INTERVAL TYPE-2 INTUITIONISTIC FUZZY LOGIC SYSTEM FOR TIME SERIES AND IDENTIFICATION PROBLEMS - A COMPARATIVE STUDY

Imo Eyoh[1], Jeremiah Eyoh[2] and Roy Kalawsky[2]

[1]Department of Computer Science, University of Uyo, Akwa Ibom State, Nigeria
[2]School of Electrical, Electronics and Systems Engineering, AVRRC Research Group, Loughborough University, UK

ABSTRACT

This paper proposes a sliding mode control-based learning of interval type-2 intuitionistic fuzzy logic system for time series and identification problems. Until now, derivative-based algorithms such as gradient descent back propagation, extended Kalman filter, decoupled extended Kalman filter and hybrid method of decoupled extended Kalman filter and gradient descent methods have been utilized for the optimization of the parameters of interval type-2 intuitionistic fuzzy logic systems. The proposed model is based on a Takagi-Sugeno-Kang inference system. The evaluations of the model are conducted using both real world and artificially generated datasets. Analysis of results reveals that the proposed interval type-2 intuitionistic fuzzy logic system trained with sliding mode control learning algorithm (derivative-free) do outperforms some existing models in terms of the test root mean squared error while competing favourable with other models in the literature. Moreover, the proposed model may stand as a good choice for real time applications where running time is paramount compared to the derivative-based models.

KEYWORDS

Interval type-2 intuitionistic fuzzy set, Sliding mode control algorithm, Intuitionistic fuzzy set, Intuitionistic fuzzy index.

1. INTRODUCTION

The classical FS including both the T1FS [1] and T2FS [2]are defined using the membership functions. Literature has it that classical FSs are complementary sets [3]. This implies that to compute the non-membership of a classical FS, one has to take the complement of the set so defined. This may not always be the case in real life contexts because people are often times hesitant to pin-point or specify a single numerical value as doing so indicate strong commitments or evidence.

This brings the idea of intuitionistic FS (IFS) introduced by Atanassov in 1986 [4]. With IFS, a set can be described by three components namely: membership function, non-membership function and hesitation index. With these three representations, IFS becomes a more appropriate tool for dealing with imprecise and vague information. While the non-membership function captures additional information, the hesitation index makes the set description very intuitive and close to human intelligence than classical FS.

DOI: 10.5121/ijfls.2020.10101

However, using single membership and non-membership values to model a concept may not capture enough information and uncertainty in the intuitionistic set definition. This is because changes in the perception of the same linguistic term will lead to frequent re-tuning of the membership and non-membership functions of IFSs. This may result to waste of resources and sub-optimal performance of the system under some environmental and operation conditions.

Atanassov and Gargov [5] extended the concept of the IFS to interval valued intuitionistic fuzzy set (IVIFS),with interval values similar to the notion of classical interval valued fuzzy set (IVFS) [6]. To obtain an IVIFS, the sum of upper membership and upper non-membership must not exceed 1. Based on the salient points discussed above for IFS and IVIFS, Eyoh *et al.,* [7] introduced a rule-based IT2IFLS that merges the capabilities of IT2FLS with those of IFS. The IT2IFLS relaxes the single restriction of the IVIFS by allowing two constraints namely that the sum of upper membership and lower non-membership must not exceed 1, similarly the sum of lower membership and upper non-membership must not exceed 1. This makes IT2IFS more flexible than IVIFS.

RELATED WORK

Several methods such as gradient descent (GD), extended Kalman filter (EKF), decoupled EKF (DEKF) and the hybrid (DEKF and GD) have been applied so far for the optimization of IT2IFLS. The optimization may be structure or parameter optimization.In [7] IT2IFLS is proposed and the parameters of the model are updated using GD back-propagation method (a first order derivative-based learning algorithm). The proposed model is applied to non-linear system prediction with good results. The same model is also applied to time series prediction [8]. Recently, Luo et al., [9], proposed an evolving recurrent interval type-2 intuitionistic fuzzy neural network (IT2IFNN) and applied the model for online learning and time series prediction. The parameters of the IT2IFNN in [9] are optimized using EKF (a second-order derivative-based method). In [10], the DEKF is adopted for the optimization of the parameters of the IT2IFLS. Using the DEKF allows the parameters of the model to be grouped into vectors such as antecedent and consequent vectors so that interactions are allowed at the second order. Experimental results reveal that EKF-based learning models perform better than the GD-based in terms of prediction accuracy. In [11], a hybrid model of GD back-propagation and DEKF is employed for the adjustment of the parameters of IT2IFLS and the model applied to system identification problem.Recently also, Yuan and Luo [12] proposed an online evolving interval type-2 intuitionistic fuzzy LSTM-Neural Networks (eIT2IF-LSTMNN). In [12] and [13], the parameters of the models are optimized using GD-back-propagationand applied for regression problems.Other studies on IT2IFS is the one reported in [14], where arithmetic operations are defined for IT2IFS using generalized trapezoidal type-2 intuitionistic fuzzy numbers. The proposed model is used to solve a transportation problem. In this work, only the parameters of the IT2IFLS are optimized. According to [15], membership function (non-membership function) parameters of FLSs are very important in deciding the overall performance of the system.

The weakness of GD and EKF-based models is that the GD and EKF-based methods are both derivative-based approaches and involve the computation of partial derivatives for both the membership and non-membership functions which is tedious and time consuming. For large scale computation, these derivative-based approaches may not be the best choice.Unlike the derivative-based models, using SMC learning algorithm to update the IT2IFLS parameters is straightforward and relieves the system of the computational burden of calculating the partial derivatives of these parameters.

This paper proposes the optimization of the IT2IFLS parameters using derivative-free methodology with application to time series and prediction problems. In this study, a sliding mode control learning of the parameters of IT2IFLS is developed and used to solve time series and identification problems for the first time. The antecedent and the consequent parameters update benefits from the proposed learning algorithm.

The rest of the paper is structured as follows: In section 2, some underlying concepts are defined. Section 3 takes a view of interval type-2 intuitionistic fuzzy logic system. In Section 4, sliding mode control learning algorithm is defined with the update rule using sliding mode algorithm derived in Section 5. Experimental evaluation is carried out in Section 6 with the conclusion drawn in Section 7.

2. DEFINITIONS

DEFINITION 1: An IFS is totally defined by membership function, non-membership function and hesitation index (π) of element, x ∊ X such that $0 \leq \mu_{A^*}(x) + \nu_{A^*}(x) + \pi(x) = 1$[4].

