A New Approach for Transformation-Based Fuzzy Rule Interpolation

Tianhua Chen, Changjing Shang, Jing Yang, Fangyi Li, and Qiang Shen

***Abstract*—Fuzzy rule interpolation (FRI) is of particular significance for reasoning in the presence of insufficient knowledge or sparse rule bases. As one of the most popular FRI methods, transformation-based fuzzy rule interpolation (TFRI) works by constructing an intermediate fuzzy rule, followed by running scale and move transformations. The process of intermediate rule construction selects a user-defined number of rules closest to an observation that does not match any existing rule, using a distance metric. It relies upon heuristically computed weights to assess the contribution of individual selected rules. This process requires a move operation in an effort to force the intermediate rule to overlap with an unmatched observation, regardless of what rules are selected and how much contribution they may each make. It is, therefore, desirable to avoid this problem and also to improve the automation of rule interpolation without resorting to the user's intervention for fixing the number of closest rules. This article proposes such a novel approach to selecting a subset of rules from the sparse rule base with an embedded rule weighting scheme for the automatic assembling of the intermediate rule. Systematic comparative experimental results are provided on a range of benchmark datasets to demonstrate statistically significant improvement in the performance achieved by the proposed approach over that obtainable using conventional TFRI.**

***Index Terms*—Active set-based solution, adaptive network-based fuzzy inference system (ANFIS), closest rule selection, fuzzy rule interpolation (FRI).**

I. Introduction

Being one of the cornerstones of soft computing, fuzzy set theory enables the tolerance of imprecision, uncertainty, and approximation in data and knowledge, which many problems in real life involve that conventional Boolean representation cannot handle. In particular, fuzzy-rule-based systems [1]–[3] have been very successful in a wide range of real-world applications (e.g., [4]–[7]). In order for such systems to work, a dense fuzzy rule base is normally required to cover the entire input space such that any incoming new observations may at least partially overlap with certain existing rules to derive appropriate consequents. However, there are many problems where knowledge about the domain is rather incomplete so that only a sparse rule base may be available. In this case, an input observation may not match and fire any of the existing rules from the sparse rule base, thereby leading to no conclusion. Fuzzy rule interpolation (FRI) explicitly addresses this common restriction in fuzzy systems, where the given rule base is sparse and unable to fully cover the input space.

A number of important FRI approaches and their variations have been proposed in the literature, which can be generally categorized into two classes. The first performs interpolation by directly manipulating the antecedents of those rules, which are deemed closest to (but do not match) the given observation. The consequent of the interpolated result may, therefore, be viewed as the combined logical outcome of those rules involved. Typical approaches in this group are based on the exploitation of the concept of α-cuts [8]–[12]. The approach works basically by propagating distance measures between the α-cuts of the observed antecedent fuzzy sets and their counterparts in those rules to compute the α-cuts of the interpolated consequent, which are then assembled through the use of the resolution principle [13] to construct the final interpolated outcome.

The second category is based on the application of analogical reasoning (that is, similar observations lead to similar consequents) [14], [15]. Such an approach first creates an intermediate rule with its antecedents constructed under the guidance of the given observation. It then imposes the similarity measure computed between the observation and the intermediate rule antecedent over the consequent deduced by firing the intermediate rule. A particular and popular example of this category is the scale and move transformation-based fuzzy rule interpolation (TFRI) [16], which has led to a number of advanced theoretical developments and applications [11], [17]–[32]. A key concept used in TFRI is the representative value of a fuzzy set, which captures important geometric characteristics of the fuzzy set (e.g., shape and location). This type of FRI works via exploiting the information provided by such representative values to construct the required intermediate rule. Further to the aforementioned two categories, analytical and closed-form solutions to FRI [22], [33] have been proposed.

TFRI is popular, but the interpolated results of running it may be significantly affected by the way in which an intermediate rule is constructed. In the general framework of TFRI, the construction of the intermediate rule typically starts by selecting a user-defined number of closest rules with respect to the unmatched

Manuscript received October 19, 2018; revised June 12, 2019, August 2, 2019, and September 13, 2019; accepted October 9, 2019. Date of publication October 29, 2019; date of current version December 1, 2020. *(Corresponding author: Qiang Shen.)*

T. Chen is with the Department of Computer Science, School of Computing and Engineering, University of Huddersfield, HD1 3DH Huddersfield, U.K. (e-mail: t.chen@hud.ac.uk).

C. Shang, J. Yang, F. Li, and Q. Shen are with the Department of Computer Science, Faculty of Business and Physical Sciences, Aberystwyth University, SY23 3FL Aberystwyth, U.K. (e-mail: cns@aber.ac.uk; jiy6@aber.ac.uk; fal2@aber.ac.uk; qqs@aber.ac.uk).

Color versions of one or more of the figures in this article are available online at http://ieeexplore.ieee.org.

Digital Object Identifier 10.1109/TFUZZ.2019.2949767

observation. The fuzzy values of the antecedent features within an intermediate rule are each obtained by taking a weighted combination of those corresponding fuzzy sets involved in the antecedents of the selected rules. However, the selection of the closest rules is purely based on a distance measure, which may not be sufficiently indicative of the most relevant rules for interpolation, especially when none of the existing rules is close enough to the observation. In addition, the use of such heuristically generated weights may lead to the implementation of undesirable move transformation, leading to counterintuitive interpolated outcomes. This is because no matter which rules to select, the move transformation will always force the resulting intermediate rule to overlap with the unmatched observation. Note that, in the FRI literature, there exist alternative methods that do not require the choice of a certain number of nearest neighboring rules to perform interpolation, but work by simply taking all the rules of the fuzzy rule base into consideration (e.g., [34], [35]). Nonetheless, all TFRI methods need the selection of certain nearest rules in order to function.

Having taken notice of the above observation, this article presents an alternative approach by examining what subset of rules should be selected and how much each of such selected rules should contribute to constructing the intermediate rule, thereby making the interpolation process more robust. In particular, the proposed method is able to select a subset of rules that assemble the intermediate rule, which overlaps with the observation without incurring further move transformation. This is followed by a procedure, which determines the weight that each individual selected rule is expected to contribute toward the intermediate rule, with the weighting scheme converted into a system of simultaneous bounded linear equations. To have a fair comparison over different methods for intermediate rule construction, systematic experimental results of applying these methods to support Takagi–Sugeno–Kang (TSK) fuzzy inference on a range of benchmark datasets are provided. Statistical analyses of the results are carried out, demonstrating the efficacy of the proposed work.

The remainder of this article is organized as follows. Section II reviews the underlying algorithm of TFRI. Section III further justifies the technical reasons for the present new approach. Section IV describes the proposed methodology. Section V discusses comparative experimental results. Section VI concludes this article and outlines ideas for further improvement.

II. Background

This section briefly reviews the interpolation procedures involved in the core of conventional TFRI. Without losing generality, suppose that a sparse rule base R consisting of a set of fuzzy rules $r_k, k = 1, 2, \ldots, |R|$, is given for an n-dimensional problem, with the rule r_k represented as follows:

$$\text{If } x_1 \text{ is } A_1^k \text{ and } \ldots \text{ and } x_n \text{ is } A_n^k, \text{Then } y^k \tag{1}$$

where $x_i, i \in \{1, 2, \ldots, n\}$, is the ith antecedent (or input) feature, which is described by the fuzzy value A_i^k, and y^k is the consequent of the fuzzy rule.

A. Representation of Representative Values

For simplicity, triangular fuzzy membership functions (MFs) are adopted herein to represent fuzzy sets (although any piecewise linear fuzzy representation may be used as an alternative if preferred). Let A_i be a tuple denoting a triangular fuzzy set $A_i = (a_{i1}, a_{i2}, a_{i3})$, where a_{i1} and a_{i3} are the left and right support vertexes, and a_{i2} is the normal point of the fuzzy set. The representative value $\text{Rep}(A_i)$ that denotes the overall geometric shape and location of the fuzzy set A_i in its corresponding domain is defined by the following:

$$\text{Rep}(A_i) = \frac{a_{i1} + a_{i2} + a_{i3}}{3}. \tag{2}$$

Note that the point associated with the center of gravity is typically used to define the representative value of an MF, but the above is used for computational simplicity.

B. Selection of Closest Rules

Given an observation $o^* = (A_1^*, \ldots, A_k^*, \ldots, A_n^*)$, with A_k^* denoting the kth feature value of the observation, the distance between a rule r^k and the observation is calculated as the aggregated distance of all individual antecedent features, as follows:

$$d(r^k, o^*) = \sqrt{\sum\nolimits_{i=1}^{n} d(A_i^k, A_i^*)^2} \tag{3}$$

where $d(A_i^k, o_k^*)$ is the normalized distance of the otherwise absolute distance measure, ensuring compatibility across all antecedent features, such that

$$d(A_i^k, A_k^*) = \frac{|\text{Rep}(A_i^k) - A_i^*|}{\max_{A_i} - \min_{A_i}} \tag{4}$$

where $|\text{Rep}(A_i^k) - A_i^*|$ is the absolute difference between the observed feature value A_i^* and the representative value of the fuzzy set A_i^k for the corresponding attribute x_i, and $\min_{A_i}$ and $\max_{A_i}$ denote the maximal and minimal value of x_i, respectively, jointly delimiting the domain bound of x_i. Once the distances between the given observation and all existing rules in the sparse rule base are calculated, the $l (l \geqslant 2)$ rules that have the minimal distances are determined and selected as the closest l rules to the observation. The number of rules l to be selected is determined by the user.

C. Construction of Intermediate Rule

Let $\omega_{A_i^k}$ denote the weight of the ith antecedent fuzzy set A_i^k of r^k such that

$$\omega_{A_i^k} = \frac{\omega'_{A_i^k}}{\sum_{i=1}^{l} \omega'_{A_i^k}} \tag{5}$$

and be termed the normalized displacement factor, where $\omega'_{A_i^k}$ represents the similarity between the antecedent fuzzy set A_i^k and the corresponding fuzzy value within the observation, which is defined by

$$\omega'_{A_i^k} = \frac{1}{d(A_i^k, A_i^*) + 1}. \tag{6}$$

The intermediate fuzzy terms A_i'' over $i, i \in \{1, \ldots, n\}$, are constructed from the antecedents of the l closest rules. These are then moved to A_i' such that they have the same representative values as those of A_i^*

$$A_i' = A_i'' + \delta_{A_i}(\max{}_{A_i} - \min{}_{A_i}) \tag{7}$$

where

$$A_i'' = \sum_{k=1}^{l} \omega_{A_i^k} A_i^k \tag{8}$$

$$\delta_{A_i} = d(A_i^*, A_i''). \tag{9}$$

From this, by analogy, the moved intermediate consequent y' can be computed with the parameters ω_{y^k} and δ_y through aggregation of the n corresponding values of A_i', such that

$$y' = \sum_{k=1}^{l} \omega_{y^k} y^k + \delta_y(\max{}_y - \min{}_y) \tag{10}$$

where ω_{y^k} and δ_y are calculated by

$$\omega_{y^k} = \frac{1}{n} \sum_{i=1}^{n} \omega_{A_i^k} \tag{11}$$

$$\delta_y = \frac{1}{n} \sum_{i=1}^{n} \delta_{A_i}. \tag{12}$$

D. Scale and Move Transformations

The aim of carrying out scale and move transformations in TFRI is to ensure that transformed antecedent feature values of the intermediate rule will coincide with their corresponding fuzzy values in the unmatched observation given. The transformations are implemented in two stages.

I) *Scale operation:* transform from A_i' to $\hat{A}_i'$ termed as the scaled intermediate fuzzy set, in an effort to determine the scale rate s_{A_i}.

