Stacked-Structure-Based Hierarchical Takagi-Sugeno-Kang Fuzzy Classification Through Feature Augmentation

Ta Zhou, Hisao Ishibuchi, *Fellow, IEEE*, and Shitong Wang

***Abstract*—In this paper, a new stacked-structure-based hierarchical Takagi–Sugeno–Kang (TSK) fuzzy classifier called SHFA-TSK-FC with both promising performance and high interpretability is proposed to tackle with the shortcoming of the existing hierarchical fuzzy classifiers in interpreting the outputs and fuzzy rules of intermediate layers. In order to achieve the enhanced classification performance, each component unit, which is a zero-order TSK fuzzy classifier, in SHFA-TSK-FC is organized in a stacked way such that all the input features of the original training samples plus the interpretable augmented features, corresponding to the interpretable output of each previous component unit, are fed as the input features of the current component unit. These augmented features can essentially open the manifold structure of the original input space such that the enhanced classification performance can be expected. In designing each component unit, its analytical solution to the consequent parts of fuzzy rules therein is obtained quickly by using the least learning machine such that SHFA-TSK-FC becomes scalable for large datasets. Its high interpretability is guaranteed by randomly selecting the input features and randomly choosing the fixed five Gaussian membership functions for the selected input features in the premise of each fuzzy rule. Experimental results on real-life datasets and an application case demonstrate the enhanced or at least comparable classification performance and high interpretability of SHFA-TSK-FC.**

***Index Terms*—Feature augmentation, interpretability, learning algorithm, least learning machine, stacked hierarchical structure, TSK fuzzy classifier.**

Manuscript received March 23, 2017; revised July 1, 2017, August 3, 2017, and September 28, 2017; accepted September 30, 2017. Date of current version November 22, 2017. This work was supported in part by Japan Society for the Promotion of Science (JSPS), in part by the National Natural Science Foundation of China under Grants 61272210, 61572236, 61300151, and NSFC-JSPS Grant 6161101250, in part by the Natural Science Foundation of Jiangsu Province under Grant BK20161268, and in part by the Fundamental Research Funds for the Central Universities (JUDCF13030). *(Corresponding author: Shitong Wang.)*

T. Zhou is with the School of Digital Media, Jiangnan University, Wuxi 214122, China, and also with the School of Electrical and Information Engineering, Jiangsu University of Science and Technology, Zhenjiang 100083, China (e-mail: jkdzhout@just.edu.cn).

H. Ishibuchi is with Department of Computer Science and Engineering, Southern University of Science and Technology, Shenzhen 518055, China, and also with the Department of Computer Science and Intelligent Systems, Osaka Prefecture University, Osaka 599-8531, Japan (e-mail: hisaoi@cs.osakafu-u.ac.jp).

S. T. Wang is with the School of Digital Media, Jiangnan University, Wuxi 214122, China, and also with the JiangSu Key Laboratory of Media Design and Software Technologies, WuXi 214122, China (e-mail: wxwangst@aliyun.com).

Digital Object Identifier 10.1109/TETCI.2017.2761915

I. Introduction

Due to their high classification performance and high interpretability, Takagi-Sugeno-Kang (TSK) fuzzy classifiers [1]–[7], [9], [30] have been making great success in many practical applications including data-driven prediction techniques in financial prediction, image processing and medical informatics, adaptive fuzzy control techniques for uncertain nonlinear systems [8], [10]–[12]. Various optimization approaches have been developed for modeling TSK fuzzy classifiers. Typical approaches include the Levenberg-Marquardt approaches [13], neurofuzzy approaches based on gradient-decent optimization techniques [14], genetic approaches for balancing the tradeoff between model complexity and classification accuracy [4], [7], [15], [30], [41]. In particular, in order to make them useful in real-world applications, both high classification accuracy and high interpretability are quite often desired in data driven TSK fuzzy classifiers. To this end, we attempt to address the following issue in this study: (1) inappropriate fuzzy partitions, which may heavily degrade the interpretability of the premises of fuzzy rules (i.e., if-parts) and hence can affect the classification performance of TSK fuzzy classifiers, and (2) the rule-explosion problem (i.e., curse of dimensionality), which often occurs in the design of a TSK fuzzy classifier for a high dimensional classification task.

In most TSK fuzzy classifiers, the number of fuzzy rules is usually assumed to be the same as the pre-specified number of clusters in the input samples. Thus fuzzy partitions can be determined by using various clustering methods such as the k-means [16], [22], FCM and its variants [17], [18], Gath-Geva [19], Gustafson-Kessel [20] and so on. However, it is not easy for us to appropriately pre-specify the number of fuzzy rules in practical applications. Although some mechanisms such as pruning [21] have been developed to reduce the size of fuzzy classifiers, each fuzzy rule generated by clustering-based approaches usually have antecedent conditions on all inputs in the if-parts of fuzzy rules. Thus the length of each fuzzy rule is usually the same as the number of features in classification tasks. Moreover, since a fuzzy set for specifying each antecedent condition is generated independently from other fuzzy rules without considering the overlap with other fuzzy sets, each resultant fuzzy rule generated by clustering-based approaches does not usually have high interpretability.

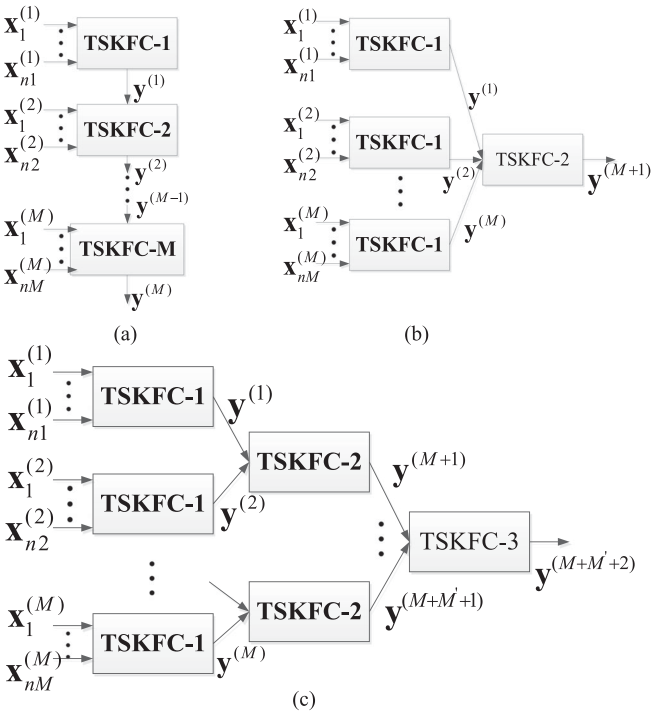

Fig. 1. Structures of hierarchical TSK fuzzy classifiers: (a) incremental, (b) aggregated, (c) cascaded (three stages involved).

Quite often, a fuzzy grid is taken to partition an input space and generate fuzzy rules in the design of TSK fuzzy classifiers. However, such a grid will result in the rule-explosion problem [23] such that the interpretability of TSK fuzzy classifiers will be severely degraded by the increase in the number of features. In order to overcome this rule-explosion problem, a hierarchical TSK fuzzy classifier was firstly proposed by Zhou *et al.* in [24] where each component TSK fuzzy classifier has only a few inputs. Since then, a lot of hierarchical fuzzy systems have been developed [24]. In general, when these existing hierarchical fuzzy systems are used for classification, they can be divided into *incremental, aggregated and cascaded,* as shown in Fig. 1 where a TSK fuzzy classifier TSKFC-i is taken as the ith component unit at the ith layer.

However, in the existing hierarchical TSK fuzzy classifiers, it is too hard to understand the output from each component TSK fuzzy classifier, which is fed as an input into component TSK fuzzy classifiers in the next layer. Consequently, due to the existence of intermediate variables, it is difficult to interpret the meaning of each fuzzy rule in component TSK fuzzy classifiers in intermediate and output layers where outputs from the previous layers are fed as inputs into the current layer. This difficulty becomes severe with the increase in the number of layers in hierarchical TSK fuzzy classifiers.

In order to address the above-mentioned difficulties, we should consider how to design a hierarchical TSK fuzzy classifier such that high accuracy, comprehensible intermediate outputs and highly interpretable fuzzy rules can be achieved. In what follows, *by the interpretability* we mean that *a hierarchical fuzzy TSK classifier has all the comprehensible intermediate outputs and all the highly interpretable fuzzy rules in its each component TSK fuzzy classifier*. We propose such a novel hierarchical fuzzy classifier design method based on the following ideas:

1) To partition each input feature into five fixed fuzzy sets with Gaussian membership functions. Their centers are fixed at [0, 0.25, 0.5, 0.75, 1]. Obviously, they may be associated with respective linguistic explanations: *very bad, bad, medium, good, very good*, even if all the widths of these five Gaussian membership functions may be different.
2) To select randomly less than or at most all the total input features for each fuzzy rule in each component unit of the proposed hierarchical classifier. This is to avoid the curse of dimensionality since a high-dimensional input space may be covered by a small number of short fuzzy rules. The use of short fuzzy rules is not only for reducing the number of fuzzy rules but also for enhancing the interpretability of each fuzzy rule (e.g., it is much easier to understand a fuzzy rule with only three antecedent conditions than that with eight antecedent conditions).
3) To design a stacked structure by means of feature augmentation for a data driven hierarchical TSK fuzzy classifier. Thanks to the stacked generalization principle [25], the resultant hierarchical TSK fuzzy classifier indeed has the enhanced classification performance. In such a stacked structure, the output of the previous component unit is augmented to the previous input space and then this augmented input space is fed as the input into the current component unit. Obviously, from the second layer to the last layer, the original input space is always taken as a part of the input space, which is completely different from the standard hierarchical models in which either the outputs of intermediate component units or the outputs of intermediate component units plus *only* a part of the original input space are fed as the input into the next component unit. Since each component unit is a zero-order TSK fuzzy classifier (see Section II), its output has a concise interpretation from the perspective of the certainty factor that a sample belongs to some class. As a result, when it is taken as an augmented feature, the premises of the resultant fuzzy rules still keep comprehensible with the randomly selected five fixed Gaussian membership functions. In other words, such a stacked structure with feature augmentation effectively circumvent the shortcoming that an intermediate variable in the existing hierarchical fuzzy classifiers is not comprehensible.
4) To employ our recent work, i.e., least learning machine (LLM) [26]–[28], to train each component unit quickly. By assigning random weights between the first layer and hidden layers and searching for the analytical solution of the ridge regression for the weights between the last hidden layer and the output layer, LLM can achieve the fast learning for feedforward neural networks. Because a zero-order TSK fuzzy classifier in each component unit of SHFA-TSK-FC can be viewed as a feedforward neural network, it is not hard to have such an idea that each component unit can be quickly trained by using LLM for both small/medium-sized and even large datasets.

Based on the above ideas, we design a stacked-structure based TSK fuzzy classifier which can be trained in the same manner as deep learning neural networks using the stacked generalization principle. We call this special hierarchical fuzzy classifier SHFA-TSK-FC (Stacked-structure-based TSK Fuzzy Classifier with Feature Augmentation) in this study. The contributions of our work can be summarized as the following three aspects:

1) As a new hierarchical TSK fuzzy classifier, the stacked-structure-based TSK fuzzy classifier SHFA-TSK-FC with promising performance and high interpretability is proposed. In SHFA-TSK-FC, the outputs of the previous component units, which are zero-order TSK fuzzy classifiers, are augmented as a part of the current input space so as to open the manifold structure existing in the original input space in a stacked way. In addition, the fast learning of each component unit of SHFA-TSK-FC can be achieved for both small/medium sized and even large datasets, by means of LLM.
2) The high interpretability of SHFA-TSK-FC can be ascribed to the following two aspects: 1) the fuzzy rules can be understood as those with only the original input features plus the interpretable augmented features in the premises of fuzzy rules. Hence, the premises of all fuzzy rules in SHFA-TSK-FC always have concise physical meanings. 2) The premise of each fuzzy rule in each component unit is generated by randomly selecting the input features, randomly choosing Gaussian membership functions for the selected input features. In such a way, the rule length of each fuzzy rule (i.e., the number of antecedent conditions) and the total number of fuzzy rules are significantly reduced.
3) The proposed hierarchical TSK fuzzy classifier SHFA-TSK-FC is carried out on real-life datasets and an application case. The experimental results have witnessed its enhanced or at least comparable performance and high interpretability.

The remaining sections of the paper are arranged as follows. In Section II, related studies including the classical TSK fuzzy classifier and least learning machine are briefly introduced. In Section III, the details of SHFA-TSK-FC are stated. Extensive experimental results are reported in Section IV. Section V concludes the paper.