When the intuitionistic index is 0, the IFS changed to classical FS. Many approaches have been adopted in the literature for constructing membership and non-membership functions of IFS. Some of the approaches include those reported in [16][17][18]. The IFS Gaussian membership and non-membership function in definitions proposed in [17] are adopted in this paper and are defined as follows:

$$\mu(x) = (1 - \pi_c(x)) * \exp\,(-\,\frac{x-c}{2\sigma})^2 \qquad (1)$$

$$\nu(x) = (1 - \pi_{var}(x)) - \mu(x) \qquad (2)$$

DEFINITION 2: An IT2IFS, $\tilde{A}^*$ is a variant of IFS but membership and non-membership functions are themselves fuzzy and defined as $\{\bar{\mu}_{\tilde{A}^*}(x), \underline{\mu}_{\tilde{A}^*}(x)\}$ and $\{\bar{\nu}_{\tilde{A}^*}(x), \underline{\nu}_{\tilde{A}^*}(x)\}$ respectively for all x $\in X$ with constraints: $0 \leq \bar{\mu}_{\tilde{A}^*}(x) + \underline{\nu}_{\tilde{A}^*}(x) \leq 1$ and $0 \leq \underline{\mu}_{\tilde{A}^*}(x) + \bar{\nu}_{\tilde{A}^*}(x) \leq 1$ [19]

For IT2IFS, two footprints of uncertainties (FOUs) are utilized which are membership function FOU and non-membership function FOU (see Figure 1).

$$FOU_{\mu}\,(\tilde{A}^*) = \cup_{\forall x \in X}[\underline{\mu}_{\tilde{A}^*}, \bar{\mu}_{\tilde{A}^*}] \qquad (3)$$

$$FOU_{\nu}\,(\tilde{A}^*) = \cup_{\forall x \in X}[\underline{\nu}_{\tilde{A}^*}, \bar{\nu}_{\tilde{A}^*}] \qquad (4)$$

As shown in Figure 1, for designing the non-membership function FOU, the lower membership function becomes the upper non-membership function while the upper membership function becomes the lower non-membership function [20].

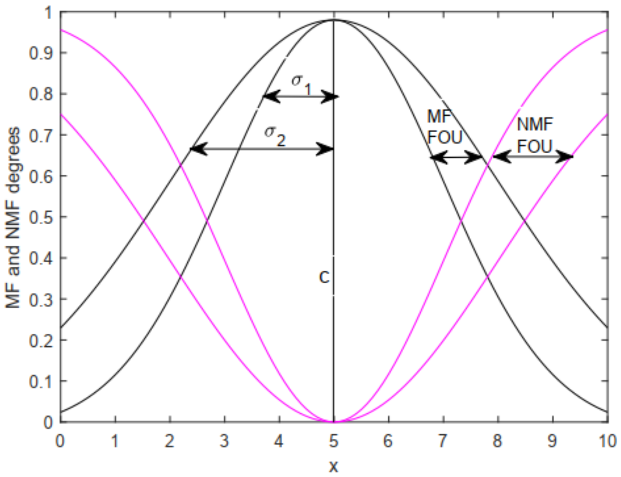


Figure 1: An IT2 intuitionistic Gaussian membership and non-membership functions FOU

3. Interval type-2 intuitionistic fuzzy logic system

An interval type-2 fuzzy logic system consist of intuitionistic - fuzzifier, rulebase, inference engine and output processing block (type reduction and defuzzifier) as shown in Figure 2. The intuitionistic fuzzifier converts crisp exogenous inputs into IT2IFS, with exactly four components namely lower membership, upper-membership, lower non-membership and upper non-membership. Here, singleton fuzzification is considered. The hesitation index ensures that the sum of lower membership and upper non-membership functions in the input partition space is less than or equal to 1 and similarly, the sum of upper membership and lower non-membership functions is less than or equal to 1. In this study, the approach for constructing membership and non-membership functions reported in [17] is adopted and modified to reflect the type-2 version as follows:

$$\bar{\mu}_{ik}(x_i) = \exp\left(-\frac{x-c}{2\bar{\sigma}_{2,ik}}\right)^2 * (1-\pi_c(x_i)) \tag{5}$$

$$\underline{\mu}_{ik}(x_i) = \exp\left(-\frac{x-c}{2\underline{\sigma}_{1,ik}}\right)^2 * (1-\pi_c(x_i)) \tag{6}$$

$$\bar{\nu}_{ik}(x_i) = (1-\bar{\pi}_{var}(x_i)) - \underline{\mu}_{ik}(x_i) \tag{7}$$

$$\underline{\nu}_{ik}(x_i) = (1-\underline{\pi}_{var}(x_i)) - \bar{\mu}_{ik}(x_i) \tag{8}$$

where π_c and $\underline{\pi}_{var}$ are the intuitionistic fuzzy indices for center and variance respectively [17]. The π_c and $\underline{\pi}_{var}$ are small numbers in the range of 0 and 1. The intuitionistic fuzzy indices can be chosen by the user or randomly generated. In this study, π_c and $\underline{\pi}_{var}$ are randomly generated in the interval [0,1].

The intuitionistic fuzzy indices use in this study are expressed for IT2IFLS as follows:

$$\pi_c(x) = \max(0, (1-(\mu_{\tilde{A}^*}(x) + \nu_{\tilde{A}^*}(x)))) \tag{9}$$

$$\overline{\pi}_{\text{var}}(x) = \max\ (0,\ (1 - (\overline{\mu}_{\tilde{A}*}(x) + \underline{v}_{\tilde{A}*}(x)))) \tag{10}$$

$$\underline{\pi}_{\text{var}}(x) = \max\ (0,\ (1\text{-}\ (\underline{\mu}_{\tilde{A}*}(x) + \overline{v}_{\tilde{A}*}(x)))) \tag{11}$$

such that $0 \leq \pi_{\text{c}}(x) \leq 1$ and $0 \leq \pi_{\text{var}}(x) \leq 1$

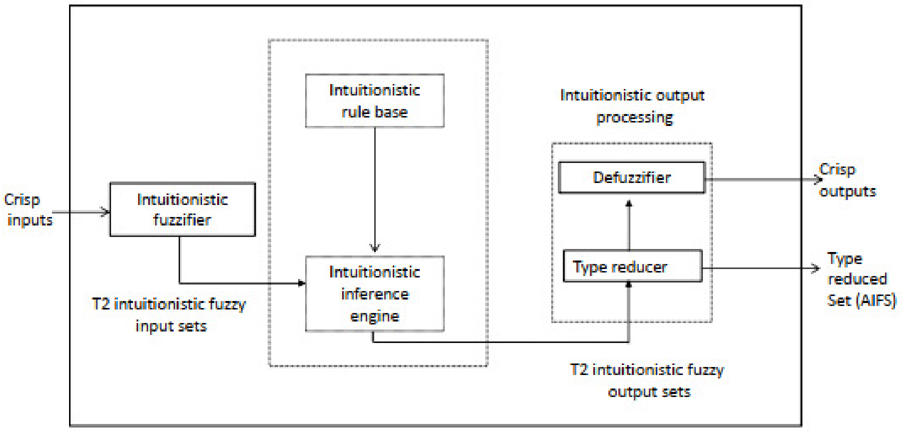

Intuitionistic rule base
Intuitionistic output processing
Defuzzifier
Crisp outputs
Crisp inputs
Intuitionistic fuzzifier
Intuitionistic inference engine
Type reducer
Type reduced Set (AIFS)
T2 intuitionistic fuzzy input sets
T2 intuitionistic fuzzy output sets

Figure 2: Interval type-2 intuitionistic fuzzy logic system [20]

The generic IT2IFLS *IF-THEN* rule is represented as follows:

$$R_k : \text{IF } x_i \text{ is } \tilde{A}^*_{ik} \text{ and } \ldots \text{ and } x_n \text{ is } \tilde{A}^*_{nk} \text{ THEN } y_k = \textstyle\sum_{i=1}^{n} w_{ik}\, x_i + b_k \tag{12}$$

