FUZZ-IEEE 2009, Korea, August 20-24, 2009

T2-HyFIS-Yager: Type 2 Hybrid Neural Fuzzy Inference System Realizing Yager Inference

S.W. Tung, C. Quek and C. Guan

Abstract— **The Hybrid neural Fuzzy Inference System (HyFIS) is a five layers adaptive neural fuzzy inference system, based on the Compositional Rule of Inference (CRI) scheme, for building and optimizing fuzzy models. To provide the HyFIS architecture with a firmer and more intuitive logical framework that emulates the human reasoning and decision-making mechanism, the fuzzy Yager inference scheme, together with the self-organizing gaussian Discrete Incremental Clustering (gDIC) technique , were integrated into the HyFIS network to produce the HyFIS-Yager-gDIC . This paper presents T2-HyFIS-Yager, a Type-2 Hybrid neural Fuzzy Inference System realizing Yager inference, for learning and reasoning with noise corrupted data. The proposed T2-HyFIS-Yager is used to perform time-series forecasting where a non-stationary time-series is corrupted by additive white noise of known and unknown SNR to demonstrate its superiority as an effective neuro-fuzzy modeling technique.**

I. Introduction

Information uncertainties are inherent in everyday life, from the natural linguistic fuzziness at the cognitive level to the measurement inaccuracies at the empirical level. All of these uncertainties translate into uncertainties about the fuzzy set membership functions. Traditional Type-1 fuzzy logic systems are unable to directly model such uncertainties because crisp membership grades are used for the fuzzy membership functions in the systems. On the other hand, Type-2 fuzzy logic systems [1],[5] are able to handle information uncertainties because the membership grades of the fuzzy membership functions used are also fuzzy. Such membership functions are fuzzy sets whose membership grades are Type-1 fuzzy sets, hence they are useful in incorporating information uncertainties in the systems.

The Hybrid neural Fuzzy Inference System (HyFIS) [4] is a five layers adaptive Type-1 neural fuzzy network that is used to combine numerical and linguistic information into a common framework. It adopts a two phase learning scheme. In the first phase, a fuzzy technique by Wang and Mendel [14] is used to obtain the initial fuzzy rulebase and the initial structure of the neural fuzzy system. In the second phase, a parameter learning technique using a gradient descent approach is used to tune the memberships of the input and output dimensions. Subsequently, the fuzzy Yager inference scheme [3], which accounts for a firm and intuitive logical framework that emulates the human reasoning and decision-making mechanism, is integrated into the HyFIS network. Together with the implementation of the gaussian Discrete Incremental Clustering (gDIC) [10] technique in the initialization phase of the HyFIS network which allows for self-organization of the membership functions, a self-organizing Hybrid neural Fuzzy Inference System based on Yager inference (HyFIS-Yager-gDIC) [13] is produced. The realization of the fuzzy Yager inference scheme in the HyFIS network offers a firm and intuitive logical framework, and the use of gDIC is shown to be able to robustly handle noisy data when the noise level is low.

This paper presents the Type-2 Hybrid neural Fuzzy Inference System which implements the Yager inference (T2-HyFIS-Yager), a self-organizing hybrid neural fuzzy inference system embedded with Type-2 fuzzy Yager inference. The proposed T2-HyFIS-Yager integrates the mathematical formalism of Type-2 fuzzy logic inference with the self-organizing Yager based HyFIS inference network, and allows for the robust learning and reasoning with noise corrupted data of known and unknown SNR. T2-HyFIS-Yager couples the Mamdani rule system [7] with Type-2 fuzzy inference to provide a clear interpretation to its knowledge-base and reasoning process for the comprehension of the human user.

The rest of the paper is organized as follows: the HyFIS and HyFIS-Yager-gDIC networks are briefly described in Sect. II; the operations and learning process of the proposed T2-HyFIS-Yager network are presented in Sect. III; the application of T2-HyFIS-Yager on the forecasting of a non-stationary time-series corrupted with additive white noise is described in Sect. IV; and Sect. V concludes the paper.

II. HyFIS and HyFIS-Yager-gDIC

HyFIS [4] and HyFIS-Yager-gDIC [13] are two Type-1 multilayer neural network based fuzzy systems. They express the knowledge induced from the training data by means of a set of IF–THEN Mamdani fuzzy rules. Both HyFIS and HyFIS-Yager-gDIC share the same generic architecture as shown in Fig. 1 and the network consists of five layers of nodes. Layer 1 consists of the input linguistic nodes; layer 2 consists of the antecedent nodes; layer 3 is the rule nodes; layer 4 is the consequent nodes; and layer 5 consists of the output linguistic nodes. Each input node $IV_i, i \in \{1 \dots n_1\}$ in layer 1 takes in a single input value and the input vector is represented as $x = \{x_1, \dots, x_i, \dots, x_{n_1}\}$. Each output node $OV_m, m \in \{1 \dots n_5\}$ in layer 5 produces a single output value and the output vector is represented as $y = \{y_1, \dots, y_m, \dots, y_{n_5}\}$. In addition, the vector $d =$

S.W. Tung is with the Center for Computational Intelligence, School of Computer Engineering, Nanyang Technological University, Singapore

C. Quek is with the Center for Computational Intelligence, Block N4 #2A-32, School of Computer Engineering, Nanyang Technological University, Nanyang Avenue, Singapore, Singapore 639798 (email: ashcquek@ntu.edu.sg).