II. Brief Review of TSK Fuzzy Classifier and Least Learning Machine

Because our study is based on least learning machine (LLM) [26]–[28] and the classical TSK fuzzy classifier, we give their brief reviews in this section.

The classical TSK fuzzy classifier is one of the most commonly used fuzzy classifiers. It adopts the following fuzzy rules [35].

Rule R^k: IF x_1 is $A_1^k \wedge x_2$ is $A_2^k \wedge \cdots \wedge x_d$ is A_d^k
THEN

$$f^k(\mathbf{x}) = p_0^k + p_1^k x_1 + \cdots + p_d^k x_d, \quad k = 1, 2, \ldots, K. \tag{1}$$

where A_i^k is a fuzzy subset which is subscribed by the input variable x_i for the kth rule, $\wedge$ is a fuzzy conjunction operator and K is the number of fuzzy rules. Each rule is premised on the input vector $\mathbf{x} = [x_1, x_2, \ldots x_d]^T$ and maps the fuzzy sets in the input space $A^k \subset R^d$ to a varying singleton which is denoted by $f^k(\mathbf{x})$. After the corresponding operations and defuzzification, the output of the TSK fuzzy model can be expressed as

$$y^o = \sum_{k=1}^{K} \frac{u^k(\mathbf{x})}{\sum_{k'=1}^{K} u^{k'}(\mathbf{x})} f^k(\mathbf{x}) = \sum_{k=1}^{K} \tilde{u}^k(\mathbf{x}) f^k(\mathbf{x}) \tag{2}$$

where $u^k(\mathbf{x})$ and $\tilde{u}^k(\mathbf{x})$ denote the fuzzy membership function and the normalized fuzzy membership function, respectively. They can be written as

$$u^k(\mathbf{x}) = \prod_{i=1}^{d} u_{A_i^k}(x_i) \tag{3}$$

and

$$\tilde{u}^k(\mathbf{x}) = u^k(\mathbf{x}) / \sum_{k'=1}^{K} u^{k'}(\mathbf{x}) \tag{4}$$

As the commonly used fuzzy membership function, Gaussian fuzzy membership function is quite often adopted, and it can be written as

$$u_{A_i^k}(x_i) = \exp\left(\frac{-(x_i - c_i^k)^2}{2\delta_i^k}\right) \tag{5}$$

Here, c_i^k and δ_i^k can be got by clustering techniques or other approaches. If we adopt the fuzzy c-means clustering algorithm FCM [17], c_i^k and δ_i^k can be expressed as

$$c_i^k = \sum_{j=1}^{N} u_{jk} x_{ji} \Big/ \sum_{j=1}^{N} u_{jk} \tag{6}$$

$$\delta_i^k = h \sum_{j=1}^{N} u_{jk} (x_{ji} - c_i^k)^2 \Big/ \sum_{j=1}^{N} u_{jk} \tag{7}$$

where u_{jk} denotes the fuzzy membership of the jth input sample $\mathbf{x}_j = (x_{j1}, x_{j2}, \ldots x_{jd})^T$ which belongs to the kth cluster, and h is a scale parameter given by the user.

As $f^k(\mathbf{x})$, we can use two formats in the classical TSK fuzzy classifier. When it is determined by a constant p_0^k, this kind of TSK fuzzy classifier is called zero-order TSK fuzzy classifier [1]. When it is determined by a linear function, this kind of TSK fuzzy classifier is called first-order TSK fuzzy classifier [1]. It is obvious that the output of zero-order TSK fuzzy classifier can be written as $y^o = \sum_{k=1}^{K} u^k(\mathbf{x}) p_0^k$. Once we determine the premises of fuzzy rules in the TSK fuzzy classifier and let

$$\mathbf{x}_e = (1, \mathbf{x}^T)^T \tag{8}$$

$$\tilde{\mathbf{x}}^k = \tilde{\mu}^k(\mathbf{x}) \mathbf{x}_e \tag{9}$$

$$\mathbf{x}_g = ((\tilde{\mathbf{x}}^1)^T, (\tilde{\mathbf{x}}^2)^T, \ldots, (\tilde{\mathbf{x}}^K)^T)^T \tag{10}$$

$$\mathbf{p}^k = (p_0^k, p_1^k, \ldots, p_d^k)^T \tag{11}$$

$$\mathbf{p}_g = ((\mathbf{p}^1)^T, (\mathbf{p}^2)^T, \ldots, (\mathbf{p}^K)^T)^T \tag{12}$$

the output of first-order TSK fuzzy classifier [2] can be written as

$$y^0 = \mathbf{p}_g^T \mathbf{x}_g. \tag{13}$$

In this way, the problem of the classical TSK fuzzy classifier learning can be changed into the learning of the parameters in the corresponding linear regression model [1].

The classical TSK fuzzy classifier can be naturally applied to binary classification tasks with the label set being $\{-1, +1\}$. That is to say, we can easily classify the input vector $\mathbf{x}$ into positive class if $y > 0$ or negative class otherwise. However, when the classical TSK fuzzy classifier is applied to classification tasks of c classes, a simple but effective way we prefer in this study is to assign the label set as $\{1, 2, \ldots\ldots, \mathrm{c}\}$, and then identify the class which $\mathbf{x}$ belongs to through observing which label the corresponding output y is nearest to.

Generally speaking, first-order TSK fuzzy classifiers have better classification performance than zero-order TSK fuzzy classifiers. However, first-order TSK fuzzy classifier is too hard to give clear interpretation for $(d + 1)$ parameters which are associated with the consequent parts of fuzzy rules. What is more, zero-order TSK fuzzy classifier has more concise interpretability than first-order TSK fuzzy classifier, because only one parameter p_0^k is involved. For a binary classification task with the label set $\{+1, -1\}$, the value of $p_0^k / \max_k(|p_0^k|)$ which is a real number in [0, 1] can be viewed a certainty factor of the kth fuzzy rule. For a c-class classification task with the label set $\{1, 2, \ldots, c\}$, let h^* be the nearest integer to p_0^k among $\{1, 2, \ldots, c\}$. When $1 \leq p_0^k \leq c$, the value of $2 * (0.5 - |p_0^k - h^*|)$ which is a real number in [0, 1] can be viewed as a certainty factor of the kth rule to support Class h^*. When $p_0^k < 1$ (or $c < p_0^k$), the kth fuzzy rule can be viewed as having a strong certainty factor to support Class 1 (or Class c). From these discussions, we can see that zero-order TSK fuzzy classifiers have more concise interpretability than first-order TSK fuzzy classifiers. So in this study, we prefer each component unit of SHFA-TSK-FC being a zero-order TSK fuzzy classifier.

Now let us introduce LLM in brief [27]. In [27], Wang *et al.* proposed a fast learning algorithm (i.e., LLM) which is for single-layer or multi-layer feedforward neural networks [27]. Its promising performance has been experimentally demonstrated in [26], and more advances about it can be seen in [28]. For the sake of brevity, here we shall only state LLM for a single-layer feedforward neural network. Assume that $g(\mathbf{x}, \theta_1), g(\mathbf{x}, \theta_2), \ldots, g(\mathbf{x}, \theta_{\tilde{N}})$ denote activation functions of $\tilde{N}$ hidden nodes in the hidden layer, $\theta_1, \theta_2, \ldots, \theta_{\tilde{N}}$ denote kernel parameter vectors and $\beta_1, \beta_2, \ldots, \beta_{\tilde{N}}$ denote the output weights. For the training dataset $D = \{(\mathbf{x}_i, t_i) | \mathbf{x}_i \in \mathbf{R}^d, t_i \in \mathbf{R}, i = 1, 2, \ldots, N\}$, Let $\mathbf{X} = [\mathbf{x}_1, \mathbf{x}_2, \ldots, \mathbf{x}_N]^T, \mathbf{T} = [t_1, t_2, \ldots, t_N]^T$, $\mathbf{H}_i = [g(\mathbf{x}_i, \theta_1), g(\mathbf{x}_i, \theta_2), \ldots, g(\mathbf{x}_i, \theta_{\tilde{N}})], \beta = (\beta_1, \beta_2, ..., \beta_{\tilde{N}})$.

When we try to determine these activation functions and the number of hidden nodes, LLM first randomly assigns all these parameters in the hidden layer of this single-layer feedforward neural network, then it realizes its fast learning for the parameter vector β by solving the following ridge regression problem with the given constant C.

$$\min \left(\frac{1}{2}\beta^2 + C \sum_{i=1}^{N} \xi_i^2 \right) \tag{14}$$

s.t. $(g(\mathbf{x}_i, \theta_1), g(\mathbf{x}_i, \theta_2), \ldots, g(\mathbf{x}_i, \theta_{\tilde{N}}))\beta^T = t_i + \xi_i, i = 1, 2, \ldots, N$

We can clearly see that the parameter vector β has an analytical solution $\tilde{\beta}$ which can be written as

$$\tilde{\beta}^T = \mathbf{H}^T \left(\mathbf{H}\mathbf{H}^T + \frac{1}{2C}\mathbf{I}_{N \times N} \right)^{-1} \mathbf{T} \tag{15}$$

where $\mathbf{H} = [\mathbf{H}_1, \mathbf{H}_2, \ldots, \mathbf{H}_N]^T$,$\mathbf{H}_i = [g(\mathbf{x}_i, \theta_1), g(\mathbf{x}_i, \theta_2), \ldots, g(\mathbf{x}_i, \theta_{\tilde{N}})]$ and $\mathbf{I}_{N \times N}$ is an $N \times N$ identity matrix.

There are obvious differences between LLM and BP-like learning algorithms [2]–[7]. For BP-like learning algorithms, it is essential to training all the parameters in both the hidden layer and output layer. However, according to [26]–[28], only the learning of the parameters in the output layer is desired for LLM. Due to this outstanding advantage, we use LLM to quickly train each TSK fuzzy classifier in each component unit of SHFA-TSK-FC, because a TSK fuzzy classifier can be viewed as a feedforward neural network [1], [2], [5].

III. SHFA-TSK-FC: The Proposed Hierarchical TSK Fuzzy Classifier

In this study, by means of feature augmentation, we develop a novel stacked-structure-based hierarchical TSK fuzzy classifier to achieve the enhanced classification performance and the high interpretability. The proposed classifier is motivated by the following considerations:

1) As pointed out in Trawinski's work in [30], the interpretability of a TSK fuzzy classifier may be enhanced by reducing the complexities of fuzzy rules. Although this idea can be obviously guaranteed by selecting the most significant input features and discarding others, it is not trivial to decide whether each input is required or not. What is more, it is likely that some input features are important in a certain region of an input space while they are unimportant and even ignorable in other regions. In a dynamically changing environment, the importance of each input feature may change. As a simple way of dealing with these situations, we randomly select input features for each component unit, which indeed results in the use of short fuzzy rules and hence enhances the interpretability of our SHFA-TSK-FC.
2) Unlike most hierarchical fuzzy classifiers [31]–[34] where the outputs of intermediate component units become very difficult to understand in the hidden and output layers, the outputs of intermediate 5 component units in the proposed hierarchical TSK fuzzy classifier are surely interpretable, due to the use of zero-order TSK fuzzy classifier. In some hierarchical models, input features are used not only in

the premises but also in the consequents of fuzzy rules. This may increase the complexity of learning of hierarchical fuzzy classifiers. In SHFA-TSK-FC, the interpretable output from a component unit is taken as an augmented feature in the premises i.e., then-parts of fuzzy rules of other component original input space are always kept. Thus, with the randomly selected Gaussian membership functions, antecedent conditions of fuzzy rules are always interpretable.

3) Recently, deep learning neural networks have demonstrated their outstanding classification performance in many applications. This is because their deep hierarchical structures can capture relevant high-level abstraction and characterize a training dataset very well in a layer-by-layer way. According to the stacked generalization principle, as a special kind of deep structures, stacked structure-based hierarchical models can guarantee the enhanced classification performance. Since the original input space plus interpretable augmented features are always used in its hidden layers, it is easy to understand the hidden layers in a similar manner to the input layer using the same input features. In addition, the stacked generalization principle can help us avoid solving a hard and nonconvex optimization problem which most deep learning approaches have to solve. What is more, each component unit of SHFA-TSK-FC can be quickly trained to give analytical solutions for small/medium sized and even large datasets, with the help of LLM.

A. On the Structure of SHFA-TSK-FC

The proposed hierarchical TSK fuzzy classifier SHFA-TSK-FC is composed of component units in a stacked manner, according to the stacked generalization principle [25]. Each component unit is essentially a special zero-order TSK fuzzy classifier. Therefore, before we state the entire structure of SHFA-TSK-FC, we first introduce the structure of each component unit, as shown in Fig. 2.