The generic rule can be formulated for membership and non-membership functions respectively as follows:

$$R_k^{\mu} : \text{IF } x_i \text{ is } \tilde{A}{}^{*\mu}_{ik} \text{ and } \ldots \text{ and } x_n \text{ is } \tilde{A}{}^{*\mu}_{nk} \text{ THEN } y_k = \textstyle\sum_{i=1}^{n} w_{ik}^{\mu} + b_k^{\mu} \tag{13}$$

$$R_k^{\nu} : \text{IF } x_i \text{ is } \tilde{A}{}^{*\nu}_{ik} \text{ and } \ldots \text{ and } x_n \text{ is } \tilde{A}{}^{*\nu}_{nk} \text{ THEN } y_k = \textstyle\sum_{i=1}^{n} w_{ik}^{\nu} + b_k^{\nu} \tag{14}$$

Where $\tilde{A}^*_{ik}$s, are IT2IFS, $y_k$are the k_{th} rule outputs for membership function and non-membership function, w and $b$are parameters of the consequent parts. The inference engine maps IT2IFS inputs to IT2IFS outputs using the combinations of the formulated rules. According to [7], the final output of a TSK-type IT2IFLS is defined as follows:

$$\text{y} = (1 - \beta) \textstyle\sum_{k=1}^{M} \tilde{f}_k^{\mu}\, y_k^{\mu} + (1 - \beta) \sum_{k=1}^{M} \tilde{f}_k^{\nu}\, y_k^{\nu} \tag{15}$$

where:

$$\tilde{f}_k^{\mu} = \frac{(\underline{f}_k^{\mu} + \bar{f}_k^{\mu})}{\sum_{k=1}^{M} \underline{f}_k^{\mu} + \sum_{k=1}^{M} \bar{f}_k^{\mu}} \tag{16}$$

$$\tilde{f}_k^{\nu} = \frac{(\underline{f}_k^{\nu} + \bar{f}_k^{\nu})}{\sum_{k=1}^{M} \underline{f}_k^{\nu} + \sum_{k=1}^{M} \bar{f}_k^{\nu}} \tag{17}$$

and $\tilde{f}_k^{\mu}$and $\tilde{f}_k^{\nu}$ are normalized firing signals for membership and non-membership functions respectively.

4. Sliding Mode Control Learning

The error of IT2IFLS is defined as follows:

$$E(t) = y^a(t) - y(t) \tag{18}$$

The error shows the difference between the measured output and the predicted output of the model. The zero value of the error coordinate is shown as time varying sliding surface [21] as below:

$$S(E(t)) = E(t) = y^a(t) - y(t) = 0 \tag{19}$$

With this,the system which is in a sliding mode is guaranteed to be on a sliding. In this way, the IT2IFLS output will match the actual output as close as possible for all time $t > t_h$, where t_h is the hitting time for $E(t) = 0$.

Definition 3: A sliding motion will be on a sliding manifold $S(E(t)) = E(t) = 0$ after a time t_h if the condition $S(t)\dot{S}\,(t) < 0$ is valid $\forall t$ in some nontrivial semi open subinterval of time of the form $[t,\, t_h) \subset (-\infty, t_h)$ [22][23]

5. Parameter Update

In this study, the SMC learning is used to update the parameters of IT2IFLS. The antecedent parameters are center (c), lower membership function standard deviation ($\underline{\sigma}$) and upper membership function standard deviation ($\bar{\sigma}$) while w and b are consequent parameters with β as the user defined parameter. The update rules of the IT2IFLS parameters are adapted from [24] for IT2FLS as follows for the antecedent parameters:

For the membership function parameter update, we have the following:

$$\dot{c}_{ik} = \dot{x}_i + (x_i - c_{ik})\alpha_1 sgn(E) \tag{20}$$

$$\dot{\underline{\sigma}}_{ik} = - \left(\underline{\sigma}_{ik} + \frac{\underline{\sigma}_{ik}{}^3}{(x_i - c_{ik})^2}\right)\alpha_1 sgn(E) \tag{21}$$

$$\dot{\bar{\sigma}}_{ik} = - \left(\bar{\sigma}_{ik} + \frac{\bar{\sigma}_{ik}{}^3}{(x_i - c_{ik})^2}\right)\alpha_1 sgn(E) \tag{22}$$

where α_1 is the learning rate and E is the learning error. Since the lower non-membership assumes the upper membership function and vice versa, the standard deviation update for the upper membership function applies to standard deviation update for the lower non-membership function and vice versa.

The consequent parameters are updated as follows:

$$\dot{w}_{ik} = -\, x_i \frac{(1-\beta)\tilde{f}_k^{\mu} + \beta\, \tilde{f}_k^{\nu}}{M\,((1-\beta)\tilde{f}_k^{\mu} + \beta\, \tilde{f}_k^{\nu})} * \alpha sgn(E) \tag{23}$$

Where $M = ((1\text{-}\beta)\tilde{f}_k^{\mu} + \beta \tilde{f}_k^{\nu})^{\mathrm{T}}$.

$$\dot{b}_{ik} = -\, \frac{(1-\beta)\tilde{f}_k^{\mu} + \beta\, \tilde{f}_k^{\nu}}{M\,((1-\beta)\tilde{f}_k^{\mu} + \beta\, \tilde{f}_k^{\nu})} * \alpha sgn(E) \tag{24}$$

In order to adjust the β values, the following vectors are defined: $\tilde{F}^{\mu}(t) \equiv [\underline{\tilde{F}}^{\mu}\,(t) + \overset{\approx}{\overline{F}}{}^{\mu}\,(t)]$. For the lower membership function, we obtained: $\underline{\tilde{F}}^{\mu}\,(t)$ = $[\underline{\tilde{F}}_1^{\mu}(t)\underline{\tilde{F}}_2^{\mu}(t) \ldots\ \underline{\tilde{F}}_M^{\mu}(t)]^{\mathrm{T}}$ and $\bar{\bar{\tilde{F}}}^{\mu}(t)$ = $[\bar{\bar{\tilde{F}}}_1^{\mu}(t)\bar{\bar{\tilde{F}}}_2^{\mu}(t) \ldots\ \bar{\bar{\tilde{F}}}_M^{\mu}(t)]^{\mathrm{T}}$ for upper membership function. Similarly, for the non-membership functions: $\tilde{F}^{\nu}(t) \equiv [\underline{\tilde{F}}^{\nu}\,(t) + \overset{\approx}{\overline{F}}{}^{\nu}\,(t)]$ is specified and defined as $\underline{\tilde{F}}^{\nu}\,(t)$ = $[\underline{\tilde{F}}_1^{\nu}(t)\underline{\tilde{F}}_2^{\nu}(t) \ldots\ \underline{\tilde{F}}_M^{\nu}(t)]^{\mathrm{T}}$ and $\bar{\bar{\tilde{F}}}^{\mu}(t)$ = $[\bar{\bar{\tilde{F}}}_1^{\nu}(t)\bar{\bar{\tilde{F}}}_2^{\nu}(t) \ldots\ \bar{\bar{\tilde{F}}}_M^{\nu}(t)]^{\mathrm{T}}$ for lower and upper non-membership functions respectively. Y = $[y_1\, y_2\ \ldots\ y_M\,]$ represents rule consequent output vector. Then:

$$\beta = -\, \frac{1}{Y(\underline{\tilde{F}}^{\mu} + \bar{\bar{\tilde{F}}}^{\mu} + \underline{\tilde{F}}^{\nu} + \overset{\approx}{\overline{F}}{}^{\nu})} * \alpha sgn(E) \tag{25}$$

In this study, the membership and non-membership function parameters are tuned using SMC learning algorithm.

