International Journal of Machine Learning and Cybernetics (2021) 12:2135–2150
https://doi.org/10.1007/s13042-021-01298-5

ORIGINAL ARTICLE

A comparative experimental evaluation on performance of type-1 and interval type-2 Takagi-Sugeno fuzzy models

Kehua Yuan[1] · Wentao Li[2] · Weihua Xu[2] · Tao Zhan[1] · Libo Zhang[2] · Shuai Liu[3]

Received: 28 June 2020 / Accepted: 8 March 2021 / Published online: 3 May 2021

Abstract

In the literature, there have been numerous studies demonstrating experimentally that type-2 fuzzy models outperform their type-1 counterparts. Although the advantages of these models seem to be well justified, the quantification of the improvements is not carefully evaluated and critically assessed in the existing studies. A thorough multi-objective experimental numeric evaluation of benefits of type-2 fuzzy models is still lacking. In this study, a numeric evaluation of the performance of type-1 and type-2 fuzzy models is carried out in terms of the criteria of accuracy and computing overhead, which leads to a thorough analysis of existing trade-offs between these two performance indexes. In the proposed numeric evaluation, type-2 fuzzy models are evaluated against their associated type-1 counterparts (the type-2 associated type-1 models sharing similar structure and the same development method). Three architectures of fuzzy models are involved in the comparative studies presented here: (1) fuzzy clustering method-based Takagi-Sugeno (TS) fuzzy models (Fuzzy C-Means based type-1, Fuzzy C-Means based interval type-2); (2) static TS-based fuzzy models (static type-1, A2C0, A2C1, EKFT2 and their associated type-1 models) and (3) evolving TS fuzzy models (SEIT2 and its associated type-1 counterpart, SCIT2 and its associated type-1 model). The experiments are carried out by involving 15 publicly available datasets. The accuracy of these two types of fuzzy models is assessed vis-a-vis their development time. Testing is involved to evaluate whether there are statistically significant differences between the performance of the type-2 and type-1 fuzzy models.

Keywords Accuracy · Design complexity · Experimental evaluation · Type-1 TS fuzzy model · Type-2 TS fuzzy model

1 Introduction

Fuzzy models have been recognized as important and effective architectures in handling uncertainty in system modeling and as such have been used in many areas, including control [6, 7], pattern recognition [8, 9, 48] and many others [16–20, 27, 33, 50–54]. In recent years, fuzzy models involved type-2 fuzzy sets started to become more visible showing their advantages over type-1 fuzzy models [21–23]. Based on this observation, a number of studies have been conveyed and the advantages of type-2 fuzzy models when compared with type-1 fuzzy models have been reported.

As the inherent extension of type-1 fuzzy sets, type-2 fuzzy sets and models need more parameters in their representation. At the same time, an additional type-reduction stage before defuzzification (decoding) becomes necessary to transform type-2 fuzzy sets into its numeric representative. Without any doubt, these aspects make the type-2 fuzzy inference systems more complex than the associated type-1 systems. To lower the associated complexity and reduce the

✉ Wentao Li
drliwentao@gmail.com

Kehua Yuan
yuan945527632@163.com

Weihua Xu
chxuwh@gmail.com

Tao Zhan
zhantao@swu.edu.cn

Libo Zhang
lbzhang@swu.edu.cn

Shuai Liu
liushuai870327@163.com

1 School of Mathematics and Statistics, Southwest University, Chongqing 400715, People's Republic of China

2 College of Artificial Intelligence, Southwest University, Chongqing 400715, People's Republic of China

3 School of Automation, Beijing Institute of Technology, Beijing 100081, People's Republic of China

related computing overhead, a lot of efforts of exploring efficient type-reduction methods and designing simplified architecture of type-2 fuzzy models (usage of a few type-2 fuzzy sets for the most critical input regions and type-1 fuzzy sets for the others) have been reported in the references

[24–26, 28–32, 34, 36–43]. While the computational overhead is essential, only a few of the existing studies elaborate on the advantages of type-2 fuzzy models when being compared with type-1 fuzzy systems being weighed against the associated computational cost. It is still unclear and uncarefully quantified whether the gain (advantages conveyed by the type-2 fuzzy systems) are beneficial in terms of costs (development time). Another important issue, which is missing in the existing studies, is a thorough comparative analysis. Although type-1 and type-2 fuzzy models are shown to be universal approximators [44–47], the performance of the systems is still impacted by numerous factors, for example, initial values of the parameters and training methods. Statistical testing is important to evaluate whether there are statistically significant improvements delivered by type-2 fuzzy models.

The ultimate objective of this study is to carry out a thorough comparative quantitative analysis of type-2 and type-1 fuzzy models by involving the criteria of accuracy and computing overhead. A comparative statistical analysis supports a framework in which comparative studies are completed. To make the comparison fair, the following development framework along with optimization setting are established: (1) type-1 and type-2 fuzzy models used in the comparison are developed using the same kinds of methods: if type-1 fuzzy models are developed based on Fuzzy C-Means (FCM) and the least square error method (LSE), then the interval type-2 fuzzy models are constructed based on FCM or interval type-2 FCM clustering method and the LSE, etc.; (2) fuzzy sets used in type-1 and type-2 fuzzy models are described by the same form of membership functions, say if type-2 is a Gaussian membership function with uncertain mean, then the corresponding type-1 fuzzy set is described in terms of Gaussian membership function; (3) as type-2 fuzzy models are impacted (especially in terms of the ensuing computational efficiency) by the type-reduction method, four type-reduction methods (Karnik–Mendel iterative procedure (KM) [49], q factor method (QF) [41], Nie–Tan method (NT) [35] and center of set type reducer without sorting requirement (COSTRWSR) [11]) are used in the design of interval type-2 fuzzy models (except for SEIT2, SCIT2 and EKFT2).

In this study, the structures of interval type-2 fuzzy models based on FCM and the interval type-2 FCM clustering algorithm with four kinds type-reduction and defuzzification methods (KM, NT, QF and COSTRWSR) are formulated. The accuracy of interval type-2 fuzzy models are compared with that of type-1 fuzzy models against the associated computing overhead. The nonparametric test is used to tell whether there are significant differences between the performances of interval type-2 and type-1 fuzzy models. We consider multiple input-single output fuzzy models where $x \in R^n$ and $y \in R$. The input-output data ($\mathbf{x}_k$, target$_k$) ($\mathbf{x}_k \in R^n$, target$_k \in R$, $k = 1, 2, \ldots, N$) are split into the training and testing set R_{tr} and R_{te}. In all the models, the minimized performance index Q expressing their accuracy comes as a sum of squared errors, namely

$$Q = \frac{\sum_{k=1}^{M}(FM(\mathbf{x}_k) - \text{target}_k)^2}{M}, \tag{1}$$

where FM denotes the fuzzy model under discussion, $M = N_{tr}$ or N_{te}, $N_{tr} = \text{card}(X_{tr})$ and $N_{te} = \text{card}(X_{te})$.

The paper is organized as follows. In Sect. 2, the structures of type-1 and type-2 fuzzy models are discussed. In Sect. 3, experimental studies are given. Conclusions are presented in Sect. 4.

2 Structure of the fuzzy models

In this section, we briefly recall the generally encountered structure and the design of the fuzzy models, both type-1 and type-2 being used in this comparative study.

2.1 Clustering method based fuzzy models

1. FCM based type-1 structure The Takagi-Sugeno (TS) fuzzy model is the commonly used rule-based model. Considering c rules with fuzzy sets in the condition part built around prototypes $\mathbf{v}_1, \mathbf{v}_2, \ldots, \mathbf{v}_c$, the rules assume the following form
   $$-\text{if } \mathbf{v} \text{ is } A^i, \text{then } y^i = a_0^i + a_1^i x_1 + \cdots + a_n^i x_n, \tag{2}$$

where $A^i (i = 1, 2, \ldots, c)$ is a condition part (fuzzy set) defined in the input space described by the following membership function (as a matter of fact, the formula describing this fuzzy set comes as a direct consequence of the use of the FCM algorithm) and $\mathbf{x} = [x_1, x_2, x_n]^T$,

$$\mu_{A^i}(\mathbf{x}) = \frac{1}{\sum_{j=1}^{c}\left(\frac{||\mathbf{x}-\mathbf{v}_i||}{||\mathbf{x}-\mathbf{v}_j||}\right)^{\frac{2}{m-1}}}. \tag{3}$$

The conclusion part is a local linear function described by a vector of parameters $\mathbf{a}_i = [a_0^i, a_1^i, \ldots, a_n^i]$, $\mathbf{a} = [\mathbf{a}_1^T, \mathbf{a}_2^T, \ldots, \mathbf{a}_c^T]$ ($\mathbf{a}_i^T$ is the transpose of $\mathbf{a}_i$) and $\mathbf{z}_i = \mu_{A^i}(\mathbf{x})[1, \mathbf{x}^T]^T, i = 1, 2, \ldots, c$. In the virtue of form of the performance index, the optimal solution to the estimation of the parameters of the conclusion parts is determined using the well-known expression

$$\mathbf{a}_{opt} = (\mathbf{F}_{tr}^T\mathbf{F}_{tr})^{-1}\mathbf{F}_{tr}^T\mathbf{y}_{tr}, \quad (4)$$

where $\mathbf{F}_{tr} = \begin{pmatrix} \mathbf{z}_1^T(\mathbf{x}_1) & \mathbf{z}_2^T(\mathbf{x}_1) & \dots & \mathbf{z}_c^T(\mathbf{x}_1) \\ \mathbf{z}_1^T(\mathbf{x}_2) & \mathbf{z}_2^T(\mathbf{x}_2) & \dots & \mathbf{z}_c^T(\mathbf{x}_2) \\ \dots & \dots & \dots & \dots \\ \mathbf{z}_1^T(\mathbf{x}_{N_{tr}}) & \mathbf{z}_2^T(\mathbf{x}_{N_{tr}}) & \dots & \mathbf{z}_c^T(\mathbf{x}_{N_{tr}}) \end{pmatrix}, \quad \mathbf{x}_i \in \mathbf{x}_{tr}$,
$\mathbf{y}_{tr} = [y_1, y_2, \dots, y_{N_{tr}}]^T, i = 1, 2, \dots, N_{tr}$.

This generic fuzzy model is denoted here as FCMT1-1. For further comparative analysis, we consider more advanced architectures in which we admit the local model to be quadratic with respect to the input variables assuming the form

$$\text{– if } \mathbf{x} \text{ is } A^i \text{ then } y^i = a_0^i + a_1^i x_1 + a_2^i x_2 + \dots + a_n^i x_n + a_{n+1}^i x_1^2 + \dots + a_{2n}^i x_n^2 \quad (5)$$

where $i = 1, 2, \dots, c$.

This model is denoted as FCMT1-2. Note that the FCMT1-2 is a linear model with respect to the parameters standing in the conclusion part (albeit the dimensionality of the parameter space has been increased). From the design perspective, this augmented model is developed in the same manner as FCMT1-1.

2. Clustering method based interval type-2 fuzzy model

   To compare the performance of the type-1 and type-2 fuzzy models, we introduce the clustering method based interval type-2 fuzzy models. Two forms of methods are introduced here: a) interval type-2 FCM [13] based interval type-2 fuzzy model (IT2FCMIT2-1); and b) FCM based interval type-2 fuzzy model (FCMIT2-1). For these two models, only their antecedent parts are interval type-2 fuzzy sets. Their consequent parts are composed of linear functions. They share the same structure as described below

$$\text{– if } \mathbf{x} \text{ is } \tilde{A}^i \text{ then } y^i = a_0^i + a_1^i x_1 + a_2^i x_2 + \dots + a_n^i x_n \quad (6)$$

where $\tilde{A}^i (i = 1, 2, \dots, c)$ is a condition condition part (type-2 fuzzy set) defined in the input space described by the membership functions (7) and (8) (for interval type-2 FCM based interval type-2 fuzzy models) or (9) and (10) (for FCM based interval type-2 fuzzy models) and $\mathbf{x} = [x_1, x_2, \dots, x_n]^T$,

$$\overline{\mu}_{\tilde{A}^i}(\mathbf{x}) = \max\left(\frac{1}{\sum_{j=1}^c \left(\frac{||\mathbf{x}-\mathbf{v}_i||}{||\mathbf{x}-\mathbf{v}_j||}\right)^{\frac{2}{m_1-1}}}, \frac{1}{\sum_{j=1}^c \left(\frac{||\mathbf{x}-\mathbf{v}_i||}{||\mathbf{x}-\mathbf{v}_j||}\right)^{\frac{2}{m_2-1}}} \right), \quad (7)$$

$$\underline{\mu}_{\tilde{A}^i}(\mathbf{x}) = \min\left(\frac{1}{\sum_{j=1}^c \left(\frac{||\mathbf{x}-\mathbf{v}_i||}{||\mathbf{x}-\mathbf{v}_j||}\right)^{\frac{2}{m_1-1}}}, \frac{1}{\sum_{j=1}^c \left(\frac{||\mathbf{x}-\mathbf{v}_i||}{||\mathbf{x}-\mathbf{v}_j||}\right)^{\frac{2}{m_2-1}}} \right), \quad (8)$$

in which $\mathbf{v}_i$ is the prototype obtained with the interval type-2 FCM ($i = 1, 2, \dots, c$) and m_1 and m_2 are two values of the fuzzifiers.