A.1. On the Structure of Each Component Unit: In each component unit of SHFA-TSK-FC, we directly choose five partitions which are denoted by Gaussian membership functions. The centers of these Gaussian membership functions (i.e., MF_1, MF_2, MF_3, MF_4 and MF_5) have been fixed at [0, 0.25, 0.5, 0.75, 1]. Obviously, even if their kernel widths are randomly selected, they may still have their respective linguistic explanations: *very bad, bad, medium, good, very good.* We do so because we want to follow the Kuncheva's claim [36] that if the choice of membership functions is not consistent throughout the implementation or the membership functions are of irregular shapes, then they are unlikely to associate with the linguistic labels precisely and unambiguously.

In this study, we randomly select the input features and ignore the remaining features. We do so because the interpretability of the TSK fuzzy classifier can be enhanced by reducing the complexity of the premises of fuzzy rules in this way. Hence, fuzzy rules in each component unit can be denoted in the following form:

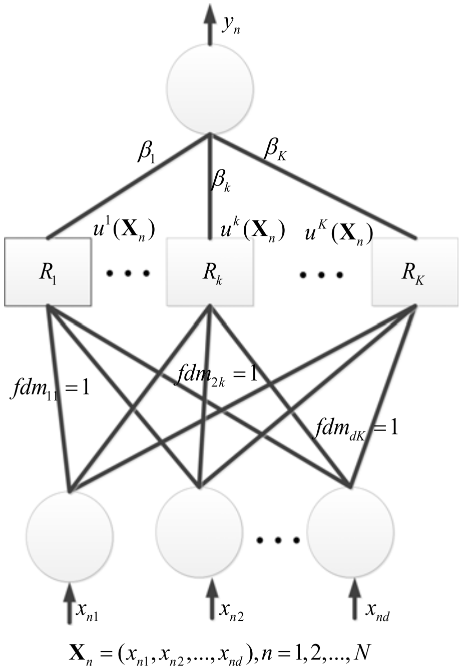

Fig. 2. A zero-order TSK fuzzy.

$$
\begin{aligned}
&If\, x_1\, is\, bad\, with\, fdm_{1k}=1 \wedge x_2\, is\, good\, with\, fdm_{2k}=1\\
&\wedge x_3\, is\, ``don't\, care''\, with\, fdm_{3k}=0 \wedge x_4\, is\, medium\\
&with\, fdm_{4k}=1
\end{aligned}
\tag{16}
$$

in which "don't care" means that the corresponding feature is ignored. In order to achieve random selection of both Gaussian membership functions and the input features, we first define two matrices (i.e., matrix FDM and RGM). Matrix FDM is called the feature decision matrix. In $FDM=[fdm_{jk}]_{d\times K}$, each element represents a decision about the corresponding input feature. $fdm_{jk}=1$ implies that the jth feature has been involved and $fdm_{jk}=0$ otherwise. The second matrix is the $d\times 5\times K$ rule generation matrix RGM. In RGM, the value of every element is randomly assigned binary value and decides which one of five Gaussian membership functions is adopted. For example,$RGM[2,1,5]=1$ indicates that the second input feature in the 5th fuzzy rule takes the Gaussian membership function about *very bad*. Based on the analysis above, the output of this particular zero-order TSK fuzzy classifier can be written as

$$
y^0=\sum_{k=1}^{K}u^k(\mathbf{x})\beta_k \tag{17}
$$

where $u^k(\mathbf{x})=\prod_{j=1}^{d}(u^k_{A_i}(x_j)\otimes fdm_{jk})$ in which $\mu^k_{A_i}(x_j)\otimes fdm_{jk}$ takes $u^k_{A_i}(x_j)$ if $fdm_{jk}=1$, 1 otherwise. Obviously, RGM can be used to determine Gaussian membership function $u^k_{A_i}(x_j)$. In most cases, FDM can be obtained from experts by means of their specific domain knowledge. However, in the proposed TSK fuzzy classifier, every element of FDM can be randomly assigned 0 or 1. More importantly, these values may not be given in advance. Such a zero-order TSK fuzzy classifier can be seen in Fig. 2. We can readily find that the structure of this zero-order TSK fuzzy classifier is similar to LLM for a single-layer feedforward neural network, since each fuzzy rule in the proposed TSK fuzzy classifier may be equivalently expressed as the corresponding hidden node.

A.2. On the Stacked Structure and Its Interpretability: As explained in the above, SHFA-TSK-FC consists of component units in a stacked way, according to the stacked generalization principle. The stacked generalization principle [25] can assure its enhanced generalization capability through constantly opening the manifold structure of the original input space. Another distinctive advantage exists in the fact that unlike most of deep learning methods, solving a hard and nonconvex optimization problem is not required. Because each component unit is a zero-order TSK fuzzy classifier, its outputs actually contain the discriminative information for both binary and multi-class data. Therefore, as an effective means of opening the manifold structure of the original input space, SHFA-TSK-FC augments the outputs of the previous component units into the original input space and then present this augmented input space as the input space into the current component unit. In this way, all the original features with their original physical meanings are kept in the current component unit. As explained in Section II, since each rule of a zero-order TSK fuzzy classifier adopted in each component unit has a interpretable output and hence the overall output of this fuzzy classifier for a sample input is interpretable for both binary and multi-class classification, the corresponding augmented fatures become concisely interpretable. What is more, SHFA-TSK-FC is suitable for both small/medium sized and even large datasets, due to the use of LLM. Here, by a large dataset, we mean the number of the samples is large and the number of dimensions is comparatively small.

Fig. 4 demonstrates the structure of SHFA-TSK-FC. Now let us state how SHFA-TSK-FC works. As shown in Fig. 4, the original training sample set X and its corresponding output set $\mathbf{T}$ are fed as the input into the first component unit. With the output set $\mathbf{Y}_1$ of the first component unit, SHFA-TSK-FC generates the augmented input space, i.e.,$[\mathbf{X}, \mathbf{Y}_1, \mathbf{T}]$, then feeds it as the input into the second component unit. With the output set $\mathbf{Y}_2$ of the second component unit, SHFA-TSK-FC generates the augmented input space, i.e.,$[\mathbf{X}, \mathbf{Y}_1, \mathbf{Y}_2, \mathbf{T}]$, then feeds it as the input into the third component unit. Such a procedure is repeated until satisfactory classification performance is achieved or the maximum depth is reached. The above training process can be graphically described in Fig. 3. Obviously, the depth of SHFA-TSK-FC is equal to the sum of one and the number of augmented features.

Now let us justify the interpretability of SHFA-TSK-FC. As shown in Fig. 4, we can easily observe the following two facts: (1) as stated in the above, the interpretable augmented features representing the outputs from the previous component units are always involved in the premises of fuzzy rules in the current component unit of SHFA-TSK-FC, therefore all the features in the premises of all fuzzy rules in SHFA-TSK-FC always have concise physical meanings. (2) Since the premise of each fuzzy rule in each component unit is generated by randomly selecting the input features, randomly choosing the interpretable Gaussian membership functions for the selected input features, each premise of each fuzzy rule is interpretable. In terms of these two facts, we can certainly claim that each component unit and hence SHFA-TSK-FC are interpretable. In other words, *once the structure of SHFA-TSK-FC is determined after training, SHFA-TSK-FC always becomes interpretable.*

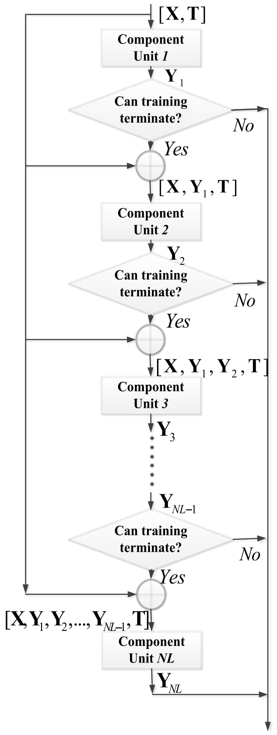

Fig. 3. The training process of SHFA-TSK-FC.

Our interpretability discussions are relative. It may be clear that our SHFA-TSK-FC classifiers are much more interpretable than back-box classifiers such as multi-layer neural networks and support vector machines. However, it is also clear that our classifiers are less interpretable than a fuzzy classifier based on a small number of short fuzzy rules with a single consequent class such as "If x_1 is *small* and x_9 is *large* then Class 1" and "If x_5 is *large* and $x_1 8$ is *medium* then Class 2". However, due to the interpretability-accuracy tradeoff [15], such a simple fuzzy classifier usually does not have high classification accuracy. Our discussions on the interpretability of the proposed approach are based on the comparison with non-hierarchical TSK fuzzy classifiers where each fuzzy rule has all the given features in its if-part (i.e., each fuzzy rule is very long) and hierarchical TSK fuzzy classifiers where short fuzzy rules are hierarchically structured (i.e., interpretation of intermediate variables is very difficult).

B. On the Learning Algorithm of SHFA-TSK-FC

Here we first state the learning algorithm of each component unit of SHFA-TSK-FC, i.e., algorithm 1, and then give the entire learning algorithm of SHFA-TSK-FC, i.e., algorithm 2.

Below we give several remarks about the above learning algorithm 1.

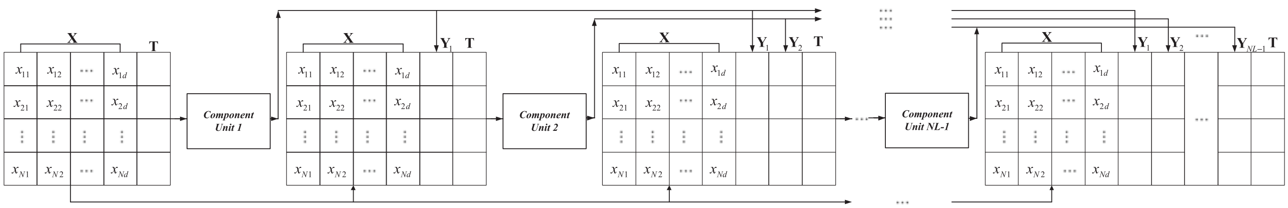
X
T
x_{11} x_{12} ⋯ x_{1d}
x_{21} x_{22} ⋯ x_{2d}
x_{N1} x_{N2} ⋯ x_{Nd}
Component Unit 1
Y_1
Component Unit 2
Y_2
Component Unit NL-1
Y_{NL-1}

Fig. 4. Structure of SHFA-TSK-FC.

Remark 1: According to LLM, here we compute the output weight β_{nl} of the *nl*th component unit using

$$\beta_{nl} = \left(\frac{1}{C}\mathbf{I} + \mathbf{H}_{nl}^T\mathbf{H}_{nl}\right)^{-1}\mathbf{H}_{nl}\mathbf{T} \tag{18}$$

instead of its alternative equivalent solution

$$\beta_{nl} = \mathbf{H}_{nl}^T\left(\frac{1}{C}\mathbf{I} + \mathbf{H}_{nl}\mathbf{H}_{nl}^T\right)^{-1}\mathbf{T} \tag{19}$$

It is obvious that the complexity of computing matrix inversion in (18) is $O(K_{nl}^3)$. However, the complexity of computing the matrix inversion in (19) is $O(N^3)$. Please note, the value of K_{dp} is generally much less than the value of N in practical applications. So algorithm 1 becomes applicable to large datasets.

Remark 2: Parameter *C* is only one important parameter which can be tuned in algorithm 1. Here the value of *C* can be chosen as a comparatively large value. In this study, we can take $C = 400$.

Now, we can easily give the entire learning algorithm of SHFA-TSK-FC, according to algorithm 1.

Now we give the remark about the above learning algorithm 2.

Remark 3: On the one hand, according to the stacked generalization principle, the classification performance can generally be enhanced with the increase of the depth NL of SHFA-TSK-FC. On the other hand, we should note that the depth of SHFA-TSK-FC is obviously dependent on the number of augmented features. In order to open the manifold structure of the training data and simultaneously make the training data be not distorted too much, we should consider that the number of augmented features must be less that the number of the original input features. What is more, our experiments reveal that NL may be a small integer for most ases. After considering these factors, we take $NL = 3$ or 4 or 5 in our experiments.

C. On Time Complexity

Here we first observe the time complexity of training the nlth component unit using algorithm 1. Its time complexity consists of the time complexity of generating the rule generation matrix RFM_{nl}, the time complexity of generating the feature decision matrix FDM_{nl}, the time complexity of computing $\mathbf{H}_{nl}$ and the time complexity of computing its LLM based output.

