

A Novel Non-Iterative Parameter Estimation Method for Interval Type-2 Fuzzy Neural Networks Based on a Dynamic Cost Function

Mojtaba Ahmadih Khanesar¹, Saima Hassan² and Erik Cambria³, Erdal Kayacan⁴

Abstract—Non-iterative methods for parameter estimation for interval type-2 neuro-fuzzy structure are fast to implement, when compared to online methods, and need no –or a few– parameters to be tuned. In this paper, a novel dynamic cost function, which defines a relationship between the current and past errors, is defined. The minimization of the aforementioned cost function results in a decreasing sequence of error which makes the proposed method numerically more stable when compared to least squares-based methods. It is a well-known phenomenon that a matrix inversion may cause problems if the matrix to be inverted is ill-defined i.e. its condition number is far bigger than one. The use of a dynamic relationship between the current and past error adds more degrees of freedom which makes it possible to improve the condition number of the matrix. Comprehensive simulation studies are presented for the prediction of financial data sets. The simulation results shows the superior numerical stability of the proposed method as the mean value of the condition number is smaller. This finding results in more accurate matrix inversion to be done in the two-step matrix inversion.

I. INTRODUCTION

Interval type-2 fuzzy logic systems (IT2FLSs) become well-known systems for their ability to deal with uncertainty, noisy and incomplete data. IT2FLSs benefit from type-2 membership functions (MFs) with fuzzy membership grades [1]. This structure demonstrated superior performance over their type-1 counterparts specially when high levels of noise and uncertainties exist in the system [2]–[5].

Among various learning algorithms, the Gradient descent-based algorithms have been dominantly utilized for training the neural networks (NNs) [6]. The back-propagation (BP) algorithms and its variants suffer from various learning issues including stop criteria, number of learning epoch and the possibility of entrapment in a local minimum during training of NNs [7] and [8]. Such issues have emerged novel non-iterative learning schemes [9]. Various least squares (LS) based methods are proposed to solve the limitations of BP algorithms and has shown its efficiency in training the NNs [10]. This method is a non-iterative method and makes it possible to find the results in two iterations. The fact

that there is no need to tune any parameters, makes this algorithm a very fast algorithm which does not necessitate any iterations to select its parameters.

The LS based method in [8] can be considered as a feed-forward NN (FFNN) where learning of the system takes place in two stages. In the first stage, the weights connecting input layer and the hidden layer are selected randomly [11], [12]. In the second stage, the connections with output neurons (weights of output layer) are analytically determined. This makes LS training of FFNN as a linear learning problem where only the connections with output neurons need to be adjusted. Solving a linear system with the random parameters reduces computation time than that of the iterative BP algorithms without sacrificing the prediction accuracy. The efficacy of this method over the support vector machines and its variants has been empirically shown [13]. Other variants of non-iterative learning algorithms are evolutionary ELM [7], optimally pruning ELM (OP-ELM) [14], and bidirectional ELM (B-ELM) [15] that tries decreases the number of hidden nodes without affecting learning effectiveness of the algorithm. In these algorithms the optimal nodes are selected in a way that the ones with lower residual error among the several randomly generated hidden nodes.

In this paper, an innovative non-iterative training algorithm to tune the parameters of IT2FLSs is introduced. Although previous non-iterative optimization methods are based on the minimization of sum of squared error, in the proposed method, the cost function is sum of squared of a discrete time dynamic relationship in the form of $s(t) = e(t) + \lambda e(t-1)$, $|\lambda| < 1$. Such cost function has been previously used in [16], but not solved using non-iterative approaches. Since absolute value of the parameter λ is less than one, minimization of the sum of squares results in decreasing error with respect to time. Although, it can be proved that existing non-iterative training methods can optimally tune the consequent part parameters of IT2FLSs, the proposed method shows superior performance with the test data. This is mainly because of the numerical stability caused by the minimization of dynamic relationship. One of the main challenges with existing non-iterative optimization is that they include matrix inversion which may cause numerical instability. However, the minimization of the proposed cost function which is based on the current and past error improves the condition number of the matrices to be inverted. The condition number in this case depends on the selection of the parameter λ which adds more degrees of freedom to the optimization method. To test the performance of the proposed training

*This work was not supported by any organization

¹M. Ahmadih Khanesar is with Faculty of Engineering, Mechanical, Materials and Manufacturing Department, University of Nottingham, NG7 2RD Nottingham, UK ezzma5@email.nottingham.ac.uk

²Saima Hassan is with Institute of Information Technology, Kohat University of Science and Technology, Kohat, KPK, Pakistan saimahassan@kust.edu.pk

⁴Erik Cambria is with the School of Computer Science and Engineering, Nanyang Technological University, Singapore cambria@ntu.edu.sg

²Erdal Kayacan is with Aarhus University, Department of Engineering, Aarhus C, DK-8000, Denmark erdal@eng.au.dk

method in dealing with real time datasets, it is applied on the prediction of SGD/USD and EURO/USD exchange rates. The simulation results show superior performance for the test data when compared to the existing non-iterative algorithms for training of the IT2FLSs.

