag{3.22}$$

where a, corresponds to the real data of the time series, p corresponds to the output of the fuzzy integrators, t is de sequence time series, and n is the number of data points of time series.

The general structure of the chromosome (individuals) represents the parameters of the MFs of fuzzy integrators. The number of parameters varies according to the type of the MFs for the type-1 fuzzy system (e.g. two parameter are needed to represent a Gaussian MF are "σ" and "μ") for this case the GA optimized 16 parameters of the type-1 fuzzy integrator (Fig. 3.14). The interval type-2 fuzzy system (e.g. three parameter are needed to represent "igaussmtype2" MF's are "σ", "μ_1" and "μ_2") for this case the GA optimized 24 parameters of the interval type-2 fuzzy integrator (Fig. 3.15). Therefore the number of parameters that each fuzzy inference system integrator has depends of the MFs type (Fig. 3.13) assigned to each input and output variables.

The optimization was performed for the parameter values of the MFs (inputs and outputs) of fuzzy integrators. The parameters for the Genetic algorithm used to optimize the fuzzy integrators are described in Table 3.1.

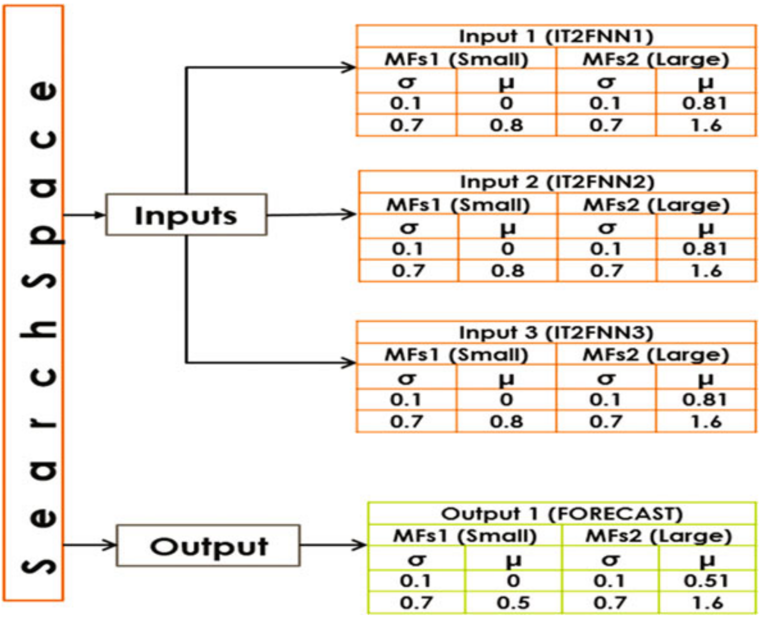

Fig. 3.14 Representation of chromosome of the GAs for the optimization of the Gaussian MFs

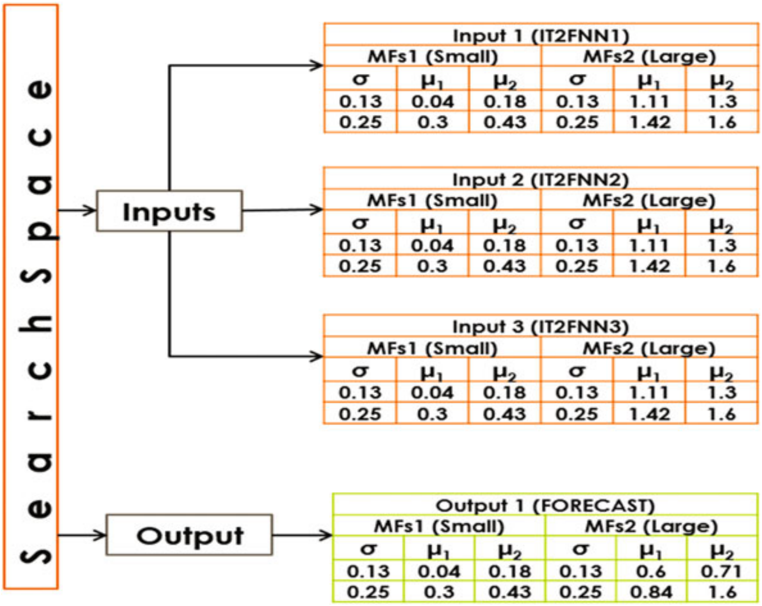

Fig. 3.15 Representation of chromosome of the GA for the optimization of the igaussmtype2 MFs

Table 3.1 The parameters of the GA used for optimization the fuzzy integrators

Representation of phenotypic	Real-values
Selection	Stochastic Universal Sample
Crossing or recombination	Discrete Recombination [0.8]
Mutation	0.1
Individuals	100
Generations	100
Iterations or run	31

3.5 Optimization of the Fuzzy Integrators with the Particle Swarm Optimization

We optimize the parameter values of the MFs in each type-1 and interval type-2 fuzzy integrators with PSO. The representation in PSO is of Real-Values and the particle size will depend on the number of the MFs that are used in each design of the fuzzy integrators.