For the FCM-based model, two groups of prototypes $(\mathbf{v}_1^1, \mathbf{v}_2^1, \dots, \mathbf{v}_c^1)$ and $(\mathbf{v}_1^2, \mathbf{v}_2^2, \dots, \mathbf{v}_c^2)$ are obtained when running the method using two values of the fuzzifier m_1 and m_2. The upper and lower membership grades are obtained as follows:

$$\overline{\mu}_{\tilde{A}^i}(\mathbf{x}) = \max\left(\frac{1}{\sum_{j=1}^c \left(\frac{||\mathbf{x}-\mathbf{v}_i^1||}{||\mathbf{x}-\mathbf{v}_j^1||}\right)^{\frac{2}{m_1-1}}}, \frac{1}{\sum_{j=1}^c \left(\frac{||\mathbf{x}-\mathbf{v}_i^2||}{||\mathbf{x}-\mathbf{v}_j^2||}\right)^{\frac{2}{m_2-1}}} \right), \quad (9)$$

$$\underline{\mu}_{\tilde{A}^i}(\mathbf{x}) = \min\left(\frac{1}{\sum_{j=1}^c \left(\frac{||\mathbf{x}-\mathbf{v}_i^1||}{||\mathbf{x}-\mathbf{v}_j^1||}\right)^{\frac{2}{m_1-1}}}, \frac{1}{\sum_{j=1}^c \left(\frac{||\mathbf{x}-\mathbf{v}_i^2||}{||\mathbf{x}-\mathbf{v}_j^2||}\right)^{\frac{2}{m_2-1}}} \right), \quad (10)$$

where $(\mathbf{v}_1^1, \mathbf{v}_2^1, \dots, \mathbf{v}_c^1)$ and $(\mathbf{v}_1^2, \mathbf{v}_2^2, \dots, \mathbf{v}_c^2)$ are the prototypes given by the FCM clustering algorithm associated wuth the fuzzification coefficients m_1 and m_2, respectively. And $\mathbf{v}_k^2 = \arg\min_{1 \le j \le c}\left(||\mathbf{v}_i^1 - \mathbf{v}_j^2||\right), i = 1, 2, \dots, c$. The closer the values of m_1 and m_2, the closer the values of $\underline{\mu}_{\tilde{A}^i}(\mathbf{x})$ and $\overline{\mu}_{\tilde{A}^i}(\mathbf{x})$. To make the interval type-2 fuzzy sets reflect a level of granularity, a higher difference between m1 and m2 is preferred. For both of the two interval type-2 fuzzy models, four type-reduction and defuzzification methods (KM, NT, COSTRWSR and QF) are considered to produce a numeric output of the model.

For KM and COSTRWSR, the left and right ends (y_l and y_r) of the type-reduced interval are given. For KM, it goes on this way (let us name it with the computation of y_l): step (1) sort the data in the primary domain in ascending order and match the associated weights; step (2) initialize the weights and defuzzified value; step (3) find the switch points; step (4) according to the switch point, update the weights and calculate the associated defuzzified value; step (5) check the termination condition and perform correspondingly. Hence the amount of computation for each step is: step (1) $c(c$-1); step (2) 5c-2; step (3) c-1; step (4) 4c-1; step (5) 2. Hence in each iteration the total computational amount for KM is $2c^2$+18c-2. For COSTRWSR, it consists of the following steps (also name the computation of y_l): step (1) check the weights and determine whether it is necessary to perform the followed steps; step (2) initialize; step (3) calculate the common terms; step (4) conditionally update λ_l^i and check the stop condition to activate the associated steps. The computational amount of COSTRWSR for each step is: step (1) c; step (2) c; step (3) 10c-3; step (4) 16c. Considering one more common term and the difference between the upper and lower weights, the total computational amount for COSTRWSR is 58c-5 in each iteration. That is to say, if we assume these two methods could convergence in less than c iterations, the computational complexity of COSTRWSR and KM are $O(c^2)$ and $O(c)$, respectively. While for NT method and QF method, their associated computational amount are $6c-3$ and $6c$, respectively. For KM and COSTRWSR, the gradient descent algorithm is used to optimize the parameters in the consequent part. For the detailed algorithm of KM refer to [1]. Here we concentrate on the calculation of the other three kinds of methods.

Detailed algorithm for COSTRWSR

COSTRWSR is introduced in [11] along with the following formulae

$$y_l = \frac{\sum_{i=1}^{c} \lambda_l^i \overline{\mu}_{\tilde{A}^i}(\mathbf{x}) y^i + \sum_{i=1}^{c} (1-\lambda_l^i) \underline{\mu}_{\tilde{A}^i}(\mathbf{x}) y^i}{\sum_{i=1}^{c} \lambda_l^i \overline{\mu}_{\tilde{A}^i}(\mathbf{x}) + \sum_{i=1}^{c} (1-\lambda_l^i) \underline{\mu}_{\tilde{A}^i}(\mathbf{x})}, \tag{11}$$

$$y_r = \frac{\sum_{i=1}^{c} \lambda_r^i \overline{\mu}_{\tilde{A}^i}(\mathbf{x}) y^i + \sum_{i=1}^{c} (1-\lambda_r^i) \underline{\mu}_{\tilde{A}^i}(\mathbf{x}) y^i}{\sum_{i=1}^{c} \lambda_r^i \overline{\mu}_{\tilde{A}^i}(\mathbf{x}) + \sum_{i=1}^{c} (1-\lambda_r^i) \underline{\mu}_{\tilde{A}^i}(\mathbf{x})}, \tag{12}$$

in which $\lambda_l^i, \lambda_r^i = 0, 1$ and $i = 1, 2, \dots, c$.

$$y = (y_l + y_r)/2. \tag{13}$$

Make

$$\tilde{\mathbf{z}}_i^{ctr}(\mathbf{x}) = 0.5\left(\frac{\lambda_l^i \overline{\mu}_{\tilde{A}^i}(\mathbf{x}) + (1-\lambda_l^i)\underline{\mu}_{\tilde{A}^i}(\mathbf{x})}{\sum_{i=1}^{c} \lambda_l^i \overline{\mu}_{\tilde{A}^i}(\mathbf{x}) + \sum_{i=1}^{c} (1-\lambda_l^i) \underline{\mu}_{\tilde{A}^i}(\mathbf{x})} + \frac{\lambda_r^i \overline{\mu}_{\tilde{A}^i}(\mathbf{x}) + (1-\lambda_r^i)\underline{\mu}_{\tilde{A}^i}(\mathbf{x})}{\sum_{i=1}^{c} \lambda_r^i \overline{\mu}_{\tilde{A}^i}(\mathbf{x}) + \sum_{i=1}^{c} (1-\lambda_r^i) \underline{\mu}_{\tilde{A}^i}(\mathbf{x})} \right) [1, \mathbf{x}^T]^T, \ (i = 1, 2, \dots, c). \tag{14}$$

Then using the gradient descent algorithm, for the $(k+1)$-th iteration, we have

$$\mathbf{a}^{(k+1)} = \mathbf{a}^{(k)} - \frac{2\left((\tilde{\mathbf{F}}_{tr}^{ctr^{(k)}})^T \tilde{\mathbf{F}}_{tr}^{ctr^{(k)}} \mathbf{a}^{(k)} - (\tilde{\mathbf{F}}_{tr}^{ctr^{(k)}})^T \mathbf{y}_{(tr)} \right)}{N_{tr}}, \tag{15}$$

in which

$$\tilde{\mathbf{F}}_{tr}^{ctr^{(k)}} = \begin{bmatrix} \left(\tilde{\mathbf{z}}_1^{ctr^{(k)}}(\mathbf{x}_1)\right)^T & \left(\tilde{\mathbf{z}}_2^{ctr^{(k)}}(\mathbf{x}_1)\right)^T & \dots & \left(\tilde{\mathbf{z}}_c^{ctr^{(k)}}(\mathbf{x}_1)\right)^T \\ \left(\tilde{\mathbf{z}}_1^{ctr^{(k)}}(\mathbf{x}_2)\right)^T & \left(\tilde{\mathbf{z}}_2^{ctr^{(k)}}(\mathbf{x}_2)\right)^T & \dots & \left(\tilde{\mathbf{z}}_c^{ctr^{(k)}}(\mathbf{x}_2)\right)^T \\ \dots & \dots & \dots & \dots \\ \left(\tilde{\mathbf{z}}_1^{ctr^{(k)}}(\mathbf{x}_{N_{tr}})\right)^T & \left(\tilde{\mathbf{z}}_2^{ctr^{(k)}}(\mathbf{x}_{N_{tr}})\right)^T & \dots & \left(\tilde{\mathbf{z}}_c^{ctr^{(k)}}(\mathbf{x}_{N_{tr}})\right)^T \end{bmatrix}, \tag{16}$$

and

$$\tilde{\mathbf{z}}_i^{ctr^{(k)}}(\mathbf{x}_j) = 0.5\left(\frac{\lambda_l^{i^{(k)}} \overline{\mu}_{\tilde{A}^i}(\mathbf{x}_j) + (1-\lambda_l^{i^{(k)}})\underline{\mu}_{\tilde{A}^i}(\mathbf{x}_j)}{\sum_{i=1}^{c} \lambda_l^{i^{(k)}} \overline{\mu}_{\tilde{A}^i}(\mathbf{x}_j) + \sum_{i=1}^{c} (1-\lambda_l^{i^{(k)}}) \underline{\mu}_{\tilde{A}^i}(\mathbf{x}_j)} \right)[1, \mathbf{x}_j^T]^T + \left(\frac{\lambda_r^{i^{(k)}} \overline{\mu}_{\tilde{A}^i}(\mathbf{x}_j) + (1-\lambda_r^{i^{(k)}})\underline{\mu}_{\tilde{A}^i}(\mathbf{x}_j)}{\sum_{i=1}^{c} \lambda_r^{i^{(k)}} \overline{\mu}_{\tilde{A}^i}(\mathbf{x}_j) + \sum_{i=1}^{c} (1-\lambda_r^{i^{(k)}}) \underline{\mu}_{\tilde{A}^i}(\mathbf{x}_j)} \right)[1, \mathbf{x}_j^T]^T, \tag{17}$$

where $\mathbf{x}_j \in \mathbf{x}_{tr}, \mathbf{y}_{tr} = [y_1, y_2, \dots, y_{N_{tr}}], i = 1, 2, \dots, c; j = 1, 2, \dots, N_{tr}$, and $k = 1, 2, \dots,$ Itera (maximum iteration).