This paper is organized as follows. The structure of IT2FLSs and existing non-iterative optimization methods are introduced in Section II. In Section III, the proposed non-iterative optimization method based on proposed dynamic cost function is discussed. Simulation results are presented in Section IV. The concluding marks are presented in Section V.

II. STRUCTURE OF THE INTERVAL TYPE 2 FUZZY LOGIC SYSTEM

Type-2 fuzzy MFs benefit from a secondary MFs which make the membership grade fuzzy. The fuzzy system, which is constructed using this type of MFs, are called generalized type-2 fuzzy logic systems. However, if the secondary membership grades are considered as either equal to *zero* or *one*, the resulting fuzzy system is called IT2FLS. This simplification makes it possible for IT2FLSs to be executed much faster than the general IT2FLSs. The main difference between T2FLSs and T1FLSs is that in IT2FLS output processing block which consists of a type-reducer followed by a defuzzifier block replaces defuzzifier block of T1FLSs (see Fig. 1).

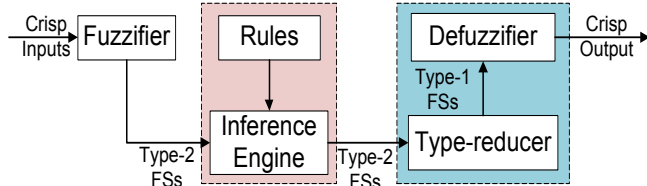


Fig. 1: Structure of the IT2FLS.

A generalized IF-THEN rule (R^n) with N number of rules for an IT2FLS is as follows [17]:

R^n : IF x_1 is $\tilde{X}_1^n \wedge x_2$ is $\tilde{X}_2^n \wedge \dots \wedge x_d$ is \tilde{X}_d^n

THEN $y^n(\mathbf{x}) = c_0^n + c_1^n x_1 + \dots + c_d^n x_d$, $n = 1, \dots, N$

The input to each fuzzy rule a vector of d -dimensional input $\mathbf{x} = [x_1, x_2, \dots, x_d]^T$, which maps the fuzzy set to a varying singleton y^n . \tilde{X}_i^n , $i = 1 \dots, d$, are the i_{th} interval type-2 fuzzy subset generated from the input variable x_i in the n_{th} rule domain with N being the total number of fuzzy rules. $\mathbf{c}^n = [c_0^n, c_1^n, \dots, c_d^n]^T$ represents the parameters of n_{th} fuzzy rule.

Since Gaussian MF are sufficiently smooth functions which are one of the most frequently used fuzzy MFs. However, in the case an interval value is considered for the center of the Gaussian MF, the resulting upper and lower Gaussian MFs are not differentiable. Such Gaussian type-2 MF with a fixed σ_i^n parameter and an interval center value that takes on values in $[m_i^{n1}, m_i^{n2}]$, [18] and [17] is as

follows:

$$\mu_{\tilde{X}_i^n}(x_i) = \exp\left[-\frac{1}{2}\left(\frac{x_i - m_i^n}{\sigma_i^n}\right)^2\right], \quad m_i^n \in [m_i^{n1}, m_i^{n2}] \quad (1)$$

The type-2 fuzzy MF $\mu_{\tilde{X}_i^n}(x_i)$ of (1) has upper fuzzy MF, $\bar{\mu}_{\tilde{X}_i^n}(x_i)$, and lower fuzzy MF, $\underline{\mu}_{\tilde{X}_i^n}(x_i)$, as follows:

$$\bar{\mu}_{\tilde{X}_i^n}(x_i) = \begin{cases} N(m_i^{n1}, \sigma_i^n; x_i), & x_i < m_i^{n1} \\ 1, & m_i^{n1} \leq x_i \leq m_i^{n2} \\ N(m_i^{n2}, \sigma_i^n; x_i), & x_i > m_i^{n2} \end{cases} \quad (2)$$

$$\underline{\mu}_{\tilde{X}_i^n}(x_i) = \begin{cases} N(m_i^{n2}, \sigma_i^n; x_i), & x_i \leq \frac{m_i^{n1} + m_i^{n2}}{2} \\ N(m_i^{n1}, \sigma_i^n; x_i), & x_i > \frac{m_i^{n1} + m_i^{n2}}{2} \end{cases} \quad (3)$$