The objective function to evaluate the performance of the PSO is similar to the Eq. (3.22). The general structure of the particles represents the parameters of the

Table 3.2 The parameters of the PSO used for optimization the fuzzy integrators

Parameters	Value
Particles	100
Iterations	65
Inertia weight "ω"	Linear decrement [0.88 − 0]
Constriction "C"	Linear increment [0.01 − 0.9]
r_1, r_2	Random
c_1	Linear decrement [2 − 0.5]
c_2	Linear increment [0.5 − 2]

MFs of the fuzzy integrators similar to the chromosome of the GAs (Figs. 3.14 and 3.15). Therefore the PSO are used to optimization the MFs (Fig. 3.13) of fuzzy integrators. The parameters for the PSO used to optimize the fuzzy integrators are described in Table 3.2.

References

1. Jang, J.S.R.: ANFIS: adaptive-network-based fuzzy inference systems. IEEE Trans. Syst. Man Cybern. 23, 665–685 (1992)
2. Jang, J.S.R.: Fuzzy modeling using generalized neural networks and Kalman filter algorithm. In: Proceedings of the Ninth National Conference on Artificial Intelligence (AAAI-91), pp. 762–767 (1991)
3. Mackey, M.C., Glass, L.: Oscillation and chaos in physiological control systems. Science **197**, 287–289 (1997)
4. Mackey, M.C.: Mackey-Glass. McGill University, Canada, http://www.sholarpedia.org/-article/Mackey-Glass_equation. 5 Sept 2009
5. Gaxiola, F., Melin, P., Valdez, F., Castillo, O.: Interval type-2 fuzzy weight adjustment for backpropagation neural networks with application in time series prediction. Inf. Sci. **260**, 1–14 (2014)
6. Melin, P., Soto, J., Castillo, O., Soria, J.: A new approach for time series prediction using ensembles of ANFIS models. Experts Syst. Appl. **39**(3), 3494–3506 (2012)
7. Pulido, M., Melin, P., Castillo, O.: Genetic optimization of ensemble neural networks for complex time series prediction. IJCNN, pp. 202–206 (2011)
8. 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. Inf. Sci. **280**, 188–204 (2014)
9. 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. Int. J. Hybrid Intell. Syst. **11**(3), 211–226 (2014)
10. Castellanos, S.G., Martínez, L.: Development of the Mexican Bond Market. In: Borensztein, E., Cowan, K., Eichengreen, B., Panizza, U. (eds.) Bond Markets in Latin America: On the Verge of a Big Bang?, pp. 51–58. MIT Press, Cambridge (2008)
11. Sidaoui, J.: The Mexican financial system: reforms and evolution 1995–2005. BIS Papers **28**, 277–293 (2006)
12. López, F., Santillán, R.J., Cruz, S.: Volatility dependence structure between the Mexican Stock Exchange and the World Capital Market. Investigación Económica **74**(293), 69–97 (2015)

13. https://es-us.finanzas.yahoo.com/q/hp?s=%5EMXX+Precios+historicos (7 May 2015)
14. Dow Jones Company. http://www.dowjones.com (10 Jan 2014)
15. Historic Dow Jones Data, Yahoo Finance, http://finance.yahoo.com (10 Jan 2014)
16. Dow Jones Indexes. http://www.djindexes.com (5 Sept 2014)
17. https://es-us.finanzas.yahoo.com/q/hp?s=%5EEDJI+Precios+historicos (8 May 2015)
18. http://business.nasdaq.com/discover/nasdaq-story/index.html (27 April 2015)
19. Blau, B.M., Van-Ness, B.F., Van-Ness, R.A.: Information in short selling: comparing NASDAQ and the NYSE. Rev. Fin. Econ. **20**(1), 1–10 (2011)
20. Pagano, M.S., Peng, L., Schwartz, R.A.: A call auction's impact on price formation and order routing: evidence from the NASDAQ stock market. J. Fin. Markets **16**(2), 331–361 (2013)
21. https://es-us.finanzas.yahoo.com/q/hp?s=%5EIXIC+Precios+historicos (9 May 2015)
22. Mendel, J.M.: Uncertain Rule-Based Fuzzy Logic Systems: Introduction and New Directions. Prentice-Hall, NJ (2001)
23. Castro, J.R., Castillo, O., Martínez, L.G.: Interval type-2 fuzzy logic toolbox. Eng. Lett. **15**(1), 89–98 (2007)