C. Guan is with the Institute for Infocomm Research, A*Star, Singapore

978-1-4244-3597-5/09/$25.00 ©2009 IEEE

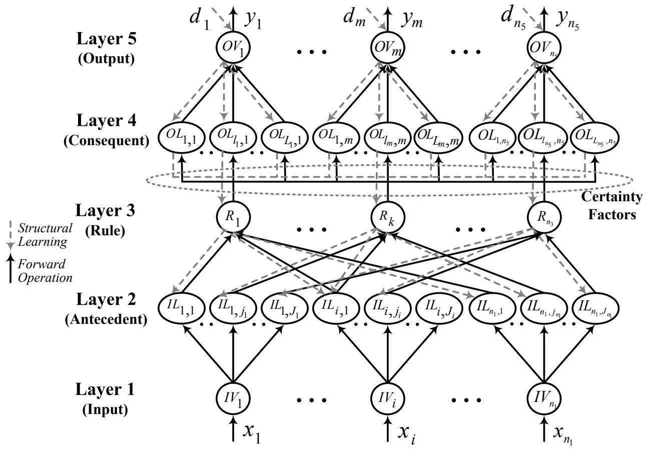

Fig. 1. Structure of HyFIS/HyFIS-Yager-gDIC.

$\{d_1, \ldots, d_m, \ldots, d_{n_5}\}$ represents the desired output during the parameter learning phase. For each input variable IV_i, it will consist of J_i number of fuzzy labels. Each antecedent node will be represented as $IL_{i,j_i}, j_i \in \{1 \ldots J_i\}$ and the total number of nodes in layer 2 is $n_2 = \sum_i J_i$. For each output variable OV_m, it will consist of L_m number of fuzzy labels. Each consequent node will be represented as $OL_{l_m,m}, l_m \in \{1 \ldots L_m\}$ and the total number of nodes in layer 4 is $n_4 = \sum_m L_m$. Layer 3 represents the rule nodes R_k where $k \in \{1 \ldots n_3\}$. Fuzzy rules with certainty factors, represented as weights in the synapses connecting layers 3 and 4 of the network, are used. The training parameters are the centres and the widths of the gaussian membership functions present in layers 2 and 4 defined by (1)

$$\mu(c, \sigma; x) = e^{-((x-c)^2/\sigma^2)} \qquad (1)$$

where c is the centre of the function and σ is the width of the function.

A. HyFIS: Hybrid neural Fuzzy Inference System

HyFIS adopts a two phase learning scheme. The first phase is the structure learning phase for knowledge acquisition. A fuzzy technique by Wang and Mendel [14] is used to derive an initial rulebase of the network. Prior knowledge about the number of clusters is required to produce an evenly spaced grid partition in the input and output dimensions. The second phase is the parameter learning phase. The input vector is propagated forward in the network following the Compositional Rule of Inference (CRI) scheme [15] and parameter tuning is performed by backpropagating the error using a gradient descending approach.

B. HyFIS-Yager-gDIC: Self-organizing Hybrid neural Fuzzy Inference System realizing Yager inference

Similar to the HyFIS network, HyFIS-Yager-gDIC also adopts a two phase learning scheme. In phase one, the self-organizing gDIC [10] is used to perform cluster partitioning in the input and output dimensions. In phase two, a supervised learning scheme based on a gradient descent learning is used to optimally tune the parameters of the membership functions. The two phase learning scheme allows HyFIS-Yager-gDIC to automatically formulate the initial fuzzy rules from raw numerical training data and subsequently evolve its structure through learning. In addition, the fuzzy Yager inference scheme [3] integrated into HyFIS-Yager-gDIC provides it with a firm and logical framework that closely emulates the human reasoning process.

III. T2-HyFIS-Yager: Type-2 fuzzy Yager inference based HyFIS

The proposed T2-HyFIS-Yager network is developed by mapping the interval Type-2 fuzzy Yager inference scheme onto the generic structure of the HyFIS/HyFIS-Yager-gDIC architecture as shown in Fig. 1. It adopts a two phase learning process. In phase one, the self-organizing extended gDIC technique is used to process noise corrupted raw numerical training data and automatically determines the number of clusters in each of the input and output dimensions. The extended Discrete Incremental Clustering (DIC) [12] technique was first proposed for trapezoidal membership functions. By implementing the extended gDIC technique for gaussian membership functions in the initialization phase of the T2-HyFIS-Yager network, prior knowledge about the number of clusters is not required. This helps to maintain a consistent representation of the fuzzy sets on a local basis, such that the number of labels in the input and output dimensions need not be the same. In phase two, a supervised learning scheme based on a gradient descending approach is used to optimally tune the parameters of the membership functions.