Detailed algorithm for QF

According to the definition of q factor method in [41] denote

$$\tilde{\mathbf{z}}_i^q(\mathbf{x}) = \left(\frac{q\underline{\mu}_{\tilde{A}^i}(\mathbf{x})}{\sum_{i=1}^{c} \underline{\mu}_{\tilde{A}^i}(\mathbf{x})} + \frac{(1-q)\overline{\mu}_{\tilde{A}^i}(\mathbf{x})}{\sum_{i=1}^{c} \overline{\mu}_{\tilde{A}^i}(\mathbf{x})} \right)[1, \mathbf{x}^T]^T, \ (i = 1, 2, \dots, c). \tag{18}$$

Making use of the gradient descent algorithm and LSE, for the $(k+1)-$th iteration, we have

$$q^{(k+1)} = q^{(k)} - \frac{d\left(\mathbf{F}_{tr}^{q^{(k)}} \mathbf{a}^{(k)} - \mathbf{y}_{tr}\right)^T \left(\mathbf{F}_{tr}^{q^{(k)}} \mathbf{a}^{(k)} - \mathbf{y}_{tr}\right)}{dq} \tag{19}$$

in which

$$\tilde{\mathbf{F}}_{tr}^{q^{(k)}} = \begin{bmatrix} \left(\tilde{\mathbf{z}}_1^{q^{(k)}}(\mathbf{x}_1)\right)^T & \left(\tilde{\mathbf{z}}_2^{q^{(k)}}(\mathbf{x}_1)\right)^T & \dots & \left(\tilde{\mathbf{z}}_c^{q^{(k)}}(\mathbf{x}_1)\right)^T \\ \left(\tilde{\mathbf{z}}_1^{q^{(k)}}(\mathbf{x}_2)\right)^T & \left(\tilde{\mathbf{z}}_2^{q^{(k)}}(\mathbf{x}_2)\right)^T & \dots & \left(\tilde{\mathbf{z}}_c^{q^{(k)}}(\mathbf{x}_2)\right)^T \\ \dots & \dots & \dots & \dots \\ \left(\tilde{\mathbf{z}}_1^{q^{(k)}}(\mathbf{x}_{N_{tr}})\right)^T & \left(\tilde{\mathbf{z}}_2^{q^{(k)}}(\mathbf{x}_{N_{tr}})\right)^T & \dots & \left(\tilde{\mathbf{z}}_c^{q^{(k)}}(\mathbf{x}_{N_{tr}})\right)^T \end{bmatrix},$$

where

$$\tilde{\mathbf{z}}_i^{q^{(k)}}(\mathbf{x}_j) = \left(\frac{q^{(k)}\underline{\mu}_{\tilde{A}^i}(\mathbf{x}_j)}{\sum_{i=1}^{c} \underline{\mu}_{\tilde{A}^i}(\mathbf{x}_j)} + \frac{(1-q^{(k)})\overline{\mu}_{\tilde{A}^i}(\mathbf{x}_j)}{\sum_{i=1}^{c} \overline{\mu}_{\tilde{A}^i}(\mathbf{x}_j)} \right)[1, \mathbf{x}_j^T]^T,$$

$$\mathbf{a}^{(k)} = \left(\left(\mathbf{F}_{tr}^{q^{(k)}}\right)^T \mathbf{F}_{tr}^{q^{(k)}} \right)^{-1} \left(\mathbf{F}_{tr}^{q^{(k)}}\right)^T \mathbf{y}_{tr},$$

and $\mathbf{x}_j \in \mathbf{x}_{tr}, \mathbf{y}_{tr} = [y_1, y_2, \dots, y_{N_{tr}}], i = 1, 2, \dots, c; j = 1, 2, \dots, N_{tr}$, and $k = 1, 2, \dots,$ Itera (maximum iteration).

Detailed algorithm for NT

According to the nature of the NT method [35], make

$$\tilde{\mathbf{z}}_i^{nt}(\mathbf{x}) = \left(\frac{\underline{\mu}_{\tilde{A}^i}(\mathbf{x}) + \overline{\mu}_{\tilde{A}^i}(\mathbf{x})}{\sum_{i=1}^{c} \underline{\mu}_{\tilde{A}^i}(\mathbf{x}) + \sum_{i=1}^{c} \overline{\mu}_{\tilde{A}^i}(\mathbf{x})} \right)[1, \mathbf{x}^T]^T, \ (i = 1, 2, \dots, c). \tag{20}$$

Then using LSE, we obtain the closed-form solution:

$$\mathbf{a}_{opt} = \left((\tilde{\mathbf{F}}_{tr}^{nt})^T \tilde{\mathbf{F}}_{tr}^{nt} \right)^{-1} (\tilde{\mathbf{F}}_{tr}^{nt})^T \mathbf{y}_{tr}, \tag{21}$$

where

$$\tilde{\mathbf{F}}^{nt}_{tr} = \begin{pmatrix} (\tilde{\mathbf{z}}^{nt}_1(\mathbf{x}_1))^T & (\tilde{\mathbf{z}}^{nt}_2(\mathbf{x}_1))^T & \dots & (\tilde{\mathbf{z}}^{nt}_c(\mathbf{x}_1))^T \\ (\tilde{\mathbf{z}}^{nt}_1(\mathbf{x}_2))^T & (\tilde{\mathbf{z}}^{nt}_2(\mathbf{x}_2))^T & \dots & (\tilde{\mathbf{z}}^{nt}_c(\mathbf{x}_2))^T \\ \dots & \dots & \dots & \dots \\ (\tilde{\mathbf{z}}^{nt}_1(\mathbf{x}_{N_{tr}}))^T & (\tilde{\mathbf{z}}^{nt}_2(\mathbf{x}_{N_{tr}}))^T & \dots & (\tilde{\mathbf{z}}^{nt}_c(\mathbf{x}_{N_{tr}}))^T \end{pmatrix},$$

$\mathbf{x}_j \in \mathbf{x}_{tr}, \mathbf{y}_{tr} = [y_1, y_2, \dots, y_{N_{tr}}]^T, j = 1, 2, \dots, N_{tr}$.

We also consider the model of which the conclusion part is made up of the quadratic variable of the inputs as following

$$- \text{ if } \mathbf{x} \text{ is } \tilde{A}^i \text{ then } y^i = a^i_0 + a^i_1 x_1 + a^i_2 x_2 + \dots + a^i_n x_n + a^i_{n+1} x_1^2 + \dots + a^i_{2n} x_n^2, \tag{22}$$

where $i = 1, 2, \dots, c$.

We mark interval type-2 fuzzy models as IT2FCMIT2-2 and FCMIT2-2 of which the prototypes are given by the interval type-2 FCM, and FCM, respectively. The crisp outputs are given by those four kinds of type-reduction and defuzzification methods (KM, QF, NT and COSTRWSR), too. For the training of this model, it is similar with that of FCMT2-1, so the detailed discussion is omitted here.

2.2 Selected type-2 fuzzy models and the induced type-1 fuzzy models

In what follows, we summarize several selected type-2 fuzzy models used in the comparative experiments. We focus here on interval-valued fuzzy models as being commonly used; they are generalized in comparison with type-1 fuzzy models but do not carry an overall computing burden associated with full-fledged type-2 fuzzy models.

1. Self-evolving Interval Type-2 Fuzzy Neural Network (SEIT2)

   This SEIT2 fuzzy neural network proposed in [1] is an interval type-2 T-S fuzzy model coming in the following form

$$\begin{aligned} &- \text{ if } x_1 \text{ is } \tilde{A}^i_1 \text{ and } x_2 \text{ is } \tilde{A}^i_2 \text{ and } \dots \text{ and } x_n \text{ is } \tilde{A}^i_n \\ &\quad \text{then } y^i = \tilde{a}^i_0 + \tilde{a}^i_1 x_1 + \tilde{a}^i_2 x_2 + \dots + \tilde{a}^i_n x_n \end{aligned} \tag{23}$$

   where $\tilde{A}^i_j$ is the interval type-2 fuzzy set with Gaussian membership function expressed as (22)-(24) and $\tilde{a}^i_0 = [b^i_0 - s^i_0, b^i_0 + s^i_0]$, $\tilde{a}^i_j = [b^i_j - s^i_j, b^i_j + s^i_j]$, $b^i_0, s^i_0, b^i_j, s^i_j \in \mathbf{R}$, $i = 1, 2, \dots, c$ and $j = 1, 2, \dots, n$.

$$\mu_{\tilde{A}^i_j} = exp\left(-(x_j - m^i_j)^2 / (2(\sigma^i_j)^2)\right), \tag{24}$$

   where $m^i_j \in [m^i_{j1}, m^i_{j2}]$, $m^i_j, m^i_{j1}, m^i_{j2} \in \mathbf{R}$, $i = 1, 2, \dots, c$ and $j = 1, 2, \dots, n$.

In more detail, the upper $\overline{\mu}_{\tilde{A}^i_j}(x_j)$ and lower $\underline{\mu}_{\tilde{A}^i_j}(x_j)$ membership function are expressed as follows.

$$\overline{\mu}_{\tilde{A}^i_j}(x_j) = \begin{cases} exp\left(-\frac{(x_j - m^i_{j1})^2}{2(\sigma^i_j)^2}\right), & x_j \le m^i_{j1}; \\ 1, & m^i_{j1} < x_j < m^i_{j2}; \\ exp\left(-\frac{(x_j - m^i_{j2})^2}{2(\sigma^i_j)^2}\right), & x_j \ge m^i_{j2}. \end{cases} \tag{25}$$

$$\underline{\mu}_{\tilde{A}^i_j}(x_j) = \begin{cases} exp\left(-\frac{(x_j - m^i_{j1})^2}{2(\sigma^i_j)^2}\right), & x_j \ge \frac{m^i_{j1} + m^i_{j2}}{2}; \\ exp\left(-\frac{(x_j - m^i_{j2})^2}{2(\sigma^i_j)^2}\right), & x_j < \frac{m^i_{j1} + m^i_{j2}}{2}. \end{cases} \tag{26}$$

This model involves two types of learning: the structure learning and the parameter learning. It is a self-evolving model meaning that initially there are no rules in the rule base of SEIT2. The rules are generated after receiving the training data, which is considered as the structure learning proceeds. When a rule (assume it is the i_0-th rule) is generated, the parameters (m^i_{j1}, m^i_{m2} and σ^i_j in the condition part, b^i_0, s^i_0, b^i_j and s^i_j in the conclusion part $i = 1, 2, \dots, i_0$ and $j = 1, 2, \dots, n$) will be tuned, which is the parameter learning part. The parameters of the condition part are tuned by the gradient descent algorithm. While for the conclusion part, the parameters are optimized by the Kalman filter. The KM method [2, 10] is used as the type-reduction method.

Induced SEIT1

The type-1 fuzzy model associated with SEIT2 could be formulated as follows, and here mark it as SEIT1,

$$\begin{aligned} &- \text{ if } x_1 \text{ is } A^i_1 \text{ and } x_2 \text{ is } A^i_2 \text{ and } \dots \text{ and } x_n \text{ is } A^i_n \\ &\quad \text{then } y^i = a^i_0 + a^i_1 x_1 + a^i_2 x_2 + \dots + a^i_n x_n \end{aligned} \tag{27}$$

where A^i_j is the type-1 fuzzy set with Gaussian membership function expressed as follows and $a^i_0, a^i_j \in \mathbf{R}$,

$$\mu_{A^i_j} = exp\left(-\frac{(x_j - m^i_j)^2}{2(\sigma^i_j)^2}\right) \tag{28}$$

in which $m^i_j, \sigma^i_j \in \mathbf{R}$, $i = 1, 2, \dots, c$ and $j = 1, 2, \dots, n$.

For SEIT1, it has the same number of rules with SEIT2. For the training of SEIT1, the parameters of the condition part (antecedent part) are trained with the gradient descent algorithm; while for those of the conclusion part (consequent part), they are trained with the Kalman filter algorithm.

2. Self-evolving Compensatory Interval Type-2 Fuzzy Neural Network (SCIT2)

This SCIT2 fuzzy neural network introduced in [3] is regarded as an interval type-2 fuzzy model whose rules come in the following form

$$\begin{aligned} &- \text{if } \left(x_1 \text{ is } \tilde{A}_1^i \text{ and } x_2 \text{ is } \tilde{A}_2^i \text{ and } \dots \text{ and } x_n \text{ is } \tilde{A}_n^i\right)^{1-\gamma^i+\gamma^i/n} \\ &\quad \text{then } y^i = \tilde{a}_0^i + \tilde{a}_1^i x_1 + \tilde{a}_2^i x_2 + \cdots + \tilde{a}_n^i x_n \end{aligned} \tag{29}$$

where $\tilde{A}_k^i$ is interval type-2 fuzzy set and the definition of its membership function refer to (22)–(24); $a_0^i, a_j^i \in textbfR$ are numeric values, $\gamma^i \in [0,1]$ is a compensation degree, $i = 1, 2, \dots, c$ and $j = 1, 2, \dots, n$.