II) *Move operation:* transform from $\hat{A}_i'$ to A_i^* to obtain a move ratio m_{A_i}.

Given a triangular intermediate fuzzy set $A_i' = (a_{i1}', a_{i2}', a_{i3}')$, the scale rate s_{A_i} is calculated by

$$s_{A_i'} = \frac{a_{i3}^* - a_{i1}^*}{a_{i3}' - a_{i1}'} \tag{13}$$

which essentially expands or contracts the support length of A_i': $a_{i3}' - a_{i1}'$ so that it becomes the same as that of the observation A_i^*. The scaled intermediate fuzzy set $\hat{A}_i'$ is obtained as follows:

$$\begin{cases} \hat{a}_{i1}' = \dfrac{(1+2s_{A_i})a_{i1}' + (1-s_{A_i})a_{i2}' + (1-s_{A_i})a_{i3}'}{3} \\ \hat{a}_{i2}' = \dfrac{(1-s_{A_i})a_{i1}' + (1+2s_{A_i})a_{i2}' + (1-s_{A_i})a_{i3}'}{3} \\ \hat{a}_{i3}' = \dfrac{(1-s_{A_i})a_{i1}' + (1-s_{A_i})a_{i2}' + (1+2s_{A_i})a_{i3}'}{3}. \end{cases} \tag{14}$$

The move operation shifts the position of $\hat{A}_i'$ to the same as that of A_i^*, with the move ratio $m_{A_i'}$ determined by

$$m_{A_i'} = \begin{cases} \dfrac{3(a_{i1}^* - \hat{a}_{i1}')}{\hat{a}_{i2}' - \hat{a}_{i1}'}, & \text{if } a_{i1}^* > \hat{a}_{i1}' \\ \dfrac{3(a_{i1}^* - \hat{a}_{i1}')}{\hat{a}_{i3}' - \hat{a}_{i2}'}, & \text{otherwise .} \end{cases} \tag{15}$$

Once all scale and move parameters are computed over i, the required factors for analogically modifying the intermediate consequent y' are heuristically calculated by finding the averages, as follows:

$$s_{y'} = \frac{1}{n} \sum_{i=1}^{n} s_{A_i'} \tag{16}$$

$$m_{y'} = \frac{1}{n} \sum_{i=1}^{n} m_{A_i'}. \tag{17}$$

The scaled result $\hat{z}'$ of the intermediate consequent y' is then calculated as follows:

$$\begin{cases} \hat{y}_1' = \dfrac{(1+2s_{y'})y_1' + (1-s_{y'})y_2' + (1-s_{y'})y_3'}{3} \\ \hat{y}_2' = \dfrac{(1-s_{y'})y_1' + (1+2s_{y'})y_2' + (1-s_{y'})y_3'}{3} \\ \hat{y}_3' = \dfrac{(1-s_{y'})y_1' + (1-s_{y'})y_2' + (1+2s_{y'})y_3'}{3}. \end{cases} \tag{18}$$

Finally, the interpolated consequent is obtained by applying the averaged move ratio such that

$$\begin{cases} y_1^* = \hat{y}_1' + m_{y'}\gamma \\ y_2^* = \hat{y}_2' - 2m_{y'}\gamma \\ y_3^* = \hat{y}_3' + m_{y'}\gamma \end{cases} \tag{19}$$

$$\gamma = \begin{cases} \frac{\hat{y}_2' - \hat{y}_1'}{3}, & \text{if } m_{y'} > 0 \\ \frac{\hat{y}_3' - \hat{y}_2'}{3}, & \text{otherwise.} \end{cases} \tag{20}$$

III. Motivations and Objectives

As can be seen from (7), in conventional TFRI, the intermediate term A_i' is obtained by moving A_i'' with a location shift of $\delta_{A_i}(\max_{A_i} - \min_{A_i})$, where A_i'' is a linear combination of those fuzzy sets $A_i^k, k = 1, \ldots, l$, respectively, selected from the l closest rules, weighted by the normalized distance to the unmatched observation. This move operation is devised in an effort to ensure that the representative value of each antecedent fuzzy set in the intermediate rule equals that of its counterpart in the observation. Whether the selected rules are used for

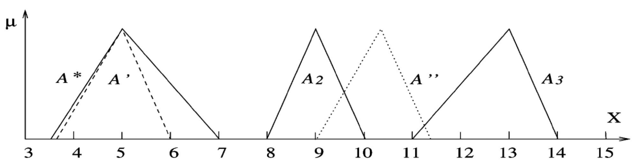


Fig. 1. Case 1—Extrapolation.

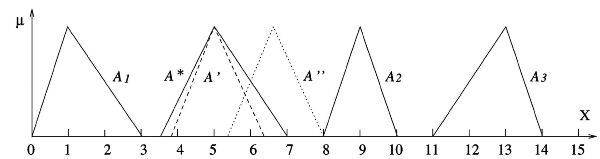


Fig. 2. Case 2—Interpolation.

extrapolation or interpolation [although mathematically, both take the same form as (7)], this requirement will in general lead to a significant problem in performing TFRI, as discussed in the following subsections.

A. Case 1—Extrapolation

Extrapolation deals with situations where the selected closest fuzzy rules all geometrically lie on one side of the hypergraphical plot depicting the given observation. For illustration, Fig. 1 shows such a case in a single input problem space, where A_2 and A_3 are the two antecedent fuzzy values taken from the rules r^2 and r^3, which are the closest to the observation A^*, thereby being selected to construct the intermediate rule. Obviously, A'' will lie in between A_2 and A_3, given that A'' is simply a weighted combination of A_2 and A_3. Therefore, A'' is required to be moved left to make its representative value equaling to that of the observation.

Denote the value of a certain antecedent feature x_i within the unmatched observation o^* as A_i^*. Given a selected rule subset R^* of l rules, the following equation should hold if the weighted linear combination of the antecedent values for x_i is required to be of the same representative value as its counterpart in the observation:

$$\mathrm{Rep}(A_i^*) = w_1\mathrm{Rep}(A_i^1) + \cdots + w_2\mathrm{Rep}(A_i^k) + \cdots + w_l\mathrm{Rep}(A_i^l). \quad (21)$$

For extrapolation, the representative values $\mathrm{Rep}(A_i^k), k = 1, \ldots, l$, are all on one side of that of the observation $\mathrm{Rep}(A_i^*)$. Thus, if $\forall r^k \in R^*, \mathrm{Rep}(A_i^*) < \mathrm{Rep}(r_i^k)$, while supposing that $\mathrm{Rep}(A_i^1) < \mathrm{Rep}(A_i^k), \forall k \neq 1$, then $\mathrm{Rep}(A_i^*) = \sum_{k=1}^{l} w_k\mathrm{Rep}(A_i^k) < \sum_{k=1}^{l} w_k\mathrm{Rep}(A_i^1) = \mathrm{Rep}(A_i^1)$, given that $\sum_k^l w_k = 1$, $w_k \in [0, 1]$. This means that the representative value of the corresponding interpolated term will be greater than that of the observation no matter what weight vector is used. Vice versa, if the representative values of all selected fuzzy terms are smaller than that of the observation, the representative value of the resulting interpolated term will be smaller than that of the observation also, regardless of the choice for the weighting vector. This analysis indicates that for extrapolation in conventional TFRI, without performing the move operation, no solution can be obtained if an intermediate rule is to be created whose representative values are to be exactly the same as those of the corresponding features in the observation.

B. Case 2—Interpolation

Interpolation deals with situations where the selected fuzzy rules flank (i.e., geometrically lie on both sides of) the observation, as illustrated in Fig. 2 for a single input problem. In this example, A_1, A_2, and A_3 are the fuzzy sets taken from the rules r^1, r^2, and r^3, which are deemed to be the closest to the observation, based on distance measures, and which are selected to construct the intermediate rule. Although there is a chance that the representative value of A'' may happen to be equal to that of the observation, it is generally difficult to guarantee this, as the weights used for linear combination are heuristically calculated. Therefore, as with Case 1, the move operation is also necessary. This is readily shown given the previously proven case for extrapolation, as, mathematically, both rule interpolation and extrapolation are calculated in exactly the same way.

C. Summary of Justifications

As discussed above, whether the existing TFRI is employed for interpolation or extrapolation, the move operation is required to ensure the representative values of the intermediate fuzzy rule to be the same as those of their counterparts within the observation. The question is whether there exists a mechanism that is able to construct the intermediate rule, by combining a set of carefully selected rules with appropriate weights, without involving the compulsory move operation. That is, whether it is feasible to utilize just the information available from the existing sparse rule base itself to create the intermediate rule, in order to perform transformation-based FRI. A positive answer to this also allows for automatic determination of the number of closest rules without human intervention, which is otherwise required by TFRI, be it for interpolation or extrapolation.

Inspired by these observations, this article proposes a novel optimized transformation-based fuzzy rule interpolation (OT-FRI) technique, improving upon the original TFRI method in two folds.

1) To automatically select a set of useful rules such that its antecedent fuzzy values lie on both sides of an unfired observation, instead of using a user-defined number of closest rules that are judged by a fixed distance measure.
2) To search for a weighting vector such that the resultant interpolated fuzzy rule overlaps with the unmatched observation, instead of using heuristic weights that inevitably requires an additional move operation.

IV. Automated Selection of Fuzzy Rules for Interpolation

In order to construct the intermediate rule without implementing the undesirable move operation, this section shows how a set of useful rules can be automatically selected from the original sparse rule base.

A. Initial Analysis

For clarity, examine just one certain individual feature x_i first. In this case, the representative value of the intermediate term A_i' may equal to that of its counterpart A_i^* given in the observation o^*, only if one of the following scenarios is satisfied, where R^* stands for the set of selected rules:

i) if $\exists r^k \in R^*$, such that $\text{Rep}(A_i^k) > \text{Rep}(A_i^*)$, and $\text{Rep}(A_i^{k'}) < \text{Rep}(A_i^*)$, for all $r^{k'} \in R^*, k' \neq k$;
ii) if $\exists r^k \in R^*$, such that $\text{Rep}(A_i^k) < \text{Rep}(A_i^*)$, and $\text{Rep}(A_i^{k'}) > \text{Rep}(A_i^*)$, for all $r^{k'} \in R^*, k' \neq k$;
iii) if $\exists r^k \in R^*$, such that $\text{Rep}(A_i^k) = \text{Rep}(A_i^*)$.

The first and second scenarios correspond to the situations where the representative values of the fuzzy terms in the selected rules flank that of their corresponding term in the observation, and the third scenario describes the case where one (or more) of the selected fuzzy terms, whose representative value happens to be exactly the same as that of its counterpart within the observation.

Suppose that the original sparse rule base is denoted by R. The aim here is to devise a method that automatically searches for a subset of fuzzy rules $R^* \subset R$. This search is constrained such that the representative values of the antecedent feature terms of the intermediate rule being constructed using these rules will equal or closely approximate those of the corresponding terms in the unmatched observation. To aid in the illustration of the underlying ideas of this work, suppose that there are $|R| = 7$ existing rules in the sparse rule base for a six-dimensional problem. Table I specifies the kth closest rule for the ith feature value given an unmatched observation o^* based on the distance measure between representative values, together with a sign expressing the proximity relation between $\text{Rep}(A_i^k)$ and $\text{Rep}(A_i^*)$, where "$+$" denotes $\text{Rep}(A_i^k) > \text{Rep}(A_i^*)$, "$=$" denotes $\text{Rep}(A_i^k) = \text{Rep}(A_i^*)$, and "$-$" denotes $\text{Rep}(A_i^k) < \text{Rep}(A_i^*)$.