A. Implementation of Type-2 Fuzzy Yager Inference Scheme

The connectionist structure of the T2-HyFIS-Yager network is based on a set of IF–THEN Mamdani fuzzy rules that are formulated from the noise corrupted training data. The k-th fuzzy rule is expressed in the form

$$\begin{aligned} R_k :\ & \text{IF } x_1 \text{ is } \tilde{IL}_{(1,j_1)_k} \text{ and } \ldots \text{ and } x_i \text{ is } \tilde{IL}_{(i,j_i)_k} \\ & \text{and } \ldots \text{ and } x_{n_1} \text{ is } \tilde{IL}_{(n_1,j_{n_1})_k} \\ & \text{THEN } y_1 \text{ is } \tilde{OL}_{(l_1,1)_k} \text{ and } \ldots \text{ and } y_m \text{ is } \tilde{OL}_{(l_m,m)_k} \\ & \text{and } \ldots \text{ and } y_{n_5} \text{ is } \tilde{OL}_{(l_{n_5},n_5)_k} \end{aligned}$$

where $\tilde{IL}_{(i,j_i)_k}$ is the Type-2 j_i-th input label associated with the i-th input variable that is connected to R_k and $\tilde{OL}_{(l_m,m)_k}$ is the Type-2 l_m-th output label associated with the m-th output variable that is connected to R_k.

In the proposed T2-HyFIS-Yager network, the training parameters are the centers of the left and right formation gaussian functions of the Type-2 fuzzy labels present in layers 2 and 4 of the network as shown in Fig. 2(I). Each fuzzy label in the antecedent layer and consequent layer is defined by its footprint of uncertainty [8] given as (2)

$$\mu_{\tilde{A}}(x) = \left[\underline{\mu}_{\tilde{A}}(x), \overline{\mu}_{\tilde{A}}(x)\right] \qquad (2)$$

where $\tilde{A}$ denotes the Type-2 fuzzy set, and $\underline{\mu}_{\tilde{A}}(x)$ and $\overline{\mu}_{\tilde{A}}(x)$ are the lower and upper membership functions of $\tilde{A}$

respectively. They are defined as in (3) and (4)

$$\underline{\mu}_{\tilde{A}}(x) = \begin{cases} \mu_R(c_R, \sigma; x) & \text{if } x \leq \frac{c_L + c_R}{2} \\ \mu_L(c_L, \sigma; x) & \text{otherwise} \end{cases} \quad (3)$$

$$\overline{\mu}_{\tilde{A}}(x) = \begin{cases} \mu_L(c_L, \sigma; x) & \text{if } x \leq c_L \\ 1 & \text{if } c_L < x \leq c_R \\ \mu_R(c_R, \sigma; x) & \text{if } x > c_R \end{cases} \quad (4)$$

where $\mu_L(c_L, \sigma; x)$ and $\mu_R(c_R, \sigma; x)$ refers to the left and right formation gaussian functions respectively as defined in (1), and c_L and c_R are the centres of the left and right functions respectively.

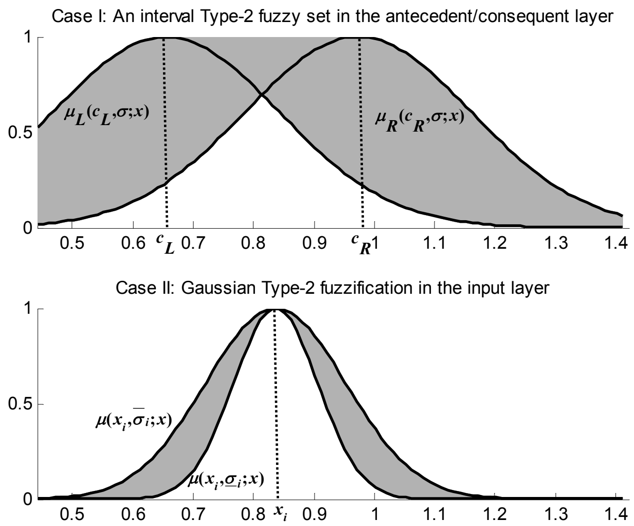

Fig. 2. (I) An interval Type-2 fuzzy set in the antecedent/consequent layer denoted by its left and right formation gaussian functions (II) Gaussian Type-2 fuzzification in the input layer.