The firing strength of a rule corresponding to an input set is an interval value, $F^n = [\underline{f}^n, \bar{f}^n]$, where

$$\underline{f}^n(x) = \underline{\mu}_{\tilde{X}_1^n}(x_1) \star \underline{\mu}_{\tilde{X}_2^n}(x_2) \star \dots \star \underline{\mu}_{\tilde{X}_d^n}(x_d) = \prod_{i=1}^d \underline{\mu}_{\tilde{X}_i^n}(x_i) \quad (4)$$

and

$$\bar{f}^n(x) = \bar{\mu}_{\tilde{X}_1^n}(x_1) \star \bar{\mu}_{\tilde{X}_2^n}(x_2) \star \dots \star \bar{\mu}_{\tilde{X}_d^n}(x_d) = \prod_{i=1}^d \bar{\mu}_{\tilde{X}_i^n}(x_i) \quad (5)$$

The output of the IT2FLS is obtained as follows:

$$\begin{aligned} Y &= [y_l, y_r] \\ &= \int_{y^1 \in [\underline{y}_1, \bar{y}_1]} \dots \int_{y^N \in [\underline{y}_N, \bar{y}_N]} \\ &\int_{f_1 \in [\underline{f}_1, \bar{f}_1]} \dots \int_{f_N \in [\underline{f}_N, \bar{f}_N]} 1 \left/ \frac{\sum_{n=1}^N f_n y^n}{\sum_{n=1}^N f_n} \right. \end{aligned} \quad (6)$$

where:

$$y^n = c_0^n + \sum_{i=1}^d c_i^n x_i = \sum_{i=0}^d c_i^n x_i \quad (7)$$

and $x_0 = 1$.

The indices l and r in the IT2 fuzzy sets $[y_l, y_r]$ represent left and right limits, respectively. There exists different approaches to calculate the outputs of IT2FLSs [19], among which Enhanced Karnik-Mendel (EKM) method [20] is used because of its efficacy and most frequently usage. Assume $\mathbf{y} = [y^1, \dots, y^N]^T$ as the original rule-ordered consequent values, and $\tilde{\mathbf{y}} = [\tilde{y}^1, \dots, \tilde{y}^N]^T$ as the reordered consequent values, where $\tilde{y}^1 \leq \tilde{y}^2 \leq \dots \leq \tilde{y}^N$. The parameters \underline{f}_n and \bar{f}_n are also reordered corresponding to $\tilde{\mathbf{y}}$. The resulting vectors are represented by $\tilde{\underline{f}}_n$ and $\tilde{\bar{f}}_n$, respectively. Finally, outputs y_l and y_r in (6) are obtained as follows.

$$y_l = \frac{\sum_{n=1}^L \tilde{\bar{f}}_n \tilde{y}^n + \sum_{n=L+1}^N \tilde{\underline{f}}_n \tilde{y}^n}{\sum_{n=1}^L \tilde{\bar{f}}_n + \sum_{n=L+1}^N \tilde{\underline{f}}_n} \quad (8)$$

and

$$y_r = \frac{\sum_{n=1}^R \tilde{\underline{f}}_n \tilde{y}^n + \sum_{n=R+1}^N \tilde{\bar{f}}_n \tilde{y}^n}{\sum_{n=1}^R \tilde{\underline{f}}_n + \sum_{n=R+1}^N \tilde{\bar{f}}_n} \quad (9)$$

where L and R are computed using several iterations of EKM. Detailed step by step procedure of EKM model is presented in in [21] and [22]. The final crisp value of IT2FLS is obtained using the mean of y_l and y_r as follows.

$$y = \frac{(y_l + y_r)}{2} \quad (10)$$

A. Basic Non-iterative Optimization Algorithm

Type-2 fuzzy extreme learning algorithm (T2FELA) is a successful non-iterated algorithm designed for SLFNs using interval type-2 fuzzy premise parts [23]. This algorithm is summarized as follows.

For a training tuple in the form of $(\mathbf{x}_i, \mathbf{y}_i)_{i=1}^D \subset R^d \times R^m$, an arbitrary SLFN with \tilde{N} number of hidden nodes is represented as follows.

$$\sum_{j=1}^{\tilde{N}} \beta_{j,k} g_j(\mathbf{x}_i) = \sum_{j=1}^{\tilde{N}} \beta_j g(\mathbf{x}_i; \mathbf{w}_j, b_j) = \mathbf{y}_{i,k}, i = 1, \dots, D$$

$$k = 1, \dots, m$$

where $g(\mathbf{x}_i; \mathbf{w}_j, b_j)$ is the activation function acting on the hidden layer and \mathbf{w}_j and b_j are its parameters. The prediction error is considered as follows:

$$e_{ik} = \mathbf{y}_{i,k} - \sum_{j=1}^{\tilde{N}} \beta_{j,k} g_j(\mathbf{x}_i) \quad i = 1, \dots, D$$