For this problem case, if the features are considered individually one at a time, the use of the least number (i.e., 2) of closest rules with regard to the value of the first feature x_1 will lead to the selection of the rules r^2 and r^3, forming a flanking case for this feature. However, if r^2 and r^3 were indeed selected, then they would result in the same sign for the antecedent values of x_2 and x_6. Thus, in computing the required weight vector, no matter what values are assigned to w_2 and w_3, the interpolated antecedent feature values will not lie within the area covered by the corresponding features of r^2 and r^3. As such, the final interpolated result will not be within the space bounded by the outcomes of r^2 and r^3, making an incorrect interpolation. In order to generate valid solutions, it is, therefore, necessary to simultaneously consider all rule antecedent features, rather than one by one.

Reflecting on the above discussion, a method for automatically selecting closest fuzzy rules out of the given sparse rule base can be introduced as follows.

1) Generate a proximity table like Table I in the example, where the first column lists the feature values (for the features $x_i, i = 1, 2, \ldots, n$) of the unmatched observation o^*, each of the rest columns represents a fuzzy rule $r^k, k = 1, \ldots, |R|$, and each cell specifies the order of how close the value A_i^k of the ith feature in the kth closest rule is to A_i^*, based on the use of a certain distance measured. The sign associated with r^k denotes the relative position holding between A_i^k and A_i^*, as indicated previously.

TABLE I
EXAMPLE PROXIMITY TABLE

i-th feature in o^*	k-th closest rule for value of i-th feature in ascending order						
A_1^*	r^2+	r^3-	r^4+	r^5+	r^1-	r^7+	r^6-
A_2^*	r^3+	r^4-	r^5+	r^1+	r^7-	r^2+	r^6+
A_3^*	r^6-	r^7+	r^1+	r^3+	r^2-	r^4+	r^5-
A_4^*	r^1-	r^3-	r^2+	r^4+	r^5+	r^7-	r^6+
A_5^*	$r^3=$	r^2+	r^5+	r^1-	r^4-	r^7+	r^6-
A_6^*	r^2+	r^3+	r^6-	r^7-	r^1+	r^4-	r^5+

2) Obtain a temporary rule base R_i^* for each observed feature value A_i^*, by adding those rules in the table iteratively from the leftmost rightwards, given that the rules in the table have been ranked in ascending order based on the distance measures. Once either of scenarios (i)–(iii) is found, stop adding rules. For the present illustration, $R_1^* = \{r^2, r^3\}$, $R_2^* = \{r^3, r^4\}$, $R_3^* = \{r^6, r^7\}$, $R_4^* = \{r^1, r^3, r^2\}$, $R_5^* = \{r^3\}$, and $R_6^* = \{r^2, r^3, r^6\}$.
3) Assign R^* to the temporary rule base R_i^* of the largest cardinality, i.e., $R^* = R_i^*,\ \forall i \in \{1, 2, \ldots, n\}, |R_i^*| > |R_{i'}^*|$ and $i \neq i'$. In the case where there are multiple feature values that involve the same highest number of selected rules, to ensure full coverage, find the union of these multiple sets. For the present example, the rules required to perform interpolation are $R^* = R_4^* \bigcup R_6^* = \{r^1, r^3, r^2\} \bigcup \{r^2, r^3, r^6\} = \{r^1, r^3, r^2, r^6\}$.
4) Remove a row if the A_i^* of that row is consistent with the currently selected rule subset R^*, namely if either of scenarios (i)–(iii) is satisfied. For shorthand, denote the notion of consistency checking between an observed feature value A_i^* and a corresponding rule set R^* by

$$R^*(A_i^*) = \begin{cases} \text{True}, & \text{if either of scenarios (i)–(iii) is satisfied} \\ \text{False}, & \text{otherwise.} \end{cases} \tag{22}$$

Back to the example, this deletes the rows starting with A_1^*, A_3^*, A_4^*, A_5^*, and A_6^*.
5) Perform the following for each of the remaining rows, by looping from the leftmost (closest) rule until reaching the rightmost (furthest) rule, skipping if $r^k \in R^*$, else updating $R^* = R^* \bigcup \{r^k\}$ if $R^* \bigcup \{r^k\}(A_i^*)$ is True, otherwise doing nothing. For the illustrative example, the first rule encountered for the (only remaining) row starting with A_2^* that is not in R^* is r^4, the union of $\{r^4\}$ and R^* results in a case satisfying scenario (ii), and therefore, R^* is updated such that $R^* = \{r^1, r^3, r^2, r^6, r^4\}$. As no further checking is possible, the loop terminates.

Note that the addition of new rules in running the above procedure is done in order. As such, the aforementioned "sets" of rules and the "union" of a multielement set R^* with a single-element set $\{r^k\}$ do not follow the strict mathematical definition of the corresponding concepts or operations. Instead, the elements in such a set are ordered, and any union needs to retain such ordering with the single element listed as the last of the existing multielement set that it is being merged into. From this, a postpruning step can be proposed as follows.

1) Start from the rightmost (or the last) rule in R^* and update $R^* = R^*/\{r^k\}$ if the removal of the rule r^k does not violate the consistency across all observed feature values.
2) Iterate the above process until every rule in R^* has been checked.

For the running example, the removal of the last rule r^4 would violate the opposite sign condition; therefore, it should be retained. The process goes on to check r^6, the removal of which does not violate the consistency relation for any of the features involved. Hence, $R^* = R^*/\{r^6\} = \{r^1, r^3, r^2, r^4\}$. Then, r^2 is checked, and its removal would cause inconsistence for A_3, given that r^6 has already been removed and r^2 is the only rule that provides a negative sign. Thus, r^2 should be kept. The pruning procedure then checks r^3, which is ok to be deleted, resulting in the updated $R^* = \{r^1, r^2, r^4\}$. Eventually, r^1 is checked and required to be retained. The final result is $R^* = \{r^1, r^2, r^4\}$. Such a postpruning ensures that the set of rules required to perform interpolation for a certain observation is minimal.

B. Rule Selection Algorithm

Generalizing the above initial analysis and the associated illustrative example leads to Algorithms 1 and 2. These algorithms, respectively, formalize the procedures that are utilized in this work to automatically determine, and then to prune to the minimum, a number of rules that are required to compute the intermediate rule for interpolation, given an observation that does not match any of the existing rules in the sparse rule base. No subsequent move operation is necessary. This is different from the original approach of TFRI that works by selecting a subset of closest rules whose cardinality is manually set, while requiring a subsequent move operation as per (7) before the interpolation is implemented.

An exceptional case is that the inclusion of all rules in the sparse rule base still fails to simultaneously provide a flanking scenario for all antecedent features. This implies that the observation is not within the space bounded by any subset of rules from the current rule base. In this event, no matter what weight vector is to be assigned, a linearly weighted combination of any existing rules cannot produce an interpolated rule that exactly coincides with the observation. However, an approximation solution may be sought for this, as described next.

C. FRI With Selected Fuzzy Rules

Given an observation o^* with no match or insufficient matching degrees against the rules in the sparse rule base, the process of interpolation is invoked. This starts with a search for its closest rule subset $R^* = \{r^1, r^2, \ldots, r^l\}$, as detailed in the preceding section. The key to performing interpolation in TFRI is to derive an intermediate fuzzy rule with the representative values of the fuzzy sets describing its antecedent features equaling to their counterparts in o^*. Formally, this can be represented as a system

Algorithm 1: Selection of Candidate Rules.

% Step 1
Generate a proximity table
% Step 2
Set $|R^*| = 0$
for each observed feature value $A_i^* \in o^*, i = 1, \ldots, n$ **do**
  Set $|R_i^*| = 0$
  for each rule $r^k, k = 1, \ldots, l$ in the ith row of the proximity table **do**
    $R_i^* = R_i^* \bigcup \{r^k\}$
    if $R_i^*(A_i^*)$ **then**
      break;
    end if
  end for
end for
% Step 3
for each row starting with $A_i^*, i = 1, \ldots, n$ **do**
  if $|R_i^*| \geq |R_{i'}^*|, \forall i' = 1, \ldots, n$, and, $i \neq i'$ **then**
    $R^* = R^* \cup R_i^*$
  end if
end for
% Step 4 and 5
for each row starting with $A_i^*, i = 1, \ldots, n$ **do**
  if $R_i^*(A_i)$ **then**
    Remove the ith row from the table
  else
    for each rule $r^k \in R$ in the ith row **do**
      if $r^k \notin R^* \wedge R^* \bigcup \{r^k\}(A_i^*)$ **then**
        $R^* = R^* \bigcup \{r^k\}$
      end if
    end for
  end if
end for

of simultaneous linear equations as follows:

$$\begin{cases} \text{Rep}(A_1^*) = \sum_{k=1}^{l} w_k \text{Rep}(A_1^k) \\ \vdots \\ \text{Rep}(A_i^*) = \sum_{k=1}^{l} w_k \text{Rep}(A_i^k) \\ \vdots \\ \text{Rep}(A_n^*) = \sum_{k=1}^{l} w_k \text{Rep}(A_n^k) \end{cases} \quad (23)$$

where $\text{Rep}(A_i^*)$ is the representative value of the fuzzy set denoting the input feature x_i in o^*, A_i^k is the ith antecedent value in the rule r^k, and w_k is the weight associated with r^k.

This system of linear equations can be simplified in the form of matrix representation such that

$$Aw = A^* \quad (24)$$

where $w = [w_1, \ldots, w_k, \ldots, w_l]^T$ is the weighting vector, $A^* = [\text{Rep}(A_1^*), \ldots, \text{Rep}(A_i^*), \ldots, \text{Rep}(A_n^*)]^T$ is the vector of representative values for the observation o^*, and A is an $n \times l$ matrix

Algorithm 2: Postpruning.

Input R^* from selection of candidate rules
for each rule $r^k, k = |R^*|, \ldots, 2, 1$ **do**
  isRemoved = True;
  for each observed feature value $A_i^* \in o^*, i = 1, \ldots, n$ **do**
    if $\neg(R^*/\{r^k\}(A_i^*))$ **then**
      isRemoved = False;
      continue;
    end if
  end for
  if *isRemoved* **then**
    $R^* = R^*/\{r^k\}$;
  end if
end for

with its generic entry A_{ik} $(i \in \{1, \ldots, n\}, k \in \{1, \ldots, l\})$ signifying the representative value $\text{Rep}(A_i^k)$ for the ith feature value of the rule r^k:

$$A = \begin{bmatrix} \text{Rep}(A_1^1) & \cdots & \text{Rep}(A_1^l) \\ \vdots & \ddots & \vdots \\ \text{Rep}(A_n^1) & \cdots & \text{Rep}(A_n^l) \end{bmatrix}. \tag{25}$$

That is to say, if a certain weight vector can be determined that satisfies (24), an intermediate rule can then be constructed. The resulting representative values of its antecedent feature terms will be exactly the same as those for the corresponding feature values A_i^* as given in the unmatched observation, without applying any further move operation. Following conventional representation of a weighting system, without losing generality, the bounding constraints below are introduced such that $\sum_{i=1}^{l} w_i = 1, w_i \in [0, 1]$.

The above constraints are introduced as with those imposed over the weighting vector in the original T-FRI method, upon which this work aims to improve. This is a common practice followed by other related work (e.g., [18], [19], [21], [23]). Further to helping assess directly how much contribution each selected fuzzy rule may make to the overall formulation of the resulting intermediate rule, such constraints also make the subsequent computation simpler in deriving the weighting vector. This is in comparison with the possible alternatives that typically use the Euclidean norm, requiring more complicated computation.