6. Performance Evaluation

This study analysis two real world datasets and two artificially generated datasets and compares the performances with existing studies in the literature. For all experiments, two IT2IFS are utilized in order to reduce the complexity of the model. The value of the user-defined parameter β is kept at 0.5 for all experiments which are carried out using *MATLAB*© 2019.

6.1. Australian new electricity market

The Australian new electricity market otherwise known as New South Wales (NSW) electricity market data is a real world dataset. Following from [25], four weeks corresponding to four Australian seasons of the year are adopted. These seasons are weeks in January, May, August and October given as in [25]:

- January 24 - January 30
- May 24 - May 30
- August 24 - August 30
- October 24 - October 30

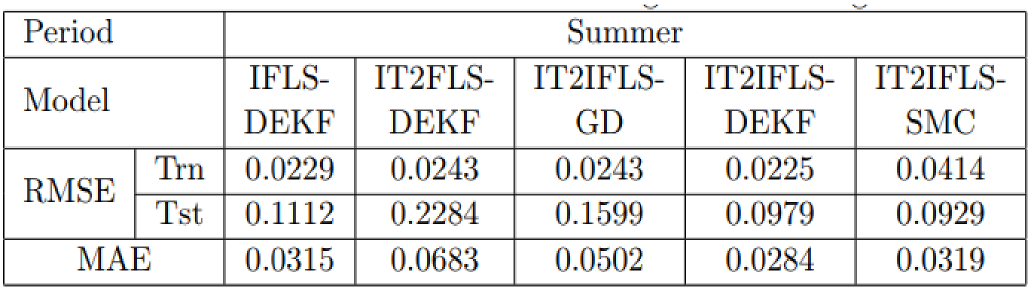
The NSW electricity dataset was analysed in [25] using wavelet neural network. According to [25], the NSW electricity market in the year 2008 was chosen for the analysis because it was the largest. In [10], the same dataset was analysed using IT2IFLS trained with the decoupled extended Kalman filter algorithm. To aid comparison with other learning algorithms, the computational arrangement is the same as that used in [10] with 336 input data for each season, 231 data points are used for training while 100 data points are used for testing. The input-output vector is [*x*(*t*-4), *x*(*t*-3), *x*(*t*-2), *x*(*t*-1); *x*(*t*+1)], where *x*(*t*+1) is the desired output. For the four inputs NSW electricity dataset, 16 rules are generated with 192 parameters. The dataset for each season is normalized to fall within the range of zero and one and the performance computed over 30 simulations with the root mean squared error (RMSE) and mean absolute error (MAE) as the performance metrics. The RMSE and MAE are computed as below:

$$\text{RMSE} = \sqrt{\frac{1}{N}\sum_{i=1}^{N}(y^{a} - y^{p})^{2}} \qquad (26)$$

$$\text{MAE} = \frac{1}{N}\sum_{i=1}^{N}|y^{a} - y^{p}| \qquad (27)$$

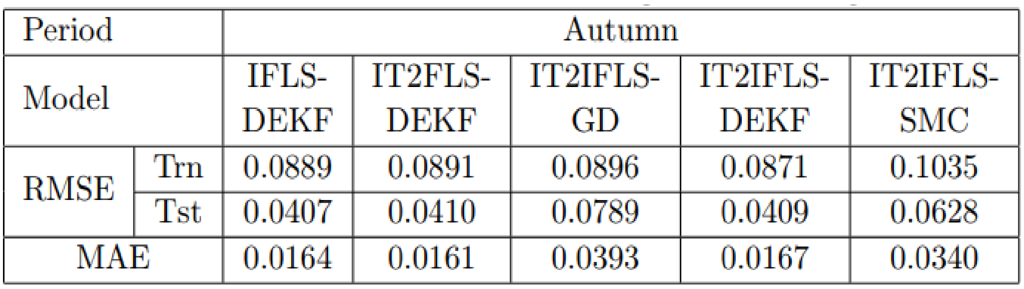
where N is the total number of test data, y^{a} and y^{p} are the actual and predicted outputs respectively.

Table 1: Performance of different models during Summer season.



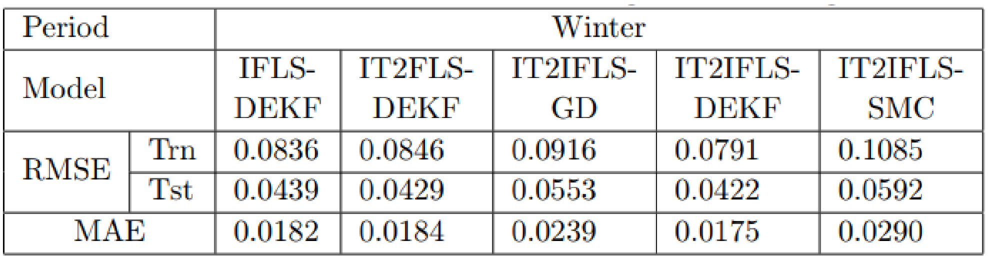
Period		Summer				
Model		IFLS-DEKF	IT2FLS-DEKF	IT2IFLS-GD	IT2IFLS-DEKF	IT2IFLS-SMC
RMSE	Trn	0.0229	0.0243	0.0243	0.0225	0.0414
	Tst	0.1112	0.2284	0.1599	0.0979	0.0929
MAE		0.0315	0.0683	0.0502	0.0284	0.0319

Table 2: Performance of different models during Autumn season.



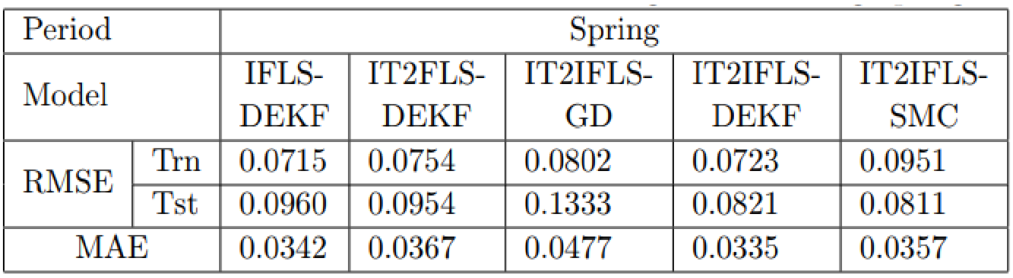
Period		Autumn				
Model		IFLS-DEKF	IT2FLS-DEKF	IT2IFLS-GD	IT2IFLS-DEKF	IT2IFLS-SMC
RMSE	Trn	0.0889	0.0891	0.0896	0.0871	0.1035
	Tst	0.0407	0.0410	0.0789	0.0409	0.0628
MAE		0.0164	0.0161	0.0393	0.0167	0.0340

Table 3: Performance of different models during Winter season.



Period		Winter				
Model		IFLS-DEKF	IT2FLS-DEKF	IT2IFLS-GD	IT2IFLS-DEKF	IT2IFLS-SMC
RMSE	Trn	0.0836	0.0846	0.0916	0.0791	0.1085
	Tst	0.0439	0.0429	0.0553	0.0422	0.0592
MAE		0.0182	0.0184	0.0239	0.0175	0.0290

Table 4: Performance of different models during Spring season.