1) Layer 1 – Gaussian Type-2 Fuzzification: Gaussian Type-2 fuzzification is employed in the input layer of the T2-HyFIS-Yager network to effectively handle the uncertainties in the training data that is corrupted by noise. Each input node $IV_i, i \in \{1 \ldots n_1\}$ in layer 1 takes in a single input value x_i and produces a gaussian Type-2 fuzzy set $\tilde{X}_i$ as shown in Fig. 2(II). $\tilde{X}_i$ is given by its footprint of uncertainty as in (5)

$$\mu_{\tilde{X}_i}(x) = \left[\underline{\mu}_{\tilde{X}_i}(x), \overline{\mu}_{\tilde{X}_i}(x)\right] \quad (5)$$

where the lower and upper memberships of $\tilde{X}_i$ are defined as in (6) and (7)

$$\underline{\mu}_{\tilde{X}_i}(x) = \mu(x_i, \underline{\sigma}_i; x) \quad (6)$$

$$\overline{\mu}_{\tilde{X}_i}(x) = \mu(x_i, \overline{\sigma}_i; x) \quad (7)$$

respectively. $\mu(x_i, \underline{\sigma}_i; x)$ and $\mu(x_i, \overline{\sigma}_i; x)$ are gaussian functions as defined in (1) with center x_i, where $\underline{\sigma}_i$ and $\overline{\sigma}_i$ define the lower and upper uncertainty bounds associated with the noise corrupted training data x_i. In this paper, it is empirically assumed that $\underline{\sigma}_i = 0.3\sigma_i^{\text{train}}$ and $\overline{\sigma}_i = 0.5\sigma_i^{\text{train}}$ where σ_i^{train} is the statistical variance of the input x_i that is computed from the training data.

2) Layer 2 – Antecedent Matching: Each antecedent node in layer 2 of the T2-HyFIS-Yager network is denoted as $\tilde{IL}_{i,j_i}, i \in \{1 \ldots n_1\}, j_i \in \{1 \ldots J_i\}$ as defined in (2), and it is the Type-2 j_i-th input label associated with the i-th input variable. The antecedent matching in the network essentially computes the degree of dissimilarity f_{i,j_i} between the gaussian Type-2 fuzzified input $\tilde{X}_i$ as given in (5) and the corresponding Type-2 input label $\tilde{IL}_{i,j_i}$. The degree of dissimilarity is an interval Type-1 fuzzy set given as in (8)

$$\begin{aligned} f_{i,j_i} &= \left[\underline{f}_{i,j_i}, \overline{f}_{i,j_i}\right] \\ &= 1 - \sqcup_x \left[\mu_{\tilde{X}_i}(x) \sqcap \mu_{\tilde{IL}_{i,j_i}}(x)\right] \end{aligned} \quad (8)$$

where $\sqcap$ and $\sqcup$ denote the meet and join operations [9] on Type-2 fuzzy sets respectively. Computationally, we have

$$\underline{f}_{i,j_i} = 1 - \sup_x \left[\overline{\mu}_{\tilde{X}_i}(x) \star \overline{\mu}_{\tilde{IL}_{i,j_i}}(x)\right] \quad (9)$$

$$\overline{f}_{i,j_i} = 1 - \sup_x \left[\underline{\mu}_{\tilde{X}_i}(x) \star \underline{\mu}_{\tilde{IL}_{i,j_i}}(x)\right] \quad (10)$$

where $\star$ is the min operator. $\underline{\mu}_{\tilde{IL}_{i,j_i}}(x)$ and $\overline{\mu}_{\tilde{IL}_{i,j_i}}(x)$ are the lower and upper membership functions of the Type-2 fuzzy set $\tilde{IL}_{i,j_i}$, as defined in (3) and (4), respectively.

3) Layer 3 – Rule Fulfillment: The set of IF-THEN Mamdani fuzzy rules that are induced from the training data are defined in the rule layer of the proposed T2-HyFIS-Yager network. The fuzzy rules denote the uncertain fuzzy relations that characterize the input-output mappings of the noise corrupted training data. Each rule node $R_k, k \in \{1 \ldots n_3\}$ functions to compute the overall degree of similarity between the inputs and the antecedents of the k-th fuzzy rule. The overall degree of similarity F_k is an interval Type-1 fuzzy set defined as in (11)

$$\begin{aligned} F_k &= \left[\underline{f}_k, \overline{f}_k\right] \\ &= 1 - \sqcup_{i=1}^{n_1} f_{(i,j_i)_k} \end{aligned} \quad (11)$$

where $f_{(i,j_i)_k}$ is the output of the antecedent node $\tilde{IL}_{i,j_i}$ that is connected to R_k, as shown in (8). Computationally, we have

$$\underline{f}_k = 1 - \cup_{i=1}^{n_1} \overline{f}_{(i,j_i)_k} \quad (12)$$

$$\overline{f}_k = 1 - \cup_{i=1}^{n_1} \underline{f}_{(i,j_i)_k} \quad (13)$$

where $\underline{f}_{(i,j_i)_k}$ and $\overline{f}_{(i,j_i)_k}$ denote the lower and upper bounds of the output of the antecedent node $\tilde{IL}_{i,j_i}$, as shown in (9) and (10) respectively, that is connected to R_k.

4) Layer 4 – Consequent Derivation: Each consequent node in layer 4 of the T2-HyFIS-Yager network is denoted as $\tilde{OL}_{l_m,m}, m \in \{1 \ldots n_5\}, l_m \in \{1 \ldots L_m\}$ as defined in (2), and it is the Type-2 l_m-th output label associated with the m-th output variable. Considering the firing effect of a single fuzzy rule R_k, the inferred fuzzy output set $\tilde{Y}^{R_k}_{(l_m,m)}$ from $\tilde{OL}_{l_m,m}$ is given as in (14)

$$\mu_{\tilde{Y}^{R_k}_{(l_m,m)}}(y) = 1 - \left[F_k \sqcap \left(1 - \mu_{\tilde{OL}_{l_m,m}}(y)\right)\right] \quad (14)$$

where F_k is the uncertain firing strength of the rule R_k due to the noise corrupted inputs, as defined in (11).