In so doing, the task of constructing an intermediate rule construction is converted into that of solving a system of simultaneous linear equations

$$Aw = A^*, \text{s.t.} \sum_{i=1}^{l} w_i = 1, w_i \in [0, 1]. \tag{26}$$

Without attempting to directly resolve this system of bounded linear equations (which would otherwise require the consideration of whether the system is underdetermined, overdetermined, or square), (26) can be transformed into an optimization problem using the least squares method. This can be implemented by minimizing the sum of the squares of the residuals in the results of every single equation as follows:

$$\arg\min_{w} ||Aw - A^*||_2, \text{s.t.} \sum_{i=1}^{l} w_i = 1, w_i \in [0, 1]. \tag{27}$$

The minimization of $||Aw - A^*||_2$ is equivalent to that of its squared difference $||Aw - A^*||_2^2$, which can be further decomposed such that

$$\begin{aligned} ||Aw - A^*||_2^2 &= (Aw - A^*)^T (Aw - A^*) \\ &= (A^T w^T - A^{*T})(Aw - A^*) \\ &= \frac{1}{2} w^T (2A^T A) w + (-2A^T A^*)^T w + A^{*T} A^*. \end{aligned} \tag{28}$$

This fits perfectly into the problem of quadratic programming as being of the form

$$\arg\min_{v} \frac{1}{2} v^T H v + c^T v \tag{29}$$

where the vector v plays the role of w, H does that of $(2A^T A)$, c corresponds to $(-2A^T A^*)^T$, and the constant term $A^{*T} A^*$ has no effect upon the minimum being sought and is, therefore, omitted during the optimization process.

D. Optimal Weight Computation

In order to solve the above transformed quadratic programming problem, the popular active set method [36] is applied herein. In general, the solution procedure involves two phases. The first calculates an initial feasible start point, and the second executes an iterative test and generation process regarding the feasible points, which eventually converge to the solution sought. There are many versions of the active set method that are similar in structure, with the underlying method adopted in this article taken from the classical work of [37] and [38] (which has been modified for both linear and quadratic programming [39]). For the present work, this algorithm is taken as a tool to implement the proposed approach and is outlined below for completeness. Note that alternative versions of the method may be employed if preferred, but an investigation into their use is beyond the scope of this article.

In running the adopted algorithm, an active set matrix S_k is maintained as an estimate of active constraints on the boundaries of the solution point. At each iteration k, S_k is updated and used to form a basis upon which to determine the search direction d_k, which attempts to minimize a given optimization objective function. The potentially feasible subspace for d_k is constructed from a basis Z_k, whose columns are orthogonal to the estimate of the active set S_k with $S_k Z_k = 0$. Thus, the resulting search direction d_k is guaranteed to remain on the boundaries of the active constraints.

Once Z_k is found, the objective function is optimized along the search direction at d_k, with d_k being a linear combination of the columns of $d_k = Z_k p$ for a certain vector p, within the null space of the active constraints. The quadratic objective function can be viewed as a function of p, by substituting for d_k as follows:

$$q(p) = \frac{1}{2}p^T Z_k^T H Z_k p + c^T Z_k p. \quad (30)$$

This can be differentiated with respect to p yielding

$$\nabla q(p) = Z_k^T H Z_k p + Z_k^T c \quad (31)$$

where $\nabla q(p)$ is referred to as the projected gradient of the quadratic objective function. The minimum of the function $q(p)$ in the subspace defined by Z_k occurs when $\nabla q(p) = 0$, which is the solution of the system of linear equations under consideration, assuming that the matrix H is positive definite.

Whether such an active set method converges is generally determined by the manner, in which the parameter α (named step length) in the following is updated at each iteration:

$$v_{k+1} = v_k + \alpha d_k, d_k = Z_k p. \quad (32)$$

Given the quadratic nature of the objective function, there are only two types of choice for α. The choice of a step of unity along d_k is the correct one to make, entailing the objective to reach the minimum of the function that is restricted to the null space of S_k. If such a step is taken without violating the constraints, then the final solution is found. Otherwise, the step along d_k is less than unity, and the process of modifying the weights iterates according to (32). The distance to the constraint boundaries in any direction d_k is given by

$$\alpha = \min_{i \in 1,\ldots,m} \left\{ \frac{-(A_i v_k - b_i)}{A_i d_k} \right\}. \quad (33)$$

From this, Lagrange multipliers λ_k are calculated that satisfy the nonsingular set of linear equations

$$S_k^T \lambda_k = c. \quad (34)$$

If all λ_k over k are positive, the vector v composed of the corresponding v_k is the optimal solution for the original problem, i.e., the vector of weights, w required for interpolation. However, if any λ_k is negative, and it does not correspond to an equality constraint, then the element corresponding to it is deleted from the active set, and a new iteration is carried out.

Note that the active set method has been shown to be convergent for strictly convex quadratic programming problems [40]. In the event that a problem is infeasible with overly stringent constraints, the method is still able to produce a result that minimizes the constraint violation in the worst case. More discussions on the convergence proof and properties of this method can be found in [40] and [41].

V. Experimental Analyses

This section presents and discusses the results of systematic comparative experimental investigations, supported with statistical analyses.

A. Experimental Setup

Experiments are performed on ten real-valued benchmark datasets taken from UCI [42] and KEEL [43] data repository, with all feature values normalized to fall within [0, 1]. A summary of the characteristics of these datasets is given in Table II.

TABLE II
Summary of Datasets Used

Data Set	# of Features	# of Instances
airfoil	6	1503
autoMPG6	5	392
CPU	6	209
delta_ail	5	7129
diabetes	2	43
ele1	2	495
ele2	4	1056
friedman	5	1200
laser	4	993
plastic	2	1650

Stratified hold-out validation is employed for result analysis, in order that there are potentially more unmatched instances for testing as opposed to the use of conventional tenfold cross validation. In a hold-out validation, a given dataset is partitioned into two subsets. Of the two, one is used to perform training to generate a fuzzy rule base, while the other subset is retained as the testing data for assessing the performance of the trained fuzzy systems. This validation process is then repeated ten times in order to lessen the impact of random factors; results of these $10\times$ hold-out validations are then averaged to produce each final experimental outcome, as reported in the following.

The experiments on comparisons between the optimized TFRI (as proposed in this work and denoted by OTFRI hereafter) and conventional TFRI are focused on applications to regression problems. To ensure fair comparison regarding the use of the intermediate rules constructed in different ways, they are verified through running the first-order TSK systems (which are of more representation power than Mamdani systems). In particular, weighted consequents propagated from the rule antecedents are directly computed, purposefully avoiding any subsequent move and scale transformation processes that remain the same to both TFRI approaches. For presentational simplicity, in the following, an intermediate rule built by the use of a certain TFRI method is referred to as an interpolated rule using that method.

The popular adaptive network-based fuzzy inference system (ANFIS) [44] is used to generate a sparse fuzzy rule base for experimental verification. Also, for fair comparison, each ANFIS is initialized with a simple and common grid partitioning, with each specified by a given MF. As such, the dimensionality of a rule base is k^n, where k is the partition granularity for each of the n input features. ANFIS is then trained using a hybrid learning method combining gradient descent and least squares estimation, which involves parameter optimization in terms of both MFs and coefficient parameters. To reduce the adverse impact of the curse of dimensionality [45] as the number of input features increases, and also to facilitate a wide range of experimental computations, only two uniformly divided triangular MFs are employed for datasets whose number of features is greater than four, otherwise three, as shown in Fig. 3. More details of ANFIS and the training method are beyond the scope of this article, but can be found in [44].

Once an original rule base is learned (by reading off from a trained ANFIS), to emphasize on the potential of FRI, a sparse rule base is created by randomly removing a certain fixed number of rules from the original rule base (see more below). Note that

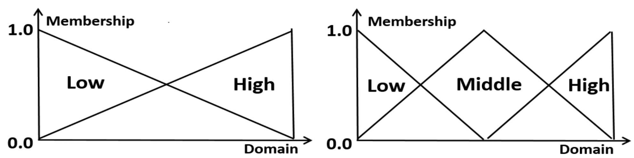


Fig. 3. Simple partitioning of feature spaces.

the removal of certain learned rules is purely introduced for evaluation purposes. In real application, especially for situations where training samples are limited, no such rule removal is carried out, but the coverage of learned ANFIS over the problem domain may be sparse in the first place. The rule removal here is set to see whether interpolation is indeed able to provide approximate inference results, in comparison to cases where richer domain knowledge is available.

Throughout the experimental investigations, a testing instance o^* is regarded unmatched if the matching degree $\prod_i^n \mu_{A_i^k}(A_i^*) < 0.5^n$, for all rules from the (sparse) rule base, where A_i^k represents the fuzzy set for the ith feature x_i in rule k, and A_i^* stands for the observed value of x_i. To quantitatively access the performance of each compared approach, the measure of root-mean-square errors (RMSEs) is adopted as the performance index, which is defined by

$$\text{RMSE} = \sqrt{\frac{\sum_i^I (y_i - \hat{y}_i)^2}{I}} \quad (35)$$

where I is the number of unmatched testing instances, y_i is the underlying (or ground) true output value, and $\hat{y}_i$ is the value predicted with the interpolated rule.

B. Illustrative Example

Before running systematical experiments for comparison, this subsection illustrates the main working procedures of the proposed OTFRI and its associated byproducts. The dataset utilized for this example is the Electric Length 1 dataset, i.e., ele1, with 495 instances and two numerical predictors to facilitate graphical illustration. This illustration is based on one single random run with the data partitioned into two subsets. The first is the training set consisting of 253 instances utilized to obtain a set of TSK fuzzy rules learned by ANFIS. With equally spaced three triangular MFs predefined for each feature and the first-order representation assumed for the rule consequent, the resulting full rule base is shown in Table III. This is followed by a random removal of four rules from the full rule base (of a size of 9), leading to a rather sparse rule base, consisting of rules 4, 6, 7, 8, and 9.

Fig. 4 shows all testing instances in the two-dimensional plane. In this figure, the horizontal and vertical axes represent the normalized values of the first attribute x_1 and those of the second attribute x_2, respectively, and the triangles refer to fired instances, while the circles are deemed unmatched as their partial matching degrees with any given rule are less than 0.5^2. As such, in the absence of approximately half of the all the rules required to cover the problem space, a large number of instances (233 out of 242) fail to directly find matching rules from the (sparse) rule base. Fuzzy interpolation is, therefore, resorted to work on these unmatched instances.

TABLE III
FULL RULE BASE USED FOR ILLUSTRATION

Rule	First Rule Antecedent	Second Rule Antecedent	Rule Consequent
1	[-0.500,-0.005,0.505]	[-0.402,0.005,0.453]	[0.331,2.864,0.009]
2	[-0.500,-0.005,0.505]	[-0.005,0.407,0.799]	[2.348,2.622,-0.996]
3	[-0.500,-0.005,0.505]	[0.477,0.816,1.209]	[30.787,5.782,-4.440]
4	[0.040,0.495,1.000]	[-0.402,0.005,0.453]	[1.116,15.108,-0.243]
5	[0.040,0.495,1.000]	[-0.005,0.407,0.799]	[0.738,13.801,-6.371]
6	[0.040,0.495,1.000]	[0.477,0.816,1.209]	[36.998,12.958,-27.027]
7	[0.5000,1.000,1.500]	[-0.402,0.005,0.453]	[0.376,0.167,0.602]
8	[0.5000,1.000,1.500]	[-0.005,0.407,0.799]	[0.041,-0.021,0.436]
9	[0.5000,1.000,1.500]	[0.477,0.816,1.209]	[-0.076,-0.038,-0.076]

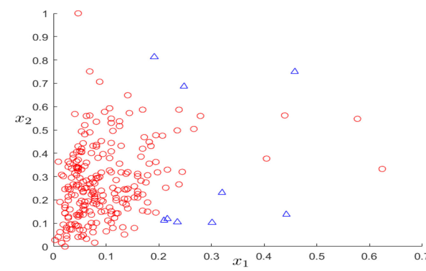


Fig. 4. Fired (triangular) instances and unfired (circles).