Period		Spring				
Model		IFLS-DEKF	IT2FLS-DEKF	IT2IFLS-GD	IT2IFLS-DEKF	IT2IFLS-SMC
RMSE	Trn	0.0715	0.0754	0.0802	0.0723	0.0951
	Tst	0.0960	0.0954	0.1333	0.0821	0.0811
MAE		0.0342	0.0367	0.0477	0.0335	0.0357

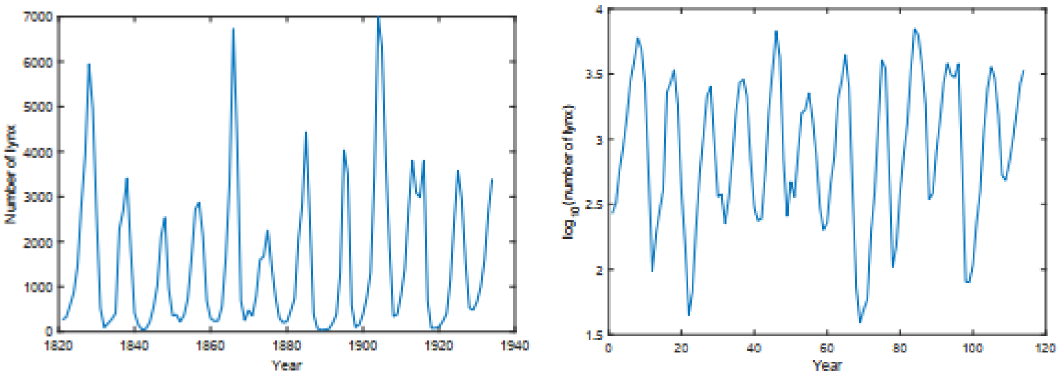
As indicated in tables 1 to 4, IT2IFLS-SMC outperforms other IT2IFLS models in terms of the test RMSE during the Summer and Spring seasons and competes favourable with other models in the other instances. The DEKF trained models tend to exhibit better prediction accuracy in terms of training RMSE in most instances while the type-1 version do perform better in some cases, an indication that a type-1 variant of FLS may work well in some instances depending on the type of data and level of uncertainty in the data.There is a loss in the performance SMC trained IT2IFLS in terms of the training RMSE. However, the computational cost of IT2IFLS-SMC is much more impressive than other competing models in all the cases. For example, the running times for IT2IFLS-SMC, GD and DEKF for one instance of NSW electricity data are 25.78secs, 28.03secs and 44.75secs respectively. Hence IT2IFLS-SMC may stands as an appropriate choice in real time applications where running time is paramount.However, using the SMC learning algorithm with the sign function exhibitedsome chattering effects in the system during the evaluation of the model. This might also explain the poor performance of IT2IFLS-SMC in the training RMSE.

6.2. Canadian Lynx Dataset

The Canadian lynx time series dataset consist of the number of lynx trapped in the McKenzie river annually in Northern Canada and is taken from period 1821 to 1934. To aid comparison with existing studies [26][27][28][10] the logarithms to base 10 of the Canadian time series is adopted with a periodicity of 10 years. The Canadian time series has a total of 114 sample points where 100 instances are used for training and the rest are used for testing. For a fair comparison with previous studies, the experiment is run for 2000 epochs. The quality of prediction is measured using the mean squared error (MSE) and MAE. The MSE is computed as below:

$$\text{MSE} = \frac{1}{N}\sum_{i=1}^{N}(y^{a} - y^{p})^{2} \tag{28}$$

While the MAE is as shown in (26).



: (a) (b)

Figure 3 (a) shows original Canadian lynx data and (b) transformed Canadian lynx data ($\log_{10)}$

Table 5: Comparison of IT2IFLS with other models on Canadian lynx data

Models	MSE(Test)	MAE(Test)
Zhang's ARIMA [26]	0.020486	0.112255
ANN [26]	0.020466	0.112109
ANN (p,d,q) [29]	0.013609	0.089625
Zhang's Hybrid ARIMA/ANNs model [26]	0.017233	0.103972
Hybrid ARIMA/ ERNN model [30]	0.009	-
SETAR [31]	0.014	-
FNN [31]	0.009	-
Generalised Hybrid ARIMA/ANNs model [32]	0.00999	0.085055
ANN/PNN model [27]	0.014872	0.079628
ARIMA/PNN model [27]	0.011461	0.084381
MNM-ANN-DEA [33]	0.00663	-
GA-BPNN [28]	0.013599	0.081477
DE-BPNN [28]	0.012899	0.080542
ANN Ensemble [34]	0.00715	-
RBF-AR [35]	0.0073	-
ADE-BPNN [28]	0.010392	0.070723
GMDH [36]	0.0082	0.0634
LSSVM [36]	0.0074	0.0657
GLSSVM [36]	0.0056	0.0552
L&NL-ANN [37]	0.006	-
IT2IFLS-GD	0.00463	0.0205
IT2IFLS-SMC	0.0045	0.0212

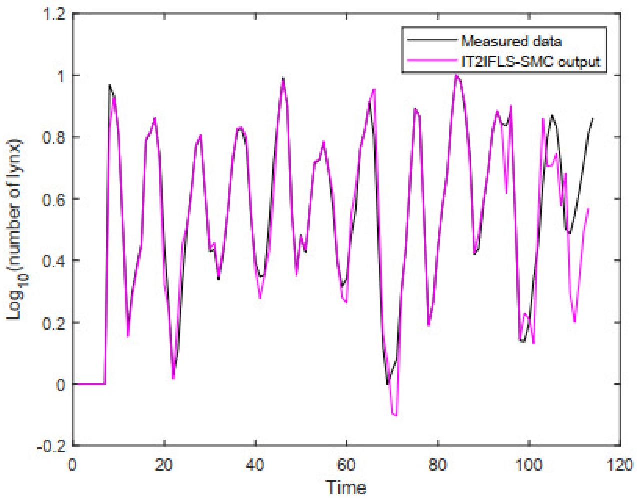


Figure 4: Actual and predicted output of Canadian lynx data

Figure 4shows the actual and predicted values of Canadian lynx data. As shown in Table 5, IT2IFLS-SMC performs better than other existing models in the literature. The SMC-based IT2IFLS prediction accuracy is very close to the GD-based IT2IFLS in this problem domain.

6.3. System identification problem 1

Here, comparison of IT2IFLS-SMC learning with other existing methods in the literature using a system identification problem is conducted. In the first non-linear dynamic system identification, we adopt a second-order time-varying system defined as:$y(t+1) = f(y(t), y(t\text{-}1), y(t\text{-}2), u(t), u(t\text{-}1))$

where $$\mathrm{f}(x_{1,}x_{2,}x_{3,}x_{4,}x_5\,) = \frac{x_{1,}x_{2,}\;\; x_3\,,\, x_5\,(x_3 - b) + cx_4}{a + x_2^2 + x_3^2} \qquad (29)$$

and *a, b*, and *c* are time-varying parameters defined as below:

$$a(t)\;\; = 1.2\;\; \text{-}\; 0.2\cos(2\pi t/T) \qquad (30)$$

$$b(t)\;\; = 1.0\;\; \text{-}\; 0.4\sin(2\pi t/T) \qquad (31)$$

$$c(t)\;\; = 1.0\;\; \text{-}\; 0.4\sin(2\pi t/T) \qquad (32)$$

Here, T = 1000 represents the total number of sample points. The parameters a, b and c take the value 1 as reported in [9]. Similar to [38], in order to keep a manageable number of parameters, only two inputs values are utilised which are u(t) and $y(t)$ with $y(t+1)$ as the desired output. The plot of $u(t)$ versus $y(t)$ is as shown in Figure 5.