The fuzzy Yager inference scheme adopts the disjunctive model of fuzzy relation, where conclusions from multiple, parallel rules have to be combined in a conjunctive manner [11]. Since $\tilde{OL}_{l_m,m}$ may serve as output to more than one fuzzy rule, the overall inferred fuzzy output set $\tilde{Y}_{l_m,m}$ from $\tilde{OL}_{l_m,m}$ due to the activation of the rulebase of the proposed T2-HyFIS-Yager network is given as in (15)

$$\mu_{\tilde{Y}_{l_m,m}}(y) = \sqcap_{k\in K_{l_m,m}} \mu_{\tilde{Y}^{R_k}_{(l_m,m)}}(y) \qquad (15)$$

where $K_{l_m,m}$ denotes the set of fuzzy rules in T2-HyFIS-Yager that share the same output fuzzy label $\tilde{OL}_{l_m,m}$ as consequent.

The overall inferred fuzzy output set $\tilde{Y}_{l_m,m}$ is a Type-2 fuzzy set, and we can re-express it in terms of its footprint of uncertainty as in (16)

$$\mu_{\tilde{Y}_{l_m,m}}(y) = \left[\underline{\mu}_{\tilde{Y}_{l_m,m}}(y), \overline{\mu}_{\tilde{Y}_{l_m,m}}(y)\right] \qquad (16)$$

where $\underline{\mu}_{\tilde{Y}_{l_m,m}}(y)$ and $\overline{\mu}_{\tilde{Y}_{l_m,m}}(y)$ are the lower and upper membership functions of the Type-2 fuzzy set $\tilde{Y}_{l_m,m}$. Computationally, we have

$$\underline{\mu}_{\tilde{Y}_{l_m,m}}(y) = \left[1 - \cup_{k\in K_{l_m,m}} \overline{f}_k\right] \vee \underline{\mu}_{\tilde{OL}_{l_m,m}}(y) \qquad (17)$$

$$\overline{\mu}_{\tilde{Y}_{l_m,m}}(y) = \left[1 - \cup_{k\in K_{l_m,m}} \underline{f}_k\right] \vee \overline{\mu}_{\tilde{OL}_{l_m,m}}(y) \qquad (18)$$

where $\vee$ is the max operator. $\underline{\mu}_{\tilde{OL}_{l_m,m}}(y)$ and $\overline{\mu}_{\tilde{OL}_{l_m,m}}(y)$ are the lower and upper membership functions of the Type-2 fuzzy set $\tilde{OL}_{l_m,m}$, as defined in (3) and (4), respectively.

5) Layer 5 – Type Reduction and Defuzzification: Each output node $OV_m, m \in \{1 \dots n_5\}$ in layer 5 of the proposed T2-HyFIS-Yager network performs two functions: (1) type-reduction of the overall inferred Type-2 fuzzy set $\tilde{Y}_m$ to the corresponding Type-1 fuzzy set Y_m, where $\tilde{Y}_m = \cup_{l_m=1}^{L_m} \tilde{Y}_{l_m,m}$ is the combination of all the inferred fuzzy output sets $\tilde{Y}_{l_m,m}$, as defined in (16), of its fuzzy labels; and (2) the defuzzification of this Type-1 fuzzy set Y_m to a crisp output value y_m. In this paper, the height type reduction (HTR) [2] method is adopted for the proposed T2-HyFIS-Yager network. To perform HTR, each inferred fuzzy output set $\tilde{Y}_{l_m,m}$ of the output label node $OL_{l_m,m}$ is replaced by a Type-2 singleton $\tilde{Y}^{\star}_{l_m,m}$. $\tilde{Y}^{\star}_{l_m,m}$ is a Type-2 singleton fuzzy set whose domain consists of a single point $y^{\star}_{l_m,m}$, and the membership grade of $y^{\star}_{l_m,m}$ is an interval set in $[0, 1]$. For T2-HyFIS-Yager, $y^{\star}_{l_m,m}$ is defined to be the midpoint of the domain of $\tilde{Y}_{l_m,m}$ such that its membership grade is given as $\left[\underline{\mu}_{\tilde{Y}_{l_m,m}}(y^{\star}_{l_m,m}), \overline{\mu}_{\tilde{Y}_{l_m,m}}(y^{\star}_{l_m,m})\right]$ where

$$\underline{\mu}_{\tilde{Y}_{l_m,m}}(y^{\star}_{l_m,m}) = \min_y \underline{\mu}_{\tilde{Y}_{l_m,m}}(y) \qquad (19)$$