This model exhibits a self-evolving architecture. Its design involves structural and parametric learning. The learning proceeds in a similar manner as in the SEIT2 architecture. In this model, the defuzzified (decoded) output is expressed in the following way

$$y = \frac{q\sum_{i=1}^{c} \underline{\varphi}^i y^i}{\sum_{i=1}^{c} \underline{\varphi}^i} + \frac{(1-q)\sum_{i=1}^{c} \overline{\varphi}^i y^i}{\sum_{i=1}^{c} \overline{\varphi}^i}, \tag{30}$$

where $\underline{\varphi}^i = \left(\prod_{j=1}^{n} \underline{\mu}_{\tilde{A}_j^i}\right)^{1-\gamma^i+\gamma^i/n}$, $\overline{\varphi}^i = \left(\prod_{j=1}^{n} \overline{\mu}_{\tilde{A}_j^i}\right)^{1-\gamma^i+\gamma^i/n}$, $i = 1, 2, \dots, c$ and q is an adjustable parameter [23].

Induced SCIT1

The corresponding compensatory type-1 fuzzy structure of SCIT2 come as follows and mark it as SCIT1

$$\begin{aligned} &- \text{if } x_1 \text{ is } A_1^i \text{ and } x_2 \text{ is } A_2^i \text{ and } \dots \text{ and } x_n \text{ is } A_n^i \\ &\quad \text{then } y^i = a_0^i + a_1^i x_1 + a_2^i x_2 + \cdots + a_n^i x_n \end{aligned} \tag{31}$$

where A_j^i is type-1 fuzzy set and the definition of its membership function refer to (27); $a_0^i, a_j^i \in \mathbf{R}$ are numeric values, $\gamma^i \in [0,1]$ is a compensation degree, $i = 1, 2, \dots, c$ and $j = 1, 2, \dots, n$.

It has the same number of rules as SCIT2. And also, the parameters in the antecedent part are trained with the gradient descent method, while for those in the consequent part, they are trained with the Kalman–Filter algorithm.

3. Extended Kalman filter based learning Algorithm for Type-2 Fuzzy Logic Systems (EKFT2)

The rules of this model are described in the following form [4],

$$\begin{aligned} &- \text{if } x_1 \text{ is } \tilde{A}_1^i \text{ and } x_2 \text{ is } \tilde{A}_2^i \text{ and } \dots \text{ and } x_n \text{ is } \tilde{A}_n^i \\ &\quad \text{then } y^i = \tilde{a}_0^i + \tilde{a}_1^i x_1 + \tilde{a}_2^i x_2 + \cdots + \tilde{a}_n^i x_n \end{aligned} \tag{32}$$

where $\tilde{A}_j^i$ is interval type-2 fuzzy set and its membership function is defined as (31)-(33); $a_0^i, a_j^i \in \mathbf{R}, i = 1, 2, \dots, c$ and $j = 1, 2, \dots, n$.

$$\mu_{\tilde{A}_j^i}(x_j) = \begin{cases} \left(1 - |(x_j - e_j^i)/d_j^i|^{h_j^i}\right)^{\frac{1}{h_j^i}}, & |x_j - e_j^i| \le |d_j^i|; \\ 0, & \text{otherwise.} \end{cases} \tag{33}$$

where $h_j^i \in [h_{j2}^i, h_{j1}^i]$, $h_j^i \in \mathbf{R}$, $h_{j1}^i > 1$, $0 < h_{j2}^i < 1$, $i = 1, 2, \dots, c$ and $j = 1, 2, \dots, n$.

The lower and upper membership functions come in the form,

$$\underline{\mu}_{\tilde{A}_j^i}(x_j) = \begin{cases} \left(1 - |(x_j - e_j^i)/d_j^i|^{h_{j2}^i}\right)^{\frac{1}{h_{j2}^i}}, & |x_j - e_j^i| \le |d_j^i|; \\ 0, & \text{otherwise.} \end{cases} \tag{34}$$

$$\overline{\mu}_{\tilde{A}_j^i}(x_j) = \begin{cases} \left(1 - |(x_j - e_j^i)/d_j^i|^{h_{j1}^i}\right)^{\frac{1}{h_{j1}^i}}, & |x_j - e_j^i| \le |d_j^i|; \\ 0, & \text{otherwise.} \end{cases} \tag{35}$$

In this model, the number of rules is specified in advance and the parameters ($e_j^i, d_j^i, h_{j1}^i, d_{j2}^i, a_0^i$ and a_j^i, $i = 1, 2, \dots, c; j = 1, 2, \dots, n$) are optimized by the extended Kalman filter. The defuzzified output is expressed as follows.

$$y = \sum_{i=1}^{c} (\underline{f}^i + \overline{f}^i) y^i \Big/ \sum_{i=1}^{c} (\underline{f}^i + \overline{f}^i), \tag{36}$$

where $\underline{f}^i = \prod_{j=1}^{n} \underline{\mu}_{\tilde{A}_j^i}$, $\overline{f}^i = \prod_{j=1}^{n} \overline{\mu}_{\tilde{A}_j^i}$, $i = 1, 2, \dots, c$. We denote it as the NT method [35].

Induced EKFT1

The type-1 fuzzy structure against with EKFT2 is defined as follows and denote it as EKFT1,

$$\begin{aligned} &- \text{if } x_1 \text{ is } A_1^i \text{ and } x_2 \text{ is } A_2^i \text{ and } \dots \text{ and } x_n \text{ is } A_n^i \\ &\quad \text{then } y^i = a_0^i + a_1^i x_1 + a_2^i x_2 + \cdots + a_n^i x_n \end{aligned} \tag{37}$$

where A_j^i is interval type-1 fuzzy set, $a_0^i, a_j^i \in \mathbf{R}$, $i = 1, 2, \dots, c$ and $j = 1, 2, \dots, n$. Its membership function is defined as follows.

$$\mu_{A_j^i}(x_j) = \begin{cases} 1 - |(x_j - e_j^i)/d_j^i|, & |x_j - e_j^i| \le |d_j^i|; \\ 0, & \text{otherwise,} \end{cases} \tag{38}$$

where $d_j^i, e_j^i \in \mathbf{R}$, $i = 1, 2, \dots, c$ and $j = 1, 2, \dots, n$.

It exhibits the same number of rules as EKFT2. All of the parameters (e_j^i, d_j^i, a_0^i and $a_j^i, i = 1, 2, \ldots, c$ and $j = 1, 2, \ldots, n$) are optimized with the use of the extended Kalman filter algorithm.

The following two forms of standard interval type-2 fuzzy models are introduced in [12]. They are constructed as following.

4. Antecedents Type-2 and Consequent Crisp Number Fuzzy Model (A2C0) and Antecedents Type-2 and Consequent Type-1 Fuzzy Model (A2C1)

A2C0

The interval type-2 fuzzy model of which the consequent part is linear function is described as follows.

$$\begin{aligned} &- \text{if } x_1 \text{ is } \tilde{A}_1^i \text{ and } x_2 \text{ is } \tilde{A}_2^i \text{ and } \ldots \text{ and } x_n \text{ is } \tilde{A}_n^i \\ &\quad \text{then } y^i = a_0^i + a_1^i x_1 + a_2^i x_2 + \cdots + a_n^i x_n \end{aligned} \tag{39}$$

where $\tilde{A}_j^i$ is interval type-2 fuzzy set and the definition of its membership function refer to (22)-(24); $a_0^i, a_j^i \in \mathbf{R}$, $i = 1, 2, \ldots, c$ and $j = 1, 2, \ldots, n$.

A2C1

The interval type-2 fuzzy structure where its condition part is type-1 fuzzy sets is defined as follows

$$\begin{aligned} &- \text{if } x_1 \text{ is } \tilde{A}_1^i \text{ and } x_2 \text{ is } \tilde{A}_2^i \text{ and } \ldots \text{ and } x_n \text{ is } \tilde{A}_n^i \\ &\quad \text{then } y^i = \tilde{a}_0^i + \tilde{a}_1^i x_1 + \tilde{a}_2^i x_2 + \cdots + \tilde{a}_n^i x_n \end{aligned} \tag{40}$$

where $\tilde{A}_j^i$ is interval type-2 fuzzy set and the definition of its membership function refer to (22)-(24); $\tilde{a}_0^i = [b_0^i - s_0^i, b_0^i + s_0^i], \tilde{a}_j^i = [b_j^i - s_j^i, b_j^i + s_j^i]$,$b_0^i, s_0^i, b_j^i, s_j^i \in \mathbf{R}$, $i = 1, 2, \ldots, c$ and $j = 1, 2, \ldots, n$.

For A2C1 and A2C0, the number of their rules is pre-specified. The parameters standing in the antecedent and consequent parts are both trained with the gradient descent algorithm. For each of them (A2C1 and A2C0), four kinds of type reduction and defuzzification methods are applied. These four kinds of methods are KM, NT, QF and COSTRWSR.

Standard Type-1 Fuzzy Logic Model (ST1)

The type-1 fuzzy model which has the same number of rules with A2C0 and A2C1, could be formulated as follows, and we mark it as ST1.

$$\begin{aligned} -&\text{if } x_1 \text{ is } A_1^i \text{ and } x_2 \text{ is } A_2^i \text{ and } \ldots \text{ and } x_n \text{ is } A_n^i \\ &\text{then } y^i = a_0^i + a_1^i x_1 + a_2^i x_2 + \cdots + a_n^i x_n \end{aligned} \tag{41}$$

where A_j^i is the type-1 fuzzy set with Gaussian membership function expressed as (27) and $a_0^i, a_j^i \in \mathrm{R}, i = 1, 2, \ldots, c$ and $j = 1, 2, \ldots, n$.

3 Experimental studies

For the fuzzy models discussed above, 15 datasets are used for carrying out a thorough performance analysis. The publicly available datasets coming from the UCI machine learning repository [14], the KEEL-dataset repository [15] and Statlib-Datasets Archive, are considered. Furthermore a commonly used Mackey-Glass time series is experimented with. A concise description of the data sets used in the experiments are listed in Table 1. It should be noted here that we use abbreviation for each dataset as: Auto MPG (APG); Boston Housing (BoH); Concrete Slump (CSl); Diabetes (Dia); Estimate of the percentage of body fat (EPB); Istanbul Stock Exchange (ISE); Laser (Las); Daily electricity energy (Dee); NNGC1 (NNG); NO2 (No2); Plastic (Pla); Pm 10 (P10); Quaake (Qua); Yacht Hydrodynamics (YaH); Mackey-Glass (McG).

For the interval type-2 structures equipped with q factor type reduction method, the data set is split into the training, validation, and testing data (60%-10%-30%) . The value of q is initialized with 10 random numbers (with uniform distribution) lying in [0, 1]. The one which minimizes the average of the performance index of the training and validation set is chosen. For all of the type-1 fuzzy models and the interval type-2 fuzzy models of which the type reduction method is KM, NT or COSTRWSR (not q factor method), the data set is split into the training and testing data (70%-30%). The experiments are repeated 15 times. As shown in (1), the mean square error (MSE) is used to evaluate the performance of the model. Experiments are completed based on Matlab 2017b running on a Lenovo Thinkpad desktop computer with Intel Core i7-6700 CPU @ 3.4GHz and 16GB memory, Windows 7 Professional system.