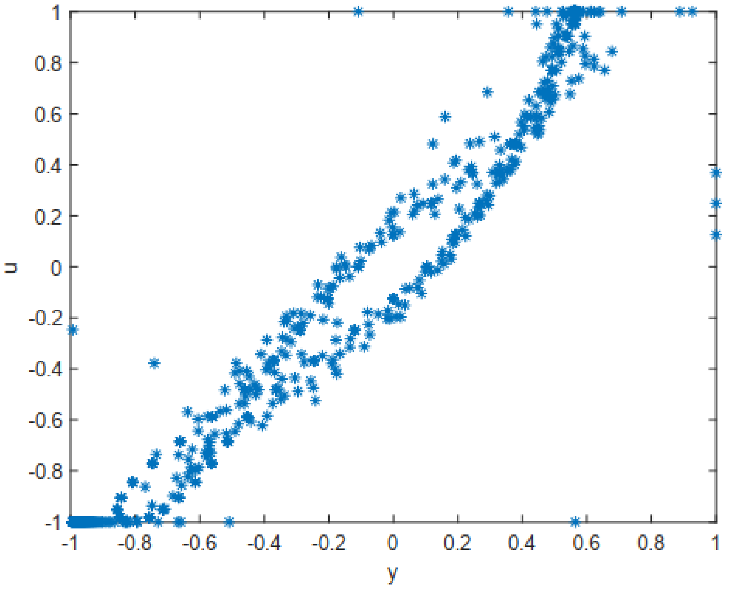


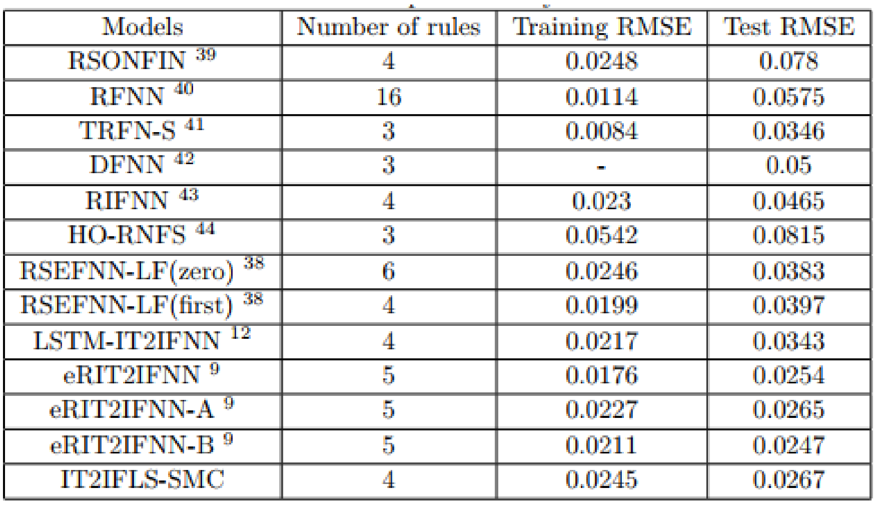
Figure 5: Plot showing the relationship between the two input u and y for system identification 1.

The following control signal is used to test the performance of the non-linear system.

$$\begin{cases} \sin\left(\frac{\pi t}{25}\right) & t < 250 \\ 1.0 & 250 \le t < 500 \\ -1.0 & 500 \le t < 750 \\ 0.3\sin\left(\frac{\pi t}{25}\right) + 0.1\sin\left(\frac{\pi t}{32}\right) + 0.6\sin\left(\frac{\pi t}{10}\right) & 750 \le t < 1000 \end{cases} \tag{33}$$

The simulation is conducted for 1000 time steps with 100 training epochs. The performance metrics used is the RMSE. For the non-linear system identification, 4 rules are obtained with 40 tunable parameters.

Table 6: Performance comparison of IT2IFLS with other models on system identification 1



Models	Number of rules	Training RMSE	Test RMSE
RSONFIN [39]	4	0.0248	0.078
RFNN [40]	16	0.0114	0.0575
TRFN-S [41]	3	0.0084	0.0346
DFNN [42]	3	-	0.05
RIFNN [43]	4	0.023	0.0465
HO-RNFS [44]	3	0.0542	0.0815
RSEFNN-LF(zero) [38]	6	0.0246	0.0383
RSEFNN-LF(first) [38]	4	0.0199	0.0397
LSTM-IT2IFNN [12]	4	0.0217	0.0343
eRIT2IFNN [9]	5	0.0176	0.0254
eRIT2IFNN-A [9]	5	0.0227	0.0265
eRIT2IFNN-B [9]	5	0.0211	0.0247
IT2IFLS-SMC	4	0.0245	0.0267

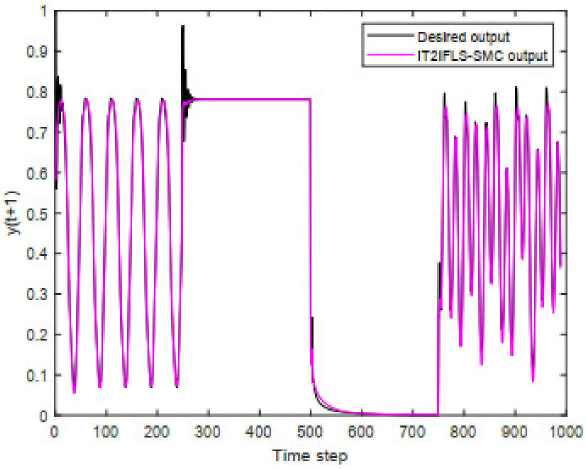


Figure 6: Actual and predicted outputs for system identification problem 1

Shown in Table 6 is the performance comparison of IT2IFLS-SMC with other learning models in the literature. As shown in Table 6, the performance of IT2IFLS is comparative to those reported in the literature especially the evolving classical IT2FLS methods reported in [9]. Figure 6 shows the actual and predicted outcome for system identification problem 1.

6.4. System identification problem 2

For the second identification problem, a non-linear dynamic plant with longer input delay is adopted and defined as:

$$y(t+1)=0.72y(t) + 0.025y(t-1)u(t-1) + 0.01u^2(t-2)+0.2u(t-3) \tag{34}$$

Equation (34) involves a one-step ahead prediction using two previous outputs and four previous inputs. The test signal in (34) is adopted to test the quality of prediction. Similar to [12], we adopt the same training data and time steps as in system identification 1. To reduce the complexity of the system, two inputs $u(t)$ and$y(t)$are passed into the IT2IFLS. Shown in Figure 6 is the relationship between the two inputs u and y for system identification 2.