The time complexity of generating matrix RGM_{nl} is $O(5(d + nl)K_{nl})$, where d is the number of features and K_{nl} is the number of fuzzy rules. For the time complexity of generating the feature decision matrix FDM_{nl}, it is obviously $O((d + nl)K_{nl})$. In terms of steps 2.2, 2.3 and 2.4 in algorithm 1, $O(5N(d + nl)^2K_{nl})$ will be required to compute the matrix $\mathbf{H}_{nl}$. Step 2.5 of algorithm 1 requires $O(K_{nl}^3 + NK_{nl} + N)$ to compute β_{nl}. Obviously, the time complexity of step 2.6 of algorithm 1 is $O(NK_{nl})$. Therefore, the time complexity of training each component unit becomes

$$\begin{aligned} &O(5(d + nl)K_{nl} + (d + nl)K_{nl} + 5N(d + nl)^2K_{nl} + K_{nl}^3 \\ &\quad + \ NK_{nl} + N + NK_{nl}) \approx O(5N(d + nl)^2K_{nl} + K_{nl}^3) \end{aligned}$$

Since the depth of SHFA-TSK-FC is NL, so with the use of LLM, the time complexity of training SHFA-TSK-FC (i.e., algorithm 2) roughly is $O(\sum_{nl=1}^{NL}(5N(d + nl)^2K_{nl} + K_{nl}^3))$, which is still linearly dependent on the size N of the training set when N is considerably large while the number K_{nl} of fuzzy rules in each component unit is comparatively small. In particular, when each component unit owns the same number K of fuzzy rules, the time complexity of algorithm 2 will become $O(\sum_{nl=1}^{NL}(5N(d + nl)^2K + K^3))$

Let us go back to observe the time complexity of the classical zero-order and first-order TSK fuzzy classifiers in Section II. When FCM [17] is used to determine the premises of fuzzy rules in these two classifiers, its time complexity is $O(NKd)$ where K denotes the number of fuzzy rules in both TSK fuzzy classifiers. Once all the premises are determined, all the consequent parts of fuzzy rules can be estimated by solving the corresponding linear regression system (i.e., (8)–(13)) with a conventional quadratic programming solver, thereby resulting in the time complexity of $O(N \sim N^{2.3})$ [1]. Therefore, the entire time complexity of both zero-order and first-order TSK classifiers becomes $O(N(Kd+1)) \sim O(N(Kd + N^{1.3}))$. In general, K in the classical zero-order and first-order TSK fuzzy classifiers is much bigger than each K_{nl} in SHFA-TSK-FC. By comparing their time complexities, we can find that when N is not big, the time complexity of SHFA-TSK-FC may perhaps be higher than the classical zero-order and first-order TSK fuzzy classifiers. When N becomes big, SHFA-TSK-FC becomes very competitive, since both zero-order and first-order TSK fuzzy classifiers often use a lot of fuzzy rules, thereby leading to poor interpretability.

IV. Experiments and Results

In this section, experimental results are presented to demonstrate the enhanced performance of SHFA-TSK-FC. We adopt twenty-one datasets including both small/medium and large datasets and do comparative study between SHFA-TSK-FC, zero-order and first-order TSK fuzzy classifiers, and two evolutionary fuzzy classifiers in KEEL software toolbox, i.e.,

Algorithm 1: Learning algorithm of the nl-th component unit of SHFA-TSK-FC.

```
Step 1-Initialization:
  (1.1) Fuzzify all input features into five Gaussian membership functions GMF1, GMF2, GMF3,
        GMF4 and GMF5, respectively, with their fixed centers at [0, 0.25, 0.5, 0.75, 1] and their
        linguistic explanations: very low, low, medium, high, very high. Randomly generate the
        corresponding five kernel widths, i.e.,$\sigma_k \in \mathbf{R}^+$, $k = 1, 2, \ldots, 5$.
  (1.2) Initialize the feature decision matrix $FDM_{nl}$, the rule generation matrix $RGM_{nl}$.
Step 2-Training:
  (2.1) Compute the values of Gaussian membership functions for each feature $x_{ij}$:
            $u(k, x_{ij}) = \exp(-(x_{ij} - a_k)^2 / 2\sigma_k^2)$,    (20)
        where $i = 1, 2, \ldots, N, j = 1, 2, \ldots, d$, kernel widths $\sigma_k \in \mathbf{R}^+ a_k \in \{0, 0.25, 0.5, 0.75, 1\}$,
        $k = 1, 2, \ldots, 5$.
  (2.2) Compute the following value of each feature in a fuzzy rule, in term of the rule generation matrix
        $RGM_{nl}$, by
        $v_{jl}(x_{ij}) = \begin{cases} 1 - \prod_{k=1}^{5} (1 - RGM_{nl}(j, k, l) u(k, x_{ij})), & fdm_{jl} = 0 \\ 1 & fdm_{jl} = 1 \end{cases}$    (21)
        where $i = 1, 2, \ldots\ldots, N, j = 1, 2, \ldots\ldots, d$,
        $k = 1, 2, \ldots\ldots, 5, l = 1, 2, \ldots\ldots, K_{nl}$.
  (2.3) Compute the following value of the if-part of a fuzzy rule
            $w_{il} = \prod_{j=1}^{d} v_{jl}(x_{ij})$    (22)
        where $w_{il} \in R^{d*K_{nl}}, i = 1, 2, \ldots\ldots, N$,
        $l = 1, 2, \ldots\ldots, K_{nl}$
  (2.4) Construct the rule layer output matrix $\mathbf{H}_{nl}$
            $\mathbf{H}_{nl} = \begin{bmatrix} w_{11} & \cdots & w_{1K_{nl}} \\ \vdots & \ddots & \vdots \\ w_{N1} & \cdots & w_{NK_{nl}} \end{bmatrix}_{N \times K_{nl}}$    (23)
  (2.5) Compute the output weight vector $\beta_{nl}$ of the $nl$th component unit, by using the least
        learning machine LLM
            $\beta_{nl} = \left( \frac{1}{C} \mathbf{I} + \mathbf{H}_{nl}^T \mathbf{H}_{nl} \right)^{-1} \mathbf{H}_{nl} \mathbf{T}$    (24)
        where $C$ is the given regularization parameter and $\mathbf{I}$ is an $K_{nl} \times K_{nl}$ identity matrix.
  (2.6) Compute the whole output matrix $\mathbf{Y}_{nl}$
            $\mathbf{Y}_{nl} = \mathbf{H}_{nl} \beta_{nl}$    (25)
```

Algorithm 2: Learning algorithm of SHFA-TSK-FC.

```
Step 1-Initialization:
    Input the training set $\mathbf{X} = [\mathbf{x}_1, \mathbf{x}_2, \ldots, \mathbf{x}_N]^T$, the corresponding class label set
    $\mathbf{T} = [t_1, t_2, \ldots, t_N]^T$, where $\mathbf{x}_n \in \mathbf{R}^d$, $t_n \in \{+1, -1\}$ for a binary classification task,
    otherwise $t_n \in \{1, 2, \ldots.., c\}$ for a multi-class classification task, in which n = 1,2, .....,N,
    and c (>2) is the number of classes, the number of fuzzy rules in each component unit is assumed
    to be $K$ and the number of component units in SHFA-TSK-FC is assumed to be NL.
Step 2-Training:
  (2.1) Train the first component unit with the training set $\mathbf{X}$ and the corresponding output set $\mathbf{T}$
        and then form the whole output matrix $\mathbf{Y}_1$, by running algorithm 1 with nl = 1
  (2.2) For $nl = 2$ to $NL$ do
    (2.2.1) Generate the augmented input space
                $\mathbf{X}_{nl} = [\mathbf{X}, \mathbf{Y}_1, \mathbf{Y}_2, \ldots, \mathbf{Y}_{nl-1}]$    (26)
            and present it as the input to the nlth component unit
    (2.2.2) Run algorithm 1 for the $nl$th component unit and obtain the corresponding output $\mathbf{Y}_{nl}$
    (2.2.3) If ($\|\mathbf{Y}_{nl} - \mathbf{Y}_{nl-1}\|_2^2 \leq \varepsilon$), then terminate the training process
            (i.e., exit loop and go to step 2.4) where $\varepsilon$ is an arbitrarily small positive constant
            ($\varepsilon = 1.0$ E $- 4$ in our experiments)
    (2.2.4) Else set $nl = nl + 1$.
  (2.3) end for
  (2.4) Output the structure of SHFA-TSK-FC with its fuzzy rules and parameters in every component unit.
```

FURIA & C4.5. KEEL (Knowledge Extraction based on Evolutionary Learning) is a free software (GPLv3) Java suite which empowers the user to assess the behavior of evolutionary learning and soft computing based techniques for different kind of data mining problems: regression, classification, clustering, pattern mining and so on. KEEL software toolbox can be downloaded from *http:www.keel.es/download.php* The adopted datasets are ten binary-class UCI datasets in Table I [38] and ten multi-class UCI datasets [38] in Table VI and one large dataset, i.e., the *Airline* dataset [43]. For the sake of the space of the paper, more details about them are omitted and can be readily seen from their respective webpages [38]. The *Airline* dataset consists of flight arrival and departure details for all commercial flights with the USA, from October 1987 to April 2008 [43]. This is a large dataset. There are nearly 120 million records in total, and takes up 1.6 gigabytes of space compressed and 12 gigabytes [43]. We have taken four years data from the year 1990 to 1993. In our experiments, all the datasets are normalized. We take seventy-five percent of the samples of each dataset for training and the remaining part of each dataset for testing. We use both classification accuracy and training/testing time as

TABLE I
ON ELEVEN BINARY-CLASS DATASETS

Datasets	No. of training samples	No. of testing samples	No. of features	No. of classes
Balloons (BAL)	57	19	5	2
Liver (LIV)	259	86	6	2
Climate-Model-Simulation-Crashes (CLI)	405	135	21	2
Wdbc (WDB)	427	142	15	2
Blood-transfusion (BLO)	561	187	5	2
Diabetes (DIA)	576	192	8	2
Seismic-bumps (SEI)	1938	646	19	2
Mushroom (MUS)	3047	1015	22	2
Magic04 (MAG)	14265	4755	11	2
Adult (ADU)	36631	12210	15	2
Airline (AIR)	300000	100000	29	2

the performance indices to evaluate the performance of all the comparison methods, in which the classification accuracy is defined as the ratio of the number of samples correctly classified to the total number of samples. All experiments are carried out on a computer with E5-2609 v2 2.5 GHZ CPU (2 processors) with 64 GB memory.

A. *On Binary-Class Datasets*

To observe the performance of SHFA-TSK-FC, ten UCI binary-class datasets and the *Airline* dataset [43] are adopted, as listed in Table I. As we know well, although various classifiers have been developed, for example, BP neural networks and support vector machines, we prefer the commonly used zero-order and first-order TSK fuzzy classifiers [1], [2] as the comparison methods, since both classification accuracy and interpretability can be simultaneously observed from them while other classifiers such as SVM and BP neural networks behave like black boxes. Below we state their respective parameter settings of these three classifiers. Because both FCM [39] and SVM [39] are involved in both zero-order and first-order TSK fuzzy classifiers, the regularization parameter in SVM is set by grid search from 0.01 to 100 with the interval being 0.1 and the number of clusters in FCM is assumed to be equivalent to the number of fuzzy rules and the value of the scale parameter r can be set by grid search from 0.01 to 100 with the interval being 0.1. For classifiers FURIA and C4.5, all the parameters take their respective default values in KEEL software toolbox. In SHFA-TSK-FC, the number of fuzzy rules in each component unit ranges from 2 to 4 with the interval being 1 for the *Balloons* dataset; from 1 to 5 with the interval being 1 for the *Liver* dataset; from 3 to 5 with the interval being 1 for the *Climate-Model-Simulation-Crashes* dataset; from 3 to 10 with the interval being 1 for the *Wdbc* dataset; from 2 to 8 with the interval being 1 for the *Blood-transfusion* dataset; from 3 to 5 with the interval being 1 for the *Diabetes* dataset; from 10 to 40 with the interval being 5 for the *Seismic-bumps* dataset; from 2 to 20 with the interval being 1 for the *Mushroom* dataset; from 5 to 30 with the interval being 5 for the *Magic04* dataset, from 10 to 250 with the interval being 10 for the *Adult* dataset and from 150 to 400 with the interval being 50 for the *Airline* dataset.

Due to the use of random selection for both input features and fuzzy membership functions, we may have multiple choices for the structure of SHFA-TSK-FC on a dataset. In order to demonstrate the behavior of SHFA-TSK-FC fairly, here we organize its experimental results from three aspects:

1) the structures of SHFA-TSK-FC under the best training accuracies on these eleven datasets. This is to demonstrate the interpretability of SHFA-TSK-FC.
2) the average number of fuzzy rules, average training/testing classification accuracy and average training/testing time for 10 trials on each dataset. This is to demonstrate the average performance of SHFA-TSK-FC.
3) the performance change of SHFA-TSK-FC with the increase of the number of augmented features. This is to demonstrate how the change of the stacked structures has impact on the performance of SHFA-TSK-FC.