For anyone in the 15 datasets, the Wilcoxon rank-sum test [5] is used to tell whether there are significant differences between the accuracy of one type-2 fuzzy model and its associated type-1 fuzzy model. For example, to tell whether there are significant differences between the accuracy of SEIT2 and SEIT1, the values of Q for 15 times experiments of SEIT2 are compared with those of SEIT1 with Wilcoxon rank-sum test. As it is usually believed that the type-2 fuzzy model could outperform the type-1 model, the alternative hypothesis against the null hypothesis H_0 that $\tilde{\mu}_1 = \tilde{\mu}_2$ is $H_1 : \tilde{\mu}_1 > \tilde{\mu}_2$, where $\tilde{\mu}_1$ and $\tilde{\mu}_2$ are the median mean squared errors of the type-1 and type-2 fuzzy models, respectively. As these two samples (consider the 15 values of Q of SEIT2 as one sample and for those of SEIT1 as another sample) are

Table 1 List of datasets used in the experiments

Name	Features	Cases	Origin of the data
APG	8	398	http://archive.ics.uci.edu/ml/datasets/Auto+MPG
BoH	13	506	http://lib.stat.cmu.edu/datasets/boston
CSl	10	103	http://archive.ics.uci.edu/ml/datasets/Concrete+Slump+Test
Dia	2	43	http://sci2s.ugr.es/keel/dataset.php?cod=45
EPB	14	252	http://lib.stat.cmu.edu/datasets/bodyfat
ISE	8	563	http://archive.ics.uci.edu/ml/datasets/ ISTANBUL+STOCK+EXCHANGE
Las	4	993	http://sci2s.ugr.es/keel/dataset.php?cod=47
Dee	6	365	http://sci2s.ugr.es/keel/dataset.php?cod=46
NNG	4	370	http://sci2s.ugr.es/keel/dataset_smja.php?cod=942
No2	7	500	http://lib.stat.cmu.edu/datasets/NO2.dat
Pla	2	1650	http://sci2s.ugr.es/keel/dataset.php?cod=74
P10	7	500	http://lib.stat.cmu.edu/datasets/PM10.dat
Qua	3	2178	http://sci2s.ugr.es/keel/dataset.php?cod=75
YaH	7	308	http://archive.ics.uci.edu/ml/datasets/Yacht+Hydrodynamics
McG	2	500	Matlab

of equal size, we just take the size of the sample of type-1 as n_1 and that of type-2 as n_2, hence we have $n_1 = n_2 = 15$. For the significant test of accuracy of FCMIT2-1 vs FCMT1-1, IT2FCMIT2-1 vs FCMT1-1, SCIT2 vs SCIT1, A2C1 vs ST1, A2C0 vs ST1, etc., the test is completed in the same way as SEIT2 vs SEIT1 for any dataset in 15 datasets.

For the purpose of convenience, the abbreviation of the type-2 fuzzy structures used in the experiments is listed in Tables 2 and 3. These abbreviations are used in all figures.

As we are concerned with a thorough comparative analysis, we consider the relative accuracy and standard deviation (reported for the two classes of models) and express it as the ratio,

$$r_p = (1 - p_{T2}/p_{T1}) \times 100\%, \tag{42}$$

where p_{T1} and p_{T2} stand for the performances of the type-1 and type-2 fuzzy model, respectively; and $p = Q$ or σ, in which σ is the standard deviation of Q's for each model (type-1 or type-2).

The development time required to design the model on a basis of the existing experimental data is the second criterion being taken into account. We express it as the following ratio,

$$r_t = (1 - t_{T1}/t_{T2}) \times 100\%, \tag{43}$$

where t_{T1} and t_{T2} stand for the development time of the type-1 and type-2 fuzzy model, respectively. The values of r_t close to 100% indicate that there is a high computing overhead associated with the construction of type-2 fuzzy model.

To assess whether the improvements of the accuracy and stability delivered by the interval type-2 fuzzy models match its cost (pay more development time) when compared with those of type-1 fuzzy models. The values of $r_Q - r_t$ and ratios of $r_Q > r_t$ ($r_{r_Q > r_t}$) associated with each data set and each interval type-2 structure are displayed in Figs. 1 and 2, respectively. While the values of $r_{r_\sigma > r_t}$ for each data set and each interval type-2 fuzzy model are displayed in Figs. 3 and 4, accordingly. When the value of $r_Q - r_t$ is larger than 0, this means compared with the saving time ratio associated with type-1 fuzzy model when compared with type-2 fuzzy model, the ratio of the improvement conveyed by the type-2 fuzzy model when compared with type-1 fuzzy model is higher. Then we can say that the cost associated with type-2 fuzzy models

Table 2 List of abbreviation of type-2 fuzzy structure used in the following figures

Fuzzy structure	Type reduction method	Abbreivation
FCMIT2-1	KM	F21-KM
	NT	F21-NT
	QF	F21-QF
	COSTRWSR	F21-SR
FCMIT2-2	KM	F22-KM
	NT	F22-NT
	QF	F22-QF
	COSTRWSR	F22-SR
A2C0	KM	20-KM
	NT	20-NT
	QF	20-QF
	COSTRWSR	20-SR
SEIT2	KM	EI2
EKFT2	NT	KF2

Table 3 List of abbreviation of type-2 fuzzy structure used in the following figures

Fuzzy structure	Type reduction method	Abbreivation
IT2FCMIT2-1	KM	I21-KM
	NT	I21-NY
	QF	I21-QF
	COSTRWSR	I21-SR
IT2FCMIT2-2	KM	I22-KM
	NT	I22-NY
	QF	I22-QF
	COSTRWSR	I22-SR
A2C1	KM	21-KM
	NT	21-NY
	QF	21-QF
	COSTRWSR	21-SR
SCIT2	QF	CI2

matches its gain. While if $r_Q - r_t$ is negative, then the cost fails to meet the gain. Obviously, the higher of the positive value of $r_Q - r_t$ and $r_{r_Q > r_t}$, the better the cost matches the gain for the type-2 fuzzy models. This is similar with that of $r_{r_\sigma > r_t}$.

With the results shown in Fig. 1, it could be concluded that for the training set, the value of $r_{r_Q > r_t}$ is lower than 20%, while for testing set, the number of datasets for $r_{r_Q > r_t} > 50\%$ is lower than 2 in 15 datasets. Meaning that, for most datasets (for more than 73% datasets), the gain cannot meet the associated cost for the type-2 fuzzy models when compared with the corresponding type-1 fuzzy systems.

As is shown in Fig. 2, for the A2C0, A2C1, SEIT2 (EIT2 in figures) and SCIT2 (CIT2 in figures), the value of $r_{r_Q > r_t}$ is zero for the training set. While for the testing set, only when A2C0 and A2C1 are equipped with NT method, the value of $r_{r_Q > r_t}$ is lower than 10% and for the rest, $r_{r_Q > r_t}$ is zero. For the first order and second order fuzzy clustering method based interval type-2 fuzzy models, for all of the structures the value of $r_{r_Q > r_t}$ is lower than 10% for training set. While for the testing set, this value is higher than 30% for F21-NT, I21-NT, F22-NT and I22-NT when $m = 1.5$ and I21-NT when $m = 2.5$. For EKFT2 (KF2 in figures), $r_{r_Q > r_t}$ is around 30% for both the training and testing set.

As can be seen in Fig. 3, similar with the results shown in Fig. 1, the value of $r_{r_Q > r_t}$ is lower than 30% for all of the training sets and the number of $r_{r_Q > r_t} > 50\%$ is no more than 3 in 15 datasets.

With the results shown in Fig. 4, it is easy to note that for the training set, the ratio of $r_\sigma > r_t$ is lower than 20% for all of the structures, except for the EKFT2 (KF2 in figures). While for the testing set, for structures F21-NT, I21-KM, I21-NT, I21-QF, I21-SR, F22-NT, F22-QF, F22-SR, I22-NT, I22-QF and I22-SR when $m = 1.5$ and EKFT2 (KF2 in figures), the ratio of $r_\sigma > r_t$ is no less than 40%, while for the rest interval type-2 structures, this ratio is less than 40%.

As visualized in Fig. 5, for all of the training set, the ratio when the performances of interval type-2 fuzzy models show significant differences when compared with type-1 fuzzy models is lower than 40%, while for the testing set, the number of this ratio which is no less than 50% is no more than 3 in 15 datasets.

With the results shown in Fig. 6, the ratio when the performances of interval type-2 fuzzy systems show significant differences when compared with type-1 fuzzy systems is higher than 50% for F22-SR (when $m = 1.5$, 2, 2.5 and 3), FS-QF (when $m = 3$) and SEIT2 (EI2 in figures) for the training set, but this ratio is lower than 25% for these structures for the testing set. For most structures, the ratio when the performances of interval type-2 fuzzy systems show significant difference when compared with type-1 fuzzy systems for the testing set is higher than that of the training set.

The average ratio of $r_Q > r_t$, $r_\sigma > r_t$ and ratio for cases when the performances of the type-2 fuzzy models show significant differences when compared with those of type-1 fuzzy systems are shown in Fig. 7. Obviously, for training set, both $r_{r_Q > r_t}$ and $r_{r_\sigma > r_t}$ are lower than 10%, and for testing set, these ratios are lower than 20% and 30%, respectively. For the ratios of cases when the performances of type-2 fuzzy systems show significant differences when compared with type-1 fuzzy models are lower than 20% and 25% for training and testing set, accordingly.

Overall, the ratios of $r_Q > r_t$, $r_\sigma > r_t$ and the ratio for cases when the performance of the type-2 fuzzy models show significant differences when compared with those of type-1 fuzzy systems are lower than 50%, no matter against each data set or each structure or the average ratio. Though for EKFT2 (KF2 in figures) and its associated type-1 structure, the ratio of $r_Q > r_t$ and $r_\sigma > r_t$ are around 40%, the ratio for cases when the performances of the type-2 fuzzy models show significant differences when compared with type-1 fuzzy systems are lower than 20%. This results from that these kinds of structures are occupied with a fast type-reduction and defuzzification (decoding) method which may cause higher value of $r_{r_Q > r_t}$ and $r_{r_\sigma > r_t}$, but these interval type-2 structures does not produce significant differences when compared with its associated type-1 structures. For SEIT2 (EI2 in figures) and its associated type-1 structure, as SEIT2 uses KM as its type-reduction method

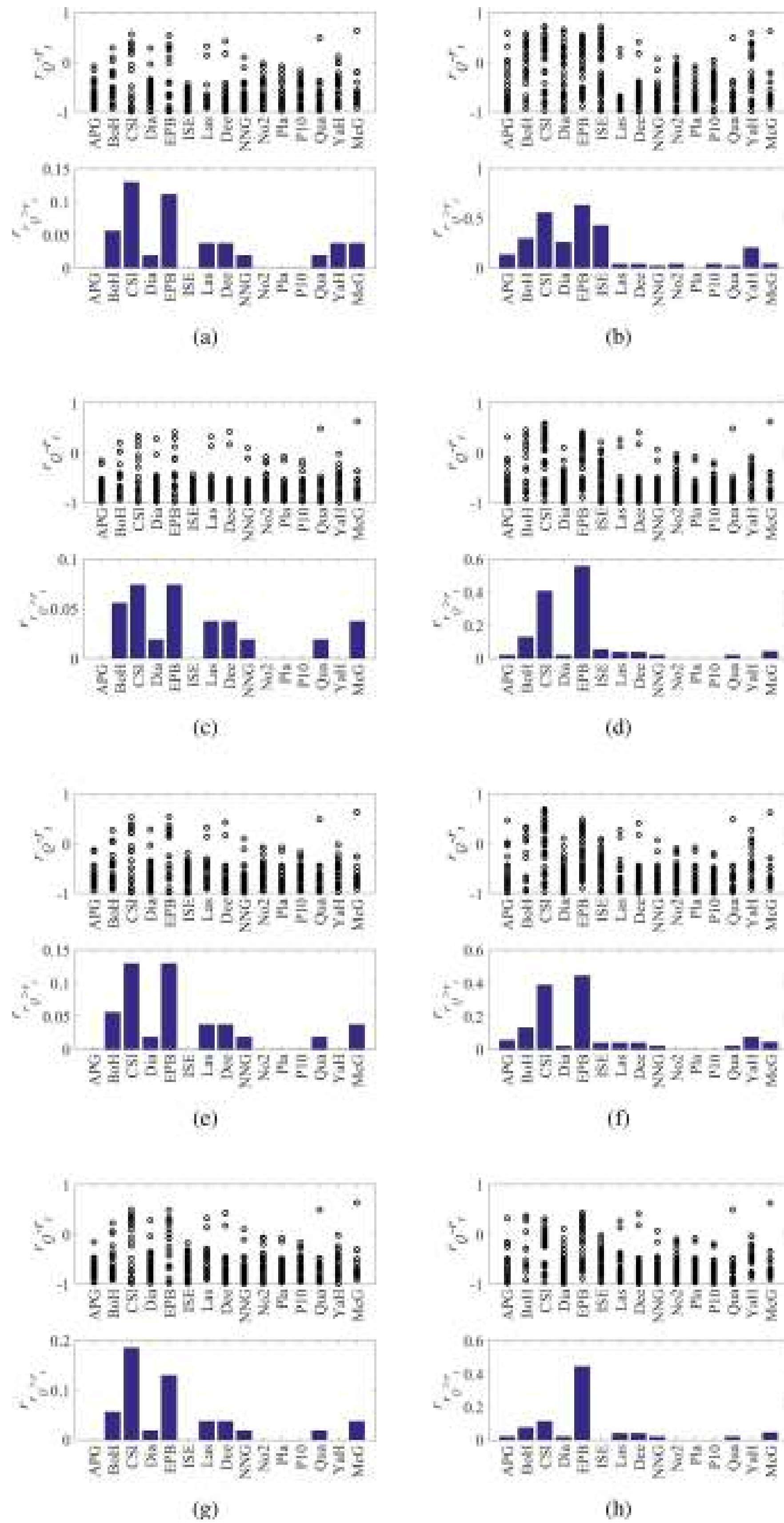

Fig. 1 Plots of $r_Q - r_t$ and ratio when $r_Q > r_t$ for each data set, **a**, **c**, **e** and **g** for training data, while **b**, **d**, **f** and **h** for testing data; **a**, **b** with $m = 1.5$, **c**, **d** with $m = 2$, **e**, **f** with $m = 2.5$ and **g**, **h** with $m = 3$

