Article

Modeling the Renewable Energy Systems by a Self-Evolving Nonlinear Consequent Part Recurrent Type-2 Fuzzy System for Power Prediction

Jafar Tavoosi[1,*], Amir Abolfazl Suratgar[2], Mohammad Bagher Menhaj[2], Amir Mosavi[3*], Ardashir Mohammadzadeh4 and Ehsan Ranjbar2

1. Dept. of Electrical Engineering, Faculty of Eng., Ilam University, Ilam, Iran
2.Center of Excellence on Control and Robotics, Dept. of Electrical Engineering, Amirkabir University of Technology, 424 Hafez Ave, Tehran, Iran.
3. Faculty of Civil Engineering, Technische Universität Dresden, 01069 Dresden, Germany
4. Department of Electrical Engineering; University of Bonab, Bonab, Iran
Corresponding: j.tavoosi@ilam.ac.ir , amir.mosavi@mailbox.tu-dresden.de

Abstract: A novel Nonlinear Consequent-Part Recurrent-Type-2 Fuzzy System (NCPRT2FS) is presented for renewable energy systems modeling. Not only does this paper present a new architecture of type-2 fuzzy system (T2FS) for identification and behavior prognostication of an experimental solar cell set and a wind turbine, but also it brings forward an exquisite technique to acquire an optimal number of membership functions (MFs) and the corresponding rules. Using nonlinear functions in the "Then" part of fuzzy rules, introducing a new mechanism in structure learning, using adaptive learning rate and convergence analysis of the learning algorithm are the innovations of this paper. Another novel innovation is using some optimization techniques (including pruning fuzzy rules, initial adjustment of MFs). Eventually, solar cell photo-voltaic and wind turbine are deemed as case studies. The experimental data are exploited and the consequent yields emerge so persuasive. The root mean square error is less than 0.006 and the number of fuzzy rules is equal or less than 4 rules, which indicates the very good performance of the presented fuzzy neural network. Finally, the obtained model is used for the first time for a geographical area for feasibility of renewable energies.

Keywords: Self-Evolving; Nonlinear Consequent-Part; Convergence Analysis; Renewable Energy; Type-2 Fuzzy; artificial intelligence

1. Introduction

Neural networks share lots of significant benefits such as landmark computation ability, parallel processing and adaptation. The fuzzy systems are competent of utilization of the expert knowledge entitled by "if-then rules" and own actual parameter concepts. As everyone knows, mathematical modeling is a substantial preliminary in many control issues. On the other hand, prediction, simulation and modeling of complicated systems established upon physical and chemical principals appear so industrious in such a way that they will not yield consolidated mathematical forms [1]. One may suggest system identification as a solution to cope with this problematic issue.

This method puts the mathematical equations at the access utilizing input-to-output data analysis for increasing the efficiency of dynamic process calculations [2]. Computational intelligence lies among the efficient methods with an excellent fulfillment. Many papers have recently been published on fuzzy modeling and identification. Nonlinear system identification, founded on fuzzy and neuro-fuzzy models, was surveyed [3]. Computational intelligence gets so feasible in the area of renewable energy [4]. For design MPPT control [5], solar water heater selection [6], Photovoltaic system failure diagnosis [7] and solar power plant location alternatives [8] the computational intelligence has been used. Neural networks were also used by Grahovac et al. [9] in order to model and make anticipation of bio-ethanol generation from intermediates and byproducts yielded in beet-to-sugar procedure. The productivity of the neuro-fuzzy controller in extraction of the maximum yield by flow and energy optimization was demonstrated by Khiareddine et al. [10] in comparison with fuzzy and algorithm controllers. It was asserted that the neuro fuzzy controller is worthy of being implemented in experimental agricultural station at Sahline in Tunisia. Ocario et al. [11] testified wind power forecasts in the Portuguese system exploiting a novel hybrid evolutionary-adaptive methodology. Etemadi et al. [12] predicted the wind power produced by data-driven fuzzy modeling.

Type-2 fuzzy (T2F) logic which appears more capable and flexible in comparison to type-1 has been being inquired during the last ten years. A novel method was suggested for general T2F clustering by Doostparast et al. [13]. Some other applications of T2F sets can be find in textile engineering [14] and aerospace engineering [15]. Fuzzy c-means clustering and high order cognitive map were exerted by Lu in order to model and predict time series on the basis of type-1 fuzzy sets [16]. T2FS identification has engrossed so many researchers [17-23]. Abiyev et al. [17] took advantage of T2F clustering to organize construction of a wavelet type-2 TSK fuzzy neural system. They brought forth an adaptive law to update parameters of antecedent part and ultimately employed gradient learning algorithm to bring parameters of the descendant part up to date. T2FSs were applied for elicitation of fuzzy rules and casting derogatory features off [24]. The proposed mechanism took benefit of self-evolution capability in such a way that identification of the integral structure of the network would get efficient and there would be no requirement for initial start-up of network structure. The antecedent part and modulation parameters are trained in order to hold parameter learning in the network true utilizing error of back-propagation. Tuning parameters of the resultant part, the rule-ordered Kalman filter algorithm assists in network sharpness amelioration. Genetic algorithm [25] and PSO [26] are among the learning mechanism of T2F neural networks which have been conversed and scrutinized so far. Research development on T2F systems has brought about their vast usages in so many various fields such as time-series prediction [27], DC motor control [28], clinical practice guideline encryption [29], pattern recognition [30], robot control [31] and control of nonlinear systems [32,33]. A new smart type reduction is held forth in [34]. A T2FS learned through its type-1 counterpart in [35]. The Learning process was held true merging and extending the type-1 membership functions. Henceforth, the novel constructed T2FS went under implementation on a programmable chip.

 doi:10.20944/preprints202103.0178.v1

It is worthy notifying that most of the control engineers and system analyzers get actual systems represented in nonlinear dynamics; not only do these system outputs momentarily turn dependent upon the input, but also they appear reliant on the delayed inputs/outputs. This leads to a responsible consideration of both external and internal dynamics as a non-negligible essential remark in system modeling. Delayed inputs/outputs have to be used in external dynamics. Another feedback denoted as "recurrent neuron" have to be exerted in internal one either. Wu et al. [36] presented the solution of recurrent fuzzy neural network for the problematic temporal classification. Not only does this paper contribute to minimization of the cost function utilizing recurrent fuzzy neural network, but also it proposes maximization of the discriminability adopting a novel approach. Some modern recurrent fuzzy systems get presented in [37]. The special kind of neural network in the resultant part functions input variables in a nonlinear manner. Hardly have there been studies on recurrent T2F systems so far. Some of them are surveyed in following. A contributive recurrent interval T2FS is held forth in order to identify nonlinear systems in [30]. The novel technique requires initial knowledge about system order and the number of delayed inputs as well. Furthermore, the convergence issue in the learning algorithm is not taken into consideration and conversed even theoretically. Juang et al. [15] put forth another contributive recurrent T2F neural network to model dynamical systems. That there is not any rule pruning, is the major defect with their work. It could bring about extremely overlapped fuzzy sets. Soft switching of nonlinear model is superior to linear one in order to identify nonlinear systems [1]. Consequently, our suggested technique is established upon nonlinear resultant part in fuzzy rules. Rarely may one find comprehensive works on nonlinear consequent part in fuzzy systems; however, some of the studies on the arena are shortly surveyed in following. Reduction of the number of rules was carried out by Moodi in a fuzzy system using TSK fuzzy model accompanied by nonlinear consequent part [38]. The resultant of a rule is supposed to comprise a linear term and a nonlinear one. In their attempts, the numerous rules get decreased and the model precision simultaneously shows increase at the cost of complication abundance in the fuzzy model. The NFNN was constructed applying fuzzy rules which merge nonlinear functions. Linear consequent part requires more rules on the ground of achieving the aspired precision during modeling complicated nonlinear processes. The more number of rules is equal with the more number of neurons [39]. Some recently work on T2F neural networks can be seen in many applications such as 2DOF robot control [40], 3 parallel robot control [41], PMSM control [42], water temperature control [43,44], environmental temperature control [45] and UAV control [46]. Tavoosi and Badamchizadeh [47] proposed a T2S with linear "then part" for dynamic modeling. Their pivotal contribution was rule pruning in such a way that increase in learning speed would be targeted to attain reduction of the parameters in both MF parameters and descendant parts. Tavoosi et al [48,49] made another contribution to the issue holding forth a novel technique for analyzing stability of one class of T2F systems. Another analyzing method of stability was also suggested by Jahangiri et al. [50]. Suratgar and Nikravesh [51] proposed a modern technique of fuzzy linguistic modeling and the integral stability analysis as well. In [52] a fuzzy neural network has been used for the wind speed forecasting. In [53] a comparison between ANFIS and autoregressive method for wind speed/power prediction has been done. In [54] a fuzzy control on basis of predictive technique for a governing system has been presented. In [55] a multilayer perceptron is combined with an adaptive fuzzy system to forecast performance of a wind turbine. Some disadvantages and shortcomings of the works studied above are: lack of convergence proof, long training time (not usable in online applications), high complexity of the model, lack of proper accuracy. On the other hand, so far no applied research has been done to use renewable energies in the Ilam region.

So, this paper proposes NCPRT2FS for nonlinear system identification. The nonlinear systems here are the same as solar cells and wind turbines. The objective of identifying the system is to use it to specify the efficiency of the renewable energy system in the Ilam region. The innovations of this article that set it apart from other works are as follows: 1- Using nonlinear functions in the "Then" part of fuzzy rules. 2- Introducing a new mechanism in structure learning. 3- Using adaptive learning rate (Different from the previous works of others). 4- Convergence analysis of the T2F neural network learning algorithm presented in this article. 5- Finally, in this article, some optimization techniques (including pruning fuzzy rules, initial adjustment of MFs, etc.) are performed. The paper gets sectioned in six divisions. Section 2 comes next surveying T2F logic shortly. Section 3 entails inspection of the structure of NCPRT2FS finally presenting identification of structure and parameters. Learning convergence of NCPRT2FS is subsumed relying upon Lyapunov theory in section 4. Section 5 holds forth simulative identification studies taking into account solar cell photo-voltaic and wind turbine as the case studies and utilizing their experimental data.

2. A Review on Type-2 Fuzzy Sets and Systems

Firstly, Zadeh brought forward type-1 fuzzy logic, and introduced T2F one in order to improve resolution in some problems of type-1 ten years later. He deemed a fuzzy set which its MF was a fuzzy one entitled "type-2 fuzzy set". T2F sets may be usually exploited when the determination of accurate membership function turns so arduous. For instance, some time series prediction lie among the problematic cases which necessitate usage of T2F sets. Hence, exploiting T2F sets emerges so advantageous in order to describe some system behaviors.

Certain defects with type-1 fuzzy sets were scrutinized by Castro et al. [56]. Research on T2F systems were so confined before years of 1998. Critical and controversial questions and debate on T2F logic and its usage commenced after publication of a book which contained solidarity and intersection of T2F sets [57]. Extensive information on T2FS computation, such as defuzzification and type reduction, was suggested by Mendel [58]. A general T2F set, $\tilde{A}$, may be specified by (1):

$$\tilde{A} = \int_{x\in X} \mu_{\tilde{A}}(x)/x = \frac{\int_{x\in X}\left[\int_{\mu\in J_x}\frac{f_x(\mu)}{\mu}\right]}{x} \quad (1)$$

where $\mu_{\tilde{A}}(x)$ is a secondary MF, J_x represents the primary membership of $x \in X$, with $\mu \in J_x$, and $f_x(\mu) \in [0,1]$ denotes a secondary membership. The primary and secondary MFs in Gaussian form are illustrated in Fig. 1.