First, SHFA-TSK-FC obtains the corresponding best training accuracies for these eleven datasets, i.e., 98.94%, 92.57%, 99.17%, 80.55%, 78.70%, 79.77%, 94.40%, 96.97%, 90.41%, 82.05% and 62.41%. The corresponding structure of SHFA-TSK-FC are 3-2, 5-4-3-1-1, 5-4-3, 8-7-3, 4-3-2, 5-4-3, 40-35-30-25-10, 8-5-3-2, 30-25-20-5, 150-120-80-10 and 400-300-200, respectively for these eleven datasets. Please note, here we denote the structure of SHFA-TSK-FC by the number of fuzzy rules in the first component unit-the number of fuzzy rules in the second component unit-. . . -the number of fuzzy rules in the last component unit. For example, 3-2 means that SHFA-TSK-FC consists of two component units, and 3 and 2 fuzzy rules are taken respectively in the first and second component units.

Next, let us observe the average performance of SHFA-TSK-FC on these eleven datasets. With the same number of the component units for each dataset, we run SHFA-TSK-FC ten times by slightly changing the number of fuzzy rules at each component unit ten times, then present the experimental results about average number of fuzzy rules, average training/testing classification accuracy and average training time and the testing time (in seconds) in standard deviation, which are denoted by "*mean(standard deviation)*" in the corresponding Tables II and III in which — means that the corresponding classifiers do not work any more or run extremely slowly (more than 4 hours) in our experimental surrounding, or that the corresponding classifiers (i.e., FURIA and C4.5) run in Java rather than MATLAB, for 10 trials for each classifier on each dataset. We notice that SHFA-TSK-FC performs better than both zero-order and first-order TSK fuzzy classifiers and achieves the best average training accuracies of 98.25%, 98.83%, 78.38%, 79.51% and 96.88%, and the best average testing accuracies of 68.42%, 97.18%, 71.59%, 74.48% and 95.62%, respectively, for five datasets from Table II. FURIA is the best for only one dataset in the sense of both training accuracy and testing accuracy, and C4.5 demonstrates its superiority over SHFA-TSK-FC only on the datasets *SEI* and *ADU*. According to Table II, we find that the number of rules determined from the leaves generated by C4.5 is significantly more than those of other classifiers in most cases.

TABLE II
AVERAGE NUMBER OF FUZZY RULES AND/OR AVERAGE CLASSIFICATION ACCURACIES (%) OF FIVE CLASSIFIERS ON ELEVEN DATASETS

Datasets	zero-order TSK fuzzy classifier			first-order TSK fuzzy classifier			FURIA			C4.5			SHFA-TSK-FC		
	Rules	Training	Testing	Rules	Training	Testing	Rules	Training	Testing	Leaves	Training	Testing	Rules	Training	Testing
BAL	2.50	43.86 (1.24)	52.63 (3.72)	2.50	49.12 (7.44)	**68.42** (0)	3.00	74.36 (0.04)	**68.42** (0.05)	2	77.63 (0.01)	61.84 (0.01)	3.75	**98.25** (4.96)	**68.42** (3.72)
CLI	5.00	91.60 (0.69)	91.11 (2.09)	4.50	98.27 (0)	**92.59** (1.57)	7.50	**99.07** (0)	91.85 (0.02)	10	**99.07** (0)	91.66 (0.02)	15.75	92.35 (0.41)	91.85 (1.28)
WDB	4.50	84.07 (0.99)	83.80 (2.99)	4.20	84.54 (1.32)	80.99 (0)	7.00	94.44 (0)	77.27 (0.03)	11	97.97 (0.01)	69.69 (0)	9.25	**98.83** (0.33)	**97.18** (0.31)
BLO	5.75	78.25 (0)	75.72 (1.89)	6.00	80.11 (0.05)	79.69 (0.81)	5.25	**82.56** (0)	**82.21** (0.03)	6	81.95 (0.01)	81.86 (0.02)	16.25	80.23 (1.02)	79.91 (5.45)
LIV	4.50	67.95 (0.54)	58.14 (1.64)	4.50	72.31 (6.53)	71.43 (6.73)	10.20	66.73 (0.05)	61.71 (0.02)	14	62.00 (0.07)	60.75 (0.05)	9.00	**78.38** (1.64)	**71.59** (4.58)
DIA	5.50	65.28 (1.96)	66.67 (4.12)	5.25	73.61 (2.45)	72.92 (2.94)	4.50	77.52 (0.04)	71.95 (0.04)	19	78.85 (0.01)	72.98 (0)	11.75	**79.51** (1.78)	**74.48** (3.33)
SEI	21.50	93.71 (0.01)	92.77 (0.01)	15.25	93.94 (0.02)	92.94 (0.01)	25.25	95.65 (0.01)	95.45 (0.04)	75	**96.62** (0.01)	**96.62** (0.02)	120.50	94.31 (0.46)	93.96 (0.78)
MUS	25.25	91.51 (2.58)	89.61 (1.56)	21.00	95.66 (0.02)	93.11 (0.41)	20.00	97.43 (7.38)	97.43 (7.38)	15	96.43 (7.38)	95.43 (7.38)	12.75	**96.88** (0.76)	**95.62** (1.02)
MAG	70.25	74.34 (0.25)	72.89 (1.03)	50.20	75.01 (0.01)	74.91 (0.02)	25.50	85.88 (5.78)	**84.61** (0)	206	**90.99** (5.26)	84.44 (0.02)	75.25	89.57 (3.53)	75.15 (7.92)
ADU	175.75	76.27 (0.15)	74.24 (1.29)	152.50	79.87 (1.47)	76.25 (0.82)	177.75	83.39 (0)	83.30 (0)	746	**87.74** (0.03)	**85.53** (3.75)	360.00	80.61 (1.90)	80.29 (2.08)
AIR	---	---	---	---	---	---	8	60.82 (0.01)	60.35 (0.01)	1834	77.29 (0.01)	68.26 (3.29)	750.00	**61.25** (1.13)	**61.27** (2.24)
Mean	32.05	76.68 (0.84)	75.76 (2.03)	26.59	80.24 (1.93)	80.32 (1.33)	28.60	85.70 (1.33)	81.42 (0.76)	267.09	86.93 (1.28)	80.08 (1.13)	125.84	**88.89** (1.68)	**82.85** (3.05)

TABLE III
AVERAGE TRAINING TIME AND TESTING TIME OF FIVE CLASSIFIERS ON ELEVEN DATASETS

Datasets	zero-order TSK fuzzy classifier		first-order TSK fuzzy classifier		FURIA		C4.5		SHFA-TSK-FC	
	Training	Testing	Training	Testing	Training	Testing	Training	Testing	Training	Testing
BAL	0.05 (0.01)	0	0.06 (0.01)	0	—	—	—	—	0.06 (0.01)	0.01
CLI	1.09 (0.09)	0.02	6.56 (0.57)	0.08	—	—	—	—	2.13 (0.62)	0.06
WDB	0.70 (0.06)	0.11	0.64 (0.03)	0.11	—	—	—	—	1.97 (0.23)	0.15
BLO	0.45 (0.02)	0.03	0.98 (0.08)	0.06	—	—	—	—	0.98 (0.05)	0.04
LIV	2.44 (0.09)	0.10	5.52 (0.27)	0.03	—	—	—	—	0.63 (0.08)	0.01
DIA	4.53 (0.33)	0.12	24.83 (0.36)	0.04	—	—	—	—	1.76 (0.86)	0.11
SEI	13.54 (0.23)	0.02	14.64 (0.17)	0.12	—	—	—	—	12.90 (0.78)	1.45
MUS	31.01 (5.42)	0.25	42.69 (1.81)	0.48	—	—	—	—	70.87 (2.57)	1.87
MAG	83.07 (1.18)	2.14	5.35e+03 (29.98)	46.98	—	—	—	—	287.07 (33.63)	58.93
ADU	276.57 (13.64)	8.11	1.54e+04 (74.25)	74.23	—	—	—	—	1.73e+03 (33.68)	252.55
AIR	—	—	—	—	—	—	—	—	1.39e+04 (57.85)	2.10e+03
Mean	41.35 (2.11)	1.09	2084.59 (10.69)	12.21	—	—	—	—	210.84 (9.67)	31.52

On the other hand, SHFA-TSK-FC indeed wins zero-order and first-order TSK fuzzy classifiers on five datasets in the sense of both average training and testing accuracy. Moreover, SHFA-TSK-FC is better than zero-order and first-order TSK fuzzy classifiers in the sense of the average testing accuracy in most cases, which implies SHFA-TSK-FC has promising generalization performance and more concise interpretability by means of its fuzzy rules. By means of its simple structure of SHFA-TSK-FC, we also find that SHFA-TSK-FC runs faster than or at most comparably to first-order TSK fuzzy classifier but slower than zero-order TSK fuzzy classifier in most cases, as illustrated in Table III.

Now, let us observe the performance change of SHFA-TSK-FC with the increase of the number of augmented features. Table IV lists the corresponding average training and testing accuracy for 10 trials for each dataset. As pointed out in the third remark of Section III-B, our extensive experiments reveal that SHFA-TSK-FC can achieve satisfactory performance in most cases through feature augmentation with a range from to 2 to 4 (hence the depth of SHFA-TSK-FC is from 3 to 5). In particular, within this range, if the number of the original features in a dataset is comparatively large and the classification accuracy of SHFA-TSK-FC with current augmented features is still quite close to the classification accuracy of the comparison classifiers (i.e., zero-order and first-order TSK fuzzy classifiers), we will report more experimental results of SHFA-TSK-FC with more than current augmented features.

For example, there are 5 features in the *Blood-transfusion* dataset. With one augmented feature, SHFA-TSK-FC obtains the training accuracy being 78.79%. Since 78.79% is close to 80.11% obtained by first-order TSK fuzzy classifier and is larger than 78.25% obtained by zero-order TSK fuzzy classifier, the second augmented feature may be considered to further observe SHFA-TSK-FC. SHFA-TSK-FC accordingly obtains the training accuracy being 80.23%.

Since 80.23% is bigger than 78.79%, we continue to observe SHFA-TSK-FC with the third augmented feature and then obtain the training accuracy being 80.25% which implies that SHFA-TSK-FC with three augmented features does not have an obvious increase in the training accuracy and that SHFA-TSK-FC at this time achieves obvious superiority over zero-order and first-order TSK fuzzy classifiers. Another example deals with the *Mushroom* dataset in which 22 features are involved. With the first augmented feature, SHFA-TSK-FC obtains the training

TABLE IV
AVERAGE TRAINING ACCURACIES AND TESTING ACCURACIES (%) OF SHFA-TSK-FC WITH DIFFERENT AUGMENTED FEATURES ON ELEVEN DATASETS

Datasets	The first augmented feature		The second augmented feature		The third augmented feature		The fourth augmented feature		The fifth augmented feature		The sixth augmented feature	
	Training	Testing	Training	Testing	Training	Testing	Training	Testing	Training	Testing	Training	Testing
BAL	91.23 (4.56)	63.16 (4.02)	98.25 (4.96)	68.42 (3.72)	---	---	---	---	---	---	---	---
CLI	91.36 (0.19)	88.89 (1.01)	92.10 (0.29)	91.85 (1.10)	91.60 (0.30)	91.11 (1.12)	92.10 (0.62)	91.85 (1.09)	92.35 (0.41)	91.85 (1.28)	---	---
WDB	98.36 (0.29)	95.77 (0.39)	98.83 (0.21)	96.48 (0.63)	98.83 (0.33)	97.18 (0.31)	---	---	---	---	---	---
BLO	78.79 (0.99)	72.19 (3.39)	80.23 (1.02)	79.91 (5.45)	80.25 (0.05)	79.77 (0.68)	---	---	---	---	---	---
LIV	76.06 (1.26)	65.12 (3.09)	78.38 (1.64)	71.59 (4.58)	78.41 (1.04)	71.00 (2.03)	---	---	---	---	---	---
DIA	77.26 (1.18)	66.71 (2.19)	77.78 (0.98)	74.48 (2.37)	79.51 (1.78)	74.48 (3.33)	---	---	---	---	---	---
SEI	93.29 (0.50)	92.57 (0.61)	93.81 (0.51)	93.81 (0.66)	93.24 (0.64)	92.26 (0.98)	93.96 (0.55)	93.70 (0.43)	94.31 (0.46)	93.96 (0.78)	---	---
MUS	95.37 (0.59)	96.90 (0.99)	95.88 (0.58)	97.57 (1.08)	96.90 (0.66)	95.47 (1.11)	96.88 (0.76)	95.62 (1.02)	---	---	---	---
MAG	81.29 (2.52)	64.23 (6.97)	83.37 (2.91)	66.20 (5.95)	85.43 (3.07)	75.15 (7.17)	89.57 (3.53)	75.15 (7.92)	---	---	---	---
ADU	76.83 (1.11)	76.76 (0.88)	80.12 (0.91)	80.65 (1.24)	76.92 (2.27)	76.99 (1.53)	80.61 (1.90)	80.29 (2.08)	---	---	---	---
AIR	60.58 (1.17)	59.71 (0.98)	61.01 (0.92)	60.74 (1.02)	61.25 (1.18)	61.27 (6.52)	---	---	---	---	---	---