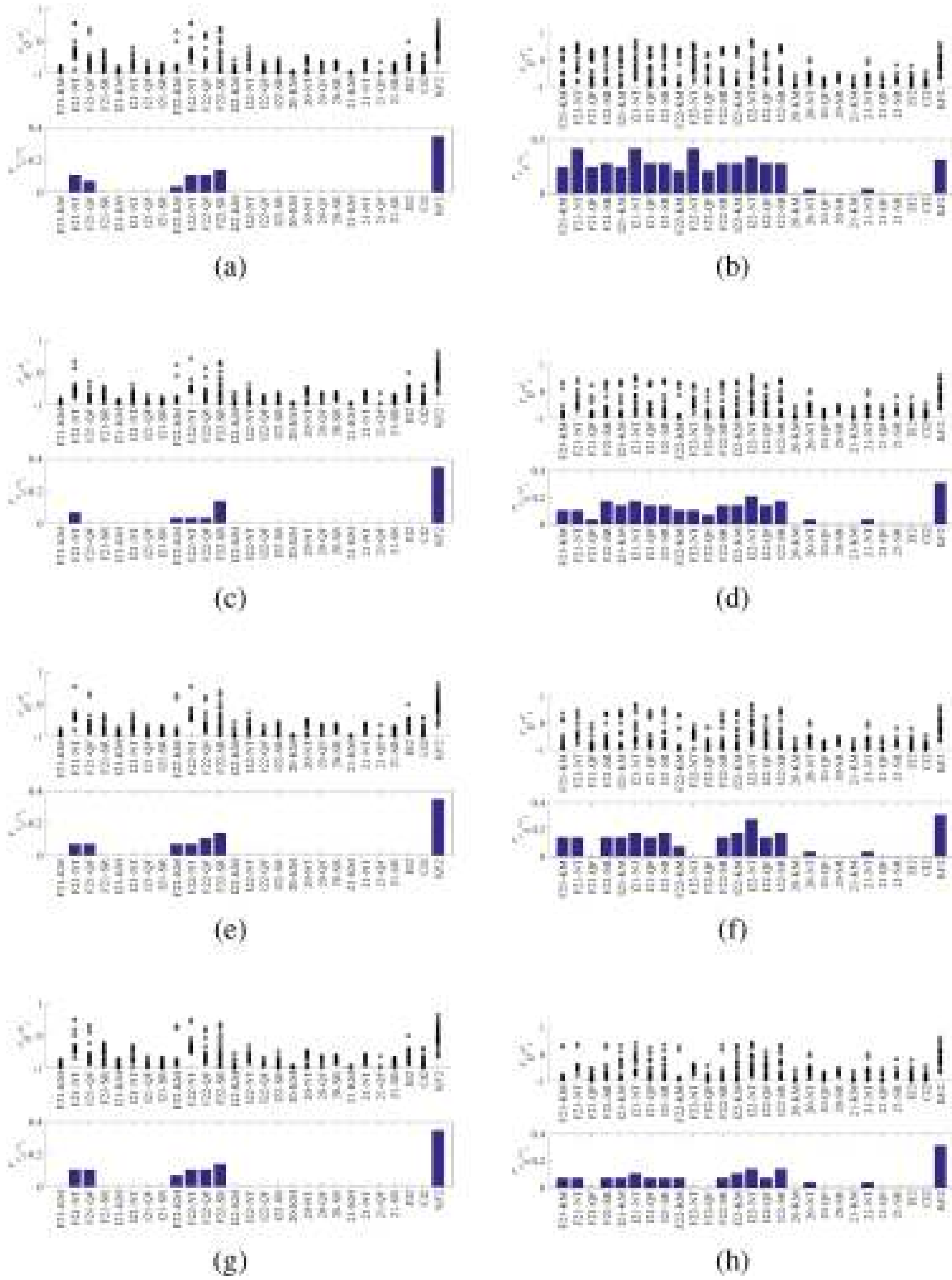

Fig. 2 Plots of $r_Q - r_t$ and ratio when $r_Q > r_t$ for each type-2 structure, **a**, **c**, **e** and **g** for training data, while **b**, **d**, **f** and **h** for testing data; **a**, **b** with $m = 1.5$, **c**, **d** with $m = 2$, **e**, **f** with $m = 2.5$ and **g**, **h** with $m = 3$

which develops slowly, this results in lower values of $r_Q > r_t$ and $r_\sigma > r_t$ which are both equal to zero, but higher ratios (> 50%) of cases of when the performances of the SEIT2 show significant differences when compared with those of its associated type-1 fuzzy structure for the training set. For A2C0 and A2C1 (both of these structures are equipped with four kinds of type-reduction methods) and their associated type-1 structures, the values for $r_{r_Q > r_t}$ and $r_{r_\sigma > r_t}$ and the ratio for cases when the performances of the type-2 fuzzy models show significant differences when compared with those of type-1 fuzzy systems are lower than 20% for both training and testing set. And for most of the fuzzy clustering based interval type-2 structures (also they are equipped with 4 kinds of type-reduction methods) and their associated type-1 structures, the ratios of $r_Q > r_t$ and $r_\sigma > r_t$ and ratio for cases when the performances of the type-2 fuzzy models show significant differences when compared with type-1 fuzzy systems for the testing set are higher than those of the training set.

Though there some improvements can be observed when the performances of interval type-2 fuzzy models are compared with those of type-1. The average ratios of cases for

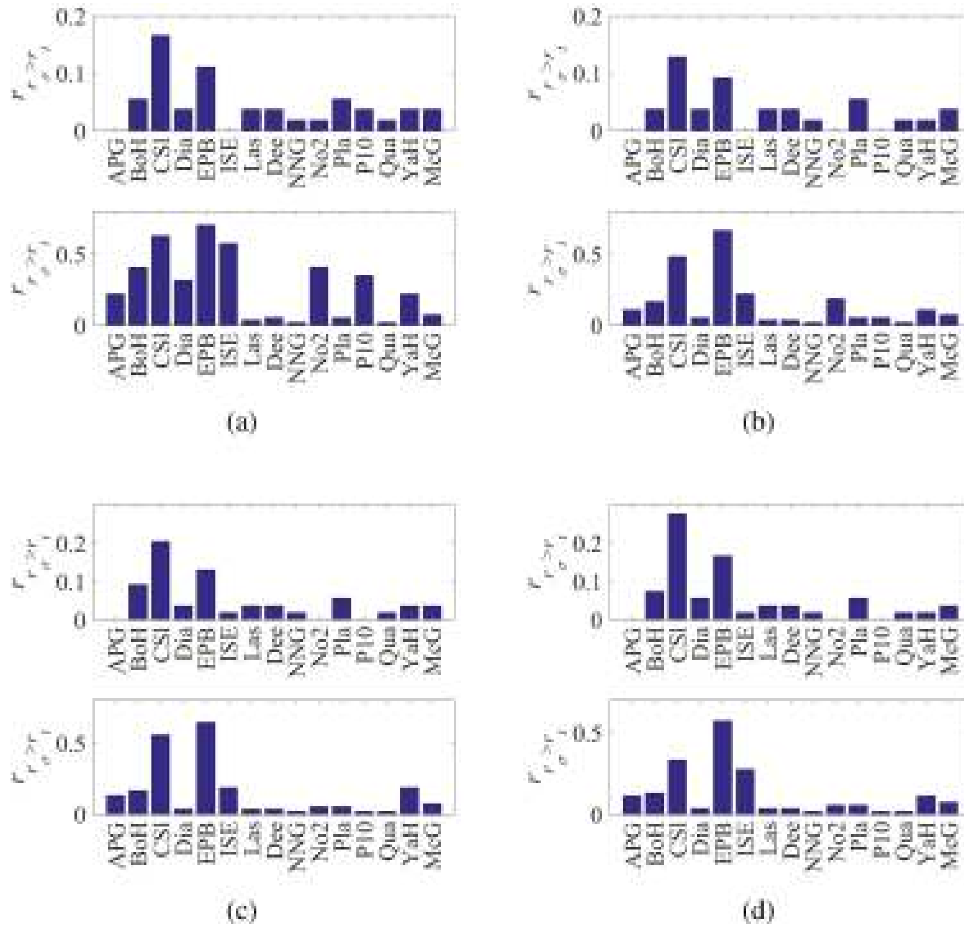

Fig. 3 Plots of ratio when $r_\sigma > r_t$ for each data set **a** $m = 1.5$, **b** $m = 2$, **c** $m = 2.5$ and **d** $m = 3$; results of training data are displayed in the first row, results of testing data are shown in the second row

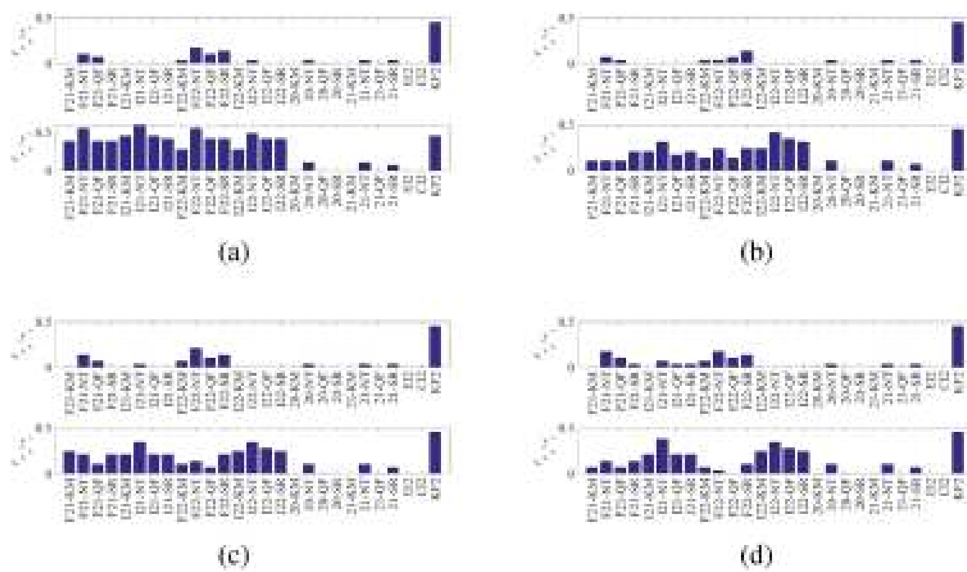

Fig. 4 Plots of ratio when $r_\sigma > r_t$ for each type-2 structure, **a** $m = 1.5$, **b** $m = 2$, **c** $m = 2.5$ and **d** $m = 3$; results of training data are displayed in the first row, results of testing data are shown in the second row