$$\overline{\mu}_{\tilde{Y}_{l_m,m}}(y^{\star}_{l_m,m}) = \min_y \overline{\mu}_{\tilde{Y}_{l_m,m}}(y) \qquad (20)$$

respectively. That is, $\tilde{Y}^{\star}_{l_m,m} = \mu_{\tilde{Y}^{\star}_{l_m,m}}(y^{\star}_{l_m,m}) = \left[\underline{\mu}_{\tilde{Y}_{l_m,m}}(y^{\star}_{l_m,m}), \overline{\mu}_{\tilde{Y}_{l_m,m}}(y^{\star}_{l_m,m})\right]$ is employed to denote the inferred fuzzy output set $\tilde{Y}_{l_m,m}$. The type-reduced set Y_m of OV_m is then given as in (21)

$$Y_m = \int_{\rho_{1,m}} \cdots \int_{\rho_{L_m,m}} 1 \Big/ \frac{\sum_{l_m=1}^{L_m} y^{\star}_{l_m,m}\left(\frac{1-\rho_{l_m,m}}{1+\sigma_{l_m,m}}\right)}{\sum_{l_m=1}^{L_m}\left(\frac{1-\rho_{l_m,m}}{1+\sigma_{l_m,m}}\right)} \qquad (21)$$

where $\rho_{l_m,m} \in \tilde{Y}^{\star}_{l_m,m}$, $\sigma_{l_m,m}$ is the width of the left/right gaussian formation function of the Type-2 fuzzy label $\tilde{OL}_{l_m,m}$, and $Y_m = \left[Y_m^{\min}, Y_m^{\max}\right]$ is an interval Type-1 fuzzy set from the HTR process. The iterative algorithm described in [2] is used to empirically compute $Y_m^{\min}$ and $Y_m^{\max}$, and the crisp output value y_m is obtained through the defuzzification of Y_m by $y_m = \frac{1}{2}[Y_m^{\min} + Y_m^{\max}]$.

B. Parameter Learning Phase

The supervised learning algorithm of the T2-HyFIS-Yager is based on a gradient descent approach to minimize the error function

$$E = \frac{1}{2}\sum_X \sum_m [d_m - y_m]^2 \qquad (22)$$

where X is the training data set.

IV. Experimental Results

The evaluation of the performance of T2-HyFIS-Yager is done by applying it to the forecasting of a Mackey-Glass chaotic time-series. The strength and potential of the proposed T2-HyFIS-Yager is revealed through a comparison with the HyFIS and the HyFIS-Yager-gDIC models.

A. Prediction of Chaotic Dynamic System: Mackey-Glass

The chaotic time-series is generated by a delay differential equation

$$\frac{\partial x(t)}{\partial t} = \frac{\alpha x(t-\tau)}{1 + x^{\gamma}(t-\tau)} - \beta x(t) \qquad (23)$$

which was first investigated by Mackey and Glass [6]. In this study, 1000 input-output data samples which consist of four past values of $x(t)$ are used, i.e.,

$$[x(t-18), x(t-12), x(t-6), x(t)\ ; x(t+6)]\ .$$

There are four input dimensions and one output dimension to the system. The first 500 samples were used as the training set, while the remaining 500 pairs were used to test the performance of T2-HyFIS-Yager. Six different cases are considered in this study: the first case consists of data as described above, while noise of 3%, 4%, 5% and 6% of the power of the original signal are added in the next four cases, and the last case is when random level noise ranging from 3% to 6% of the power of the original signal is added. A measure of prediction accuracy is given by the root mean square error ($RMSE$) defined as in (24)

$$RMSE = \left[\frac{1}{n_5}\sum_{m=1}^{n_5}(d_m - y_m)^2\right]^{1/2}\ . \qquad (24)$$

TABLE I
COMPARISON ON MACKEY-GLASS PREDICTION

Type of Data	Method	$RMSE_{test}$
No Noise	HyFIS	0.0220
	HyFIS-Yager-gDIC	**0.0190**
	T2-HyFIS-Yager	0.0694
3% Noise	HyFIS	0.0266
	HyFIS-Yager-gDIC	**0.0236**
	T2-HyFIS-Yager	0.0694
4% Noise	HyFIS	0.0257
	HyFIS-Yager-gDIC	**0.0229**
	T2-HyFIS-Yager	0.0694
5% Noise	HyFIS	0.0639
	HyFIS-Yager-gDIC	**0.0597**
	T2-HyFIS-Yager	0.0838
6% Noise	HyFIS	0.1915
	HyFIS-Yager-gDIC	0.1526
	T2-HyFIS-Yager	**0.1134 (-25.7%)**
Unknown Noise Level	HyFIS	0.1296
	HyFIS-Yager-gDIC	0.1176
	T2-HyFIS-Yager	**0.0997 (-15.2%)**