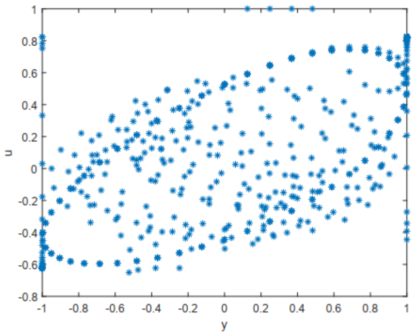
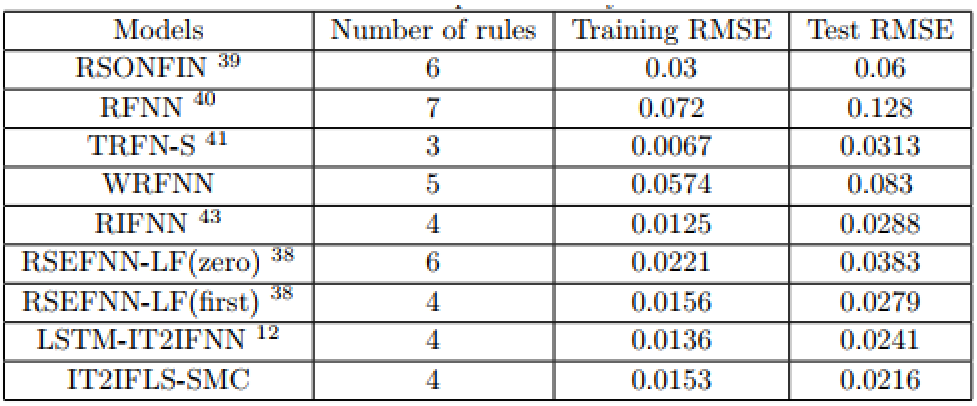


Figure 7: Plot showing the relationship between the u and y in system identification problem 2

Table 7:Performance comparison of IT2IFLS on system identification problem 2 with other models in the literature



Models	Number of rules	Training RMSE	Test RMSE
RSONFIN [39]	6	0.03	0.06
RFNN [40]	7	0.072	0.128
TRFN-S [41]	3	0.0067	0.0313
WRFNN	5	0.0574	0.083
RIFNN [43]	4	0.0125	0.0288
RSEFNN-LF(zero) [38]	6	0.0221	0.0383
RSEFNN-LF(first) [38]	4	0.0156	0.0279
LSTM-IT2IFNN [12]	4	0.0136	0.0241
IT2IFLS-SMC	4	0.0153	0.0216

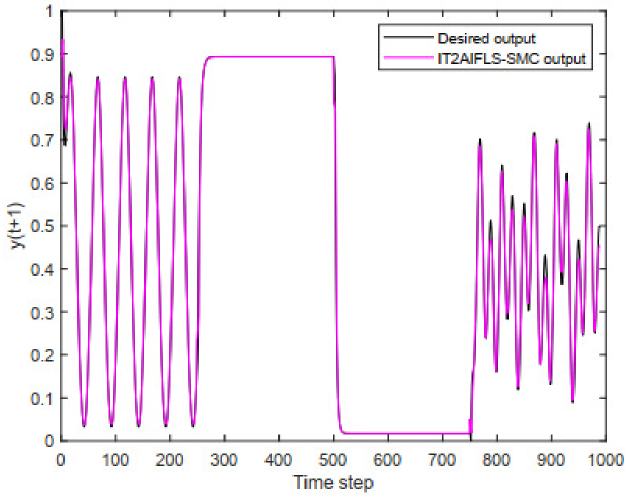


Figure 8: Actual and predicted outputsfor system identification problem 2

Figure 7 shows the actual and predicted output for system identification problem 2. As shown in the figure, the predicted output closely follows the actual output indicating a good learning performance. As shown in Table 7, IT2IFLS-SMC performs well in this problem instance in terms of the test RMSE compared to other existing models in the literature.

7. CONCLUSION

This paper analysis time series and system identification problems using IT2IFLS trained with SMC learning algorithm. As demonstrated in the experimental studies, IT2IFLS-SMC outperforms some existing studies in the literature and competes favourably with others in some problem instances in terms of the test RMSE. For the training RMSE, IT2IFLS-SMC performs poorly in many cases. A careful look at all the Australian seasons prediction, IT2IFLS-SMC performs poorly compared to other learning models in terms of the training RMSE.

In the future, it will be worthwhile to look at using other derivative-free learning algorithms and possibly hybridize them and evaluate their predictive strengths. It will also be interesting to consider other membership functions outside the conventional membership functions such as elliptic and diamond-shaped membership functions for interval type-2 intuitionistic fuzzy set.

REFERENCES

[1] Zadeh, Lotfi A (1965) "Fuzzy sets." Information and control, Vol. 8 No.3, pp338-353.

[2] Zadeh, Lotfi Asker (1975) "The concept of a linguistic variable and its application to approximate reasoning—I." Information sciences, Vol. 8, No. 3, pp199-249.

[3] Naim, Syibrah, and Hani Hagras (2014) "A type 2-hesitation fuzzy logic based multi-criteria group decision making system for intelligent shared environments." Soft Computing, Vol. 18, No. 7,pp1305-1319.

[4] Atanassov, Krassimir T. (1986) "Intuitionistic fuzzy sets." Fuzzy Sets and Systems, Vol. 20, No. 1, pp 87-96.

[5] K. Atanassov and G. Gargov (1989)"Interval valued intuitionistic fuzzy sets", Fuzzy sets and systems, Vol. 13, No. 3, pp343-349.

[6] Gorzalczany, Marian B. (1987) "A method of inference in approximate reasoning based on Fuzzy sets and systems, Vol. 21, No. 1,pp1-17.

[7] Eyoh, Imo, Robert John, and Geert De Maere (2016) "Interval type-2 intuitionistic fuzzy logic system for non-linear system prediction." 2016 IEEE International Conference on Systems, Man, and Cybernetics (SMC). IEEE.

[8] Eyoh, Imo, Robert John, and Geert De Maere (2017) "Time series forecasting with interval type-2 intuitionistic fuzzy logic systems." 2017 IEEE International Conference on Fuzzy Systems (FUZZ-IEEE). IEEE.

[9] Luo, Chao, Chenhao Tan, Xingyuan Wang, and Yuanjie Zheng (2019) "An evolving recurrent interval type-2 intuitionistic fuzzy neural network for online learning and time series prediction." Applied Soft Computing, 78,pp150-163.

[10] Eyoh, Imo, Robert John, and Geert De Maere (2017) "Extended Kalman filter-based learning of interval type-2 intuitionistic fuzzy logic system." 2017 IEEE International Conference on Systems, Man, and Cybernetics (SMC). IEEE.

[11] Eyoh, Imo, et al. (2018) "Hybrid learning for interval type-2 intuitionistic fuzzy logic systems as applied to identification and prediction problems." IEEE Transactions on Fuzzy Systems, Vol. 26, No. 5,pp2672-2685.

[12] Yuan, Wang, and Luo, Chao (2019) "Online Evolving Interval Type-2 Intuitionistic Fuzzy LSTM-Neural Networks for Regression Problems." IEEE Access 7, pp35544-35555.

[13] Eyoh, Imo, Robert John, and Geert De Maere (2018) "Interval type-2 A-intuitionistic fuzzy logic for regression problems." IEEE Transactions on Fuzzy Systems Vol. 26, No. 4,pp2396-2408.

[14] Jana, Dipak Kumar (2016) "Novel arithmetic operations on type-2 intuitionistic fuzzy and its applications to transportation problem." Pacific Science Review A: Natural Science and Engineering,Vol. 18, No. 3, 178-189.

[15] Deepak, S., et al. (2018) "Optimization of Gaussian Membership Functions using Unscented Kalman Filter." 2018 International Conference on Advances in Computing, Communications and Informatics (ICACCI). IEEE.

[16] Radhika, C., and R. Parvathi (2016) "Intuitionistic fuzzification functions." Global Journal of Pure and Applied Mathematics, Vol. 12, No. 2, pp1211-1227.

[17] Hajek, Petr, and Vladimir Olej (2015) "Intuitionistic fuzzy neural network: The case of credit scoring using text information." Engineering Applications of Neural Networks. Springer, Cham, 2015. pp337-346.