As an initial step of OTFRI (see Algorithms 1 and 2), a proximity table is built for each of the unmatched instances. Taking the unmatched instance (0.577, 0.548, 0.850) as an example, rules 6 and 7 are returned by the constructed proximity table and heuristic search (see Table III for their semantics). Note that the selected rule subset does not include rule 8, which is the closest rule otherwise to be returned by the use of the conventional distance metric, as given in [16]. For shorthand, denote this rule subset as {rule_6, rule_7} (and similarly for the other rules to be mentioned below). The construction of the interpolated rule is then invoked by calculating a weighting vector that minimizes the distance between this unmatched instance and the linear combination of the two fuzzy rules selected above. The elements of the resultant weighting vector (which is optimized by the active set method) are 0.706 and 0.294 for rules 6 and 7, respectively. The interpolated rule is finally assembled by aggregating all corresponding individual antecedent feature values of the selected fuzzy rules, modified by their respective weights.

In contrast, if conventional TFRI is employed, with the number of closest rules for selection manually set to 2, the rule subset of $\{rule\_6$ and $rule\_8\}$ is selected for the construction of the interpolated rule. The construction process works purely based on the exploitation of distance measures. For direct comparison, the resultant interpolated fuzzy rule produced by TFRI and that obtained using OTFRI as described above are both shown in Table IV.

The way of OTFRI constructing interpolated fuzzy rules leads to a significantly different outcome from that of TFRI. Fig. 5 shows a simple comparison for the illustrative example, over the whole set of 233 unfired instances. In particular, the horizontal axis in the figure projects the values of the first antecedent

TABLE IV
INTERPOLATED FUZZY RULES

Interpolated Rule	First Antecedent Feature	Second Antecedent Feature	Rule Consequent
TFRI	[0.093,0.567,1.070]	[0.165,0.542,0.935]	[19.220,6.649,-13.991]
OTFRI	[0.175,0.643,1.147]	[0.219,0.577,0.987]	[26.240,9.201,-18.911]

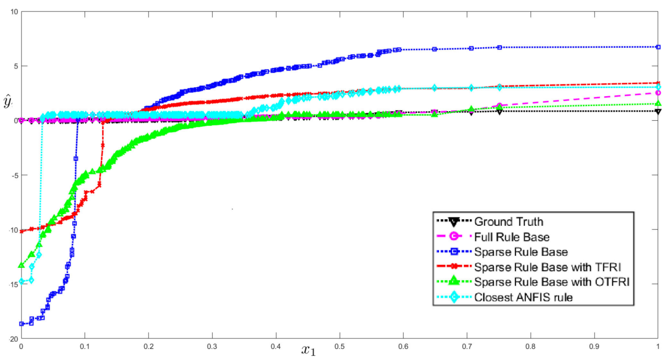


Fig. 5. Illustrative performance comparison.

feature x_1, and the vertical axis depicts the consequent values $\hat{y}$, obtained by running different approaches with the use of various rule bases. As demonstrated by this figure, OTFRI running on the sparse rule base can excellently approximate the ground truth for the middle range of the domain and even outperform the result of using the full rule base over the final part of the domain range. It also beats the possible alternative approach by obtaining the estimated consequent through simply firing the closest rule in the sparse rule base, over most parts of the domain. Note that over the full domain range, using just a sparse rule base is overall underperformed as compared to the use of the full rule base learned by ANFIS. This is of course expected given that a large proportion of the rules are randomly removed from the full rule base.

As explicitly stated previously, the purpose of this work is to show that when only a sparse rule base is available, the proposed approach can improve upon the popular state-of-the-art TFRI. Fig. 5 clearly illustrate this, with OTFRI beating TFRI over almost the whole range of the feature domain. Instead of using just one particular example, in order to demonstrate that such superior performance over TFRI is systematic, results on a set of experiments across a variety of setting and benchmark datasets are discussed below.

C. Results and Discussion

1) With or Without FRI: Tables V and VI show the RMSEs between the ground truth and the predicted values returned by the use of different sparse rule bases, where "$m\%$ Missing" ($m \in \{10, 15, 20, 25, 30, 40, 50\}$) stands for what percentage of rules are omitted from the original learned rule base. Again, such an omission is intentional, to reveal the potential effectiveness of rule interpolation. That the proportion of missing rules is set over this range (from 10% to 50%) also helps demonstrate the robustness of the proposed approach under different settings. Numerical figures under the headings of ANFIS, ANFIS_TFRI and NFIS_OTFRI in these tables indicate the averaged RMSEs and their associated standard deviations over the results, obtained by the use of ANFIS, conventional TFRI, and the proposed approach, respectively. Note that TFRI involves the selection of a user-defined number of closest rules for interpolation, which is uniformly set to 2 here. This is both for computational simplicity and for reflecting the most recent discovery in that effective TFRI techniques typically require the use of just two closest rules to perform interpolation [18].

In Tables V and VI, an entry of N/A indicates that all testing instances have been fired with the rules given in the (sparse) rule base, thereby leaving no unmatched instances requiring FRI. For example, two or three datasets have no instances that cannot be matched by the given rules for firing when the proportion of missing rules is 10% or 15%, respectively. An interesting observation from these results is that for the delta_ail dataset, all testing instances are fired with the original rules when the sparse rule base misses 15% rules, but there are unmatched instances when the missing proportion is lower at 10%. This is due to the randomness nature of the experiments in that the particular 15% rules missed in one case are completely independent of those 10% missing rules in the other. However, in general, as the missing proportions are getting larger, more datasets will have more instances unmatchable by the given rules.

In both Tables V and VI, the lowest RMSEs are highlighted in boldface, indicating the best performance for a dataset under the same experimental setting over different approaches. When missing 10% rules, the original ANFIS only achieves two lowest errors out of ten datasets, while incurring a relatively larger standard deviation. The number of winning cases for ANFIS is reduced to just one if the missing rule proportion becomes 15% or 25%, and further to no winner at all in the case of 20% missing rules. This demonstrates that FRI significantly improves the performance of the systems running on a sparse rule base, as most of the lowest errors are achieved with interpolated rules (which are equivalent to rules being added on to the original rule bases).

As the missing proportions become higher, however, the original ANFIS starts to produce relatively more results that are more accurate, e.g., there are three or four datasets for which a better RMSE is obtained with the missing proportion being 30%, 40%, or 50%. This decrease in performance is likely attributed to the fact that a poor rule base with too many missing rules is difficult for any rule interpolation method to function well. This can be expected of course, since with more and more rules missing from the given rule base, the sparse rule base becomes even more incomplete (and may eventually become completely empty), this will gradually deteriorate the effect of interpolation as the closest rules found in such situations may be very different from the underlying true relations holding between the antecedent features and the consequent. Nevertheless, in terms of winning cases, running FRI methods still outnumber using the original ANFIS without rule interpolation, despite the reduction in the winning numbers. The question is then whether the proposed OTFRI performs better than the conventional TFRI, which is to be addressed next.

2) Comparison Between TFRI and OTFRI: The main purpose of the proposed approach is to improve upon existing

TABLE V
PERFORMANCE IN TERMS OF RMSE

Data Sets	10% **Missing**			15% **Missing**			20% **Missing**			25% **Missing**		
	ANFIS	**ANFIS_OTFRI**	**ANFIS_TFRI**	**ANFIS**	**ANFIS_OTFRI**	**ANFIS_TFRI**	**ANFIS**	**ANFIS_OTFRI**	**ANFIS_TFRI**	**ANFIS**	**ANFIS_OTFRI**	**ANFIS_TFRI**
airfoil	5.395 ± 1.274	**2.068 ± 1.139**	2.645 ± 1.898	11.484 ± 1.717	7.244 ± 2.140	**1.538 ± 0.576**	10.587 ± 1.771	8.002 ± 2.348	**3.378 ± 1.287**	N/A	N/A	N/A
autoMPG6	N/A	N/A	N/A	N/A	N/A	N/A	2.637 ± 3.164	2.248 ± 1.835	**2.197 ± 1.692**	4.798 ± 1.686	**4.112 ± 2.175**	4.990 ± 2.293
CPU	0.707 ± 0.407	**0.362 ± 0.096**	1.141 ± 0.640	0.702 ± 0.463	**0.478 ± 0.311**	1.187 ± 0.910	0.703 ± 0.462	**0.478 ± 0.311**	1.187 ± 0.910	1.022 ± 0.602	**0.884 ± 0.513**	1.778 ± 1.118
delta_ail	0.011 ± 0.036	**0.002 ± 0.008**	0.109 ± 0.345	N/A	N/A	N/A	0.047 ± 0.149	**0.018 ± 0.056**	0.062 ± 0.197	**0.111 ± 0.234**	0.315 ± 0.596	0.513 ± 0.872
diabetes	1.392 ± 1.926	**1.129 ± 1.489**	1.628 ± 1.492	1.392 ± 1.926	**1.129 ± 1.489**	1.628 ± 1.492	3.483 ± 1.199	3.395 ± 1.036	**1.947 ± 0.999**	3.483 ± 1.199	3.395 ± 1.036	**1.947 ± 0.999**
ele1	**1.930 ± 1.428**	2.090 ± 1.244	2.563 ± 1.464	**1.631 ± 1.334**	1.920 ± 1.228	5.176 ± 6.385	2.135 ± 1.343	**1.743 ± 2.030**	5.411 ± 6.230	2.135 ± 1.343	**1.743 ± 2.030**	5.411 ± 6.230
ele2	0.373 ± 0.156	0.255 ± 0.073	**0.163 ± 0.083**	0.371 ± 0.120	0.255 ± 0.111	**0.148 ± 0.030**	0.336 ± 0.131	0.257 ± 0.115	**0.148 ± 0.032**	0.390 ± 0.125	0.377 ± 0.161	**0.230 ± 0.067**
friedman	N/A	N/A	N/A	N/A	N/A	N/A	N/A	N/A	N/A	N/A	N/A	N/A
laser	1.284 ± 1.220	**0.651 ± 0.392**	2.960 ± 2.188	0.667 ± 0.391	**0.478 ± 0.281**	1.653 ± 1.259	1.561 ± 0.920	**1.220 ± 0.312**	2.617 ± 2.000	1.374 ± 0.493	**1.112 ± 0.337**	2.661 ± 1.349
plastic	**2.606 ± 1.879**	2.683 ± 1.774	2.653 ± 0.709	2.316 ± 1.160	**2.150 ± 0.879**	3.203 ± 1.219	2.951 ± 1.551	2.618 ± 1.061	**2.291 ± 0.842**	2.951 ± 1.551	2.618 ± 1.061	**2.291 ± 0.842**
average	1.7123 ± 1.0406	**1.1551 ± 0.7769**	1.7328 ± 1.1023	2.6518 ± 1.0159	**1.9506 ± 0.9200**	2.0762 ± 1.6959	2.7156 ± 1.1879	2.2200 ± 1.0114	**2.1377 ± 1.5766**	2.0329 ± 0.9040	**1.8194 ± 0.9885**	2.4776 ± 1.7214