$$k = 1, \dots, m$$

III. MODIFIED NON-ITERATIVE TRAINING BASED ON DYNAMIC COST FUNCTION

A discrete time dynamic relationship between current and past error is defined as follows:

$$s_{ik} = e_{ik} + \lambda e_{(i-1)k} \quad i = 1, \dots, D$$

$$k = 1, \dots, m \quad (11)$$

where $\lambda > 0$ is a design parameter which is chosen as $|\lambda| < 1$ to guarantee the stability of the dynamic surface. The equation (11) can be represented in the following matrix form:

$$\mathbf{H}_{2,D} \boldsymbol{\beta} + \lambda \mathbf{H}_{1,D-1} \boldsymbol{\beta} = \mathbf{Y}_{2,D} + \lambda \mathbf{Y}_{1,D-1} \quad (12)$$

where

$$\mathbf{H}_{i,j}(\mathbf{x}_i, \dots, \mathbf{x}_j; \mathbf{w}_1, \dots, \mathbf{w}_{\tilde{N}}, b_1, \dots, b_{\tilde{N}}) \quad (13)$$

$$= \begin{bmatrix} g(\mathbf{x}_i; \mathbf{w}_1, b_1) & \dots & g(\mathbf{x}_i; \mathbf{w}_{\tilde{N}}, b_{\tilde{N}}) \\ \vdots & \ddots & \vdots \\ g(\mathbf{x}_j; \mathbf{w}_1, b_1) & \dots & g(\mathbf{x}_j; \mathbf{w}_{\tilde{N}}, b_{\tilde{N}}) \end{bmatrix}_{(i-(j+1)) \times \tilde{N}}$$

$$\boldsymbol{\beta} = \begin{bmatrix} \beta_1^T \\ \vdots \\ \beta_{\tilde{N}}^T \end{bmatrix}_{\tilde{N} \times m} = \begin{bmatrix} \beta_1^1 & \dots & \beta_1^m \\ \dots & \ddots & \dots \\ \beta_{\tilde{N}}^1 & \dots & \beta_{\tilde{N}}^m \end{bmatrix}_{\tilde{N} \times m} \quad (14)$$

and

$$\mathbf{Y}_{i,j} = \begin{bmatrix} \mathbf{y}_i^T \\ \vdots \\ \mathbf{y}_j^T \end{bmatrix}_{(j-(i+1)) \times m} \quad (15)$$

where \mathbf{H} is a matrix composed of the outputs of hidden layer; generated randomly with parameters \mathbf{w}_j and b_j . $\boldsymbol{\beta}$ is the output weight matrix and \mathbf{y}^T represent the transpose of vector \mathbf{y} . The parameter $\hat{\boldsymbol{\beta}}$, the optimal values of $\boldsymbol{\beta}$ is obtained as the solution to (12) as follows.

$$\hat{\boldsymbol{\beta}} = (\mathbf{H}_{2,D} + \lambda \mathbf{H}_{1,D-1})^\dagger (\mathbf{Y}_{2,D} + \lambda \mathbf{Y}_{1,D-1}) \quad (16)$$

where \mathbf{H}^\dagger is the *Moore-Penrose generalized inverse* of matrix \mathbf{H} [24] and [25].

The proposed parameter update rule based on (16) is presented in Fig. 2.

Remark 1. Since the most important novelty of the present paper lies in its novel cost function, this cost function needs to be justified in more details. In the case this cost function made equal to zero, a relationship between the past and current error is satisfied. In this relationship, the current error is equal to the past error when it is multiplied by a value λ which is less than *one*. This means that the error has a decreasing value which may result in more stability for the training phase.

Remark 2. It is important to note that although the parameter λ adds more degrees of freedom, its selection may be challenging and may require trial and error. Based on our experiments, an appropriate selection of this parameter would be $\lambda = -0.5$. However, the tuning of this parameter may result in superior performance.

IV. SIMULATION RESULTS

A. Performance indexes

The proposed algorithm is tested according to the most common performance indexes namely mean absolute percentage error (MAPE), adopted mean absolute percentage error (AMAPE) [26], symmetric mean absolute percentage error (SMAPE) and root mean squared error (RMSE).

$$MAPE = \frac{1}{N} \sum_{t=1}^N \left| \frac{A_t - F_t}{A_t} \right| \times 100 \quad (17)$$

$$AMAPE = \frac{1}{N} \sum_{t=1}^N \left| \frac{A_t - F_t}{\frac{1}{N} \sum_{t=1}^N A_t} \right| \times 100 \quad (18)$$

$$RMSE = \sqrt{\frac{1}{N} \sum_{t=1}^N (A_t - F_t)^2} \quad (19)$$

where N represent test sample number. The parameters A_t and F_t represent actual and foretasted values at time instance t .