[18] Mahapatra, G. S., and T. K. Roy (2013). "Intuitionistic fuzzy number and its arithmetic operation with application on system failure." Journal of Uncertain Systems, Vol. 7, No. 2, pp92-107.

[19] Nguyen, Dzung Dinh, Long Thanh Ngo, and Long The Pham (2013) "Interval type-2 fuzzy c-means clustering using intuitionistic fuzzy sets." 2013 Third World Congress on Information and Communication Technologies (WICT 2013). IEEE.

[20] Eyoh, Imo (2018) Interval type-2 Atanassov-intuitionistic fuzzy logic for uncertainty modelling. Diss. University of Nottingham, 2018.

[21] Kayacan, Erdal, and Mojtaba Ahmadieh Khanesar (2015) Fuzzy neural networks for real time control applications: concepts, modeling and algorithms for fast learning. Butterworth-Heinemann.

[22] Ahmed, Sevil, Nikola Shakev, Andon Topalov, Kostadin Shiev, and Okyay Kaynak (2012) "Sliding mode incremental learning algorithm for interval type-2 Takagi–Sugeno–Kang fuzzy neural networks." Evolving Systems,Vol. 3, No. 3,pp179-188.

[23] Kayacan, Erdal, and Okyay Kaynak (2012) "Sliding mode control theory-based algorithm for online learning in type-2 fuzzy neural networks: application to velocity control of an electro hydraulic servo system." International Journal of Adaptive Control and Signal Processing Vol. 26, No. 7,pp645-659.

[24] Kayacan, Erkan, Erdal Kayacan, and Mojtaba Ahmadieh Khanesar (2014) "Identification of nonlinear dynamic systems using type-2 fuzzy neural networks—A novel learning algorithm and a comparative study." IEEE Transactions on Industrial Electronics, Vol. 62, No. 3,pp1716-1724.

[25] Areekul, Phatchakorn, et al. (2011) "Neural-wavelet approach for short term price forecasting in deregulated power market." Journal of International Council on Electrical Engineering, Vol. 1, No.3 (2011): pp331-338.

[26] Zhang, G. Peter (2003) "Time series forecasting using a hybrid ARIMA and neural network model." Neurocomputing, 50, pp159-175.

[27] Khashei, Mehdi, and Mehdi Bijari (2012) "A new class of hybrid models for time series forecasting." Expert Systems with Applications, Vol. 39, No. 4,pp4344-4357.

[28] Wang, Lin, Yi Zeng, and Tao Chen (2015) "Back propagation neural network with adaptive differential evolution algorithm for time series forecasting." Expert Systems with Applications, Vol. 42, No. 2, pp855-863.

[29] Khashei, Mehdi, and Mehdi Bijari (2010) "An artificial neural network (p, d, q) model for timeseries forecasting." Expert Systems with applications, Vol. 37, No.1, pp 479-489.

[30] Aladag, Cagdas Hakan, Erol Egrioglu, and Cem Kadilar (2009) "Forecasting nonlinear time series with a hybrid methodology." Applied Mathematics Letters, Vol. 22, No. 9, pp1467-1470.

[31] Kajitani, Yoshio, Keith W. Hipel, and A. Ian McLeod (2005) "Forecasting nonlinear time series with feed-forward neural networks: a case study of Canadian lynx data." Journal of Forecasting, Vol. 24, No. 2, pp105-117.

[32] Khashei, Mehdi, and Mehdi Bijari (2011) "A novel hybridization of artificial neural networks and ARIMA models for time series forecasting." Applied Soft Computing, Vol. 11, No. 2,pp2664-2675.

[33] Bas, Eren (2016). "The training of multiplicative neuron model based artificial neural networks with differential evolution algorithm for forecasting." Journal of Artificial Intelligence and Soft Computing Research, Vol. 6, No.1,pp5-11.

[34] Adhikari, Ratnadip, and R. K. Agrawal (2013) "A homogeneous ensemble of artificial neural networks for time series forecasting." arXiv preprint arXiv:1302.6210.

[35] Gan, Min, and Hui Peng (2012) "Stability analysis of RBF network-based state-dependent autoregressive model for nonlinear time series." Applied Soft Computing, Vol. 12, No.1,pp174-181.

[36] Samsudin, R., P. Saad, and A. Shabri (2011) "A hybrid GMDH and least squares support vector machines in time series forecasting." Neural Network World, Vol. 21, No. 3, pp251.

[37] Yolcu, Ufuk, Erol Egrioglu, and Cagdas H. Aladag (2013) "A new linear and nonlinear artificial neural network model for time series forecasting." Decision support systems, Vol. 54, No. 3,pp1340-1347.

[38] Juang, Chia-Feng, Yang-Yin Lin, and Chiu-Chuan Tu (2010) "A recurrent self-evolving fuzzy neural network with local feedbacks and its application to dynamic system processing." Fuzzy Sets and Systems, Vol. 161, No. 19, pp2552-2568.

[39] Juang, Chia-Feng, and Chin-Teng Lin (1999) "A recurrent self-organizing neural fuzzy inference network." IEEE Transactions on Neural Networks, Vol. 10, No. 4, pp828-845.

[40] Lee, Ching-Hung, and Ching-Cheng Teng (2000) "Identification and control of dynamic systems using recurrent fuzzy neural networks." IEEE Transactions on fuzzy systems, Vol. 8, No.4, pp349-366.

[41] Juang, Chia-Feng (2002) "A TSK-type recurrent fuzzy network for dynamic systems processing by neural network and genetic algorithms." IEEE Transactions on Fuzzy Systems, Vol. 10, No.2,pp155-170.

[42] Mastorocostas, Paris A., and John B. Theocharis (2002) "A recurrent fuzzy-neural model for dynamic system identification." IEEE Transactions on Systems, Man, and Cybernetics, Part B (Cybernetics) Vol. 32, No. 2, pp176-190.

[43] Juang, Chia-Feng, Yang-Yin Lin, and Ren-Bo Huang (2011) "Dynamic system modeling using a recurrent interval-valued fuzzy neural network and its hardware implementation." Fuzzy Sets and Systems, Vol. 179, No.1, pp83-99.

[44] Theocharis, J. B. (2006) "A high-order recurrent neuro-fuzzy system with internal dynamics: Application to the adaptive noise cancellation." Fuzzy Sets and Systems, Vol. 157, No.4, pp471-500.

AUTHORS

Eyoh, Imo Jeremiah received the B.Sc. and M.Sc. degrees in Computer Science from the University of Uyo and Ibadan, all in Nigeria in 1999 and 2005 respectively. She obtained her PhD in Computer Science from the University of Nottingham, United Kingdom in 2018. She is currently working in the Department of Computer Science, University of Uyo, as a Senior lecturer. Her main research interest is uncertainty modelling using fuzzy logic. In particular, she works in the area of type-2 fuzzy logic (classical and intuitionistic) with application to diverse problem domains. She has published many papers in reputable national and international journals and conferences. She is a member of Computer Professionals Registration Council of Nigeria (CPN) and Organisation for Women in Science for the Developing World (OWSD). She is also a member of IEEE.

Eyoh, Jeremiah Effiong received the B.Sc in Computer Engineering from the University of Uyo in 2006, Nigeria. He received the M.Sc in Control Engineering from the University of Sheffield, United Kingdom in 2008. He is currently working towards the Ph.D. degree in Electrical, Electronics and Systems Engineering at the Loughborough University, United Kingdom.
Short Biography