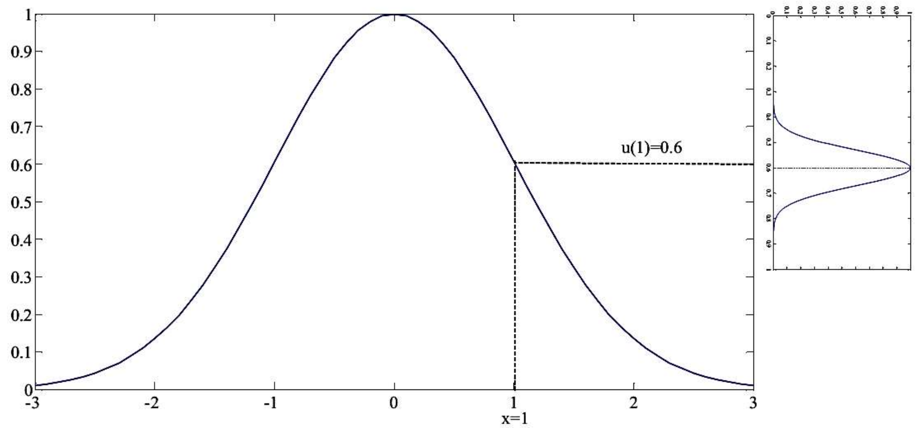

Figure 1. Primary and secondary MF.

Note that, the secondary MFs will be interval sets and the fuzzy set would be interval T2F ones while $f_x(\mu) = 1, \forall \mu \in J_x \subseteq [0,1]$. For more explanation, a crisp number would be fuzzified in two stages supposing that Gaussian MF were exerted to attain a T2F number. First off,

$$\mu_1 = \exp\left(-0.5.\frac{(x-M)^2}{{\sigma_x}^2}\right) \quad (2)$$

where μ_1 is primary membership, M and σ_x are the primary mean and spread of Gaussian MF, respectively, then

$$\mu_2(x,\mu_1) = \exp\left(-0.5.\frac{(a-\mu_1(x))^2}{{\sigma_m}^2}\right) \quad (3)$$

where $\mu_2(x,\mu_1)$ is secondary degree, $a \in [0,1]$ is the domain of secondary MF for each x and σ_m is the secondary spread of Gaussian MF.

Simple and special sort of general T2F sets turn as interval T2F one. Figure 2 depicts two interval T2F sets. A fuzzy set specified by a MF of Gaussian form with the mean value of m and a standard deviation amount in interval of $[\sigma_1,\sigma_2]$ is demonstrated on Fig 2.a.Two cases of interval T2F sets are given in Fig 2. Fig 2.b illustrates a fuzzy set with a MF of Gaussian form encompassing a distinct standard deviation of σ. However, the mean value is quite uncertain and adopts values in the interval of $[m_1, \ m_2]$.

Preprints (www.preprints.org) | NOT PEER-REVIEWED | Posted: 5 March 2021 doi:10.20944/preprints202103.0178.v1

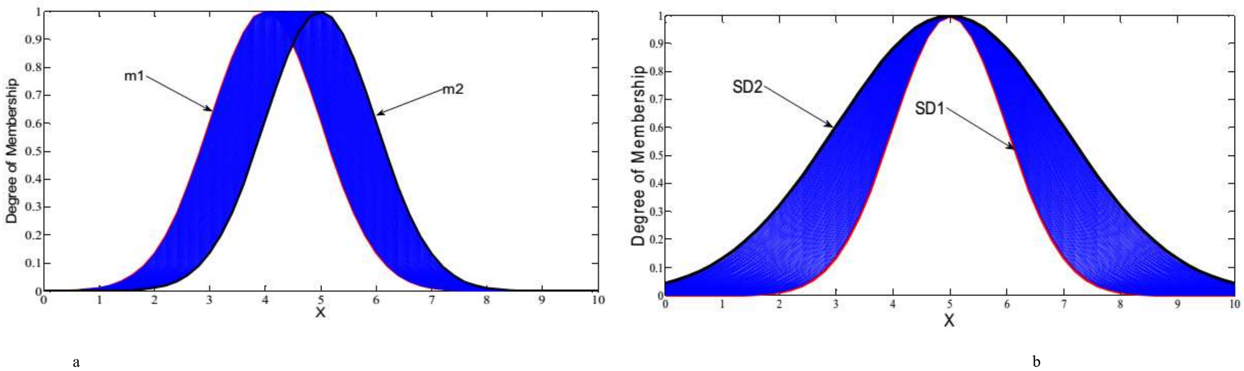

Figure 2. a) Uncertainty in width b) uncertainty in center.

MF of Gaussian form with determined standard deviation of σ and uncertain mean, seen in Fig. 2.a, is applied through all this paper.

2.1. Type-2 Fuzzy Systems

One may gain a crisp number defuzzifying the type-1 fuzzy [59] system-output whereas T2FS yields a T2F set. That is the reason one has to make endeavor to succeed in reduction of fuzzy set type from two to one in a process entitled "Type Reduction". The process is a challenging issue of high significance in T2F systems [60]. Figure 3 displays the structure of a T2F system.

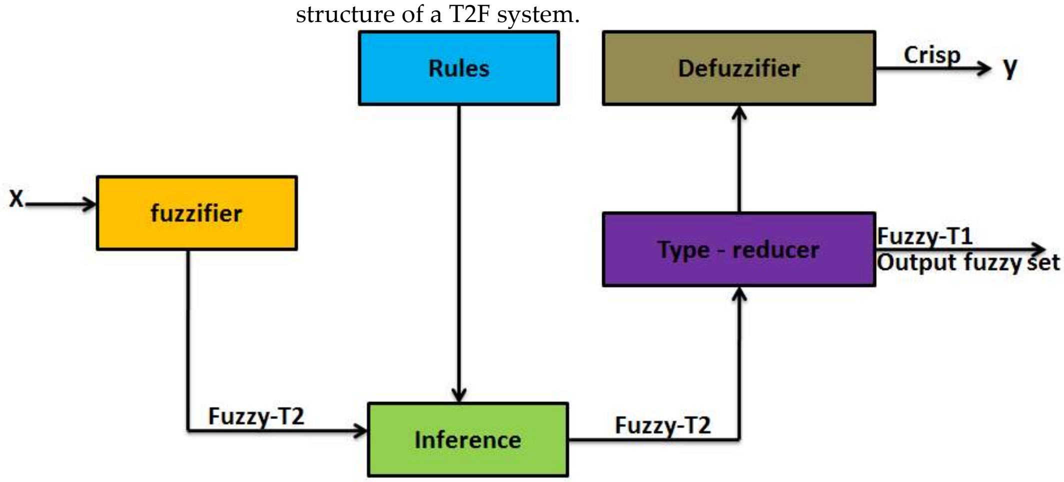

Figure 3. The structure of a T2F system.

As it could be easily grasped through Fig 4, construction of the T2FS will be the same as organization of type-1 if the "Type-Reduction" block is neglected.

3. The Proposed NCPRT2FS

Section 3 tries to consolidate the nonlinear descendant or resultant part of recurrent T2F systems into a formula. Reviewing two informative useful points mentioned later, the descriptive equation of (1) establishes the kth rule;

1) Type-2 TSK fuzzy systems, usually yield a polynomial constructive of the inputs,

2) The output and its coefficients in are type-1 fuzzy sets [61].

This paper recommends a novel NCPRT2FS which its total construction is illustrated in Fig. 4. As one may see, the system obviously embodies seven layers. Generally speaking, the kth rule would be demonstrated in following terms in a first-order interval type-2 linear TSK model with M rules and n inputs:

$$R^k: if\ x_1\ is\ \tilde{A}_1^k\ and\ \dots and\ x_n\ is\ \tilde{A}_n^k\ then\ \tilde{y}_k = C_{k,0} + C_{k,1}x_1 + \dots + C_{k,n}x_n \quad (5)$$

where $k = 1, \dots, M$ is the rules number, $x_i (i = 1, \dots, n)$ are inputs, $\tilde{y}_k$ is output of the kth rule. $\tilde{y}_k$ is an interval type-1 fuzzy set $\tilde{A}_i^k$ are antecedent sets, $C_{k,i} \in [c_{k,i} - s_{k,i}, c_{k,i} + s_{k,i}]$ represent consequent sets, that $c_{k,i}$ represents the center of $C_{k,i}$ and $s_{k,i}$is the spread of $C_{k,i}$.

In this paper, nonlinear consequent part is taken into account. The resulting kth rule in NCPRT2FS which has got two antecedent variables and three outputs with delayed time-shift ranging from one unit to 3 in the descendant part is demonstrated in (2):

$$R^k: if\ x_1\ is\ \tilde{A}_1^k\ and\ x_2\ is\ \tilde{A}_2^k\ then$$

$$\tilde{y}_k = C_{k,0} + C_{k,1}x_1 + C_{k,2}x_2 + C_{k,3}y(t-1) + C_{k,4}x_1x_2 + C_{k,5}x_1y(t-1) + C_{k,6}x_2y(t-1)$$

$$+C_{k,7}x_1^2 + C_{k,8}x_2^2 + C_{k,9}y^2(t-1) + C_{k,10}x_1x_2y(t-1) \quad (6)$$

One may make an extension to fuzzy rule (2) considering n antecedent variables and time-delayed outputs in descendant part with delaying shift in time ranging from one unit to m units. n may be designed remarking nonlinearity degree and complexity of the unknown system which is going to be identified next.

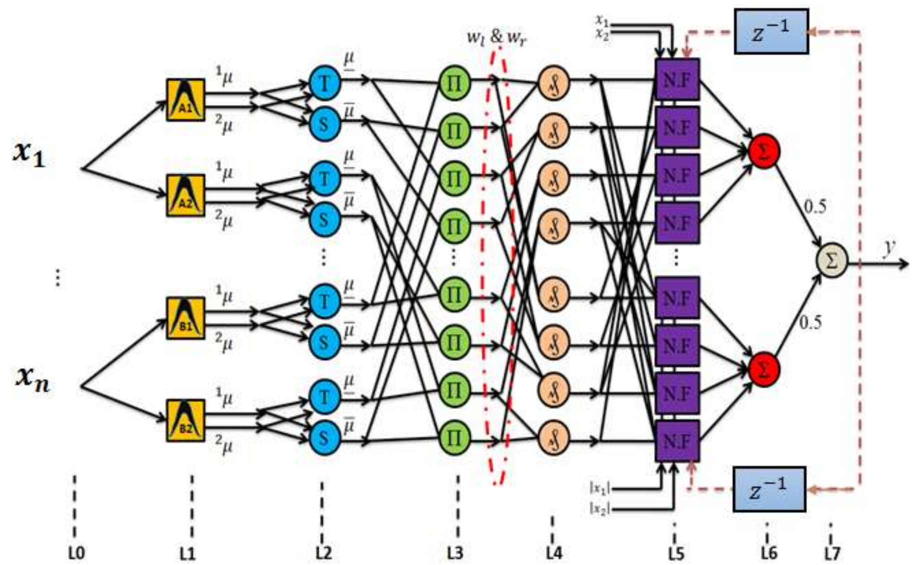

Figure 4. The structure of the proposed NCPRT2FS.

The layers' details are as:

Layer 0: This layer is inputs layer.