$r_Q > r_t$ and $r_\sigma > r_t$ are lower than 10% and 30% for training set and testing set, respectively. This entails that the computing overhead related with interval type-2 fuzzy models have cast a heavy burden, which originates because of more parameters needed to describe the membership functions of the interval type-2 fuzzy sets and the additional type-reduction phase when transforming the interval type-2 fuzzy sets to numeric outcomes. Although, for a few cases, some significant improvements could be observed when the performance of interval type-2 fuzzy models is compared with

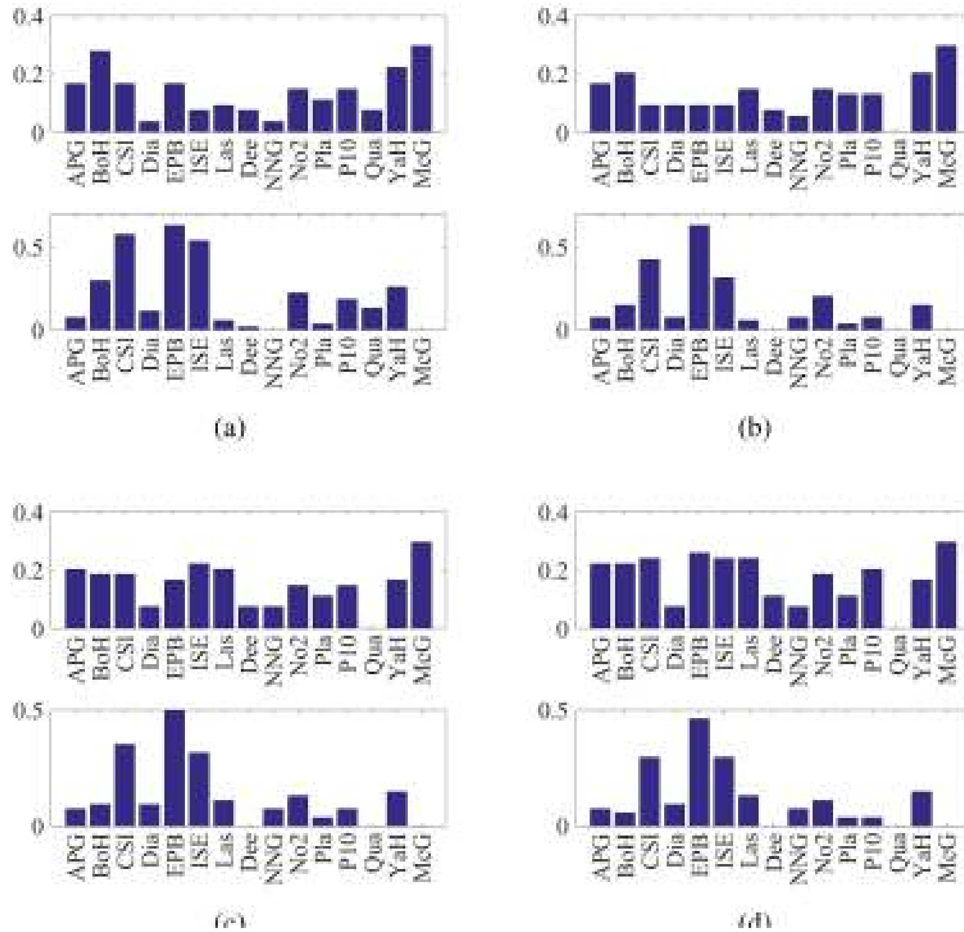

Fig. 5 Ratios of cases when the values of Q of type-2 models show significant difference (critical value equals to 0.05) when compared with those of the associated type-1 models for each dataset **a**, **b** with $m = 1.5$, **c**, **d** with $m = 2$, **e**, **f** with $m = 2.5$ and **g**, **h** with $m = 3$; results of training data are displayed in the first row, results of testing data are shown in the second row

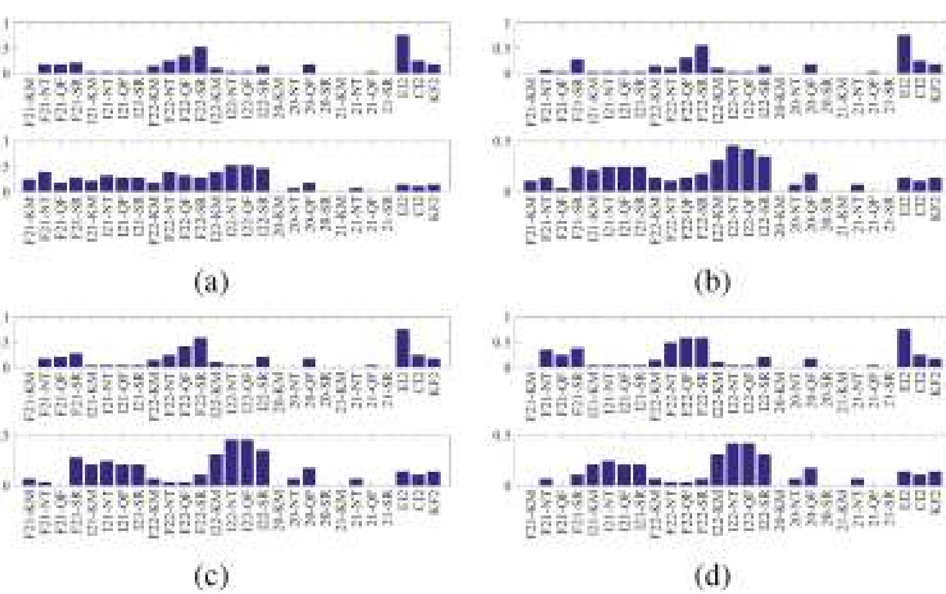

Fig. 6 For each type-2 structure, ratios of cases when its values of Q show significant difference (critical value equals to 0.05) when compared with those of the associated type-1 model **a** with $m = 1.5$, **b** with $m = 2$, **c** with $m = 2.5$ and **d** with $m = 3$; results of training data are displayed in the first row, results of testing data are shown in the second row

the performance of type-1 fuzzy models, the ratios for these cases are lower than 20% and 30% for training set and testing set, correspondingly. In most cases, there are no significant improvements when the performances of interval type-2 fuzzy models are compared with the quality delivered by the type-1 models.

When contrasting type-2 fuzzy models with their type-1 counterparts, as reported in Table 4 and 5, the improvement

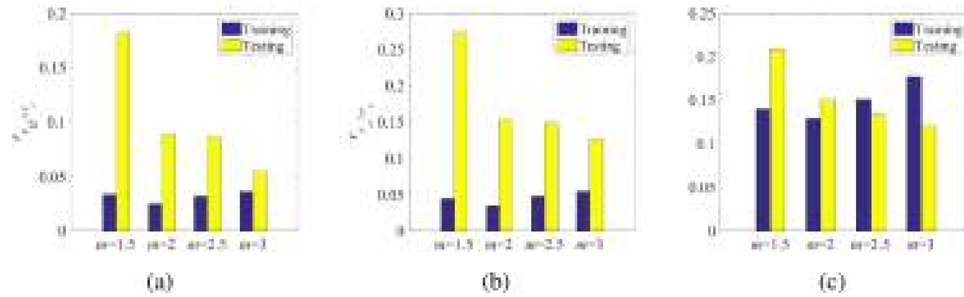

Fig. 7 **a** Ratios of cases when $r_Q > r_t$; **b** ratios of cases when $r_\sigma > r_t$ and (*c*) ratios of cases when the performances of the type-2 models show significant difference with those of type-1 models for training and testing data ($m = 1.5, 2, 2.5$ and 3)

Table 4 Average ratios of improvements (r_Q) and saving time (r_t) for each dataset when $m = 1.5$ and $m = 2$

Dataset	$m = 1.5$			$m = 2$		
	r_Q		r_t (%)	r_Q		r_t (%)
	Training (%)	Testing (%)		Training (%)	Testing (%)	
APG	− 4.71	4.12	72.50	− 4.24	− 8.62	77.25
BoH	− 8.78	22.68	75.03	− 10.06	3.43	78.23
CSl	2.68	52.17	63.97	− 1.76	37.71	68.92
Dia	9.21	23.45	70.64	− 4.74	− 3.24	76.07
EPB	4.76	64.55	69.91	2.03	58.75	73.19
ISE	− 13.35	40.35	65.33	− 16.03	8.21	69.32
Las	− 16.42	− 12.75	68.64	− 7.52	− 6.56	72.18
Dee	− 3.78	0.75	73.64	− 5.05	− 1.03	77.07
NNG	− 10.01	− 3.27	72.83	− 8.80	− 2.61	77.25
No2	− 6.68	8.73	69.75	− 6.73	0.60	75.03
Pla	− 5.30	− 6.85	72.83	− 5.21	− 6.77	78.24
P10	− 7.59	8.93	71.44	− 8.51	− 6.44	76.13
Qua	− 0.90	0.15	77.28	− 1.39	− 0.58	80.15
YaH	− 1.65	14.82	64.71	− 0.39	0.94	70.28
McG	0.50	− 0.40	68.69	0.61	− 0.96	73.48

Table 5 Average ratios of improvements (r_Q) and saving time (r_t) for each dataset when $m = 2.5$ and $m = 3$

Dataset	$m = 2.5$			$m = 3$		
	r_Q		r_t (%)	r_Q		r_t (%)
	Training (%)	Testing (%)		Training (%)	Testing (%)	
APG	− 3.18	− 6.71	75.14	− 2.98	− 11.31	76.44
BoH	− 10.10	− 1.98	75.88	− 9.26	− 2.61	77.48
CSl	4.55	39.59	64.86	− 3.29	13.04	67.26
Dia	− 7.46	− 5.31	72.18	− 6.64	− 5.39	72.82
EPB	6.87	44.91	69.98	10.11	37.61	72.07
ISE	− 7.38	3.70	65.44	− 3.92	0.80	67.95
Las	− 7.99	− 4.99	68.85	− 6.67	− 4.25	71.01
Dee	− 5.34	− 2.65	73.66	− 4.06	− 2.14	74.75
NNG	− 8.48	− 2.53	74.00	− 8.41	− 2.57	74.92
No2	− 5.99	− 4.49	72.26	− 5.36	− 6.42	74.12
Pla	− 5.39	− 6.77	76.30	− 5.32	− 6.88	77.29
P10	− 7.19	− 9.01	73.34	− 7.81	− 11.06	75.04
Qua	− 1.38	− 0.40	78.34	− 1.35	− 0.34	79.25
YaH	− 1.17	− 5.11	66.28	− 0.52	− 2.03	68.45
McG	0.53	− 0.60	70.63	0.53	− 0.69	71.89

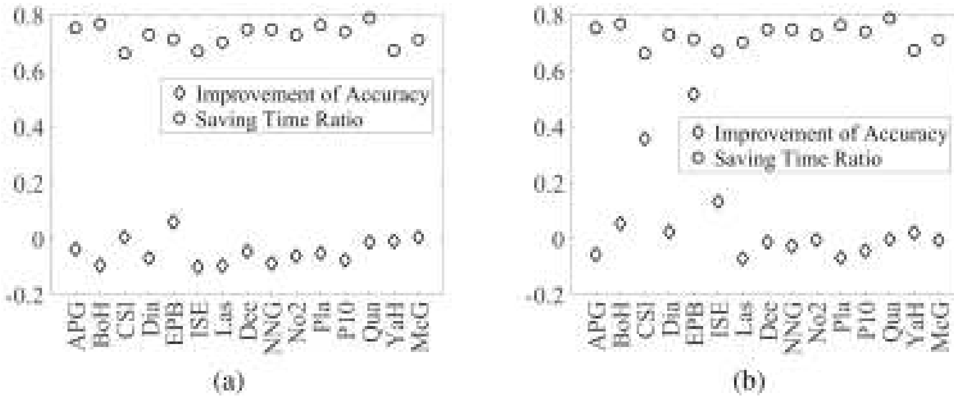

Fig. 8 Average ratios of improvement of accuracy (when performances in terms of accuracy of type-2 are compared with those of type-1) against average saving time ratio (when development time of type-1 is compared with that of type-2 fuzzy models) **a** for training set and **b** for testing set

of accuracy is in-between −4.54 and 5.39% (on average) for training and testing sets, respectively. These findings are associated with the significant computing overhead of around 72.62% (on average). These values are data-dependent as visualized in Fig. 8. In a nutshell, the completed experiments demonstrate that the gain in accuracy is quite limited (or in some cases some loses of accuracy have been reported) while a substantially higher computing overhead and a significantly increased complexity of the overall design process has been noted.