Symmetric mean absolute percentage error (SMAPE) is an error indicator based on the relationship between absolute error and the average value of real and predicted values. This performance index is represented as follows:

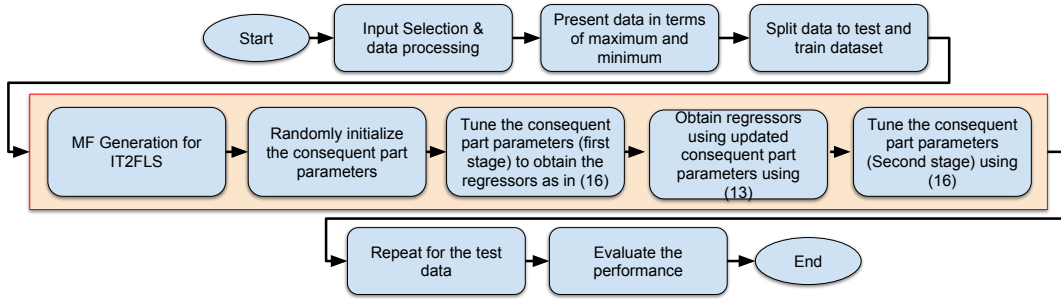


Fig. 2: Sequential diagram of the proposed training algorithm

$$\text{SMAPE} = \frac{1}{N} \sum_{t=1}^n \frac{|F_t - A_t|}{(|A_t| + |F_t|)/2} \times 100 \quad (20)$$

where N , A_t and F_t are defined in the same way as in (19).

B. Prediction of the SGD/USD exchange rate

Cross border activities and foreign exchange market require the prediction of exchange rate between major currencies across the globe. Accurate forecast of the exchange rate results in considerable decrease in risks and makes the overall financial activity profitable. Singapore Dollar as one of the major currencies is chosen here to test the prediction accuracy of the proposed approach when it comes to financial data. The data is chosen from 3 – Jan – 1994 to 29 – Apr – 2016. Totally, the number of entries in investigated data set is equal to 5735 samples. The missing data appears in the dataset as a result of holidays, in which the latest exchange rate is used to fill the missing data entry. The data samples from 3 – Jan – 1994 till 14 – Jan – 2005 are used for training and the data samples from 15 – Jan – 2005 till 29 – Apr – 2016 are used for testing of the proposed method.

Six individual fuzzy rules are used to predict the financial data with their center and sigma variables being completely random variables. The data set is normalized between *zero* and *one* and the centers are uniformly distributed in this interval. The sigma values are chosen uniformly from the interval of [0.2, 0.5]. In order to have statistically meaningful results, the experiments are repeated for 30 times and the mean values are reported in the Table I. As can be seen from this table, performance of the proposed method is better than T2FELA considerably. Moreover, the prediction performance for test data is illustrated in Fig. 3a. This figure shows that it is possible to use the data points between 3 – Jan – 1994 and 14 – Jan – 2005 and the trained system is capable of predicting the exchange rate of 15 – Jan – 2005 till 29 – Apr – 2016. Figure 3b depicts that the graph of identified data with respect to real data is very close to $y = x$ line.

As mentioned earlier, the reason behind better performance of the proposed algorithm over T2FELA is that the addition of a parameter to the cost function makes it more stable. In this case, the mean value of condition number for the proposed method is obtained as 1.8619×10^6 while it is being equal to 1.1345×10^7 for T2FELA. Similar to the previous case, this fact clearly shows the reason why

TABLE I: Results of comparison between the proposed SMC based non-iterative optimization method and T2FELA.

Index	Test Data		Train Data	
	The proposed method	T2FELA	The proposed method	T2FELA
MAPE %	0.4325	0.7353	0.2505	0.4715
AMAPE %	0.4477	0.7561	0.2465	0.4642
RMSE	0.0044	0.0074	0.0024	0.0045
SMAPE %	0.4342	0.7374	0.2505	0.4715

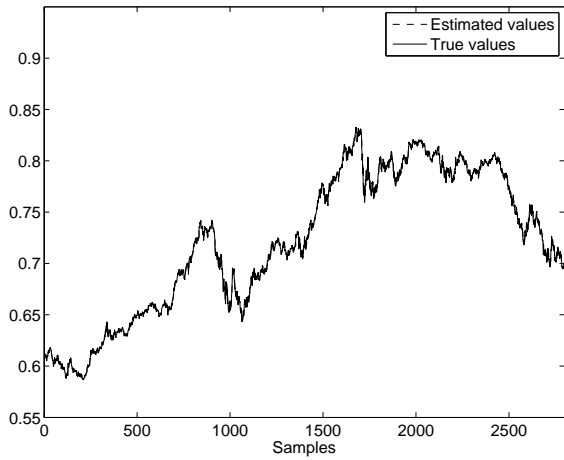
the results obtained by the proposed method outperform the results obtained by T2FELA.