B. Simulations

The three models, HyFIS, HyFIS-Yager-gDIC and the proposed T2-HyFIS-Yager, are used to perform time-series forecasting when (1) there is no noise in the signal, (2) noise of known SNR is added into the signal, and (3) noise of unknown SNR is added into the signal. The results are shown in Table I. From Table I, the HyFIS-Yager-gDIC model outperforms the HyFIS model in both the noise-free and the noisy data. The performance of the HyFIS-Yager-gDIC model is the best out of the three models when the signal is noise-free and when the noise level is low in the signal. However, when the noise level is above a threshold, the proposed T2-HyFIS-Yager significantly outperforms the other two models in the $RMSE$ achieved based on the test data. A reduction of 25.7% in the $RMSE_{test}$ is achieved for the proposed T2-HyFIS-Yager model over the HyFIS-Yager-gDIC model when the noise level is set at 6%. Furthermore, T2-HyFIS-Yager also obtains significantly better results over the HyFIS and the HyFIS-Yager-gDIC models when noise of unknown SNR is present in the data. For the unknown noise level data, there is a reduction of 15.2% in the $RMSE_{test}$ for the proposed T2-HyFIS-Yager model when compared to the HyFIS-Yager-gDIC model. Although the proposed model do not perform as well as the Type-1 models during the zero noise and low noise cases, we can observe that the presence of low noise level in the original signal do not affect the performance of T2-HyFIS-Yager as seen by the constant $RMSE_{test}$ value of 0.0694 obtained by T2-HyFIS-Yager for the zero-noise, 3%-noise and 4%-noise cases.

Further investigations are shown in Figs. 3–4. Fig. 3 shows the $RMSE$ curves for the HyFIS-Yager-gDIC for the noisy cases (when noise of 4 %, 5 % and 6 % are added to the data). For each case, 150 epochs of training is applied. As seen from the figure, the training $RMSE$ curves show good convergence for all the three cases, indicating that the HyFIS-Yager-gDIC model is able to learn and generalize the acquired knowledge well. However, the testing $RMSE$ curves only show good convergence for the 4 % and 5 % noise cases, while the testing $RMSE$ curve do not converge for that of the 6 % noise level. This suggests that while the HyFIS-Yager-gDIC model is able to handle low levels of noise in the data, it is unable to cope when higher levels of noise are present in the data. On the other hand, Fig. 4 presents the $RMSE$ plots of the proposed T2-HyFIS-Yager model for the training and testing data for the four cases when known noise of 4%, 5%, 6%, and unknown noise are present in the data. Similarly, 150 epochs of training is applied. As seen from the figure, the training and the testing $RMSE$ curves show good convergence for all the four cases, indicating that the T2-HyFIS-Yager model is able to learn and generalize the acquired knowledge well. Comparing Fig. 3 and Fig. 4, we can conclude that the proposed T2-HyFIS-Yager model has demonstrated good efficiency in modeling inaccuracies when the noise level is high and when noise of unknown SNR is present in the original signal.

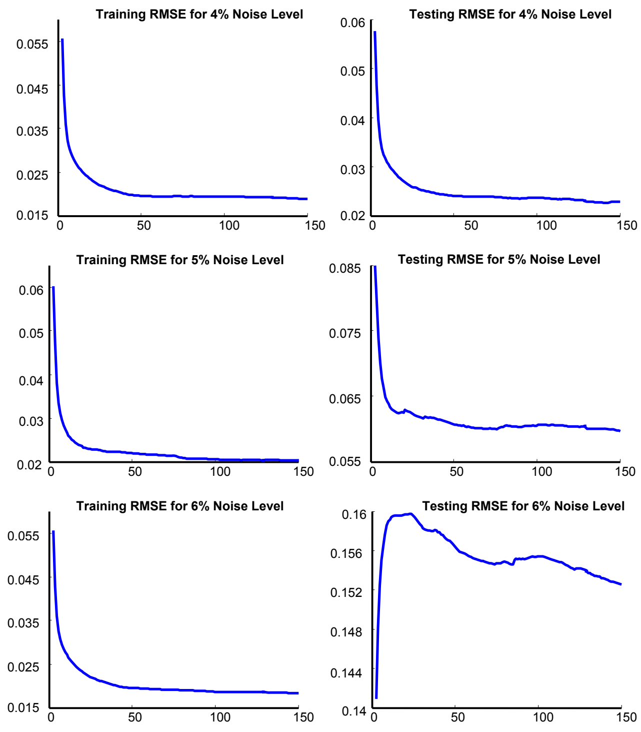

Fig. 3. $RMSE$ curves of HyFIS-Yager-gDIC for Mackey-Glass prediction.