Layer 1: The output of fuzzification are written as:

$${}^1\mu_{k,i}(x_i, [\sigma_{k,i}, {}^1m_{k,i}]) = e^{-.5\left(\frac{x_i - {}^1m_{k,i}}{\sigma_{k,i}}\right)^2} \quad (5)$$

$${}^2\mu_{k,i}(x_i, [\sigma_{k,i}, {}^2m_{k,i}]) = e^{-.5\left(\frac{x_i - {}^2m_{k,i}}{\sigma_{k,i}}\right)^2} \quad (6)$$

where $m_{k,i} \in [{}^1m_{k,i}, {}^2m_{k,i}]$ and $\sigma_{k,i}$ are uncertain mean and spread for kth rule and ith input.

Layer 2: The T-norm and S-norm are computed as:

$$\underline{\mu}_{k,i}(x_i) = {}^{1}\mu_{k,i}(x_i).\ {}^{2}\mu_{k,i}(x_i)\ ,\qquad k = 1,2,\dots,M\ \ ,i = 1,2,\dots,n \tag{7}$$

$$\bar{\mu}_{k,i}(x_i) = {}^{1}\mu_{k,i}(x_i) + {}^{2}\mu_{k,i}(x_i) - \underline{\mu}_{k,i}(x_i) \tag{8}$$

Layer 3: The rule firings ($\underline{f}^k$ and $\bar{f}^k$) are:

$$\underline{f}^k = \prod_{i=1}^{n} \underline{\mu}_{k,i}\quad ;\quad \bar{f}^k = \prod_{i=1}^{n} \bar{\mu}_{k,i} \tag{9}$$

Layer 4: The left-most/right-most firing are obtained as:

$$f_l^k = \frac{\overline{w}_l^k \bar{f}^k + \underline{w}_l^k \underline{f}^k}{\overline{w}_l^k + \underline{w}_l^k}\ ;\ \ f_r^k = \frac{\overline{w}_r^k \bar{f}^k + \underline{w}_r^k \underline{f}^k}{\overline{w}_r^k + \underline{w}_r^k} \tag{10}$$

where w are adjustable weights.

Layer 5: The rule left/right firings are:

$$\begin{aligned} y_l^k = {} & c_{k,0} + c_{k,1}x_1 + c_{k,2}x_2 + c_{k,3}y(t-1) + c_{k,4}x_1x_2 + c_{k,5}x_1y(t-1) + c_{k,6}x_2y(t-1) + c_{k,7}x_1^2 \\ & + c_{k,8}x_2^2 + c_{k,9}y^2(t-1) + c_{k,10}x_1x_2y(t-1) - s_{k,0} - s_{k,1}|x_1| - s_{k,2}|x_2| - s_{k,3}|y(t-1)| \\ & - s_{k,4}|x_1x_2| - s_{k,5}|x_1y(t-1)| - s_{k,6}|x_2y(t-1)| - s_{k,7}x_1^2 - s_{k,8}x_2^2 - s_{k,9}y^2(t-1) \\ & - s_{k,10}x_1x_2y(t-1) \end{aligned} \tag{11}$$

$$\begin{aligned} y_r^k = {} & c_{k,0} + c_{k,1}x_1 + c_{k,2}x_2 + c_{k,3}y(t-1) + c_{k,4}x_1x_2 + c_{k,5}x_1y(t-1) + c_{k,6}x_2y(t-1) + c_{k,7}x_1^2 \\ & + c_{k,8}x_2^2 + c_{k,9}y^2(t-1) + c_{k,10}x_1x_2y(t-1) + s_{k,0} + s_{k,1}|x_1| + s_{k,2}|x_2| + s_{k,3}|y(t-1)| \\ & + s_{k,4}|x_1x_2| + s_{k,5}|x_1y(t-1)| + s_{k,6}|x_2y(t-1)| + s_{k,7}x_1^2 + s_{k,8}x_2^2 + s_{k,9}y^2(t-1) \\ & + s_{k,10}x_1x_2y(t-1) \end{aligned} \tag{12}$$

Layer 6: $\hat{y}_l$ and $\hat{y}_r$ are:

$$\hat{y}_l = \frac{\sum_{k=1}^{M} f_l^k y_l^k}{\sum_{k=1}^{M} f_l^k} \tag{13}$$

$$\hat{y}_r = \frac{\sum_{k=1}^{M} f_r^k y_r^k}{\sum_{k=1}^{M} f_r^k} \tag{14}$$

Layer 7: The output is:

$$\hat{y} = \frac{\hat{y}_l + \hat{y}_r}{2} \tag{15}$$

Structure learning is realized exploiting T2F clustering in this article. As one knows, an efficacious rule and fuzzy set turn-out algorithm is suggested to procreate fuzzy rules in real-time and decrease the number of fuzzy sets in antecedent part in structure learning [62]. Structure learning appears as a great assistance in simplification of T2FS taking advantage of reduction of the fuzzy rules. Scrutinizing more, its duty is not only production of novel membership but also pruning additional MFs and rules. In the input space, a rule geometrically corresponds to a cluster. Its firing strength could be taken into account as the degree through which input data belongs to the cluster. The center of the firing strength in the NCPRT2FS is calculated by (16) since it is an interval.

$$f_k = \frac{\underline{f}^k + \bar{f}^k}{2} \tag{16}$$

And for generation a new MF, find

$$\mu_{\tilde{A}_i^k} = \frac{\underline{\mu}_{\tilde{A}_i^k} + \bar{\mu}_{\tilde{A}_i^k}}{2}\quad ,\quad i = 1,2,\dots,n \tag{17}$$

For each incoming data $\vec{x} = \{x_1,\dots,x_n\}$, calculate

$$I = \arg \max_{1\le k\le M(t)} f_k \tag{18}$$

For each newly generated rule, compute

$$I_i = \arg \max_{1\le k\le k_i(t)} \mu_{\tilde{A}_i^k} \quad , \quad i = 1,2,\dots,n \tag{19}$$

where $M(t)$ and $k_i(t)$ are the existing rules number at time t and the number of fuzzy sets in input variable i, respectively. If $I \le \emptyset_{th}$, the system generate a new rule. Where $\emptyset_{th} \in (0\ 1)$ is a threshold that defined [63]. If $I_i > \rho$, where $\rho \in [0\ 1]$, is a threshold defined before, then use the existing fuzzy set $\tilde{A}_i^{I_i}$ as the antecedent part of the new rule in input variable i. Otherwise, one could turn out a novel fuzzy set in input variable i and hold the equation, $k_i(t+1) = k_i(t)+1$, true. The number of fuzzy sets is defined by the parameter ϱ in each input variable. Fuzzy clustering is a technique to structure a fuzzy model [64]. A new T2F clustering technique which is a development of Krishnapuram and Keller Possibilistic C-Mean (PCM) [65] is suggested and described by the following equations:

$$J_m(x,\tilde{\mu},c) = min\left[\sum_{i=1}^{c}\sum_{j=1}^{N}\tilde{\mu}_{ij}^m D_{ij} + \sum_{i=1}^{c}\eta_i\sum_{j=1}^{N}\left(1-\tilde{\mu}_{ij}\right)^m\right] \tag{20}$$

$$S.T:\begin{cases} 0 < \sum_{j=1}^{N}\tilde{\mu}_{ij} < N \\ \tilde{\mu}_{ij} \in [0,1] \quad \forall i,j \\ \max \tilde{\mu}_{ij} > 0 \quad \forall j \end{cases} \tag{21}$$

where $\tilde{\mu}_{ij}$ is type-2 MF in the j^{th} data for the i^{th} cluster, Moreover, the symbols D_{ij}, c, and N are the Euclidean distance of the j^{th} data in the i^{th} cluster center, number of clusters and the number of input data, respectively. η_i is also a positive number. D_{ij} has to be as small as possible as the first term. On the other hand, the membership values in a cluster have to be as large as possible as the second term as well. They have to stay in the interval of [0 1] and their sum is confined to get smaller than the number of input data. Equation (21) appears as the descriptive term. That η_i corresponds to ith cluster and is of the order of D_{ij}, is greatly welcomed [65]. The distance to the cluster's center must be as low as possible (first term). It is desirable that η_i relate to i^{th} cluster and be of the order of D_{ij} [63].

$$\eta_i = \frac{\sum_{j=1}^{N}\tilde{\mu}_{ij}^m D_{ij}}{\sum_{j=1}^{N}\tilde{\mu}_{ij}^m} \qquad \forall i = 1,\dots,c$$

Using (20) the optimal value of centers of the clusters are achieved. The initial uncertain mean $m_{k,i}$ and standard deviation $\sigma_{k,i}$ for the $k_i(t+1)$ th interval T2F set in input variable i are

$$m_{k,i} \in [v_i - 0.1v_i, v_i + 0.1v_i]$$

$$\sigma_{k_i(t+1)i} = \beta\left|v_i - \frac{{}^1m_{I_i,i} + {}^2m_{I_i,i}}{2}\right|$$

Where, v_i is the optimal value of the clusters center, β > 0 denotes the degree of overlap between 2 fuzzy sets. In this paper, we sets β at 0.5 so that the spread of the new fuzzy set is 50% the distance between the average centers of the new fuzzy set and fuzzy set I_i, so it generate suitable overlapping between fuzzy numbers [61]. The initial of parameters in the consequent part are set to

$$\left[c_{k,0} - s_{k,0}, c_{k,0} + s_{k,0}\right] = [yd - 0.1, yd + 0.1] \quad , \quad k = 1,2,\dots,M \tag{22}$$

where yd is the desired output for input $\vec{x} = \{x_1, \dots, x_n\}$. All the other consequent parameters are zero.

By repeating the above process for each training data, it creates new rules one after the other until NCPRT2FS is finally complete. For learning of the network, adaptive learning rate backpropagation is used. The network output is calculated for each input applied. The calculated output is then compared to the target to obtain an error. Assume that the input-output data pair $\{(x_p: t_p)\}\ \forall p = 1, \dots, q,$ where p is the number of data, x and t are the input and output, respectively. NCPRT2FS output error can be expressed as follows:

$$e_p = t_p - \hat{y}_p, \tag{23}$$

$$E_p = \frac{1}{2} e_p^2 = \frac{1}{2}\left(t_p - \hat{y}_p\right)^2 \tag{24}$$

$$E = \sum_{p=1}^{q} E_p \tag{25}$$

To update the consequent part parameters, the equations (26)–(45) are used.