4 Conclusions

The comprehensive experimental study reported in the paper offers a well-supported evidence and carefully quantified comparative analysis of performance of type-2 and type-1 fuzzy models. It becomes apparent that a very limited improvement (or eventually degradation) in accuracy is significantly offset by substantial computing overhead. The number of parameters of type-2 fuzzy models is far larger than those being present in the type-1 counterpart. This calls for more advanced learning procedures; the engagement of both type reduction and decoding (defuzzification) makes the overall structure of the model complex and quite challenging with regard to the efficient usage of optimization mechanisms. This observation sounds quite pessimistic and, more radically, one may even question the relevance of type-2 fuzzy models. Nevertheless one has to put the findings in a certain perspective and revisit carefully a way in which type-2 models are evaluated and used. The accuracy criterion is numeric, viz. one evaluates the numeric manifestation of type-2 fuzzy model. The far richer format of results of this model, viz type-2 or interval-valued fuzzy set is lost and underutilized. One may conclude that a lot of development effort has not been fully taken advantage of. This phenomenon has been witnessed in case of type-1 fuzzy models but to a lesser extent. We badly need different evaluation criteria, which, as a matter of fact, are well aligned with the quantification of the quality of the non-numeric (granular) results produced by fuzzy models. Granular computing and models ensuing there [49] where the results are indeed information granules could offer an attractive and sound alternative and highlight a need for a paradigm shift. As of now, this direction has not been pursued in type-2 fuzzy modeling.

Acknowledgements This paper is supported by the National Natural Science Foundation of China (Nos. 61772002, 61976245), the Fundamental Research Funds for the Central Universities (Nos. XDJK2019B029, SWU119045, SWU119063, SWU119043)

Declarations

Conflict of interest There is no conflict of interest.

References

1. Juang CF, Tsao YW (2008) A self-evolving interval type-2 fuzzy neural network with online structure and parameter learning. IEEE Trans Fuzzy Syst 16(6):1411–1424
2. Mendel JM (2004) Computing derivatives in interval type-2 fuzzy logic systems. IEEE Trans Fuzzy Syst 12(1):84–98
3. Lin YY, Chang JY, Lin CT (2013) A TSK-type-based self-evolving compensatory interval type-2 fuzzy neural network (TSCIT2FNN) and its applications. IEEE Trans Ind Electron 61(1):159–447
4. Khanesar MA, Kayacan E, Teshnehlab M, Kaynak O (2012) Extended Kalman filter based learning algorithm for type-2 fuzzy logic systems and its experimental evaluation. IEEE Trans Ind Electron 59(11):4443–4455
5. Fay, MP, Proschan MA (2010) Wilcoxon-Mann-Whitney or t-test? On assumptions for hypothesis tests and multiple interpretations of decision rules. Stat Surv:1–39
6. Takag T, Sugeno M (1985) Fuzzy identification of systems and its application to modeling and control. IEEE Trans Syst Man Cybern 15(1):116–132
7. Sugeno M, Kang GT (1988) Structure identification of fuzzy model. Fuzzy Sets Syst 28(1):15–23
8. Xu W, Li W (2016) Granular computing approach to two-way learning based on formal concept analysis in fuzzy datasets. IEEE Trans Cybern 46(2):366–379
9. Pedrycz W, Gomide F (2007) Fuzzy systems engineering: toward human-centric computing. Wiley-IEEE Press, New York
10. Han S, Liu X (2016) Global convergence of Karnik–Mendel algorithm. Fuzzy Sets Syst 283:108–119

11. Khanesar MA, Khakshour AJ, Kaynak O, Gao H (2017) Improving the speed of center of sets type reduction in interval type-2 fuzzy systems by eliminating the need for sorting. IEEE Trans Fuzzy Syst 25(5):1193–1206
12. Liang Q, Mendel JM (1999) An introduction to type-2 TSK fuzzy logic systems. Proc IEEE Int Conf Fuzzy Syst 3:1534–1539
13. Hwang C, Rhee FCH (2007) Uncertain fuzzy clustering: interval type-2 fuzzy approach to C-means. IEEE Trans Fuzzy Syst 15(1):107–120
14. Lichman M (2020) UCI machine learning repository. http://archive.ics.uci.edu/ml
15. Alcal-Fdez J, Fernandez A, Luengo J, Derrac J, Garca S, Snchez L, Herrera F (2011) KEEL data-mining software tool: data set repository, integration of algorithms and experimental analysis framework. J Mult Valued Logic Soft Comput 17(2–3):255–287
16. Hassani H, Zarei J, Chadli M, Qiu J (2017) Unknown input observer design for interval type-2 T-S fuzzy systems with immeasurable premise variables. IEEE Trans Cybern 47(9):2639–2650
17. Zhou H, Ying H (2017) Deriving and analyzing analytical structures of a class of typical interval type-2 TS fuzzy controllers. IEEE Trans Cybern 47(9):2492–2503
18. Lee CH, Chang FY, Lin CM (2014) An efficient interval type-2 fuzzy CMAC for chaos time-series prediction and synchronization. IEEE Trans Cybern 44(3):329–341
19. Li H, Wu C, Shi P, Gao Y (2015) Control of nonlinear networked systems with packet dropouts: interval type-2 fuzzy model-based approach. IEEE Trans Fuzzy Syst 45(11):2378–2389
20. Khanesar MA, Kayacan E, Reyhanoglu M, Kaynak O (2015) Feedback error learning control of magnetic satellites using type-2 fuzzy neural networks with elliptic membership functions. IEEE Trans Cybern 45(4):858–868
21. Mendel JM, John RI, Liu F (2006) Interval type-2 fuzzy logic systems made simple. IEEE Trans Fuzzy Syst 14(6):808–821
22. Liang Q, Mendel JM (2000) Interval type-2 fuzzy logic systems: theory and design. IEEE Trans Fuzzy Syst 8(5):535–550
23. Wagner C, Hagras H (2000) Toward general type-2 fuzzy logic systems based on zSlices. IEEE Trans Fuzzy Syst 18(4):637–660
24. Karnik NN, Mendel JM (1998) Type-2 fuzzy logic systems: type-reduction. Proc IEEE Syst Man Cybern 2:2046–2051
25. Wu D (2013) Approaches for reducing the computational cost of interval type-2 fuzzy logic systems: overview and comparison. IEEE Trans Fuzzy Syst 21(1):80–99
26. Yeh CY, Roger Jeng WH, Lee SJ (2011) An enhanced type-reduction algorithm for type-2 fuzzy sets. IEEE Trans Fuzzy Syst 19(2):227–240
27. Li W, Xue X, Xu W, Zhan T, Fan B (2020) Double-quantitative variable consistency dominance-based rough set approach. Int J Approx Reason 124:1–26
28. Hu H, Wang Y, Cai Y (2012) Advantages of the enhanced opposite direction searching algorithm for computing centroid of an interval type-2 fuzzy set. Asian J Control 14(5):1422–1430
29. Khanesar MA, Khakshour AJ, Kaynak O, Gao H (2017) Improving the speed of center of sets type reduction in interval type-2 fuzzy systems by eliminating the need for sorting. IEEE Trans Fuzzy Syst 25(5):1193–1206
30. Gorzalczany MB (1988) Interval-valued fuzzy controller based on verbal model of object. Fuzzy Sets Syst 28(1):45–53
31. Liang Q, Mendel JM (2000) Equalization of nonlinear time-varying channels using type-2 fuzzy adaptive filters. IEEE Trans Fuzzy Syst 8(5):551–563
32. Wu H, Mendel JM (2002) Uncertainty bounds and their use in the design of interval type-2 fuzzy logic systems. IEEE Trans Fuzzy Syst 10(5):622–639
33. Li W, Pedrycz W, Xue X, Xu W, Fan B (2019) Fuzziness and incremental information of disjoint regions in double-quantitative decision-theoretic rough set model. Int J Mach Learn Cybern 10(10):2669–2690
34. Coupland S, John R (2007) Geometric type-1 and type-2 fuzzy logic systems. IEEE Trans Fuzzy Syst 15(1):3–15
35. Nie M, Tan WW (2008) Towards an efficient type-reduction method for interval type-2 fuzzy logic systems. IEEE Int Conf Fuzzy Syst:1425–1432
36. Begian MB, Melek WW, Mendel JM (2008) Stability analysis of type-2 fuzzy systems. IEEE Int Conf Fuzzy Syst:947–953
37. Greenfield S, Vhiclana F, Coupland S, John R (2009) The collapsing method of defuzzification for discretized interval type-2 fuzzy sets. Inf Sci 179:2055–2069
38. Li C, Yi J, Zhao D (2008) A novel type-reduction method for interval type-2 fuzzy logic systems. Fuzzy Syst Knowl Discov 1:157–161
39. Du X, Ying H (2010) Derivation and analysis of the analytical structures of the interval type-2 fuzzy-PI and PD controllers. IEEE Trans Fuzzy Syst 18(4):802–814
40. Tao CW, Taur JS, Chang CW, Chang YH (2012) Simplified type-2 fuzzy sliding controller for wing rock system. Fuzzy Sets Syst 207:111–129
41. Abiyev RH, Kaynak O (2010) Type-2 fuzzy neural structure for identification and control of time-varying plants. IEEE Trans Ind Electron 57(12):4147–4159
42. Pratama M, Lu J, Lughofer E, Zhang G, Er MJ (2017) An incremental learning of concept drifts using evolving type-2 recurrent fuzzy neural networks. IEEE Trans Fuzzy Syst 25(5):1175–1192
43. Wu D, Tan WW (2004) A simplified architecture for type-2 FLSs and its application to nonlinear control. IEEE Conf Cybern Intell Syst:485–490
44. Ying H (2008) General interval type-2 Mamdani fuzzy systems are universal approximators. North American Fuzzy Information Processing Society, pp 1–6
45. Ying H (2009) Interval type-2 Takagi-Sugeno fuzzy systems with linear rule consequent are universal approximators. North American Fuzzy Information Processing Society, pp 1–5
46. Wang LX, Mendel JM (1992) Fuzzy basis functions, universal approximation, and orthogonal least-squares learning. IEEE Trans Neural Netw 3(5):807–814
47. Wang LX (1994) Fuzzy systems are universal approximators. IEEE Trans Comput 43(11):1329–1333
48. Hu X, Pedrycz W, Wang X (2017) Granular fuzzy rule-based methods: a study in a comprehensive evaluation and construction on fuzzy models. IEEE Trans Fuzzy Syst 25(5):1342–1355
49. Karnik NN, Mendel JM (2001) Centroid of a type-2 fuzzy set. Inf Sci 132(1–4):195–220
50. Chang X, Wang Q, Liu Y, Wang Y (2017) Sparse regularization in Fuzzy C-Means for high-dimensional data clustering. IEEE Trans Cybern 47(9):2616–2627
51. Li L, Ding SX, Qiu J, Yang Y (2017) Real-time fault detection approach for nonlinear systems and its asynchronous T-S fuzzy observer-based implementation. IEEE Trans Cybern 47(2):283–294
52. Wu ZG, Dong S, Su H, Li C (2017) Asynchronous dissipative control for fuzzy Markov jump systems. IEEE Trans Cybern 48(8):2426–2436
53. Wang L, Lam HK (2018) New stability criterion for continuous-time Takagi-Sugeno fuzzy systems with time-varying delay. IEEE Trans Cybern 49(4):1551–1556
54. Li W, Pedrycz W, Xue X, Xu W, Fan B (2018) Distance-based double-quantitative rough fuzzy sets with logic operations. Int J Approx Reason 101:206–233