C. Prediction of the Euro/USD exchange rate

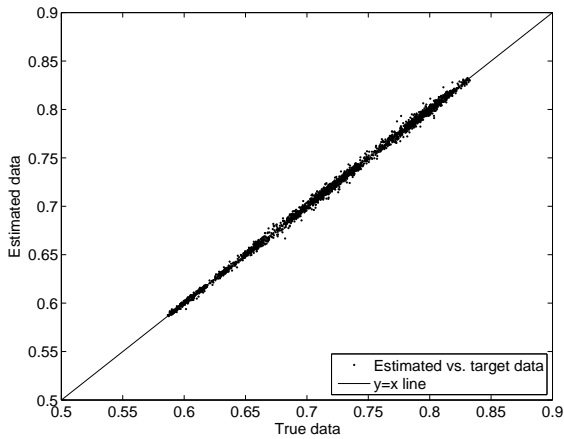
In this section, the proposed non-iterative algorithm is used to predict the daily exchange rate of Euro versus USD for 4558 days in the interval of 30 – Oct – 1998 to 15 – Aug – 2016. For missing data resulted from holidays or unavailable data samples, the latest exchange rate is used. The data samples from 30 – Oct – 1998 till 8 – Aug – 2007 is used for training and the data samples from 9 – Aug – 2007 till 15 – Aug – 2016 are used for testing the proposed method.

Four rules are considered for the fuzzy system with completely random center and sigma variables. For the prediction purpose, the exchange rates are normalized to the interval of [0, 1] and the centers are uniformly distributed in this interval. However, the indices are all evaluated based on the true data. The sigma values are chosen uniformly from the interval of [0.2, 0.5]. Similar to the previous case, Table II illustrates that the proposed method outperforms T2FELA considerably. In order to have statistically meaningful results, the experiments are repeated for 30 times and the mean values are reported in the table. Moreover, the prediction performance for test data is presented in Fig. 4a. Moreover, Figure 4c shows how identification error is distributed around *zero*. Figure 4d depicts that the graph of identified data with respect to real data is very close to $y = x$ line.

Similarly, the authors believe that the reason behind better performance of the proposed algorithm over T2FELA is the proposed method is numerically more stable. The fact that the mean value of condition number for the proposed method is obtained as 3.5774×10^8 while it is being equal to 1.9220×10^9 for T2FELA supports this idea.



(a) Testing performance of the proposed algorithm for the prediction of SGD/USD exchange rate



(b) Estimated values versus real values for the prediction of SGD/USD exchange rate

Fig. 3: Estimated values versus real values for the prediction of SGD/USD exchange rate

V. CONCLUSION

The use of dynamic cost function in this paper results in overcoming the numerical problems caused by matrix inversion needed to estimate the parameters of the consequent part of IT2FLSs when using the T2FELA. Using the proposed dynamic cost function, a novel formula is extracted for the consequent part parameters of IT2FLS. Since the proposed cost function benefits from a parameter, it has more degrees of freedom which makes it possible to obtain a more numerically stable parameter update rule which results in better approximation of the true values. The proposed method is applied to predict financial datasets. Having a good prediction is required for decreasing the risk of decisions. This is the main reason why a foreign exchange dataset is selected in this study to test the prediction capability of the proposed approach. As can be inferred from the simulation results, the proposed method makes the condition number for the matrix inversion in the estimation of the parameters of the consequent part of IT2FNN closer to *one*, which is highly

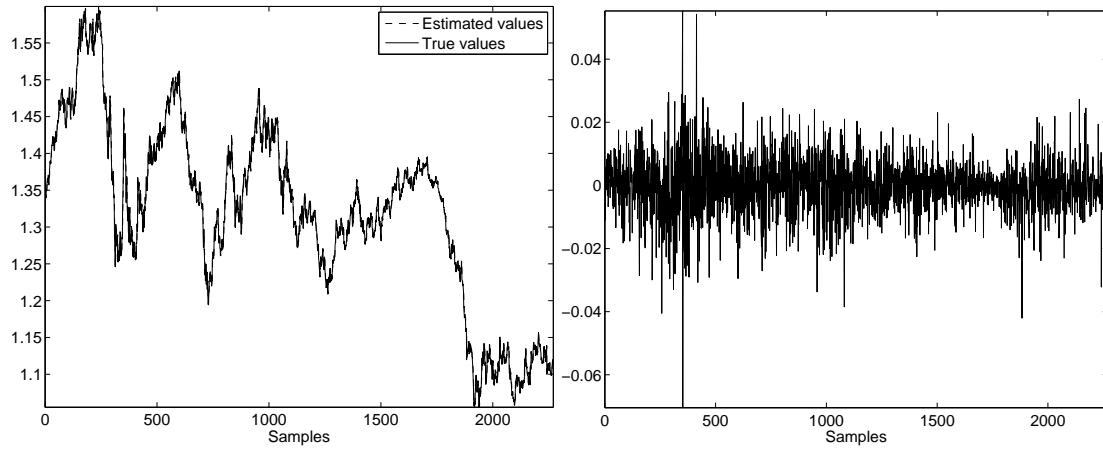
TABLE II: Results of comparison between the proposed SMC based non-iterative optimization method and T2FELA.