$$^{new}c_{k,0} = {}^{old}c_{k,0} + \eta.0.5.e_p.\left[\frac{f_l^k}{\sum_{k=1}^M f_l^k} + \frac{f_r^k}{\sum_{k=1}^M f_r^k}\right] \tag{26}$$

$$^{new}c_{k,i} = {}^{old}c_{k,i} + \eta.0.5.e_p.\left[\frac{f_l^k}{\sum_{k=1}^M f_l^k} + \frac{f_r^k}{\sum_{k=1}^M f_r^k}\right].x_i \qquad i = 1,2 \tag{27}$$

$$^{new}c_{k,3} = {}^{old}c_{k,3} + \eta.0.5.e_p.\left[\frac{f_l^k}{\sum_{k=1}^M f_l^k} + \frac{f_r^k}{\sum_{k=1}^M f_r^k}\right].y(t-1) \tag{28}$$

$$^{new}c_{k,4} = {}^{old}c_{k,4} + \eta.0.5.e_p.\left[\frac{f_l^k}{\sum_{k=1}^M f_l^k} + \frac{f_r^k}{\sum_{k=1}^M f_r^k}\right].x_1.x_2 \tag{29}$$

$$^{new}c_{k,5} = {}^{old}c_{k,5} + \eta.0.5.e_p.\left[\frac{f_l^k}{\sum_{k=1}^M f_l^k} + \frac{f_r^k}{\sum_{k=1}^M f_r^k}\right].x_1.y(t-1) \tag{30}$$

$$^{new}c_{k,6} = {}^{old}c_{k,6} + \eta.0.5.e_p.\left[\frac{f_l^k}{\sum_{k=1}^M f_l^k} + \frac{f_r^k}{\sum_{k=1}^M f_r^k}\right].x_2.y(t-1) \tag{31}$$

$$^{new}c_{k,7} = {}^{old}c_{k,7} + \eta.0.5.e_p.\left[\frac{f_l^k}{\sum_{k=1}^M f_l^k} + \frac{f_r^k}{\sum_{k=1}^M f_r^k}\right].x_1^2 \tag{32}$$

$$^{new}c_{k,8} = {}^{old}c_{k,8} + \eta.0.5.e_p.\left[\frac{f_l^k}{\sum_{k=1}^M f_l^k} + \frac{f_r^k}{\sum_{k=1}^M f_r^k}\right].x_2^2 \tag{33}$$

$$^{new}c_{k,9} = {}^{old}c_{k,9} + \eta.0.5.e_p.\left[\frac{f_l^k}{\sum_{k=1}^M f_l^k} + \frac{f_r^k}{\sum_{k=1}^M f_r^k}\right].y^2(t-1) \tag{34}$$

$$^{new}c_{k,10} = {}^{old}c_{k,10} + \eta.0.5.e_p.\left[\frac{f_l^k}{\sum_{k=1}^M f_l^k} + \frac{f_r^k}{\sum_{k=1}^M f_r^k}\right].x_1.x_2.y(t-1) \tag{35}$$

$$^{new}s_{k,0} = {}^{old}s_{k,0} + \eta.0.5.e_p.\left[\frac{f_l^k}{\sum_{k=1}^M f_l^k} - \frac{f_r^k}{\sum_{k=1}^M f_r^k}\right] \tag{36}$$

$$^{new}s_{k,i} = {}^{old}s_{k,i} + \eta.0.5.e_p.\left[\frac{f_l^k}{\sum_{k=1}^M f_l^k} - \frac{f_r^k}{\sum_{k=1}^M f_r^k}\right].|x_i| \qquad i = 1,2 \tag{37}$$

$$^{new}s_{k,3} = {}^{old}s_{k,3} + \eta.0.5.e_p.\left[\frac{f_l^k}{\sum_{k=1}^M f_l^k} - \frac{f_r^k}{\sum_{k=1}^M f_r^k}\right].|y(t-1)| \tag{38}$$

$$^{new}s_{k,4} = {}^{old}s_{k,4} + \eta.0.5.e_p.\left[\frac{f_l^k}{\sum_{k=1}^M f_l^k} - \frac{f_r^k}{\sum_{k=1}^M f_r^k}\right].|x_1 x_2| \tag{39}$$

$$^{new}s_{k,5} = {}^{old}s_{k,5} + \eta.0.5.e_p.\left[\frac{f_l^k}{\sum_{k=1}^M f_l^k} - \frac{f_r^k}{\sum_{k=1}^M f_r^k}\right].|x_1.y(t-1)| \tag{40}$$

$$^{new}s_{k,6} = {}^{old}s_{k,6} + \eta.0.5.e_p.\left[\frac{f_l^k}{\sum_{k=1}^M f_l^k} - \frac{f_r^k}{\sum_{k=1}^M f_r^k}\right].|x_2.y(t-1)| \tag{41}$$

$$^{new}s_{k,7} = {}^{old}s_{k,7} + \eta.0.5.e_p.\left[\frac{f_l^k}{\sum_{k=1}^M f_l^k} - \frac{f_r^k}{\sum_{k=1}^M f_r^k}\right].x_1^2 \tag{42}$$

$$^{new}s_{k,8} = {}^{old}s_{k,8} + \eta.0.5.e_p.\left[\frac{f_l^k}{\sum_{k=1}^{M} f_l^k} - \frac{f_r^k}{\sum_{k=1}^{M} f_r^k}\right].x_2^2 \quad (43)$$

$$^{new}s_{k,9} = {}^{old}s_{k,9} + \eta.0.5.e_p.\left[\frac{f_l^k}{\sum_{k=1}^{M} f_l^k} - \frac{f_r^k}{\sum_{k=1}^{M} f_r^k}\right].y^2(t-1) \quad (44)$$

$$^{new}s_{k,10} = {}^{old}s_{k,10} + \eta.0.5.e_p.\left[\frac{f_l^k}{\sum_{k=1}^{M} f_l^k} - \frac{f_r^k}{\sum_{k=1}^{M} f_r^k}\right].|x_1.x_2.y(t-1)| \quad (45)$$

The learning rate is indicated by η.

To update the left and right weights, the equations (46)–(49) are used.

$$^{new}\underline{w}_l^k = {}^{old}\underline{w}_l^k + \eta.0.5.e_p \cdot \frac{y_l^k - \hat{y}_l}{\sum_{j=1}^{M} f_l^j} \cdot \frac{\underline{f}^k - f_l^k}{\overline{w}_l^k + \underline{w}_l^k} \quad (46)$$

$$^{new}\overline{w}_l^k = {}^{old}\overline{w}_l^k + \eta.0.5.e_p.\frac{y_l^k - \hat{y}_l}{\sum_{j=1}^{M} f_l^j} \cdot \frac{\bar{f}^k - f_l^k}{\overline{w}_l^k + \underline{w}_l^k} \quad (47)$$

$$^{new}\underline{w}_r^k = {}^{old}\underline{w}_r^k + \eta.0.5.e_p.\frac{y_r^k - \hat{y}_r}{\sum_{j=1}^{M} f_r^j} \cdot \frac{\underline{f}^k - f_r^k}{\overline{w}_r^k + \underline{w}_r^k} \quad (48)$$

$$^{new}\overline{w}_r^k = {}^{old}\overline{w}_r^k + \eta.0.5.e_p \cdot \frac{y_r^k - \hat{y}_r}{\sum_{j=1}^{M} f_r^j} \cdot \frac{\bar{f}^k - f_r^k}{\overline{w}_r^k + \underline{w}_r^k} \quad (49)$$

And finally the equations for updating the antecedent parameters can be described as follows:

$$^{1}m_{k,i}^{new} = {}^{1}m_{k,i}^{old} + \eta.0.5.e_p.\left[\frac{y_l^k - \hat{y}_l}{\sum_{j=1}^{M} f_l^j} \cdot \frac{\partial f_l^k}{\partial\, {}^{1}m_{k,i}} + \frac{y_r^k - \hat{y}_r}{\sum_{j=1}^{M} f_r^j} \cdot \frac{\partial f_r^k}{\partial\, {}^{1}m_{k,i}}\right] \quad (50)$$

$$^{2}m_{k,i}^{new} = {}^{2}m_{k,i}^{old} + \eta.0.5.e_p.\left[\frac{y_l^k - \hat{y}_l}{\sum_{j=1}^{M} f_l^j} \cdot \frac{\partial f_l^k}{\partial\, {}^{2}m_{k,i}} + \frac{y_r^k - \hat{y}_r}{\sum_{j=1}^{M} f_r^j} \cdot \frac{\partial f_r^k}{\partial\, {}^{2}m_{k,i}}\right] \quad (51)$$

$$\sigma_{k.i}^{new} = \sigma_{k.i}^{old} + \eta.0.5.e_p.\left[\frac{y_l^k - \hat{y}_l}{\sum_{j=1}^{M} f_l^j} \cdot \frac{\partial f_l^k}{\partial \sigma_{k,i}} + \frac{y_r^k - \hat{y}_r}{\sum_{j=1}^{M} f_r^j} \cdot \frac{\partial f_r^k}{\partial \sigma_{k,i}}\right] \quad (52)$$

where,

$$\frac{\partial f_l^k}{\partial\, {}^{1}m_{k,i}} = \frac{\overline{w}_l^k \cdot \left[\bar{f}^k - {}^{2}\mu_{k,i} \cdot \prod_{l=1,l\neq i}^{n}(\bar{\mu}_{k,l})\right] + \underline{w}_l^k \cdot \underline{f}^k}{\overline{w}_l^k + \underline{w}_l^k} \cdot \frac{x_i - {}^{1}m_{k,i}}{(\sigma_{k,i})^2}, \quad (53)$$

$$\frac{\partial f_l^k}{\partial\, {}^{2}m_{k,i}} = \frac{\overline{w}_l^k \cdot \left[\bar{f}^k - {}^{1}\mu_{k,i} \cdot \prod_{l=1,l\neq i}^{n}(\bar{\mu}_{k,l})\right] + \underline{w}_l^k \cdot \underline{f}^k}{\overline{w}_l^k + \underline{w}_l^k} \cdot \frac{x_i - {}^{2}m_{k,i}}{(\sigma_{k,i})^2} \quad (54)$$

$$\frac{\partial f_l^k}{\partial \sigma_{k,i}} = \frac{\overline{w}_l^k \cdot \left[\left(\bar{f}^k - {}^{2}\mu_{k,i} \cdot \prod_{l=1,l\neq i}^{n}(\bar{\mu}_{k,l})\right) \cdot \frac{(x_i - {}^{1}m_{k,i})^2}{(\sigma_{k,i})^3}\right]}{\overline{w}_l^k + \underline{w}_l^k} + \frac{\overline{w}_l^k \cdot \left[\left(\bar{f}^k - {}^{1}\mu_{k,i} \cdot \prod_{l=1,l\neq i}^{n}(\bar{\mu}_{k,l})\right) \cdot \frac{(x_i - {}^{2}m_{k,i})^2}{(\sigma_{k,i})^3}\right]}{\overline{w}_l^k + \underline{w}_l^k} + \frac{\underline{w}_l^k \cdot \underline{f}^k \cdot \left[\frac{(x_i - {}^{1}m_{k,i})^2 + (x_i - {}^{2}m_{k,i})^2}{(\sigma_{k,i})^3}\right]}{\overline{w}_l^k + \underline{w}_l^k} \quad (55)$$

$$\frac{\partial f_r^k}{\partial\, {}^{1}m_{k,i}} = \frac{\overline{w}_r^k \cdot \left[\bar{f}^k - {}^{2}\mu_{k,i} \cdot \prod_{l=1,l\neq i}^{n}(\bar{\mu}_{k,l})\right] + \underline{w}_r^k \cdot \underline{f}^k}{\overline{w}_r^k + \underline{w}_r^k} \cdot \frac{x_i - {}^{1}m_{k,i}}{(\sigma_{k,i})^2} \quad (56)$$

$$\frac{\partial f_r^k}{\partial\, {}^{2}m_{k,i}} = \frac{\overline{w}_r^k \cdot \left[\bar{f}^k - {}^{1}\mu_{k,i} \cdot \prod_{l=1,l\neq i}^{n}(\bar{\mu}_{k,l})\right] + \underline{w}_r^k \cdot \underline{f}^k}{\overline{w}_r^k + \underline{w}_r^k} \cdot \frac{x_i - {}^{2}m_{k,i}}{(\sigma_{k,i})^2} \quad (57)$$

$$\frac{\partial f_r^k}{\partial \sigma_{k,i}} = \frac{\bar{w}_r^k \cdot \left[\left(\bar{f}^k - {}^2\mu_{k,i} \cdot \prod_{l=1,l\neq i}^{n} (\bar{\mu}_{k,l}) \right) \cdot \frac{\left(x_i - {}^1m_{k,i}\right)^2}{\left(\sigma_{k,i}\right)^3} \right]}{\bar{w}_r^k + \underline{w}_r^k} + \frac{\bar{w}_r^k \cdot \left[\left(\bar{f}^k - {}^1\mu_{k,i} \cdot \prod_{l=1,l\neq i}^{n} (\bar{\mu}_{k,l}) \right) \cdot \frac{\left(x_i - {}^2m_{k,i}\right)^2}{\left(\sigma_{k,i}\right)^3} \right]}{\bar{w}_r^k + \underline{w}_r^k} + \frac{\underline{w}_r^k \cdot \underline{f}^k \cdot \left[\frac{\left(x_i - {}^1m_{k,i}\right)^2 + \left(x_i - {}^2m_{k,i}\right)^2}{\left(\sigma_{k,i}\right)^3} \right]}{\bar{w}_r^k + \underline{w}_r^k} \tag{58}$$