TABLE VI
PERFORMANCE IN TERMS OF RMSE (CONTINUED)

Data Sets	30% **Missing**			40% **Missing**			50% **Missing**		
	ANFIS	**ANFIS_OTFRI**	**ANFIS_TFRI**	**ANFIS**	**ANFIS_OTFRI**	**ANFIS_TFRI**	**ANFIS**	**ANFIS_OTFRI**	**ANFIS_TFRI**
airfoil	11.157 ± 1.796	**4.912 ± 1.357**	6.727 ± 1.518	10.778 ± 1.570	8.799 ± 2.005	**4.328 ± 1.101**	10.284 ± 3.101	**7.160 ± 2.202**	11.622 ± 3.190
autoMPG6	4.022 ± 2.638	**3.727 ± 1.597**	4.600 ± 4.269	**4.859 ± 2.793**	5.575 ± 3.232	5.448 ± 2.232	6.656 ± 1.977	7.323 ± 2.567	**5.644 ± 1.671**
CPU	**0.262 ± 0.065**	**0.262 ± 0.065**	1.336 ± 0.910	0.309 ± 0.051	**0.303 ± 0.046**	1.319 ± 0.934	1.246 ± 0.814	**0.969 ± 0.609**	1.463 ± 0.708
delta_ail	**0.062 ± 0.092**	0.220 ± 0.314	0.292 ± 0.434	0.100 ± 0.226	**0.097 ± 0.263**	0.129 ± 0.347	**0.802 ± 0.722**	1.501 ± 1.060	2.152 ± 1.364
diabetes	**3.532 ± 2.876**	3.599 ± 1.714	5.568 ± 4.588	4.089 ± 1.349	**4.085 ± 1.169**	4.253 ± 1.368	4.089 ± 1.349	**4.085 ± 1.169**	4.253 ± 1.368
ele1	**3.575 ± 1.705**	3.832 ± 1.220	4.215 ± 2.408	6.174 ± 2.701	**3.299 ± 1.649**	5.170 ± 1.884	5.948 ± 2.902	**2.716 ± 0.708**	4.274 ± 1.662
ele2	N/A	N/A	N/A	**0.344 ± 0.134**	0.399 ± 0.185	0.853 ± 0.419	0.686 ± 0.276	**0.393 ± 0.193**	1.265 ± 0.600
friedman	2.159 ± 0.834	2.630 ± 1.229	**2.029 ± 0.629**	2.178 ± 1.528	**2.170 ± 1.041**	3.496 ± 1.639	2.971 ± 0.866	**2.883 ± 0.906**	4.334 ± 1.571
laser	1.758 ± 2.227	**1.223 ± 0.602**	1.278 ± 0.535	**1.200 ± 0.177**	1.624 ± 0.411	1.462 ± 0.685	**0.755 ± 0.188**	1.117 ± 0.246	0.900 ± 0.360
plastic	2.612 ± 1.203	**2.308 ± 0.902**	3.359 ± 1.140	**2.501 ± 1.002**	2.597 ± 0.938	3.249 ± 1.243	**2.871 ± 1.986**	2.914 ± 1.883	3.722 ± 2.586
average	3.2377 ± 1.4930	**2.5238 ± 1.0000**	3.2670 ± 1.8257	3.2532 ± 1.1530	**2.8948 ± 1.0938**	2.9706 ± 1.1853	3.6307 ± 1.4180	**3.1061 ± 1.1542**	3.9629 ± 1.5081

TABLE VII
PERFORMANCE OF INTERPOLATED FUZZY RULES

Data Sets	10% **Missing**		15% **Missing**		20% **Missing**		25% **Missing**		30% **Missing**	
	OTFRI	**TFRI**	**OTFRI**	**TFRI**	**OTFRI**	**TFRI**	**OTFRI**	**TFRI**	**OTFRI**	**TFRI**
airfoil	**2.155 ± 1.117**	2.625 ± 1.908	7.634 ± 1.806	**1.630 ± 0.575**	8.630 ± 2.217	**3.586 ± 1.415**	N/A	N/A	**4.719 ± 1.195**	6.818 ± 1.595
autoMPG6	N/A	N/A	N/A	N/A	2.190 ± 1.801	**2.170 ± 1.694**	**4.972 ± 2.626**	5.160 ± 2.292	**3.976 ± 1.391**	4.682 ± 4.340
CPU	**0.777 ± 0.269**	1.141 ± 0.640	**0.800 ± 0.313**	1.189 ± 0.919	**0.798 ± 0.312**	1.189 ± 0.919	**1.226 ± 0.681**	1.791 ± 1.126	**0.865 ± 0.388**	1.336 ± 0.910
delta_ail	**0.037 ± 0.116**	0.209 ± 0.662	N/A	N/A	0.153 ± 0.485	**0.073 ± 0.230**	**0.547 ± 1.126**	0.790 ± 1.369	0.467 ± 0.717	**0.459 ± 0.688**
diabetes	**2.274 ± 1.754**	8.427 ± 4.992	**2.364 ± 1.991**	8.427 ± 4.992	**2.461 ± 2.000**	11.022 ± 11.115	**2.693 ± 2.577**	9.234 ± 6.099	**3.046 ± 2.907**	10.956 ± 5.954
ele1	2.995 ± 1.699	**2.603 ± 1.318**	**3.925 ± 2.576**	5.627 ± 6.390	**4.223 ± 3.212**	6.018 ± 6.285	**4.223 ± 3.212**	6.018 ± 6.285	**4.609 ± 1.052**	4.686 ± 2.869
ele2	0.254 ± 0.069	**0.172 ± 0.080**	0.254 ± 0.108	**0.157 ± 0.031**	0.272 ± 0.111	**0.156 ± 0.032**	0.398 ± 0.168	**0.247 ± 0.079**	N/A	N/A
friedman	N/A	N/A	N/A	N/A	N/A	N/A	N/A	N/A	3.134 ± 1.413	**2.142 ± 0.656**
laser	**0.736 ± 0.646**	3.193 ± 2.250	**0.520 ± 0.291**	1.705 ± 1.304	**1.316 ± 0.351**	2.668 ± 2.035	**1.283 ± 0.492**	2.908 ± 1.459	**1.274 ± 0.652**	1.366 ± 0.592
plastic	**2.831 ± 1.738**	3.491 ± 0.748	**2.276 ± 0.906**	4.345 ± 1.509	3.482 ± 1.441	**3.163 ± 1.178**	3.482 ± 1.441	**3.163 ± 1.178**	**2.308 ± 0.924**	4.224 ± 1.402
average	**1.5074 ± 0.9260**	2.7327 ± 1.5749	**2.5391 ± 1.1416**	3.2969 ± 2.2458	**2.6141 ± 1.3255**	3.3381 ± 2.7669	**2.3531 ± 1.5407**	3.6637 ± 2.4858	**2.7110 ± 1.1820**	4.0742 ± 2.1119

techniques for TFRI. This can also be demonstrated by examining the results of Tables V and VI. From the viewpoint of overall performance, as reflected by the bottom row in each of these two tables, other than the situation where the missing rule proportion is 20% in which the average performance of OTFRI (2.2200) is very slightly worse than that of TFRI (2.1377), OTFRI achieves the best average results under all other settings. Even in the 20% case, OTFRI has a much less standard deviation (±1.0114 versus ±1.5766). From the viewpoint of individual winners per dataset, OTFRI also clearly outperforms TFRI with 33 versus 13 wins across all settings.

Note that TSK systems equivalently compute the final output based on a weighted combination of firing all matched rules in the rule base. Thus, it is interesting to examine the errors of the unmatched instances using interpolated fuzzy rules only, discounting the contributions made by firing those rules from a given sparse rule base. This is in order to verify whether the combination of the original reduced rule base and interpolated fuzzy rules (by either method) happens to work well. Tables VII and VIII list the differences between the ground truth and the results of firing just the interpolated rules, in terms of the resulting RMSEs, using the same experimental settings as those used to obtained the results of Tables V and VI.

The average errors of the interpolated results by OTFRI are always lower than those achievable by TFRI under all settings, beating TFRI systematically without a single loss. From the viewpoint of winners over the individual datasets under various missing rule settings, OTFRI achieves 43 wins against 18 wins obtained by TFRI. Both of these general outcomes are consistent with those shown earlier in Tables V and VI (which are obtained on the basis of the errors measured over the entire rule base, including the interpolated rules). This is not surprising, since

TABLE VIII
PERFORMANCE OF INTERPOLATED FUZZY RULES (CONTINUED)

	40% Missing		50% Missing	
Data Sets	**OTFRI**	**TFRI**	**OTFRI**	**TFRI**
airfoil	8.943 ± 2.229	**4.589 ± 1.195**	**7.476 ± 2.311**	11.858 ± 3.230
autoMPG6	6.736 ± 3.117	**5.678 ± 2.193**	8.392 ± 2.728	**5.740 ± 1.663**
CPU	**1.127 ± 1.034**	1.319 ± 0.934	**1.292 ± 0.562**	1.478 ± 0.714
delta_ail	**0.103 ± 0.312**	0.241 ± 0.583	**1.965 ± 1.304**	2.586 ± 1.592
diabetes	**1.791 ± 2.101**	11.177 ± 6.094	**4.827 ± 2.134**	12.692 ± 8.025
ele1	**3.077 ± 1.356**	5.289 ± 2.042	**2.356 ± 0.846**	4.274 ± 1.635
ele2	**0.432 ± 0.217**	1.048 ± 0.490	**0.407 ± 0.196**	1.286 ± 0.608
friedman	**2.627 ± 1.490**	3.742 ± 1.772	**3.464 ± 1.257**	4.554 ± 1.694
laser	1.767 ± 0.454	**1.555 ± 0.736**	1.722 ± 0.536	**0.989 ± 0.396**
plastic	**2.978 ± 0.928**	3.559 ± 1.301	**3.337 ± 1.912**	4.032 ± 2.677
average	**2.9580 ± 1.3238**	3.8197 ± 1.7341	**3.5237 ± 1.3785**	4.9491 ± 2.2235

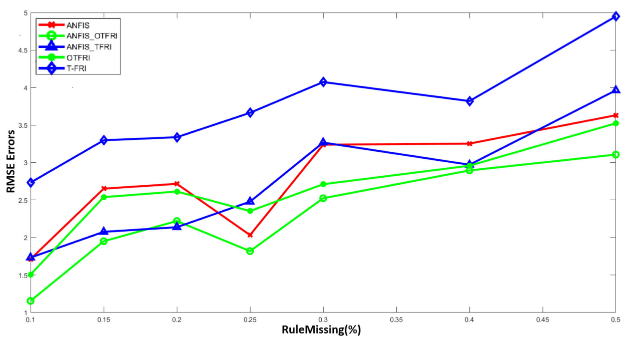


Fig. 6. Overall performance under various settings.

the errors across the whole rule base are more affected by the interpolated rules than by the rules that do not fire the unmatched instances to sufficient degrees (smaller than 0.5^n). These results collectively demonstrate that OTFRI improves upon the conventional TFRI method.

Fig. 6 depicts the overall average errors for different approaches under various settings. In general, the error becomes larger as the proportion of missing rules gets larger, but again this is expected since the uncovered problem space becomes larger. Occasional oscillations are due to the randomness in the removal of the original rules. However, what is important is the general outcome in that the error rates of ANFIS_OTFRI and OTFRI are both below those of their counterparts (i.e., ANFIS_TFRI and TFRI), showing that OTFRI significantly improves over the existing TFRI approach. Such significance in performance improvement is statistically verified below.