Index	Test	Data	Train	Data
	The proposed method	T2FELA	The proposed method	T2FELA
MAPE %	0.8009	1.4422	0.5316	0.9952
AMAPE %	0.8400	1.5065	0.5198	0.9730
RMSE	0.0189	0.0332	0.0073	0.0139
SMAPE %	0.8088	1.4541	0.5316	0.9953

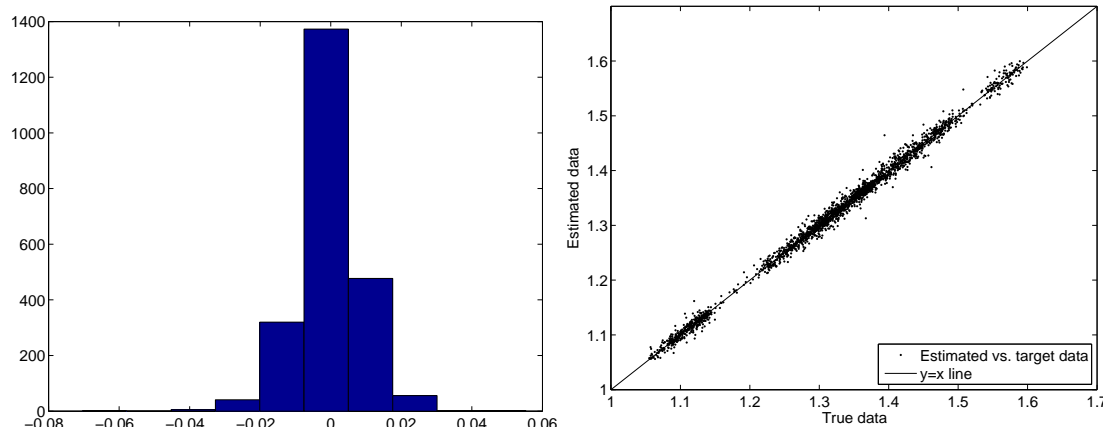
desirable. Furthermore, the proposed algorithm shown to be capable of outperforming T2FELA if appropriate parameter values are selected for this algorithm.