4. Convergence Analysis of Learning Algorithm

Lyapunov function is used to learning algorithm convergence guarantee. Define lyaponov function as

$$V_p(k) = E_p(k) = \frac{1}{2} e_p^2(k) = \frac{1}{2}\left(t_p(k) - \hat{y}_p(k)\right)^2 \tag{59}$$

Eq. (60) shows the layapunov function changes.

$$\Delta V_p(k) = V_p(k+1) - V_p(k) = \frac{1}{2}\left(e_p^2(k+1) - e_p^2(k)\right) \tag{60}$$

Next moment error is calculated from eq. (61).

$$e_p(k+1) = e_p(k) + \Delta e_p(k) \cong e_p(k) + \left[\frac{\partial e_p(k)}{\partial W}\right]^T \Delta W \tag{61}$$

In eq. (61), ΔW is parameter changing where $W = [\sigma_{k,i}, {}^1m_{k,i}, {}^2m_{k,i}, c_{k,i}, s_{k,i}]$

In equation (62) the general form of gradient-based updating is presented.

$$W(k+1) = W(k) + \Delta W(k) = W(k) + \eta.\left(-\frac{\partial E_p(k)}{\partial W}\right) \tag{62}$$

where,

$$\frac{\partial E_p(k)}{\partial W} = -e_p(k).\frac{\partial \hat{y}}{\partial W} \tag{63}$$

Eq. (60) can be rewritten as eq. (64).

$$\Delta V_p(k) = \frac{1}{2}\left(e_p^2(k+1) - e_p^2(k)\right) \tag{64}$$

$$= \frac{1}{2}\left[\left(e_p(k+1) - e_p(k)\right)\right].\left[\left(e_p(k+1) + e_p(k)\right)\right]$$

$$= \frac{1}{2}\Delta e_p(k)\left[2\left(e_p(k)\right) + \Delta e_p(k)\right]$$

$$= \Delta e_p(k)\left[e_p(k) + \frac{1}{2}\Delta e_p(k)\right]$$

$$= \left[\frac{\partial e_p(k)}{\partial W}\right]^T.\eta.e_p(k).\frac{\partial \hat{y}(k)}{\partial W}.\left\{e_p(k) + \frac{1}{2}\left[\frac{\partial e_p(k)}{\partial W}\right]^T.\eta.e_p(k).\frac{\partial \hat{y}(k)}{\partial W}\right\}$$

$$= -\left[\frac{\partial \hat{y}(k)}{\partial W}\right]^T.\eta.e_p(k).\frac{\partial \hat{y}(k)}{\partial W}.\left\{e_p(k) - \frac{1}{2}\left[\frac{\partial \hat{y}(k)}{\partial W}\right]^T.\eta.e_p(k).\frac{\partial \hat{y}(k)}{\partial W}\right\}$$

$$= -\eta.\left(e_p(k)\right)^2\left|\frac{\partial \hat{y}(k)}{\partial W}\right|^2.\left[1 - \frac{1}{2}\eta.\left|\frac{\partial \hat{y}(k)}{\partial W}\right|^2\right]$$

In order that $\Delta V_p(k) < 0$, the eq. (65) must be satisfied

$$0 < \eta < \frac{2}{max\left|\frac{\partial \hat{y}(k)}{\partial W}\right|^2} \tag{65}$$

If (65) holds for every parameter $W = \left[\sigma_{k,i}, {}^1m_{k,i}, {}^2m_{k,i}, c_{k,i}, s_{k,i}\right]$, then the algorithm is definitely convergent. We choose the initial η as:

Preprints (www.preprints.org) | NOT PEER-REVIEWED | Posted: 5 March 2021 doi:10.20944/preprints202103.0178.v1

$$\eta = \frac{1}{max\left|\frac{\partial \hat{y}(k)}{\partial W}\right|^2}$$

After all the data has been applied, the variable learning rate is determined by the following form.

$$\begin{cases} \boldsymbol{if} & \frac{RMSE\ (l)}{RMSE\ (l-1)} < 1 \quad \rightarrow \quad \eta(l) = \eta(l-1) \\ \boldsymbol{if} & \frac{RMSE\ (l)}{RMSE\ (l-1)} \geq 1 \quad \rightarrow \quad \eta(l) = 0.9 \times \eta(l-1) \end{cases}$$

Where $RMSE$ is Root Mean Square Error and l is the number of iteration.

5. Simulation results

In this paper two real renewable energy systems are used to identification. For each system, the structure of the system and the NCPRT2FS based identifier is shown in Fig. 5.

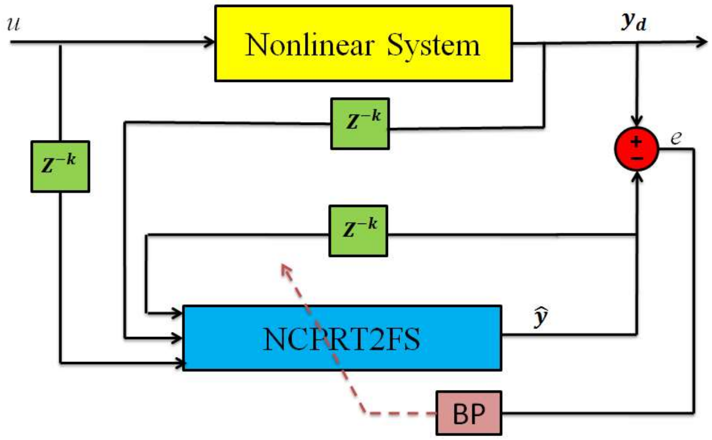

Figure 5. The structure of the system and the NCPRT2FS based identifier.

The inputs to the NCPRT2FS-based identifier are the main input and delayed system output. The parameters of NCPRT2FS structure should be adjusted to minimize plant output yd and identification yield $\hat{y}$ per all input values of x.

Example 1: Real data of a 660kw wind turbine have been taken from Iran Renewable Energy Organization (SUNA)[1]. The model of wind turbine is S47-660kw made by VESTAS (Denmark) are given in Table 1.

[1] http://www.suna.org.ir/en/home

 doi:10.20944/preprints202103.0178.v1

Figure 6. Manjil and Rudbar Wind Farm.

Table 1. Information for Example 1.

Cut-in wind speed:	4 m/s
Rated wind speed:	15 m/s
Cut-out wind speed:	25 m/s
Survival wind speed:	60 m/s
Rotor:	**Generator**:

Rotor:

Diameter:	47 m
Swept area:	1.735 m²
Number of blades:	3
Rotor speed, max:	28.5 U/min
Tipspeed:	70.1 m/s
Type:	22.9
Material:	GFK

Generator:

Type:	Asynchronous
Number:	1.0
Speed, max:	1.650 U/min
Voltage:	400 V
Grid connection:	Thyristor
Grid frequency:	50 Hz

In this example $u(k), k = 1, \ldots, 365$ is wind speed that is fed to the wind turbine system and obtains the 365 samples of $y(k)$ that is output power of the wind turbine. The other details are the same as proposed NCPRT2FS in example 1. Fig. 7 shows the identification results of the NCPRT2FS. Here the plant output (solid line) and the NCPRT2FS identifier output (dashed line) is shown.

Preprints (www.preprints.org) | NOT PEER-REVIEWED | Posted: 5 March 2021 doi:10.20944/preprints202103.0178.v1

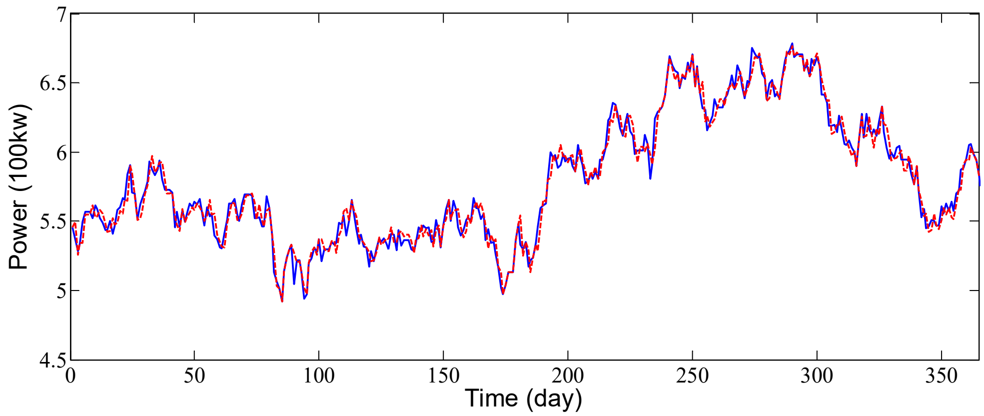

Figure 7. Identification results of the NCPRT2FS for wind turbine.

The trained NCPRT2FS is used to calculate the wind power in a place of Ilam[2]. Fig. 8 shows the wind speed of Ilam for a year. Fig. 9 shows the predicted wind power in Ilam.

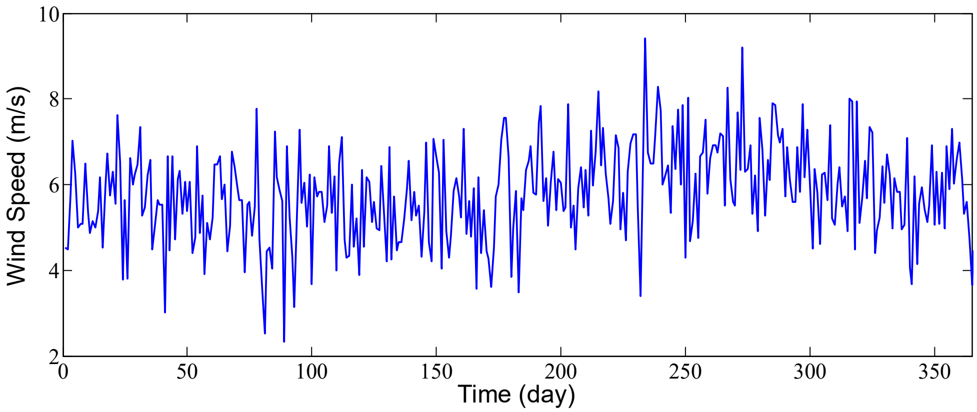

Figure 8. Wind speed of a place in Ilam for a year.