3) Statistical Tests: The performance enhancement of the proposed OTFRI over TFRI is further supported with the corresponding pairwise t-test outcomes ($p < 0.05$), as shown in Table IX. Regardless of the missing percentage of rules, OTFRI always statistically achieves more or at least equal number of winners as compared to TFRI, for each individual setting without a single exception, be it evaluated using interpolated rules only or in conjunction with the sparse rule base. Overall, when considering the results obtained on the basis of a given rule base plus the corresponding interpolated fuzzy rules, OTFRI achieves 28 wins, 21 ties, and 12 losses. When the interpolated fuzzy rules are evaluated alone without counting in any contribution of the existing sparse rule base, OTFRI performs even better with 31 wins, 20 ties, and 11 losses. Once again, these results clearly show that OTFRI significantly improves over the existing TFRI method.

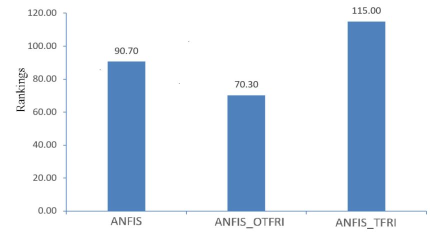


Fig. 7. Rankings of ANFIS, ANFIS_OTFRI, and ANFIS_TFRI.

To further validate the overall superior performance that OTFRI possesses over its alternatives, nonparametric statistical tests are also employed here. In particular, the Friedman Aligned Ranks test [46] is applied to detect whether there is indeed any statistically significant difference among the algorithms (namely, ANFIS, ANFIS_OTFRI, and ANFIS_TFRI) as a group. Before conducting this test, the results across all settings including 61 valid pairs are stacked together for ANFIS, ANFIS_OTFRI, and ANFIS_TFRI, with the corresponding average RMSEs listed in Table X. From the above, the Friedman Aligned Ranks test is applied with the rankings calculated, as shown in Fig. 7, where the bars are proportional to the average ranking obtained for each named algorithm. The lowest bar (which corresponds to the best algorithm statistically) achieved by ANFIS_OTFRI agrees with the smallest error that is also obtained by it, as of Table X. To examine whether significant differences indeed exist among the average errors, parameters associated with the Friedman Aligned Ranks test outputs are shown in Table XI, where the p value indicates the probability to reject the null hypothesis that there is no significant difference among the three average performances. At the significance level of $\alpha = 0.05$, the null hypothesis is rejected, indicating that there exist significant statistical differences among the results attainable by the members of the group concerned.

The Friedman Aligned Ranks test can detect significant differences within a certain group. However, it is unable to establish explicit comparisons when considering a particular control method and a set of possible alternatives. As ANFIS_OTFRI achieves the smallest error and is of the lowest ranking bar among the three compared algorithms, it is of a natural appeal to be used as the control method in comparison to ANFIS and ANFIS_TFRI. The standard Holm's procedure [46] is applied to run the test, computing the adjusted p values. The results of this investigation are presented in Table XII. Since both p values are smaller than the level of significance specified by $\alpha = 0.05$, the null hypothesis that there exists no significant performance difference between ANFIS_OTFRI and ANFIS or between ANFIS_OTFRI and ANFIS_TFRI is rejected. Thus, it can be concluded statistically that ANFIS_OTFRI significantly improves upon both ANFIS and ANFIS_TFRI.

Further to comparing at the overall rule base level, it is also interesting to investigate the performance when only interpolated rules are used. The Wilcoxon signed-rank test [47] is employed to detect whether significant differences exist between two sample means over the errors due to the use of interpolated fuzzy rules introduced by TFRI or those by OTFRI. From the statistical point of view, this test may be more reliable than t-test,

TABLE IX
COMPARISON BETWEEN OTFRI AND TFRI, WHERE b, =, AND w INDICATE OTFRI ACHIEVING STATISTICALLY BETTER, EQUIVALENT, AND WORSE PERFORMANCE, RESPECTIVELY

Data Sets	Use interpolated rules with sparse rule base							Use of interpolated rules only						
	10%	15%	20%	25%	30%	40%	50%	10%	15%	20%	25%	30%	40%	50%
airfoil	=	w	w	N/A	b	w	b	=	w	w	N/A	b	w	b
autoMPG6	N/A	N/A	=	=	b	=	w	N/A	N/A	=	=	=	=	w
CPU	b	b	b	b	b	b	=	b	=	b	b	b	b	=
delta_ail	b	N/A	=	b	=	b	b	b	N/A	b	=	=	b	b
diabetes	=	=	w	w	=	=	=	b	b	b	b	b	b	b
ele1	=	b	b	w	=	b	b	=	b	=	=	=	b	b
ele2	w	w	w	w	N/A	w	b	w	w	w	w	N/A	b	b
friedman	N/A	N/A	N/A	N/A	=	b	b	N/A	N/A	N/A	N/A	w	b	b
laser	b	b	b	b	=	=	=	b	b	b	b	=	=	w
plastic	=	b	=	=	b	b	b	w	b	=	=	b	=	=
Summary(b/=/w)	(3/4/1)	(4/1/2)	(3/3/3)	(3/2/3)	(4/5/0)	(5/3/2)	(6/3/1)	(4/2/2)	(4/1/2)	(4/3/2)	(3/4/1)	(4/4/1)	(6/3/1)	(6/2/2)

TABLE X
AVERAGE RMSE AT RULE BASE LEVEL

Methods	ANFIS	ANFIS_OTFRI	ANFIS_TFRI
Average RMSE	2.8024	2.2976	2.7245

TABLE XI
FRIEDMAN ALIGNED RANKS TEST RESULT

Comparison	Hypothesis ($\alpha = 0.05$)	p value	statistic
ANFIS, ANFIS_OTFRI, ANFIS_TFRI	Reject	4.60e-04	15.3581

TABLE XII
OUTCOME OF HOLM'S PROCEDURE WITH ANFIS_OTFRI BEING THE CONTROL METHOD

Comparison	Hypothesis ($\alpha = 0.05$)	Adjusted p value	statistic
ANFIS_OTFRI v.s. ANFIS	Reject	0.0335	2.1262
ANFIS_OTFRI v.s. ANFIS_TFRI	Reject	1.00e-05	4.6600

TABLE XIII
AVERAGE RMSE OF INTERPOLATED FUZZY RULES

Methods	OTFRI	TFRI
Average RMSE	2.6459	3.7483

TABLE XIV
WILCOXON TEST ON OTFRI AND TFRI

Comparison	Hypothesis ($\alpha = 0.05$)	p value	signrank	z value
OTFRI v.s. TFRI	Reject	1.5129e-04	418	-3.7889

as it does not assume normal distributions of the samples while being more robust for situations where exceptionally good or bad performances on a few datasets may occur.

Similar to comparisons at rule base level, the results across all settings including 61 valid pairs are stacked together for OTFRI and TFRI, with the corresponding average RMSEs presented in Table XIII. Applying the Wilcoxon signed-rank test leads to the outcomes as given in Table XIV, where the p value represents the probability that two pieces of information under comparison are of statistically equal significance. As p is much smaller than the predefined significance level (0.05), the null hypothesis that there is no statistical difference between running the two sets of interpolated rules is thus rejected. That is to say, OTFRI significantly beats TFRI in producing quality interpolated rules, reassuring the previous evaluation outcome.

VI. CONCLUSION

TFRI is a popular approach to performing inference with sparse rule bases. Existing techniques require the user to specify the number of the rules that are selected, with the employment of a distance metric, to build the intermediate fuzzy rules. They also require a move operation that forces a generated intermediate rule to overlap with an unmatched observation. This article has proposed an automatic rule selection procedure and an associated mechanism to automatically assemble the intermediate rules, without involving the move operation. In particular, the required rule weighting scheme is computed by interpreting the solution of a linear equation system as a quadratic programming problem, which is then resolved using the classical active set method. Systematic comparative experimental results have demonstrated statistically the significant performance improvement achieved by the proposed approach over conventional TFRI.

The present research is focused on the application to TSK fuzzy systems. The transference of the underlying approach to suit Mamdani inference systems forms an immediate next piece of further work. Working with Mamdani rules, which are of fuzzy consequents instead of crisp values directly produced by the TSK-style rules, would allow for a more general representation of the inferred output. This is facilitated by the employment of fuzzy values to describe the vague consequents in response to imprecise observations. Thus, an extension of the present work to Mamdani systems would help attain interpretability for situations, where a linguistic inference outcome is sought, as with other typical FRI methods in the literature [48]. Also, the current approach presumes the use of a fixed (sparse) rule base. It is very interesting to investigate how such a static rule base may be enriched through an integration with dynamic FRI [19]. Finally, the present experimental studies are based on the use of a fixed, even partition of the feature domains. Interpolation performance may be further improved if a learning mechanism can be built into the system to automatically construct the required partitions, say using a fuzzy clustering method [49].

ACKNOWLEDGMENT

The authors are grateful to the reviewers for their endorsement in this research and for their comments, which have helped improve the quality of this work.

REFERENCES

[1] R. Alcalá, M. J. Gacto, and F. Herrera, "A fast and scalable multiobjective genetic fuzzy system for linguistic fuzzy modeling in high-dimensional regression problems," *IEEE Trans. Fuzzy Syst.*, vol. 19, no. 4, pp. 666–681, Aug. 2011.

[2] 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, Aug. 2015.

[3] J. Alcalá-Fdez, R. Alcalá, S. González, Y. Nojima, and S. García, "Evolutionary fuzzy rule-based methods for monotonic classification," *IEEE Trans. Fuzzy Syst.*, vol. 25, no. 6, pp. 1376–1390, Dec. 2017.

[4] M. Antonelli, D. Bernardo, H. Hagras, and F. Marcelloni, "Multiobjective evolutionary optimization of type-2 fuzzy rule-based systems for financial data classification," *IEEE Trans. Fuzzy Syst.*, vol. 25, no. 2, pp. 249–264, Apr. 2017.

[5] Y. Fan, G. Huang, K. Huang, and B. W. Baetz, "Planning water resources allocation under multiple uncertainties through a generalized fuzzy two-stage stochastic programming method," *IEEE Trans. Fuzzy Syst.*, vol. 23, no. 5, pp. 1488–1504, Oct. 2015.

[6] P. Su, C. Shang, T. Chen, and Q. Shen, "Exploiting data reliability and fuzzy clustering for journal ranking," *IEEE Trans. Fuzzy Syst.*, vol. 25, no. 5, pp. 1306–1319, Oct. 2017.

[7] T. Nguyen, A. Khosravi, D. Creighton, and S. Nahavandi, "Classification of healthcare data using genetic fuzzy logic system and wavelets," *Expert Syst. Appl.*, vol. 42, no. 4, pp. 2184–2197, 2015.

[8] D. Tikk, I. Joó, L. Kóczy, P. Várlaki, B. Moser, and T. D. Gedeon, "Stability of interpolative fuzzy KH controllers," *Fuzzy Sets Syst.*, vol. 125, no. 1, pp. 105–119, 2002.

[9] L. Kóczy and K. Hirota, "Approximate reasoning by linear rule interpolation and general approximation," *Int. J. Approx. Reason.*, vol. 9, no. 3, pp. 197–225, 1993.

[10] L. Kóczy and K. Hirota, "Interpolative reasoning with insufficient evidence in sparse fuzzy rule bases," *Inf. Sci.*, vol. 71, nos. 1/2, pp. 169–201, 1993.

[11] K. W. Wong, D. Tikk, T. D. Gedeon, and L. T. Kóczy, "Fuzzy rule interpolation for multidimensional input spaces with applications: A case study," *IEEE Trans. Fuzzy Syst.*, vol. 13, no. 6, pp. 809–819, Dec. 2005.