REFERENCES

- [1] J. M. Mendel, "Uncertain rule-based fuzzy logic system: introduction and new directions," 2001.
- [2] Y. Liu, J. Yi, R. Yuan, and Z. Pu, "Adaptive type-2 fuzzy control for flexible air-breathing hypersonic vehicles with measurement noises," in *2018 Annual American Control Conference (ACC)*. IEEE, 2018, pp. 1485–1490.
- [3] E. Kayacan and M. A. Khanesar, *Fuzzy Neural Networks for Real Time Control Applications: Concepts, Modeling and Algorithms for Fast Learning*. Butterworth-Heinemann, 2015.
- [4] E. Kayacan, O. Kaynak, R. Abiyev, J. Tørresen, M. Høvin, and K. Glette, "Design of an adaptive interval type-2 fuzzy logic controller for the position control of a servo system with an intelligent sensor," in *international conference on fuzzy systems*. IEEE, 2010, pp. 1–8.
- [5] E. Kayacan, E. Kayacan, H. Ramon, and W. Saeys, "Towards agrobots: Identification of the yaw dynamics and trajectory tracking of an autonomous tractor," *Computers and Electronics in Agriculture*, vol. 115, pp. 78–87, 2015.
- [6] M. A. Khanesar, E. Kayacan, M. Teshnehlab, and O. Kaynak, "Levenberg marquardt algorithm for the training of type-2 fuzzy neuro systems with a novel type-2 fuzzy membership function," in *2011 IEEE symposium on advances in type-2 fuzzy logic systems (T2FUZZ)*. IEEE, 2011, pp. 88–93.
- [7] Q.-Y. Zhu, A. Qin, P. Suganthan, and G.-B. Huang, "Evolutionary extreme learning machine," *Pattern Recognition*, vol. 38, no. 10, pp. 1759 – 1763, 2005.
- [8] G.-B. Huang, Q.-Y. Zhu, and C.-K. Siew, "Extreme learning machine: Theory and applications," *Neurocomputing*, vol. 70, no. 1-3, pp. 489 – 501, 2006.
- [9] S. Hassan, M. A. Khanesar, E. Kayacan, J. Jaafar, and A. Khosravi, "Optimal design of adaptive type-2 neuro-fuzzy systems: A review," *Applied Soft Computing*, vol. 44, pp. 134–143, 2016.
- [10] G.-B. Huang, E. Cambria, K.-A. Toh, B. Widrow, and Z. Xu, "New trends of learning in computational intelligence [guest editorial]," *IEEE Computational Intelligence Magazine*, vol. 10, no. 2, pp. 16–17, 2015.
- [11] Z. Liouane, T. Lemlouma, P. Roose, F. Weis, and H. Messaoud, "An improved extreme learning machine model for the prediction of human scenarios in smart homes," *Applied Intelligence*, pp. 1–14, 2017.
- [12] X. Zhu, Z. Ni, M. Cheng, F. Jin, J. Li, and G. Weckman, "Selective ensemble based on extreme learning machine and improved discrete artificial fish swarm algorithm for haze forecast," *Applied Intelligence*, pp. 1–19, 2017.
- [13] M. Fernandez-Delgado, E. Cernadas, S. Barro, J. Ribeiro, and J. Neves, "Direct kernel perceptron (dkp): Ultra-fast kernel elm-based classification with non-iterative closed-form weight calculation," *Neural Networks*, vol. 50, pp. 60 – 71, 2014.
- [14] Y. Miche, A. Sorjamaa, P. Bas, O. Simula, C. Jutten, and A. Lendasse, "OP-ELM: Optimally pruned extreme learning machine," *Neural Networks, IEEE Transactions on*, vol. 21, no. 1, pp. 158–162, 2010.
- [15] Y. Yang, Y. Wang, and X. Yuan, "Bidirectional extreme learning machine for regression problem and its learning effectiveness," *Neural Networks and Learning Systems, IEEE Transactions on*, vol. 23, no. 9, pp. 1498–1505, 2012.



(a) Testing performance of the proposed algorithm on Euro/USD exchange rate (b) Error of the proposed algorithm on Euro/USD exchange rate



(c) Histogram of the error of the algorithm on Euro/USD exchange rate (d) Estimated values versus real values on Euro/USD exchange rate

Fig. 4: Prediction of Euro/USD exchange rate

- [16] N. Jafarpisheh and M. Teshnehlab, "Cancers classification based on deep neural networks and emotional learning approach," *IET Systems Biology*, 2018.
- [17] C.-F. Juang and Y.-W. Tsao, "A self-evolving interval type-2 fuzzy neural network with online structure and parameter learning," *Fuzzy Systems, IEEE Transactions on*, vol. 16, no. 6, pp. 1411–1424, Dec 2008.
- [18] Q. Liang and J. Mendel, "Interval type-2 fuzzy logic systems: theory and design," *Fuzzy Systems, IEEE Transactions on*, vol. 8, no. 5, pp. 535–550, 2000.
- [19] D. Wu, "Approaches for reducing the computational cost of interval type-2 fuzzy logic systems: overview and comparisons," *Fuzzy Systems, IEEE Transactions on*, vol. 21, no. 1, pp. 80–99, 2013.
- [20] D. Wu and J. M. Mendel, "Enhanced karnik–mendel algorithms," *Fuzzy Systems, IEEE Transactions on*, vol. 17, no. 4, pp. 923–934, 2009.
- [21] N. N. Karnik and J. M. Mendel, "Centroid of a type-2 fuzzy set," *Information Sciences*, vol. 132, no. 1–4, pp. 195 – 220, 2001.
- [22] J. M. Mendel, *Uncertain Rule-Based Fuzzy Logic Systems: Introduction and New Directions*. Prentice Hall, Englewood Cliffs, NJ, 2001.
- [23] Z. Deng, K.-S. Choi, L. Cao, and S. Wang, "T2fela: Type-2 fuzzy extreme learning algorithm for fast training of interval type-2 tsk fuzzy logic system," *Neural Networks and Learning Systems, IEEE Transactions on*, vol. 25, no. 4, pp. 664–676, April 2014.
- [24] D. Serre, *Matrices: Theory and Application*. New York, USA: Springer-Verlag, 2002.
- [25] C. Rao and S. Mitra, *Generalized Inverse of Matrices and its Applications*. Wiley, New York, , 1971.
- [26] P. Mandal, T. Senjyu, N. Urasaki, T. Funabashi, and A. Srivastava, "A novel approach to forecast electricity price for pjm using neural network and similar days method," *Power Systems, IEEE Transactions on*, vol. 22, no. 4, pp. 2058–2065, Nov 2007.