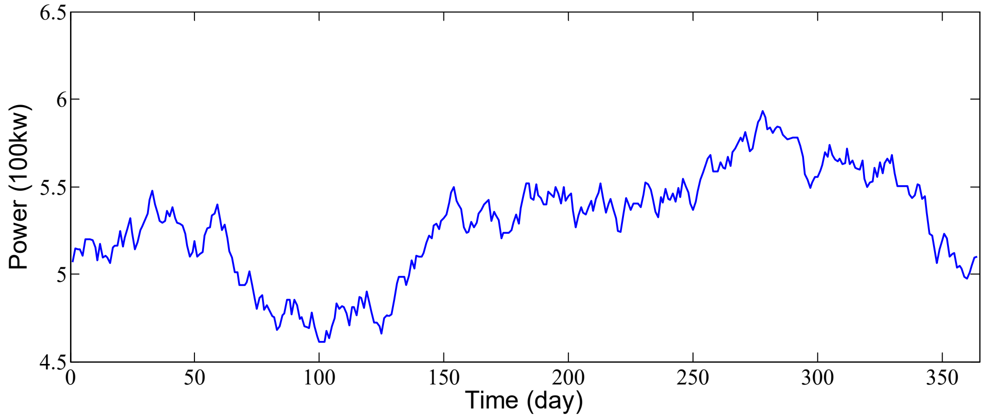

Figure 9. Predicted wind power of a place in Ilam for a year.

[2]- A city in the west of Islamic Republic of IRAN.

The final values of the parameters of NCPRT2FS are shown in Table 2.

Table 2. The final values of NCPRT2FS parameters.

		$^1m_{ij}$	$^2m_{ij}$	σ_{ij}
Antecedent parameters	u(k)	$^1m_{11} = 3.62$ $^1m_{21} = 6.13$ $^1m_{31} = 8.19$	$^2m_{11} = 4.32$ $^2m_{21} = 7.02$ $^2m_{31} = 9.51$	$\sigma_{11} = 0.38$ $\sigma_{21} = 1.10$ $\sigma_{31} = 0.89$
	y(k-1)	$^1m_{12} = 4.93$ $^1m_{22} = 5.34$ $^1m_{32} = 5.81$ $^1m_{42} = 6.11$	$^2m_{12} = 5.12$ $^2m_{22} = 5.66$ $^2m_{32} = 5.98$ $^2m_{42} = 6.48$	$\sigma_{12} = 0.21$ $\sigma_{22} = 0.09$ $\sigma_{32} = 0.36$ $\sigma_{42} = 0.18$
fourth layer adaptive weights	$\overline{w}_r^1 = 1.92$	$\underline{w}_r^1 = 1.50$	$\overline{w}_l^1 = 1.00$	$\underline{w}_l^1 = 0.63$
	$\overline{w}_r^2 = 1.66$	$\underline{w}_r^2 = 0.92$	$\overline{w}_l^2 = 0.71$	$\underline{w}_l^2 = 0.06$
	$\overline{w}_r^3 = 0.80$	$\underline{w}_r^3 = 0.70$	$\overline{w}_l^3 = 0.56$	$\underline{w}_l^3 = 0.43$
	$\overline{w}_r^4 = 1.87$	$\underline{w}_r^4 = 0.94$	$\overline{w}_l^4 = 0.85$	$\underline{w}_l^4 = 0.77$

	Rule 1	Rule 2	Rule 3	Rule 4	Rule 1	Rule 2	Rule 3	Rule 4
consequent parameters	$s_{1,0} = 0.40$	$s_{2,0} = 0.33$	$s_{3,0} = 0.27$	$s_{4,0} = 0.52$	$c_{1,0} = 1.00$	$c_{2,0} = 1.40$	$c_{3,0} = 1.00$	$c_{4,0} = 1.40$
	$s_{1,1} = 0.55$	$s_{2,1} = 0.39$	$s_{3,1} = 0.48$	$s_{4,1} = 0.43$	$c_{1,1} = 1.10$	$c_{2,1} = 1.00$	$c_{3,1} = 1.00$	$c_{4,1} = 1.00$
	$s_{1,2} = 1.00$	$s_{2,2} = 1.00$	$s_{3,2} = 1.00$	$s_{4,2} = 1.00$	$c_{1,2} = 1.00$	$c_{2,2} = 1.32$	$c_{3,2} = 0.81$	$c_{4,2} = 0.93$
	$s_{1,3} = 0.43$	$s_{2,3} = 0.39$	$s_{3,3} = 0.65$	$s_{4,3} = 0.90$	$c_{1,3} = 1.00$	$c_{2,3} = 1.00$	$c_{3,3} = 1.65$	$c_{4,3} = 1.82$
	$s_{1,4} = 0.62$	$s_{2,4} = 1.00$	$s_{3,4} = 1.00$	$s_{4,4} = 1.00$	$c_{1,4} = 1.00$	$c_{2,4} = 1.09$	$c_{3,4} = 1.00$	$c_{4,4} = 1.00$
	$s_{1,5} = 0.87$	$s_{2,5} = 0.10$	$s_{3,5} = 1.00$	$s_{4,5} = 1.00$	$c_{1,5} = 1.10$	$c_{2,5} = 1.00$	$c_{3,5} = 1.55$	$c_{4,5} = 1.90$
	$s_{1,6} = 1.00$	$s_{2,6} = 1.00$	$s_{3,6} = 1.00$	$s_{4,6} = 1.00$	$c_{1,6} = 1.00$	$c_{2,6} = 1.00$	$c_{3,6} = 1.00$	$c_{4,6} = 1.00$
	$s_{1,7} = 0.69$	$s_{2,7} = 0.66$	$s_{3,7} = 0.31$	$s_{4,7} = 0.06$	$c_{1,7} = 0.80$	$c_{2,7} = 0.72$	$c_{3,7} = 0.67$	$c_{4,7} = 0.81$
	$s_{1,8} = 0.96$	$s_{2,8} = 0.11$	$s_{3,8} = 0.54$	$s_{4,8} = 0.21$	$c_{1,8} = 1.10$	$c_{2,8} = 1.00$	$c_{3,8} = 0.92$	$c_{4,8} = 0.59$
	$s_{1,9} = 0.30$	$s_{2,9} = 0.32$	$s_{3,9} = 0.36$	$s_{4,9} = 0.98$	$c_{1,9} = 0.95$	$c_{2,9} = 0.77$	$c_{3,9} = 1.00$	$c_{4,9} = 1.00$
	$s_{1,10} = 0.35$	$s_{2,10} = 0.31$	$s_{3,10} = 0.54$	$s_{4,10} = 0.50$	$c_{1,10} = 1.00$	$c_{2,10} = 0.44$	$c_{3,10} = 0.64$	$c_{4,10} = 0.89$

Example 2: A real solar cell system is shown in Fig. 10.

(a) (b)

Figure 10. Experimental solar cell testing system (a) and a solar cell (b).

In this example $u(k), k = 1, \ldots, 600$ is solar radiation that is fed to the real solar cell system and obtain the 600 samples of $y(k)$. The other details are the same as proposed NCPRT2FS in examples 1 and 2. Fig. 11 shows the identification results of the NCPRT2FS for three solar radiations. Here the plant output (solid line) and the NCPRT2FS identifier output (dashed line) is shown.

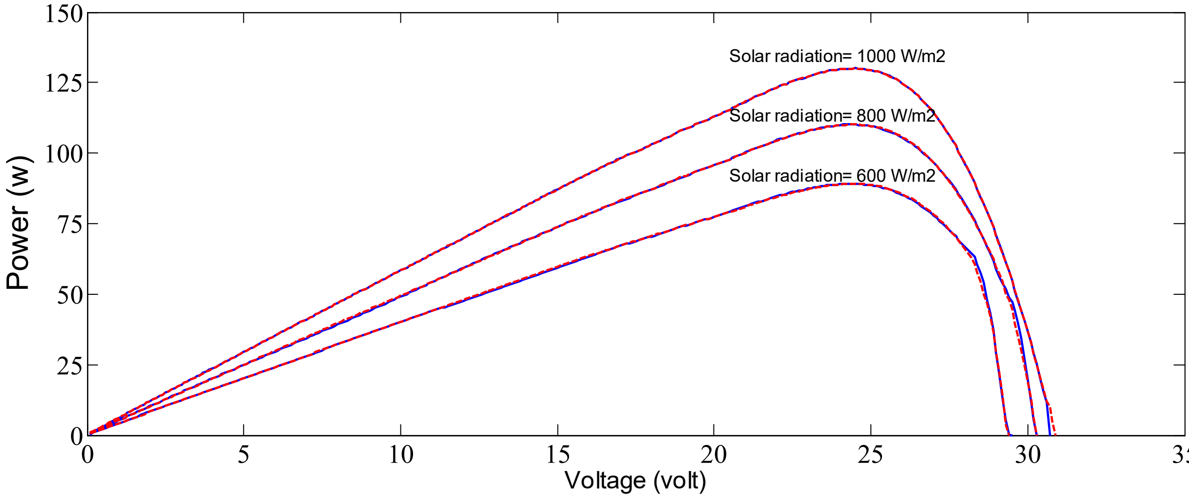

Figure 11. Identification results of the NCPRT2FS for solar cell.

After structure learning, for NCPRT2FS three rules are generated and the Root Mean Square Error (RMSE) value for the NCPRT2FS and IT2-TSK-FNN for training and test are shown in table 3. The final values of NCPRT2FS parameters are shown in Table 3.

Table 3. The final values of NCPRT2FS parameters.

		$^1m_{ij}$	$^2m_{ij}$	σ_{ij}
Antecedent parameters	u(k)	$^1m_{11} = 251$ $^1m_{21} = 598$ $^1m_{31} = 798$	$^2m_{11} = 332$ $^2m_{21} = 615$ $^2m_{31} = 949$	$\sigma_{11} = 43$ $\sigma_{21} = 12$ $\sigma_{31} = 211$
	y(k-1)	$^1m_{12} = 69$ $^1m_{22} = 82$ $^1m_{32} = 93$	$^2m_{12} = 75$ $^2m_{22} = 89$ $^2m_{32} = 97$	$\sigma_{12} = 11$ $\sigma_{22} = 5$ $\sigma_{32} = 3$
fourth layer adaptive weights	$\overline{w}_r^1 = 0.20$	$\underline{w}_r^1 = 0.06$	$\overline{w}_l^1 = 0.12$	$\underline{w}_l^1 = 0.09$
	$\overline{w}_r^2 = 1.80$	$\underline{w}_r^2 = 1.00$	$\overline{w}_l^2 = 1.42$	$\underline{w}_l^2 = 0.98$
	$\overline{w}_r^3 = 0.57$	$\underline{w}_r^3 = 0.21$	$\overline{w}_l^3 = 1.93$	$\underline{w}_l^3 = 1.10$

	Rule 1	Rule 2	Rule 3	Rule 1	Rule 2	Rule 3
consequent parameters	$s_{1,0} = 0.10$	$s_{2,0} = 0.84$	$s_{3,0} = 1.00$	$c_{1,0} = 0.56$	$c_{2,0} = 1.00$	$c_{3,0} = 1.22$
	$s_{1,1} = 0.32$	$s_{2,1} = 0.39$	$s_{3,1} = 0.37$	$c_{1,1} = 0.94$	$c_{2,1} = 1.60$	$c_{3,1} = 1.00$
	$s_{1,2} = 1.00$	$s_{2,2} = 1.00$	$s_{3,2} = 0.61$	$c_{1,2} = 1.00$	$c_{2,2} = 1.00$	$c_{3,2} = 1.00$
	$s_{1,3} = 0.22$	$s_{2,3} = 1.20$	$s_{3,3} = 0.50$	$c_{1,3} = 1.00$	$c_{2,3} = 1.77$	$c_{3,3} = 1.20$
	$s_{1,4} = 0.10$	$s_{2,4} = 0.42$	$s_{3,4} = 1.00$	$c_{1,4} = 1.61$	$c_{2,4} = 0.60$	$c_{3,4} = 1.63$
	$s_{1,5} = 0.47$	$s_{2,5} = 1.00$	$s_{3,5} = 1.00$	$c_{1,5} = 1.30$	$c_{2,5} = 1.00$	$c_{3,5} = 2.00$
	$s_{1,6} = 0.10$	$s_{2,6} = 1.00$	$s_{3,6} = 1.00$	$c_{1,6} = 1.00$	$c_{2,6} = 1.11$	$c_{3,6} = 1.00$
	$s_{1,7} = 1.20$	$s_{2,7} = 1.00$	$s_{3,7} = 0.19$	$c_{1,7} = 1.10$	$c_{2,7} = 1.50$	$c_{3,7} = 0.88$
	$s_{1,8} = 1.00$	$s_{2,8} = 0.36$	$s_{3,8} = 0.69$	$c_{1,8} = 1.60$	$c_{2,8} = 0.89$	$c_{3,8} = 0.91$
	$s_{1,9} = 1.00$	$s_{2,9} = 0.28$	$s_{3,9} = 0.11$	$c_{1,9} = 1.53$	$c_{2,9} = 0.95$	$c_{3,9} = 0.48$
	$s_{1,10} = 0.55$	$s_{2,10} = 0.35$	$s_{3,10} = 0.50$	$c_{1,10} = 0.88$	$c_{2,10} = 1.00$	$c_{3,10} = 1.00$

The trained NCPRT2FS is used to calculate the solar power of Ilam. Fig. 12 shows the solar radiation of Ilam for a year. Fig. 13 shows the predicted solar power in Ilam.