TABLE V
AVERAGE TRAINING TIME AND TESTING TIME OF SHFA-TSK-FC WITH DIFFERENT AUGMENTED FEATURES ON ELEVEN DATASETS

Datasets	The first augmented feature		The second augmented feature		The third augmented feature		The fourth augmented feature		The fifth augmented feature		The sixth augmented feature	
	Training	Testing	Training	Testing	Training	Testing	Training	Testing	Training	Testing	Training	Testing
BAL	0.05 (0.01)	0.01	0.06 (0.01)	0.01	---	---	---	---	---	---	---	---
CLI	1.79 (0.17)	0.06	2.01 (0.39)	0.06	2.31 (0.44)	0.09	2.13 (0.56)	0.09	2.13 (0.62)	0.06	---	---
WDB	1.91 (0.17)	0.09	1.99 (0.29)	0.14	1.97 (0.23)	0.15	---	---	---	---	---	---
BLO	0.91 (0.02)	0.04	0.98 (0.05)	0.04	1.02 (0.02)	0.04	---	---	---	---	---	---
LIV	0.51 (0.12)	0.01	0.63 (0.08)	0.01	0.74 (0.08)	0.01	---	---	---	---	---	---
DIA	1.53 (0.79)	0.11	1.71 (0.61)	0.14	1.76 (0.86)	0.11	---	---	---	---	---	---
SEI	12.45 (0.77)	1.45	13.58 (1.24)	1.62	13.96 (1.71)	1.67	14.44 (0.59)	1.73	12.90 (0.78)	1.87	---	---
MUS	67.25 (2.36)	1.87	71.79 (2.08)	1.99	72.38 (3.34)	1.89	70.87 (2.57)	1.87	---	---	---	---
MAG	251.36 (44.59)	50.85	269.69 (40.89)	52.62	277.27 (30.46)	55.06	287.07 (33.03)	58.93	---	---	---	---
ADU	1.31e+03 (37.80)	252.98	1.37e+03 (31.09)	249.67	1.39e+03 (44.26)	243.47	1.73e+03 (33.68)	252.55	---	---	---	---
AIR	6.09e+03 (39.71)	1.02e+03	6.32e+03 (30.28)	1.75e+03	1.39e+04 (57.85)	2.10e+03	---	---	---	---	---	---

accuracy which is very close to those obtained by first-order TSK fuzzy classifier but are better than those obtained by zero-order TSK fuzzy system. Therefore, we continue to observe SHFA-TSK-FC with the second augmented feature, and it obtains the training accuracy and testing accuracy which are much better than those obtained by zero-order and first-order TSK fuzzy classifiers. However, when SHFA-TSK-FC works with the third and fourth features, the obtained training and testing accuracies actually keep stable. In such cases, we do not go ahead to run SHFA-TSK-FC with more augmented features, and hence we do not report the corresponding experimental results.

According to Table V, we can find that SHFA-TSK-FC actually runs faster than first-order TSK fuzzy classifier but slower than zero-order TSK fuzzy classifier. In general, the training time and the testing time of SHFA-TSK-FC increase constantly with the increase of the number of augmented features. Please note, these experimental results and the concise interpretability of fuzzy rules in SHFA-TSK-FC show that SHFA-TSK-FC is a good choice for these binary-class datasets. However, since the number of augmented features actually has an impact on the classification accuracy, how to determine an appropriate number of augmented features for SHFA-TSK-FC on each dataset is still an interesting future research topic.

TABLE VI
ON TEN MULTI-CLASS DATASETS

Datasets	No. of training samples	No. of testing samples	No. of features	No. of classes
Iris (IRI)	113	37	4	3
Wine (WIN)	134	44	13	3
Page-blocks (PAG)	4105	1368	11	5
Winequality (WIQ)	3674	1224	12	7
Balance-scale (BAS)	469	156	5	3
Abalone (ABA)	3133	1044	9	3
Contraceptive-Method-Choice (CON)	1105	368	10	3
Vehicle (VEH)	635	211	19	4
Yeast (YEA)	1113	371	9	10
Car-Evaluation (CAR)	1296	432	7	4

TABLE VII
AVERAGE NUMBER OF FUZZY RULES AND/OR AVERAGE CLASSIFICATION ACCURACIES (%) OF FIVE CLASSIFIERS ON TEN MULTI-CLASS DATASETS

Datasets	zero-order TSK fuzzy classifier			first-order TSK fuzzy classifier			FURIA			C4.5			SHFA-TSK-FC		
	Rules	Training	Testing	Rules	Training	Testing	Rules	Training	Testing	Leaves	Training	Testing	Rules	Training	Testing
IRI	1.50	96.46 (0.62)	91.89 (1.91)	**1.25**	97.34 (0.01)	94.29 (0.21)	3.50	96.66 (0)	94.66 (0)	3	98.00 (0)	93.33 (0.01)	3.75	**98.23** (0.62)	**97.30** (1.35)
WIN	1.75	84.32 (1.06)	81.82 (1.60)	1.50	90.91 (1.28)	88.17 (3.04)	5.00	**99.43** (0)	**94.38** (0.03)	5	97.75 (0.02)	87.64 (0)	5.75	87.31 (2.23)	84.09 (2.27)
PAG	15.50	90.42 (0.36)	89.93 (0.19)	14.25	93.32 (1.10)	92.10 (1.96)	24.50	**98.81** (9.13)	97.07 (0.01)	33	98.63 (5.48)	**97.24** (9.14)	25.00	91.25 (0.33)	91.30 (1.88)
WIQ	17.25	47.55 (0.22)	45.26 (1.52)	15.50	51.96 (0.62)	50.00 (3.23)	24.00	58.92 (0)	52.69 (0)	404	**89.75** (0)	**53.67** (0)	45.00	53.21 (0.25)	50.41 (2.12)
BAS	15.25	56.08 (6.63)	50.64 (4.08)	12.00	91.47 (0.90)	88.46 (3.17)	16.50	88.95 (0)	83.36 (0.01)	25	89.91 (0.01)	78.40 (0.01)	45.25	**95.52** (1.40)	**91.67** (2.25)
ABA	20.50	55.54 (0.75)	52.96 (0.25)	15.50	56.43 (0.23)	54.11 (2.72)	17.50	56.21 (0)	54.59 (0.02)	174	56.77 (0.01)	56.01 (0.01)	50.00	**56.81** (0.69)	**56.32** (4.50)
CON	5.50	51.58 (0.70)	49.45 (0)	4.25	64.79 (0.25)	55.16 (2.69)	7.50	56.14 (2.97)	54.31 (0.01)	91	**75.15** (0.01)	49.35 (0.02)	15.75	60.00 (1.88)	**57.07** (1.13)
VEH	17.80	61.10 (1.67)	58.29 (0.34)	11.25	**100** (0)	70.14 (4.35)	23.00	86.05 (0.04)	71.15 (0)	31	87.82 (0.01)	69.38 (0.02)	34.00	71.34 (0.23)	**73.46** (6.03)
YEA	16.00	52.85 (0.13)	48.64 (2.50)	12.25	64.43 (1.21)	56.52 (3.66)	16.50	64.01 (0.03)	58.89 (0.04)	74	**79.31** (0.01)	55.86 (0.02)	30.00	69.00 (2.61)	**61.46** (2.38)
CAR	25.50	71.43 (1.20)	70.83 (3.60)	21.25	70.42 (0.38)	73.15 (0.16)	48.00	94.67 (0)	88.19 (0)	87	94.50 (0.01)	87.33 (0.01)	33.00	**96.45** (2.24)	**94.68** (1.96)
Mean	13.66	66.73 (1.33)	63.97 (1.60)	10.9	78.11 (0.59)	72.21 (2.52)	18.60	79.99 (1.22)	74.93 (0.01)	92.7	**86.76** (0.62)	72.82 (0.92)	28.75	77.91 (1.25)	**75.78** (2.59)

TABLE VIII
AVERAGE TRAINING TIME AND TESTING TIME OF FIVE CLASSIFIERS ON TEN MULTI-CLASS DATASETS

Datasets	zero-order TSK fuzzy classifier		first-order TSK fuzzy classifier		FURIA		C4.5		SHFA-TSK-FC	
	Training	Testing	Training	Testing	Training	Testing	Training	Testing	Training	Testing
IRI	**0.10** (0.01)	0.04	0.14 (0.03)	0.03	—	—	—	—	0.40 (0.01)	**0**
WIN	2.49 (0.41)	**0.01**	2.90 (0.05)	0.12	—	—	—	—	**0.73** (0.06)	**0.01**
PAG	**22.53** (0.50)	**0.09**	50.41 (0.76)	0.24	—	—	—	—	53.98 (1.90)	8.26
WIQ	**24.39** (0.84)	**0.29**	270.17 (1.15)	2.95	—	—	—	—	45.98 (1.42)	6.69
BAS	0.16 (0)	0.03	**0.14** (0)	**0**	—	—	—	—	0.56 (0.02)	0.06
ABA	**5.27** (0.58)	**0.57**	36.13 (0.60)	4.56	—	—	—	—	26.92 (1.45)	4.14
CON	**2.84** (0.57)	**0.06**	8.20 (0.28)	0.5	—	—	—	—	3.33 (0.04)	0.34
VEH	**1.47** (0.02)	**0.03**	41.41 (13.51)	0.17	—	—	—	—	7.07 (0.85)	0.17
YEA	**2.41** (0.50)	**0.07**	8.70 (0.35)	0.45	—	—	—	—	7.11 (0.31)	0.39
CAR	**0.38** (0.01)	**0.06**	2.23 (0.36)	0.26	—	—	—	—	2.09 (0.35)	0.25
Mean	**6.20** (0.34)	**0.13**	42.04 (1.71)	0.93	—	—	—	—	14.82 (0.64)	2.03

B. On Multi-Class Datasets

Table VI lists ten UCI multi-class datasets to further evaluate the classification performance of SHFA-TSK-FC. The same experimental organization and the same parameter settings for zero-order and first-order TSK fuzzy classifiers are adopted as in Section IV-A.

In SHFA-TSK-FC, the number of fuzzy rules ranges from 1 to 10 with the interval being 1 for the *Iris* dataset; from 1 to 15 with the interval being 1 for the *Wine* dataset; from 2 to 30 with the interval being 5 for the *Page-blocks* dataset; from 10 to 25 with the interval being 1 for the *Winequality* dataset; from 5 to 30 with the interval being 1 for the *Balance-scale* dataset; from 10 to 20 with the interval being 1 for the *Abalone* dataset; from 1 to 10 with the interval being 1 for the *Contraceptive-Method-Choice* dataset; from 5 to 10 with the interval being 1 for the *Vehicle* dataset; from 5 to 15 with the interval being 1 for the *Yeast* dataset and from 8 to 15 with the interval being 1 for the *Car-Evaluation* dataset.

Likewise in the last subsection, in order to demonstrate the interpretability of SHFA-TSK-FC on these datasets, here we also report the corresponding structures of SHFA-TSK-FC, i.e., 3-1, 3-2-1, 15-10-2, 20-15-10, 30-20, 20-15-10, 10-7-1, 10-9-8-7, 15-10-5 and 15-10-8, respectively, under the obtained best accuracies, i.e., 98.40%, 87.55%, 91.50%, 53.57%, 95.60%, 56.90%, 60.27%, 71.44%, 69.51% and 96.91%, for these ten datasets.

The average experimental results obtained by the five comparison classifiers are summarized in Tables VII–X. With a careful inspection for these tables, we can readily find that the same claim as in the above subsection still holds for these ten multi-class datasets.

C. Non-Parametric Statistical Analysis

Milton Friedman [42] developed a non-parametric statistical test, i.e., the Friedman ranking test which is used to detect differences across multiple tests. Here we apply Friedman ranking test for multiple comparisons about all the datasets listed in Tables I and VI. The value $\alpha = 0.05$ is used as the level of confidence in this test. Friedman ranking test is used to assess whether some differences exist or not in these multiple comparisons about all the datasets. Fig. 5 shows the ranking results of these five classifiers on both binary and multiple classification tasks in terms of the Friedman test.