V. CONCLUSIONS

This paper proposes a self-organizing Type-2 fuzzy Yager based Hybrid neural Fuzzy Inference System named T2-HyFIS-Yager. The proposed T2-HyFIS-Yager network integrates the mathematical formalism of Type-2 fuzzy logic inference with the self-organizing Yager based HyFIS inference network, which allows for the robust learning and reasoning

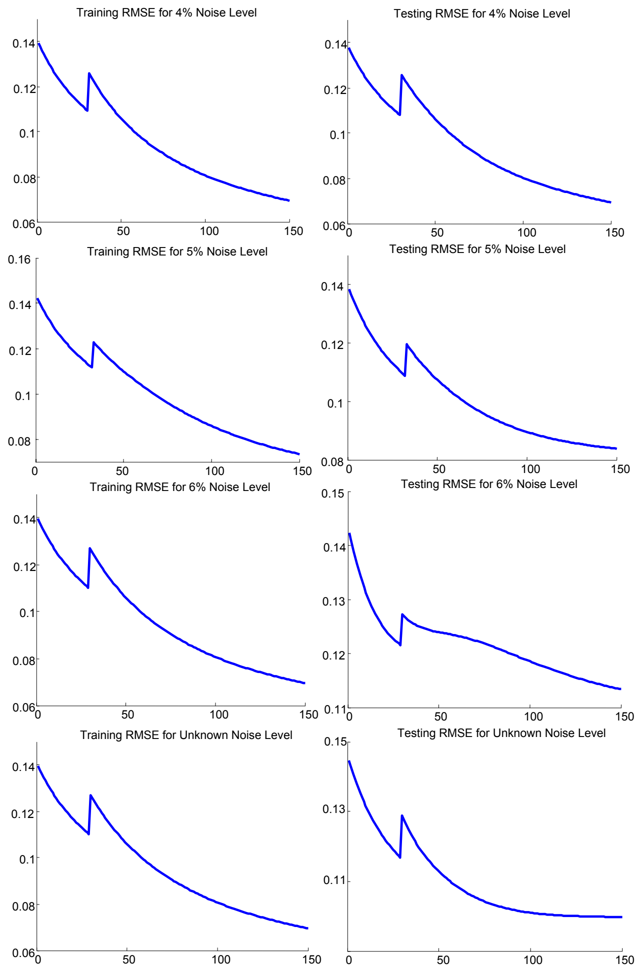

Fig. 4. $RMSE$ curves of T2-HyFIS-Yager for Mackey-Glass prediction.

with noise corrupted data. The system has been used to perform time-series forecasting and the superior performance has shown that the T2-HyFIS-Yager model is effective in modeling signals with known and unknown SNR.

References

[1] N. N. Karnik and J. M. Mendel, "Introduction to type-2 fuzzy logic systems," *Proc. IEEE World Congress on Computational Intelligence*, vol. 2, May 1998, pp. 915–920.

[2] N. N. Karnik and J. M. Mendel, "Type-2 fuzzy logic systems," *IEEE Transactions on Fuzzy Systems*, vol. 7-6, pp. 643-658, 1999.

[3] J. M. Keller, R. R. Yager, and H. Tahani, "Neural network implementation of fuzzy logic," *Fuzzy Sets and Systems*, vol. 45, pp. 1–12, 1992.

[4] J. Kim and N. Kasabov, "HyFIS: Adaptive neuro-fuzzy inference systems and their application to nonlinear dynamical systems," *Neural Networks*, vol. 12, pp. 1301–1319, 1999.

[5] Q. Liang and J. M. Mendel, "Interval type-2 fuzzy logic systems: Theory and design," *IEEE Transactions on Fuzzy Systems*, vol. 8-5, pp. 535–550, 2000.

[6] M. Mackey and L. Glass, "Oscillation and chaos in physiological control systems," *Science*, vol. 197, pp. 287–289, 1977.

[7] E. H. Mamdani, "Application of fuzzy logic to approximate reasoning using linguistic systems," *IEEE Transactions on Computers*, vol. C-26, pp. 1182–1191, 1977.

[8] J. M. Mendel and R. I. B. John, "Type-2 fuzzy sets made simple," *IEEE Transactions on Fuzzy Systems*, vol. 10-2, pp. 117–127, 2002.

[9] M. Mizumoto and K. Tanaka, "Some properties of fuzzy sets of type 2," *Information and Control*, vol. 31, pp. 312–340, 1976.

[10] M. N. Nguyen, D. Shi, and C. Quek, "Self-organizing gaussian fuzzy CMAC with truth value restriction," *ICITA*, vol. 2, pp. 185–190, 2005.

[11] R. J. Oentaryo, M. Pasquier, and C. Quek, "GenSoFNN-Yager: A novel brain-inspired generic self-organizing neuro-fuzzy system realizing yager inference," *Expert Systems with Applications*, vol. 35-4, pp. 1825–1840, 2008.

[12] W. L. Tung and C. Quek, "A brain-inspired fuzzy semantic memory model for learning and reasoning with uncertainty," *Neural Computing and Applications*, Springer, 2007.

[13] S. W. Tung, C. Quek, and C. Guan, "HyFIS-Yager-gDIC: A self-organizing hybrid neural fuzzy inference system realizing yager inference," *ICONIP*, 2008.

[14] L. X. Wang and J. M. Mendel, "Generating fuzzy rules by learning from examples," *IEEE Transactions on Systems, Man, and Cybernetics*, vol. 22, pp. 1414–1427, 1992.

[15] L. A. Zadeh, "Calculus of fuzzy restrictions," *Fuzzy Sets and Their Applications to Cognitive and Decision Processes*, pp. 1–39, 1975.