Preprints (www.preprints.org) | NOT PEER-REVIEWED | Posted: 5 March 2021 doi:10.20944/preprints202103.0178.v1

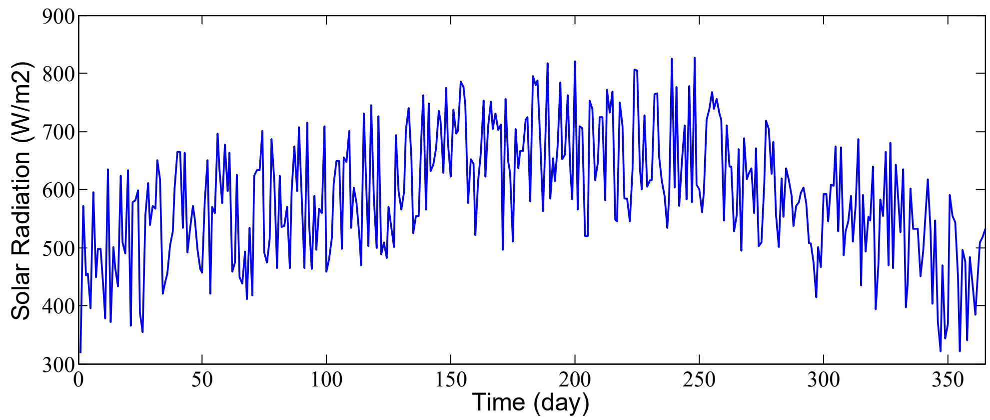

Figure 12. Solar radiation of Ilam.

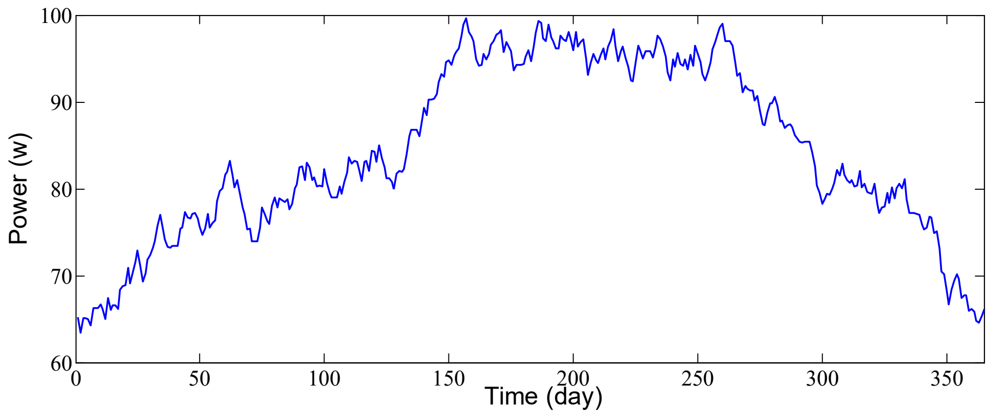

Figure 13. Predicted solar power in ILAM for a year.

Table 4. presents the comparison of our proposed method with another method (method of [46]).

Table 4. Comparison between results of the proposed method and the method of [46].

Example	Method of [46]				Proposed NCPRT2FS			
-	Rules	epochs	Run Time (s)	RMSE	Rules	epochs	Run Time (s)	RMSE
1	4	34	4	0.0159	4	31	6	0.0057
2	5	27	4	0.00759	3	39	7	0.0013

Simulation results show that the proposed NCPRT2FS has high performances in function approximation and system identification. Table 4 shows that the number of rules of the proposed NCPRT2FS is almost less than method of [53], accuracy of identification is better than [53], but the training time that achieves by average of 10 times run the program (computer processor: Dual CPU T3200 @ 2.00 GHz 2.00 GHz, RAM: 2.00 GB and MATLAB 2011a), is more than [53]. The references [23, 46] presented two different T2F neural structures. They have also been used and evaluated only to identify some theory systems. In the present paper, however, both the T2F neural network structure is different from references [23] and [53] and several experimental energy systems have been used for modeling.

6. CONCLUSION

In this paper, a novel Nonlinear Consequent-Part Recurrent T2FS (NCPRT2FS) for identification and prediction of renewable energy systems was proposed. Nonlinear consequent part helps to better model highly nonlinear systems. Recurrent structure is a useful and yet suitable choice for modeling and identification of dynamical systems. Adaptive learning rate helps to prevent the NCPRT2FS from trapping into local minima. Self-evolving structure helps to get simpler structure of NCPRT2FS by ending up with finally a minimum number of fuzzy sets and fuzzy rules. Simulation results showed that the NCPRT2FS based on backpropagation algorithm with adaptive learning rate performs better than IT2-TSK-FNN [53] in identification of highly nonlinear time-varying systems. S47-660 kw wind turbine (VESTAS company Denmark) and a solar cell were selected as case studies. After data gathering, the proposed method was finally used the experimental data for the purpose of identification. The RMSE was less than 0.006 and the number of fuzzy rules was equal and less than 4 rules, so the results easily approved the remarkable capability of the NCPRT2FS developed in the paper. In order to continue the work and look to the future, we can use the evolutionary algorithms as a complement to the proposed method for the development of the fuzzy neural network (to increase accuracy, increase convergence, etc.). On the other hand, different case studies (types of solar cells, types of wind turbines, etc.) should be identified and the appropriate renewable system can be extracted for each geographical location.

Conflict of interest: The authors declare that they have no conflict of interest.

Data Availability: The data that support the findings of this study are available from the corresponding author, (j.tavoosi@ilam.ac.ir), upon reasonable request.

Reference

[1] Quaranta, G.; Lacarbonara, W. & Masri, S.F. A review on computational intelligence for identification of nonlinear dynamical systems. *Nonlinear Dyn.* **2020**, 99, 1709–1761. https://doi.org/10.1007/s11071-019-05430-7

[2] Andrukhiv, A.; Sokil, M.; Fedushko, S.; Syerov, Y.; Kalambet, Y.; Peracek, T. Methodology for Increasing the Efficiency of Dynamic Process Calculations in Elastic Elements of Complex Engineering Constructions. *Electronics* **2021**, 10 (1) . https://doi.org/10.3390/electronics10010040

[3] Shekhovtsov, A.; Kołodziejczyk, J.; Sałabun, W. Fuzzy Model Identification Using Monolithic and Structured Approaches in Decision Problems with Partially Incomplete Data. Symmetry **2020**, 12, 1541. https://doi.org/10.3390/sym12091541

[4] Mombeini, H.; Yazdani-Chamzini, A.; Streimikiene, D.; Zavadskas, E.K. New fuzzy logic approach for the capability assessment of renewable energy technologies: Case of Iran. *Energy & Environment* **2018**, 29 (4), 511-532.

[5] Sakthivel, K.; Devaraj, B.; Banu, D.; Narmatha Selvi, R.; Agnes Idhaya, V. A hybrid wind-solar energy system with ANFIS based MPPT controller. *Journal of Intelligent & Fuzzy Systems* **2018**, 35 (2), 1579-1595.

[6] Shuli, L.; Xinwang, L.; Dongwei, L. A prospect theory based MADM method for solar water heater selection problems. *Journal of Intelligent & Fuzzy Systems* **2017**, 32 (3), 1855-1865.

[7] Kaid, I.E.; Hafaifa, A.; Guemana, M.; Hadroug, N.; Kouzou, A.; Mazouz, L. Photovoltaic system failure diagnosis based on adaptive neuro fuzzy inference approach: South Algeria solar power plant. *Journal of Cleaner Production* **2018**, 204, 169-182.

[8] Samanlioglu, F.; Ayağ, Z. A fuzzy AHP-PROMETHEE II approach for evaluation of solar power plant location alternatives in Turkey. *Journal of Intelligent & Fuzzy Systems* **2017**, 33 (2), 859-871.

[9] Grahovac, J.; Jokic, A.; Dodic, J.; Vucurovic, D.; Dodic, S. Modelling and prediction of bioethanol production from intermediates and byproduct of sugar beet processing using neural networks. *Renewable Energy* **2016**, 85, 953-958.

[10] Khiareddine, A.; Salah, C.B.; Mimouni, M.F. Power management of a photovoltaic/battery pumping system in agricultural experiment station. *Solar Energy* **2015**, 112, 319–338.

[11] Osorio, G.J.; Matias, J.C.O. ; Catalao, J.P.S. Short-term wind power forecasting using adaptive neuro-fuzzy inference system combined with evolutionary particle swarm optimization, wavelet transform and mutual information, *Renewable Energy* **2015**, 75, 301-307.

[12] Etemadi, M.; Abdollahi, A.; Rashidinejad, M.; Aalami, H.A. Wind Turbine Output Power Prediction in a Probabilistic Framework Based on Fuzzy Intervals. *Iran J Sci Technol Trans Electr Eng* **2021**, 45, 131-139. https://doi.org/10.1007/s40998-020-00359-9

[13] Torshizi, A.D.; Fazel Zarandi, M.H. A new cluster validity measure based on general type-2 fuzzy sets: Application in gene expression data clustering. *Knowledge-Based Systems* **2014**, 64, 81–93.

[14] Hesarian, M.S.; Tavoosi, J.; Hosseini, S.H. Neuro-fuzzy Modelling and Experimental Study of the Physiological Comfort of Green Cotton Fabric Based on Yarn Properties, *International Journal of Engineering* **2020**, 33(12). https://doi.org/10.5829/ije.2020.33.12c.02

[15] Tavoosi, J. A new type-2 fuzzy sliding mode control for longitudinal aerodynamic parameters of a commercial aircraft. *Journal Européen des Systèmes Automatisés* **2020**, 53(4), 479-485. https://doi.org/10.18280/jesa.530405

[16] Lu, W.; Yang, J.; Liu, X.; Pedrycz, W. The modeling and prediction of time series based on synergy of high-order fuzzy cognitive map and fuzzy c-means clustering, *Knowledge-Based Systems* **2014**, 70, 242–255.