Fig. 5 demonstrates the ranking results of the five classifiers with p-value being less than 1.0E-4. Obviously, significant differences indeed exist in all the classifiers, and C4.5 keeps the best ranking while SHFA-TSK-FC keeps the second ranking among

TABLE IX
AVERAGE TRAINING ACCURACIES AND TESTING ACCURACIES (%) OF SHFA-TSK-FC WITH DIFFERENT AUGMENTED FEATURES ON TEN MULTI-CLASS DATASETS

Datasets Datasets	The first augmented feature		The second augmented feature		The third augmented feature		The fourth augmented feature		The fifth augmented feature	
	Training	Testing	Training	Testing	Training	Testing	Training	Testing	Training	Testing
IRI	97.35 (0.78)	94.59 (1.28)	98.23 (0.62)	97.30 (1.35)	—	—	—	—	—	—
WIN	82.84 (2.19)	79.55 (1.56)	85.07 (1.82)	81.82 (1.46)	87.31 (2.23)	84.09 (2.27)	—	—	—	—
PAG	90.60 (0.39)	91.52 (1.01)	90.91 (0.52)	88.16 (1.55)	91.25 (0.33)	91.30 (1.88)	—	—	—	—
WIQ	52.86 (0.49)	46.73 (1.71)	52.72 (0.62)	50.41 (1.87)	53.21 (0.25)	50.41 (2.12)	—	—	—	—
BAS	94.46 (1.51)	93.59 (2.54)	95.52 (1.40)	91.67 (2.25)	—	—	—	—	—	—
ABA	55.47 (0.66)	47.51 (3.50)	56.43 (0.69)	53.54 (3.39)	56.81 (0.69)	56.32 (4.50)	—	—	—	—
CON	60.00 (1.61)	54.89 (1.08)	56.74 (1.46)	56.52 (0.99)	60.00 (1.88)	57.07 (1.13)	—	—	—	—
VEH	71.18 (0.55)	58.29 (2.30)	71.18 (0.71)	59.72 (2.22)	71.50 (0.38)	65.88 (3.92)	71.34 (0.23)	73.46 (6.03)	—	—
YEA	66.22 (2.56)	60.38 (1.75)	67.83 (1.96)	61.19 (2.39)	69.00 (2.61)	61.46 (2.38)	—	—	—	—
CAR	93.28 (2.19)	91.90 (1.84)	96.45 (2.24)	94.68 (1.96)	—	—	—	—	—	—

TABLE X
AVERAGE TRAINING TIME AND TESTING TIME OF SHFA-TSK-FC WITH DIFFERENT AUGMENTED FEATURES ON TEN MULTI-CLASS DATASETS

Datasets	The first augmented feature		The second augmented feature		The third augmented feature		The fourth augmented feature		The fifth augmented feature	
	Training	Testing	Training	Testing	Training	Testing	Training	Testing	Training	Testing
IRI	97.35 (0.59)	94.59	98.23 (0.62)	97.30	—	—	—	—	—	—
WIN	0.06 (0)	0	0.06 (0)	0	0.06 (0)	0	—	—	—	—
PAG	50.19 (1.19)	8.17	52.27 (1.82)	8.42	53.98 (1.90)	8.26	—	—	—	—
WIQ	43.31 (1.28)	6.62	43.82 (1.49)	6.63	45.98 (1.42)	6.69	—	—	—	—
BAL	0.47 (0.02)	0.06	0.56 (0.02)	0.06	—	—	—	—	—	—
ABA	21.03 (1.14)	2.70	25.38 (1.20)	4.15	26.93 (1.45)	4.14	—	—	—	—
CON	3.29 (0.01)	0.25	3.37 (0.01)	0.27	3.33 (0.04)	0.34	—	—	—	—
VEH	5.30 (0.60)	0.03	5.05 (0.20)	0.03	6.13 (0.55)	0.03	7.06 (0.85)	0.05	—	—
YEA	6.67 (0.27)	0.44	6.94 (0.43)	0.48	7.11 (0.31)	0.39	—	—	—	—
CAR	2.00 (0.10)	0.17	2.09 (0.35)	0.25	—	—	—	—	—	—

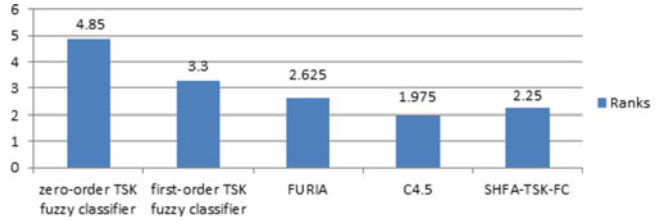

Fig. 5. Rankings of these five classifiers.

TABLE XI
HOLM POST-HOC TEST RESULTS FOR SHFA-TSK-FC VS. ZERO-ORDER AND FIRST-ORDER TSK FUZZY CLASSIFIERS AND FURIA WITH $\alpha = 0.05$

i	Classifiers	z	p	*Holm*	Hypothesis
3	zero-order TSK fuzzy classifier	5.3889	0	0.0167	Rejected
2	first-order TSK fuzzy classifier	1.8371	0.0661	0.025	Not Rejected
1	FURIA	0.6124	0.5403	0.05	Not Rejected

these classifiers. Since C4.5 is based on decision tree rather than fuzzy systems, SHFA-TSK-FC actually keeps the first among the remaining four fuzzy classifiers. Next, in order to assess whether some differences exist or not between SHFA-TSK-FC and the other classifiers except C4.5 on all the datasets, we reconduct the Friedman ranking test for all the classifiers except C4.5 and then conduct the corresponding Holm post-hoc test. The results of Holm post-hoc test results are listed in Table XI where Holm post-hoc test rejects the hypothesis of equivalence for four classifiers with $p < \alpha/i$. It seems that SHFA-TSK-FC is not better than both the first-order TSK fuzzy classifier and FURIA in this test. However, let us keep in mind that SHFA-TSK-FC has the second ranking among all the classifiers. Therefore, SHFA-TSK-FC is comparable to three fuzzy classifiers in the sense of accuracy. After considering comprehensible immediate outputs and interpretable fuzzy rules in SHFA-TSK-FC, we believe that these results actually indicate that SHFA-TSK-FC has a good trade-off between high accuracy, comprehensible intermediate outputs and interpretable fuzzy rules, therefore it is a favorable choice when a hierarchical fuzzy TSK classifier is desired in practical applications.

D. On Real-World Case: Electricity Pricing Dataset

This experiment on a real-world *Electricity Pricing* dataset [40] is arranged here for illustrating both classification performance and interpretability. It consists of 45312 instances which were collected at regular 30-min intervals during 135 weeks. Although there are nine features in the original dataset, here we use

TABLE XII
THE NUMBER OF FUZZY RULES UNDER BEST ACCURACIES (%) OF FIVE CLASSIFIERS ON THE *Electricity Pricing* DATASET

Zero-order TSK fuzzy classifier			First-order TSK fuzzy classifier			FURIA			C4.5			SHFA-TSK-FC		
Rules	Training	Testing	Rules	Training	Testing	Rules	Training	Testing	Leaves	Training	Testing	Rules	Training	Testing
150	69.43	67.61	120	72.88	71.54	7	73.28	73.12	76	76.88	74.47	87	73.59	71.92

TABLE XIII
THE TRAINING TIME AND TESTING TIME OF THREE CLASSIFIERS ON THE *Electricity Pricing* DATASET

Zero-order TSK fuzzy classifier		First-order TSK fuzzy classifier		FURIA		C4.5		SHFA-TSK-FC	
Training	Testing	Training	Testing	Training	Testing	Training	Testing	Training	Testing
182.70	35.88	8.2469e+03	14.41	—	—	—	—	535.11	95.67

TABLE XIV
RULE DESCRIPTIONS

	IF							THEN output y (class label)
	feature_1	feature_2	feature_3	feature_4	feature_5	feature_Y1	feature_Y2	is
Rule 1	very low	high	medium	Don't care	low	low	very high	0.6832(+1)
Rule 2	low	high	very high	very low	Don't care	medium	very low	−0.3631(−1)
Rule 3	medium	high	very low	Don't care	medium	very high	low	0.7662(+1)
Rule 4	Don't care	low	high	very high	very low	medium	very low	−0.8951(−1)
Rule 5	very low	Don't care	high	medium	low	low	very high	−0.9753(−1)
Rule 6	medium	low	very low	Don't care	low	very high	high	0.3763(+1)
Rule 7	Don't care	very high	high	low	medium	low	very low	0.9269(+1)
Rule 8	very high	low	high	very high	Don't care	very low	medium	−0.9928(−1)

five features, called feature_1, feature_2, feature_3, feature_4 and feature_5 in this experiment, according to [40].The first two features date the record in day of week (1–7) and half-hour period (1–48). The current demands which consist of demand in New South Wales and demand in Victoria are measured in last three features. The classification label is a binary value which indicates that the price of electricity will go up (denoted as +1) or down (denoted as −1). Because the values of one or two features in the first 17760 instances downloaded are incomplete, we process experimental data on the later 27552 instances (i.e., 82 weeks from the 54th batch to the 135th batch). Thus the performance of the proposed hierarchical TSK fuzzy classifier SHFA-TSK-FC is evaluated on the later 27552 instances in the *Electricity Pricing* dataset. For comparison, here we adopt the same experimental organization and the same parameter settings for zero-order TSK and first-order TSK fuzzy classifiers as in the above subsection. In order to design the proposed classifier SHFA-TSK-FC on the *Electricity Pricing* dataset, we determine an appropriate number of fuzzy rules ranging from 50 to 270 in this experiment. Because the interpretability of SHFA-TSK-FC deals with both its corresponding structure and fuzzy rules of SHFA-TSK-FC, here we illustrate its interpretability by reporting the obtained SHFA-TSK-FC for the best accuracy on this dataset. In terms of our experiment, the final structure of SHFA-TSK-FC for the best accuracy on this dataset is 73.59%, and the corresponding structure of SHFA-TSK-FC is 150-30-5, respectively for the *Electricity Pricing* datasets. The experimental results about accuracy, running time obtained by the three comparison classifiers are summarized in Tables XII and XIII in which "—" means that the corresponding classifiers (i.e., FURIA and C4.5) run in Java instead of MATLAB such that we do not compare the running time between them and SHFA-TSK-FC. From Table XII, it is noticeable that SHFA-TSK-FC indeed outperforms both zero-order and first-order TSK fuzzy classifiers, achieving the best training accuracy of 73.59% for the *Electricity Pricing* dataset. In terms of the training time listed in Table XIII, we can see that due to its simpler structure, SHFA-TSK-FC always runs more quickly than first-order TSK fuzzy classifier but more slowly than zero-order TSK fuzzy classifier. What is more, as for easy visible observation for the interpretability of fuzzy rules, we pick up eight rules among all the fuzzy rules obtained by SHFA-TSK-FC and then summarize them in Table XIV in which "*Don't care*" means that the corresponding feature is not selected in the corresponding fuzzy rule. As an example, we can easily express Rule 1 in Table XIV as

$$\begin{aligned}
&IF \quad feature\_1\, is\, very\, low\\
&\quad AND\; feature\_2\, is\, high\\
&\quad AND\; feature\_3\, is\; Don't\; care\\
&\quad AND\; feature\_4\, is\, low\\
&\quad AND\; feature\_5\\
&\quad AND\; feature\_Yl\, is\, low\\
&\quad AND\; feature\_Yl\, is\, low\\
&THEN \quad y = 0.6832(+1)
\end{aligned}$$

where *feature_Y1* and *feature_Y2* are two augmented features which express the outputs of the first and second component units of SHFA-TSK-FC, respectively. Obviously, such fuzzy rules have high interpretability.

V. Conclusion

By using the stacked generalization principle, the least learning machine and feature augmentation trick, a new stacked-structure-based hierarchical TSK fuzzy classifier called SHFA-TSK-FC has been developed in this study. SHFA-TSK-FC consists of zero-order TSK fuzzy classifiers as component units. Unlike in the existing hierarchical fuzzy classifiers, each component unit in SHFA-TSK-FC is organized in a stacked way such that the current component unit is fed with all the input features of the original training samples plus the interpretable augmented features, which correspond to the interpretable output of each previous component unit and can indeed open the manifold structure of the original input space such that the enhanced classification performance may be expected. With the help of the least learning machine, each component unit can have a fast analytical solution to the consequent parts of fuzzy rules, which indeed results in the very scalability of SHFA-TSK-FC. Each component unit and hence SHFA-TSK-FC achieves high interpretability by randomly selecting the input features and randomly choosing the fixed five Gaussian membership functions for the selected input features in the premise of each fuzzy rule. Our extensive experimental results on real-life datasets and an application case have indicated the power of SHFA-TSK-FC in the sense of both the enhanced or at least comparable classification performance and high interpretability.