[12] S.-M. Chen and Y.-K. Ko, "Fuzzy interpolative reasoning for sparse fuzzy rule-based systems based on α-cuts and transformations techniques," *IEEE Trans. Fuzzy Syst.*, vol. 16, no. 6, pp. 1626–1648, Dec. 2008.

[13] L. Kóczy and K. Hirota, "Ordering, distance and closeness of fuzzy sets," *Fuzzy Sets Syst.*, vol. 59, no. 3, pp. 281–293, 1993.

[14] B. Bouchon-Meunier, R. Mesiar, C. Marsala, and M. Rifqi, "Compositional rule of inference as an analogical scheme," *Fuzzy Sets Syst.*, vol. 138, no. 1, pp. 53–65, 2003.

[15] P. Baranyi, L. T. Kóczy, and T. D. Gedeon, "A generalized concept for fuzzy rule interpolation," *IEEE Trans. Fuzzy Syst.*, vol. 12, no. 6, pp. 820–837, Dec. 2004.

[16] Z. Huang and Q. Shen, "Fuzzy interpolation and extrapolation: A practical approach," *IEEE Trans. Fuzzy Syst.*, vol. 16, no. 1, pp. 13–28, Feb. 2008.

[17] S.-M. Chen and S. I. Adam, "Adaptive fuzzy interpolation based on ranking values of interval type-2 polygonal fuzzy sets," *Inf. Sci.*, vol. 435, pp. 320–333, 2018.

[18] F. Li, S. Changjing, Y. Li, J. Yang, and Q. Shen, "Fuzzy rule-based interpolative reasoning supported by attribute ranking," *IEEE Trans. Fuzzy Syst.*, vol. 26, no. 5, pp. 2758–2773, Oct. 2018.

[19] N. Naik, R. Diao, and Q. Shen, "Dynamic fuzzy rule interpolation and its application to intrusion detection," *IEEE Trans. Fuzzy Syst.*, vol. 26, no. 4, pp. 1878–1892, Aug. 2018.

[20] L. T. Koczy, "Interpolation in homogenous fuzzy signature rule bases," in *Proc. IEEE Int. Conf. Fuzzy Syst.*, 2017, pp. 1–6.

[21] L. Yang, F. Chao, and Q. Shen, "Generalized adaptive fuzzy rule interpolation," *IEEE Trans. Fuzzy Syst.*, vol. 25, no. 4, pp. 839–853, Aug. 2017.

[22] I. Perfilieva, D. Dubois, H. Prade, F. Esteva, L. Godo, and P. Hoďáková, "Interpolation of fuzzy data: Analytical approach and overview," *Fuzzy Sets Syst.*, vol. 192, pp. 134–158, 2012.

[23] S. Jin, R. Diao, C. Quek, and Q. Shen, "Backward fuzzy rule interpolation," *IEEE Trans. Fuzzy Syst.*, vol. 22, no. 6, pp. 1682–1698, Dec. 2014.

[24] S.-M. Chen and Y.-C. Chang, "Weighted fuzzy rule interpolation based on GA-based weight-learning techniques," *IEEE Trans. Fuzzy Syst.*, vol. 19, no. 4, pp. 729–744, Aug. 2011.

[25] S.-M. Chen, W.-C. Hsin, S.-W. Yang, and Y.-C. Chang, "Fuzzy interpolative reasoning for sparse fuzzy rule-based systems based on the slopes of fuzzy sets," *Expert Syst. Appl.*, vol. 39, no. 15, pp. 11961–11969, 2012.

[26] A. M. Bica, "The middle-parametric representation of fuzzy numbers and applications to fuzzy interpolation," *Int. J. Approx. Reason.*, vol. 68, pp. 27–44, 2016.

[27] E. Hosseini and A. Jafarian, "Bivariate fully fuzzy interpolation problem using artificial neural networks approach," *J. Intell. Fuzzy Syst.*, vol. 30, no. 4, pp. 2267–2275, 2016.

[28] Y. Yam and L. T. Kóczy, "Representing membership functions as points in high-dimensional spaces for fuzzy interpolation and extrapolation," *IEEE Trans. Fuzzy Syst.*, vol. 8, no. 6, pp. 761–772, Dec. 2000.

[29] S.-M. Chen and W.-C. Hsin, "Weighted fuzzy interpolative reasoning based on the slopes of fuzzy sets and particle swarm optimization techniques," *IEEE Trans. Cybern.*, vol. 45, no. 7, pp. 1250–1261, Jul. 2015.

[30] Z. C. Johanyák and S. Kovács, "Fuzzy rule interpolation based on polar cuts," in *Computational Intelligence, Theory and Applications*. Berlin, Germany: Springer, 2006, pp. 499–511.

[31] Y. Bai and D. Wang, "On the comparison of trilinear, cubic spline, and fuzzy interpolation methods in the high-accuracy measurements," *IEEE Trans. Fuzzy Syst.*, vol. 18, no. 5, pp. 1016–1022, Oct. 2010.

[32] W.-H. Chen and B.-S. Chen, "Robust stabilization design for nonlinear stochastic system with Poisson noise via fuzzy interpolation method," *Fuzzy Sets Syst.*, vol. 217, pp. 41–61, 2013.

[33] L. Yang and Q. Shen, "Closed form fuzzy interpolation," *Fuzzy Sets Syst.*, vol. 225, pp. 1–22, 2013.

[34] S. Kovacs and L. T. Koczy, "Approximate fuzzy reasoning based on interpolation in the vague environment of the fuzzy rulebase," in *Proc. IEEE Int. Conf. Intell. Eng. Syst.*, 1997, pp. 63–68.

[35] D. Vincze and S. Kovács, "Performance optimization of the fuzzy rule interpolation method 'FIVE,'" *J. Adv. Comput. Intell. Intell. Inform.*, vol. 15, no. 3, pp. 313–320, 2011.

[36] P. E. Gill, W. Murray, and M. H. Wright, *Practical Optimization*. London, U.K.: Academic, 1981.

[37] P. E. Gill, W. Murray, M. A. Saunders, and M. H. Wright, "Procedures for optimization problems with a mixture of bounds and general linear constraints," *ACM Trans. Math. Softw.*, vol. 10, no. 3, pp. 282–298, 1984.

[38] P. E. Gill, W. Murray, M. A. Saunders, and M. H. Wright, *Numerical Linear Algebra and Optimization*, vol. 1. Redwood City, CA, USA: Addison-Wesley, 1991.

[39] T. Coleman, M. A. Branch, and A. Grace, "Optimization toolbox," *For Use With MATLAB. Users Guide for MATLAB 5, Version 2, Relaese II*, 1999.

[40] P. Hungerlander and F. Rendl, "A feasible active set method for strictly convex quadratic problems with simple bounds," *SIAM J. Optim.*, vol. 25, no. 3, pp. 1633–1659, 2015.

[41] A. Forsgren, P. E. Gill, and E. Wong, "Primal and dual active-set methods for convex quadratic programming," *Math. Programming*, vol. 159(1-2), pp. 469–508, 2016.

[42] K. Bache and M. Lichman, "UCI machine learning repository," 2013. [Online]. Available: http://archive.ics.uci.edu/ml

[43] J. Alcalá *et al.*, "Keel data-mining software tool: Data set repository, integration of algorithms and experimental analysis framework," *J. Multiple-Valued Logic Soft Comput.*, vol. 17, nos. 2/3, pp. 255–287, 2010.

[44] J.-S. Jang, "ANFIS: Adaptive-network-based fuzzy inference system," *IEEE Trans. Syst., Man, Cybern.*, vol. 23, no. 3, pp. 665–685, May/Jun. 1993.

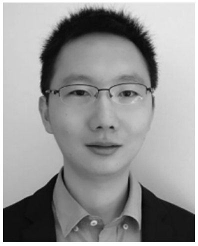
[45] W. E. Combs and J. E. Andrews, "Combinatorial rule explosion eliminated by a fuzzy rule configuration," *IEEE Trans. Fuzzy Syst.*, vol. 6, no. 1, pp. 1–11, Feb. 1998.

[46] J. Demšar, "Statistical comparisons of classifiers over multiple data sets," *J. Mach. Learn. Res.*, vol. 7, pp. 1–30, Jan. 2006.

[47] D. J. Sheskin, *Handbook of Parametric and Nonparametric Statistical Procedures*. Boca Raton, FL, USA: CRC Press, 2003.

[48] S. Kovács, "Extending the concept of fuzzy rule interpolation with the interpolation of fuzziness," in *Proc. IEEE Int. Conf. Fuzzy Syst.*, 2012, pp. 1–8.

[49] F. Klawonn, R. Kruse, and R. Winkler, "Fuzzy clustering: More than just fuzzification," *Fuzzy Sets Syst.*, vol. 281, pp. 272–279, 2015.



Tianhua Chen received the B.Sc. degree in software engineering from Fujian Normal University, Fuzhou, China, in 2012, and the M.Sc. degree in intelligent systems and Ph.D. degree in computer science from Aberystwyth University, Aberystwyth, U.K., in 2013 and 2017, respectively.

He is currently a Lecturer with the Department of Computer Science, School of Computing and Engineering, University of Huddersfield, Huddersfield, U.K. His research interests include machine learning, computational intelligence, and their applications to healthcare intelligence.

Dr. Chen served as a Lead Guest Editor for the special issue on “Medical Analytics for Healthcare Intelligence” in *Artificial Intelligence in Medicine*.

Changjing Shang received the Ph.D. degree in computing and electrical engineering from Heriot-Watt University, Edinburgh, U.K., in 1995.

She is currently a University Research Fellow with the Department of Computer Science, Faculty of Business and Physical Sciences, Aberystwyth University, Aberystwyth, U.K. Prior to joining Aberystwyth University, she was with Heriot-Watt University, Loughborough University, and Glasgow University. She has published extensively and supervised more than ten Ph.D.s/PDRAs in the areas of pattern recognition, data mining and analysis, space robotics, and image modeling and classification.

Jing Yang received the B.Sc. degree in information and computing science from Taiyuan Normal University, Taiyuan, China, in 2012, and the M.Sc. degree in computational mathematics from Northwestern Polytechnical University, Xi’an, China, in 2015. She is currently working toward the Ph.D. degree in computational intelligence jointly with the School of Computer Science and Engineering, Northwestern Polytechnical University, and the Department of Computer Science, Institute of Mathematics, Physics and Computer Science, Aberystwyth University, Aberystwyth, U.K.

Her research interests include fuzzy systems, machine learning, and image super resolution.

Fangyi Li received the B.Sc. degree in computer science and technology from Northwestern Polytechnical University, Xi’an, China, in 2014. She is currently working toward a joint Ph.D. degree in computational intelligence with the School of Computer Science and Engineering, Northwestern Polytechnical University, and the Department of Computer Science, Faculty of Business and Physical Sciences, Aberystwyth University, Aberystwyth, U.K.

Her research interests include fuzzy interpolative reasoning, machine learning, and image classification.

Qiang Shen received the Ph.D. degree in computing and electrical engineering from Heriot-Watt University, Edinburgh, U.K., in 1990, and the D.Sc. degree in computational intelligence from Aberystwyth University, Aberystwyth, U.K., in 2013.

He is the Chair of Computer Science and the Pro Vice-Chancellor Faculty of Business and Physical Sciences, Aberystwyth University. He has authored two research monographs and more than 390 peer-reviewed papers.

Dr. Shen was a recipient of the Outstanding Transactions Paper Award from the IEEE.