[17] Abiyev, R.H.; Kaynak, O.; Kayacan, E. A type-2 fuzzy wavelet neural network for system identification and control, *Journal of the Franklin Institute* **2013**, 350 (7), 1658–1685.

[18] Wang, H.; Luo, C.; Wang, X. Synchronization and identification of nonlinear systems by using a novel self-evolving interval type-2 fuzzy LSTM-neural network, *Engineering Applications of Artificial Intelligence* **2019**, 81, 79-93.

[19] Luo, C.; Tan, C.; Wang, X.; Zheng, Y. An evolving recurrent interval type-2 intuitionistic fuzzy neural network for online learning and time series prediction. *Applied Soft Computing* **2019**, 78 150-163.

[20] El-Nagar, A. Nonlinear dynamic systems identification using recurrent interval type-2 TSK fuzzy neural network – A novel structure, *ISA Transactions* **2018**, 72, 205-217.

[21] Tavoosi, J.; Suratgar, A.A.; Menhaj, M.B. Nonlinear system identification based on a self-organizing type-2 fuzzy RBFN, *Engineering Applications of Artificial Intelligence* **2016**, 54, 26-38.

[22] Anh, N.; Suresh, S.; Pratama, M.; Srikanth, N. Interval prediction of wave energy characteristics using meta-cognitive interval type-2 fuzzy inference system, *Knowledge-Based Systems* **2019**, 169 (1), 28-38.

[23] Tavoosi, J.; Suratgar, A.A.; Menhaj, M.B. Stable ANFIS2 for Nonlinear System Identification, *Neurocomputing* **2016**, 182, 235-246.

[24] Ali, F.; Kim, E.K.; Kim, Y.G. Type-2 fuzzy ontology-based semantic knowledge for collision avoidance of autonomous underwater vehicles, *Information Sciences* **2015**, 295 441–464.

[25] Sun, Z.; Wang, N.; Bi, Y. Type-1/type-2 fuzzy logic systems optimization with RNA genetic algorithm for double inverted pendulum, *Applied Mathematical Modeling* **2015**, 39 (1), 70–85.

[26] Khooban, M.H.; Alfi, A.; Abadi, D.N.M. Control of a class of non-linear uncertain chaotic systems via an optimal Type-2 fuzzy proportional integral derivative controller, *IET Science, Measurement & Technology* **2013**, 7 (1), 50–58.

[27] Lee, C.H.; Chang, F.Y.; Lin, C.M. An efficient interval type-2 fuzzy CMAC for chaos time-series prediction and synchronization. *IEEE Trans. on Cybernetics* **2014**, 44 (3), 329–341.

[28] Yu, W.S.; Chen, H.S. Interval type-2 fuzzy adaptive tracking control design for PMDC motor with the sector dead-zones, *Information Sciences* 2014, 288 (20), 108–134.

[29] Esposito, M.; Pietro, G.D. Interval type-2 fuzzy logic for encoding clinical practice guidelines, *Knowledge-Based Systems* **2013**, 54, 329–341.

[30] Lin, Y.Y.; Chang, J.Y.; Pal, N.R.; Lin, C.T. A Mutually Recurrent Interval Type-2 Neural Fuzzy System (MRIT2NFS) With Self-Evolving Structure and Parameters, *IEEE Trans. on Fuzzy Systems* 2013, 21 (3), 492–509.

[31] Tavoosi, J. A New Type-2 Fuzzy Systems for Flexible-Joint Robot Arm Control, *AUT Journal of Modeling and Simulation* **2019**, 51(2). https://doi.org/10.22060/miscj.2019.14478.5108.

[32] Asad, Y.P.; Shamsi, A.; Ivani, H.; Tavoosi, J. Adaptive intelligent inverse control of nonlinear systems with regard to sensor noise and parameter uncertainty (magnetic ball levitation system case study). *International Journal on Smart Sensing and Intelligent Systems* **2016**, 9 (1), 148-169.

[33] Tavoosi, J.; Badamchizadeh, M.A.; Ghaemi, S. Adaptive Inverse Control of Nonlinear Dynamical System Using Type-2 Fuzzy Neural Networks. *Journal of Control* **2011**, 5(2), 52-60.

[34] Fazlya, M.; Pedram, M.Z.; Salarieh, H.; Alasty, A. Parameter estimation and interval type-2 fuzzy sliding mode control of a z-axis MEMS gyroscope, *ISA Transactions* **2013**, 52 (6), 900–911.

[35] Melin, P.; Castillo, O. A review on type-2 fuzzy logic applications in clustering, classification and pattern recognition, *Applied Soft Computing* **2014**, 21, 568–577.

[36] Wu, G.D.; Zhu, Z.W. An enhanced discriminability recurrent fuzzy neural network for temporal classification problems, *Fuzzy Sets and Systems* **2014**, 237, 47–62.

[37] Tavoosi, J.; Mohammadzadeh, A. A New Recurrent Radial Basis Function Network-based Model Predictive Control for a Power Plant Boiler Temperature Control. *International Journal of Engineering* **2021**, 34(3), 667-675.

[38] Moodi, H.; Farrokhi, M. Robust observer design for Sugeno systems with incremental quadratic nonlinearity in the consequent, *International Journal of Applied Mathematics and Computer Science* **2013**, 23 (4), 711–723.

[39] Tavoosi, J. A Novel Recurrent Type-2 Fuzzy Neural Network Stepper Motor Control. *Mechatronic Systems and Control* **2021**, 49(1). https://doi: 10.2316/J.2021.201-0097

[40] Tavoosi, J.; Mohammadi, F. A New Type-II Fuzzy System for Flexible-Joint Robot Arm Control, 6th International Conference on Control, Instrumentation and Automation (ICCIA), Sanandaj, Iran, **2019**, pp. 1-4, https://doi: 10.1109/ICCIA49288.2019.9030872.

[41] Tavoosi, J.; Mohammadi, F. A 3-PRS Parallel Robot Control Based on Fuzzy-PID Controller, 6th International Conference on Control, Instrumentation and Automation (ICCIA), Sanandaj, Iran, **2019**, pp. 1-4, https://doi: 10.1109/ICCIA49288.2019.9030860.

[42] Tavoosi, J. Azami, R. A New Method for Controlling the Speed of a Surface Permanent Magnet Synchronous Motor using Fuzzy Comparative Controller with Hybrid Learning, *Computational Intelligence in Electrical Engineering* **2019**, 10(3), 57-68.
[43] Tavoosi, J. An experimental study on inverse adaptive neural fuzzy control for nonlinear systems, International *Journal of Knowledge-based and Intelligent Engineering Systems* **2020**, 24(2), 135-143.
[44] Tavoosi, J. Stable Backstepping Sliding Mode Control Based on ANFIS2 for a Class of Nonlinear Systems, *Jordan Journal of Electrical Engineering* (*JJEE*) **2020**, 6 (1), 49-62.
[45] Tavoosi, J. Sliding mode control of a class of nonlinear systems based on recurrent type-2 fuzzy RBFN. *International Journal of Mechatronics and Automation* **2020**, 7 (2), 230-240.
[46] Tavoosi, J. Hybrid Intelligent Adaptive Controller for Tiltrotor UAV, *International Journal of Intelligent Unmanned Systems*. In Press, **2020**. https://doi.org/10.1108/IJIUS-05-2020-0009
[47] Tavoosi, J.; Badamchizadeh, M.A. A class of type-2 fuzzy neural networks for nonlinear dynamical system identification. *Neural Computing & Application* **2013**, 23 (3-4) 707–717.

[48] Tavoosi, J.; Suratgar, A.A.; Menhaj, M.B. Stability analysis of recurrent type-2 TSK fuzzy systems with nonlinear consequent part. *Neural Computing and Applications* **2017**, 28 (1), 47-56.

[49] Tavoosi, J.; AA Suratgar, MB Menhaj, Stability analysis of a class of MIMO recurrent type-2 fuzzy systems. *International Journal of Fuzzy Systems* **2017**, 19(3) 895-908.

[50] Jahangiri, F.; Doustmohammadi, A.; Menhaj, M.B. An adaptive wavelet differential neural networks based identifier and its stability analysis. *Neurocomputing* **2012**, 77, 12–19.

[51] Suratgar, A.A.; Nikravesh, S.K. A New Method for Linguistic Modeling with Stability Analysis and Applications, *Intelligent Automation and Soft Computing* **2009**, 15 (3), 329-342.

[52] Li, C.; Wang, L.; Zhang, G.; Wang, H.; Shang, F. Functional-type single-input-rule-modules connected neural fuzzy system for wind speed prediction. *Journal of Automatica Sinica* **2017**, 4(4), 751-762.

[53] Karakuş, O.; Kuruoğlu, E. E.; Altınkaya, M.A. One-day ahead wind speed/power prediction based on polynomial autoregressive model. *IET Renewable Power Generation* **2017**, 11, 1430-1439.

[54] Tian, Y.; Wang, B.; Zhu, D.; Wu, F. Takagi–Sugeno fuzzy generalised predictive control of a time-delay non-linear hydro-turbine governing system. *IET Renewable Power Generation* **2019**, 13, 2338-2345.

[55] Morshedizadeh, M.; Kordestani, M.; Carriveau, R.; Ting, D.S.; Saif, M. Power production prediction of wind turbines using a fusion of MLP and ANFIS networks. *IET Renewable Power Generation* **2018**, 12(9), 1025-1033.

[56] Castro, J.R. ; Castillo, O.; Martínez, L.G. Interval Type-2 Fuzzy Logic Toolbox, *Engineering Letters* **2007**, 15:1.
[57] Mendel, J.M. Uncertain Rule-Based Fuzzy Logic Systems: Introduction and New Directions. Prentice-Hall, NJ, **2001**.
[58] Mendel, J.M. Advances in type-2 fuzzy sets and systems, *Information Sciences* **2007**, 177, 84–110.

[59] Kizielewicz, B.; Sałabun, W. A New Approach to Identifying a Multi-Criteria Decision Model Based on Stochastic Optimization Techniques. *Symmetry* **2020**, 12, 1551. https://doi.org/10.3390/sym12091551

[60] Karnik, N.N.; Mendel, J.M.; Liang, Q. Type-2 Fuzzy Logic Systems. *IEEE Trans. on Fuzzy Systems* **1999**, 7 (6) 643 – 658.
[61] Singh, M.; Srivastava, S.; Hanmandlu, M.; Gupta, J.R.P. Type-2 fuzzy wavelet networks (T2FWN) for system identification using fuzzy differential and Lyapunov stability algorithm. *Applied Soft Computing* **2009**, 9, 977–989.

[62] Juang, C.F.; Lin, Y.Y.; Tu, C.C. A recurrent self-evolving fuzzy neural network with local feedbacks and its application to dynamic system processing. *Fuzzy Sets and Systems* **2010**, 161, 2552–2568.

 doi:10.20944/preprints202103.0178.v1

[63] Lin, T.C.; Kuo, C.H.; Balas, V.E. Real-time fuzzy system identification using uncertainty bounds. *Neurocomputing* **2014**, 125, 195–216.

[64] Pedrycz, W.; Izakian, H. Cluster–centric fuzzy modeling. *IEEE Trans. on Fuzzy Systems* **2014**, 22 (6), 1585 – 1597.

[65] Krishnapuram, R.; Keller, J.M. A possibilistic approach to clustering. *IEEE Trans. on Fuzzy Systems* **1993**, 1 (2), 98–110.