There exists a large room worthy to be studied in the future. For example, an interesting topic may be to explore other feasible stacked structures for hierarchical fuzzy classifiers. What is more, for concrete and practical application scenarios, how to appropriately set several the parameters (i.e., α, C, the number of fuzzy rules in each component unit and the depth of SHFA-TSK-FC) is still an open topic, which is our on-going work.

References

[1] Z. Deng, C. Kup-Sze, F.-L. Chung, and S. Wang, "Scalable TSK fuzzy modeling for very large datasets using minimal-enclosing-ball approximation," *IEEE Trans. Fuzzy Syst.*, vol. 19, no. 2, pp. 210–226, Apr. 2011.

[2] Y. Jiang, F.-L. Chung, H. Ishibuchi, Z. Deng, and S. Wang, "Multitask TSK fuzzy system modeling by mining intertask common hidden structure," *IEEE Trans. Cybern.*, vol. 45, no. 3, pp. 534–547, Mar. 2015.

[3] Y. Zheng, H. Ling, S. Chen, and J. Xue, "A hybrid neuro-fuzzy network based on differential biogeography-based optimization for online population classification in earthquakes," *IEEE Trans. Fuzzy Syst.*, vol. 23, no. 4, pp. 1070–1083, Aug. 2014.

[4] H. Ishibuchi and Y. Nojima, "Repeated double cross-validation for choosing a single solution in evolutionary multi-objective fuzzy classifier design," *Knowl.-Based Syst.* vol. 54, pp. 22–31, 2013.

[5] Z. Deng, L. Cao, Y. Jiang, and S. Wang, "Minimax probability TSK fuzzy system classifier: A more transparent and highly interpretable classification model," *IEEE Trans. Fuzzy Syst.*, vol. 23, no. 4, pp. 813–826, Apr. 2015.

[6] O. Guenounou, B. Dahhou, and F. Chabour, "TSK fuzzy model with minimal parameters," *Appl. Soft Comput.*, vol. 30, no. 2, pp. 748–757, Jan. 2015.

[7] Y. Nojima and H. Ishibuchi, "Multiobjective fuzzy genetics-based machine learning with a reject option," in *Proc. IEEE Int. Conf. Fuzzy Syst.*, Nov. 2016, pp. 1405–1412, doi: 10.1109/FUZZ-IEEE.2016.7737854.

[8] P. C. Chang and C. Y. Fan, "A hybrid system integrating a wavelet and TSK fuzzy rules for stock price forecasting," *IEEE Trans. Syst., Man, Cybern., Part C*, vol. 38, no. 6, pp. 802–815, Nov. 2008.

[9] H. Ishibuchi, S. Mihara, and Y. Nojima, "Parallel distributed hybrid fuzzy GBML models with rule set migration and training data rotation," *IEEE Trans. Fuzzy Syst.*, vol. 21, no. 2, pp. 355–368, Apr. 2013.

[10] C. Yang, Z. Deng, K.-S. Choi, and S. Wang, "Takagi-Sugeno-Kang transfer learning fuzzy logic system for the adaptive recognition of epileptic electroencephalogram signals," *IEEE Trans. Fuzzy Syst.*, vol. 24, no. 5, pp. 1079–1094, Oct. 2016.

[11] M. Joo Er and S. Mandal, "A survey of adaptive fuzzy controllers: Nonlinearities and classifications," *IEEE Trans. Fuzzy Syst.*, vol. 24, no. 5, pp. 1095–1107, Oct. 2016.

[12] Y.-J. Liu, S. Tong, D.-J. Li, and Y. Gao, "Fuzzy adaptive control with state observer for a class of nonlinear discrete-time systems with input constraint," *IEEE Trans. Fuzzy Syst.*, vol. 24, no. 5, pp. 1147–1158, Oct. 2016.

[13] T. Pradeep, P. Srinivasu, P. S. Avadhani, and Y. V. S. Murthy, "Comparison of variable learning rate and Levenberg-Marquardt back-propagation training algorithms for detecting attacks in intrusion detection systems," *Int. J. Comput. Sci. Eng.*, vol. 3, no. 11, pp. 676–683, 2016.

[14] Y. Shi and M. Mizumoto, "Some considerations on conventional neuro-fuzzy learning algorithms by gradient descent method," *Fuzzy Sets Syst.*, vol. 112, no. 1, pp. 51–63, 2000.

[15] H. Ishibuchi and Y. Nojima, "Analysis of interpretability-accuracy tradeoff of fuzzy systems by multiobjective fuzzy genetics-based machine learning," *Int. J. Approx. Reason.*, vol. 44, no. 1, pp. 4–31, Jan. 2007.

[16] T. Kanungo, D. M. Mount, N. S. Netanyahu, C. D. Piatko, and R. Silverman, "An efficient k-means clustering algorithm: Analysis and implementation," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 24, no. 7, pp. 881–892, Jul. 2002.

[17] J. C. Bezdek, R. Ehrlich, and W. Full, "FCM: The fuzzy c-means clustering algorithm," *Comput. Geosci.*, vol. 10, no. 2, pp. 191–203, 1984.

[18] T. Chaira, "A novel intuitionistic fuzzy C means clustering algorithm and its application to medical images," *Appl. Soft Comput.*, vol. 11, no. 2, pp. 1711–1717, 2011.

[19] J. Abonyi, R. Babuška, and F. Szeifert, "Modified Gath-Geva fuzzy clustering for identification of Takagi-Sugeno fuzzy models," *IEEE Trans. Syst. Man Cybern. Part B*, vol. 32, no. 5, pp. 612–621, Oct. 2002.

[20] C. Puri and N. Kumar, "Projected Gustafson-Kessel clustering algorithm and its convergence," *Trans. Rough Sets XIV*, vol. 6600, pp. 159–182, 2011.

[21] R. Silipo, G. Bortolan, and C. Marchesi, "Fuzzy preprocessing and artificial neural network classification for the diagnostic interpretation of the resting ECG," *Comput. Cardiology*, no. 10-13, pp. 365–368, Sep. 1995.

[22] L. Zhu, F.-L. Chung, and S. Wang, "Generalized fuzzy C-means clustering algorithm with improved fuzzy partitions," *IEEE Trans. Syst. Man Cybern. Part B*, vol. 39, no. 3, pp. 578–591, Jun. 2009.

[23] L.-X. Wang, "Analysis and design of hierarchical fuzzy systems," *IEEE Trans. Fuzzy Syst.*, vol. 7, no. 5, pp. 617–624, Oct. 1999.

[24] T. Zhou, F.-L. Chung, H. Ishibuchi, and S. Wang, "Stacked blockwise combination of interpretable TSK fuzzy classifiers by negative correlation learning," *IEEE Trans. Fuzzy Syst.*, vol. 25, no. 5, pp. 1207–1221, Oct. 2017.

[25] D. H. Wolpert, "Stacked generalization," *Neural Netw.*, vol. 5, no. 2, pp. 241–259, 1992.

[26] S. Wang, F.-L. Chung, J. Wu, and J. Wang, "Least learning machine and its experimental studies on regression capability," *Appl. Soft Comput.*, vol. 21, pp. 677–684, Aug. 2014.

[27] W. Shitong and F.-L. Chung, “On least learning machine,” *J. Jiangnan Univ.*, vol. 9, pp. 505–510, Feb. 2010.

[28] S. Wang, Y. Jiang, F.-L. Chung, and P. Qian, “Feedforward kernel neural networks, generalized least learning machine, and its deep learning with application to image classification,” *Appli. Soft Comput.*, vol. 37, pp. 125–141, Dec. 2015.

[29] Y. Chen, D. Wang, and S. Tong, “Forecasting studies by designing Mamdani interval type-2 fuzzy logic systems: With the combination of BP algorithms and KM algorithms,” *Neurocomputing*, vol. 174, no. 22, pp. 1133–1146, Jan. 2016.

[30] K. Trawinski, O. Cordon, L. Sanchez, and A. Quirin, “A genetic fuzzy linguistic combination method for fuzzy rule-based multi-classifiers,” *IEEE Trans. Fuzzy Syst.*, vol. 21, no. 5, pp. 950–965, Oct. 2013.

[31] M. L. Lee, H. Chung, and F. Yu, “Modeling of hierarchical fuzzy systems,” *Fuzzy Sets Syst*., vol. 138, pp. 343–361, 2003.

[32] R. Holve, “Rule generation for hierarchical fuzzy systems,” in *Proc. Annu. Meet. North Amer. Fuzzy Info. Process. Soc.*, 2003, pp. 444–449.

[33] C. Peres, R. Guerra, R. Haber, A. Alique, and S. Ros, “Fuzzy model and hierarchical fuzzy control integration: an approach for milling process optimization,” *Comput. Ind.*, vol. 39, pp. 199–207, 1999.

[34] M. G. Joo and J. S. Lee, “Universal approximation by hierarchical fuzzy system with constrains on the fuzzy rule,” *IEEE Trans. Fuzzy Syst.*, vol. 130, no. 2, pp. 175–188, Sep. 2002.

[35] T. Takagi and M. Sugeno, “Fuzzy identification of systems and its application to modeling and control,” *IEEE Trans. Syst., Man, Cybern.*, vol. SMC-15, no. 1, pp. 116–132, Jan. 1985.

[36] L. I. Kuncheva, “How good are fuzzy if-then classifiers?” *IEEE Trans. Syst., Man, Cybern., Part B*, vol. 30, no. 4, pp. 501–509, Aug. 2000.

[37] G. L. Grinblat, L. C. Uzal, H. A. Ceccatto, and P. M. Granitto, “Solving nonstationary classification problems with coupled support vector machines,” *IEEE Trans. Neural Netw.*, vol. 22, no. 1, pp. 37–51, Jan. 2011.

[38] 2007. [Online]. Available: https://archive.ics.uci.edu/ml/datasets.html

[39] C.-C. Chuang, C.-C. Hsiao, and J.-T. Jeng, “Adaptive fuzzy regression clustering algorithm for TSK fuzzy modeling,” in *Proc. IEEE Int. Symp. Comput. Intell. Robot. Autom.*, Jul. 2003, vol. 1, pp. 201–206.

[40] M. Harries, “Splice-2 comparative evaluation: Electricity pricing,” School Comput. Sci. Eng., Univ. New South Wales, Sydney, NSW, Australia, Tech. Rep. NSW-CSE-TR-9905, 1999.

[41] Y. Nojima, Y. Takahashi, and H. Ishibuchi, “Genetic lateral tuning of membership functions as post-processing for hybrid fuzzy genetics-based machine learning,” in *Proc. Joint 7th Int. Conf. Adv. Intell. Syst., 15th Int. Symp. Soft Comput. Intell. Syst.*, Feb. 2015, pp. 667–672, doi: 10.1109/SCIS-ISIS.2014.7044847.

[42] J. Demsar, “Statistical comparisons of classifiers over multiple data sets,” *J. Mach. Learn. Res.*, vol. 7, pp. 1–30, 2006.

[43] 2017. [Online]. Available: http://stat-computing.org/dataexpo/2009/

Ta Zhou is currently working toward the Ph.D. degree in the School of Digital Media, Jiangnan University, Wuxi, China. He is also a Lecturer in the School of Electrical and Information Engineering, Jiangsu University of Science and Technology, Zhenjiang, China. His main research interests include neuro-fuzzy systems, data mining, pattern recognition, and their applications.

Hisao Ishibuchi (M’93–SM’10–F’14) received the B.S. and M.S. degrees in precision mechanics from Kyoto University, Kyoto, Japan, in 1985 and 1987, respectively, and the Ph.D. degree in computer science from Osaka Prefecture University, Sakai, Osaka, Japan, in 1992. Since 1987, he has been with Osaka Prefecture University. Since April 2017, he is also with Southern University of Science and Technology, Shenzhen, China. His research interests include fuzzy rule-based classifiers, evolutionary multiobjective optimization, memetic algorithms, and evolutionary games. He was the IEEE Computational Intelligence Society Vice-President for Technical Activities for 2010–2013. He is currently the Editor-in-Chief of the IEEE COMPUTATIONAL INTELLIGENCE MAGAZINE (2014–2019).

Shitong Wang received the M.S. degree in computer science from Nanjing University of Aeronautics and Astronautics, Nanjing, China, in 1987. He visited London University and Bristol University in U.K., Hiroshima International University and Osaka Prefecture University in Japan, Hong Kong University of Science and Technology, Hong Kong Polytechnic University, as a Research Scientist, for almost eight years. He is currently a Full Professor in the School of Digital Media, Jiangnan University, China. His research interests include artificial intelligence, neuro-fuzzy systems, pattern recognition, and image processing. He has published about 100 papers in international/national journals and has authored seven books.