New Type-2 Rule Ranking Indices for Designing Parsimonious Interval Type-2 Fuzzy Logic Systems

Shang-Ming Zhou, Robert John, Francisco Chiclana and Jonathan M. Garibaldi

Abstract— In this paper, we propose two novel indices for type-2 fuzzy rule ranking to identify the most influential fuzzy rules in designing type-2 fuzzy logic systems, and name them as R-values and c-values of fuzzy rules separately. The R-values of type-2 fuzzy rules are obtained by applying QR decomposition in which there is no need to estimate a rank as required in the SVD-QR with column pivoting algorithm. The c-values of type-2 fuzzy rules are suggested to rank rules based on the effects of rule consequents. Experimental results on a signal recovery problem have shown that by using the proposed indices the most influential type-2 fuzzy rules can be effectively selected to construct parsimonious type-2 fuzzy models while the system performances are kept at a satisfied level.

I. INTRODUCTION

Type-2 fuzzy sets initially suggested by Zadeh in 1975 [1] offer an opportunity to model higher level uncertainty in the human decision making process than the type-1 fuzzy sets [2][3]. As one of the major research topics in fuzzy system community, type-2 fuzzy sets and type-2 fuzzy logic system (T2FLS) have achieved many successful applications in various areas where uncertainties occur such as in decision making [4][5], diagnostic medicine [6], signal processing [7][8], traffic forecasting [9], mobile robot control [10], pattern recognition [11][12], intelligent control [13][14]. However, one challenge in type-1 fuzzy systems remains in T2FLSs, that is, the *curse of dimensionality*: the number of fuzzy rules required increases exponentially with the input space dimension in grid partition. Additional challenge in T2FLS modelling is that there is a higher computational overhead than type-1 FLS modelling [2]. Hence, complexity reduction techniques are urgently needed for T2FLS modelling. As a matter of fact, even in type-1 fuzzy logic system (FLS) modelling, developing parsimonious fuzzy modelling technique with as few fuzzy rules as possible is a very important research topic [15][16][17][20]. Interestingly, Liang and Mendel in the first place suggested to design parsimonious interval T2FLS (IT2FLS) by using SVD-QR decomposition method to perform rule reduction [18]. However, one issue arising in applying the SVD-QR with column pivoting algorithm to fuzzy rule reduction is the estimation of an effective rank.

In order to avoid the estimation of the rank for the SVD-QR with column pivoting algorithm, this paper proposes to apply the pivoted QR decomposition algorithm to the type-2 fuzzy rule reduction to obtain the R-values of fuzzy rules for rule ranking. But both the pivoted QR decomposition algorithm and the SVD-QR with column pivoting algorithm only employ the information from the premise parts of fuzzy rules for rule reduction, and ignore the information from the fuzzy rule consequent parts. In type-1 fuzzy system modelling, some researchers have proposed fuzzy rule reduction methods by considering the contributions of the rule consequent parts [19] [20]. In this paper, we suggest c-values of fuzzy rules with the consideration of effects of type-2 rule consequents to perform the rule reduction.

The organization of this paper is as follows. In section 2, the T2FLS is reviewed with emphasis on IT2FLS whose fuzzy sets with the secondary membership grades are unity. In section 3, we propose some new rule ranking indices for T2FLS in the interests of performing rule reduction. Section 4 gives the experimental results and section 5 concludes this paper.

II. OVERVIEW OF TYPE-2 FUZZY LOGIC SYSTEMS

Similar to the type-1 Mamdani FLS, a type-2 Mamdani FLS consists of five components including fuzzifier, rule base, fuzzy inference process, (type-reducer and) defuzzifier as depicted in Figure 1. Different from the type-1 FLS, in a T2FLS at least some of the fuzzy sets used in the antecedent and/or consequent parts and each rule inference output are type-2 fuzzy sets. Generally speaking, the T2FLS works as follows, the crisp input values first feed into the system through the fuzzifier by which the fuzzification of these inputs is carried out in singleton or non-singleton manners. The fuzzified type-2 fuzzy sets then activate the inference engine and rule base to yield output type-2 fuzzy sets by performing the union and intersection operations of type-2 fuzzy set and compositions of type-2 relations. Then a type reduction process is applied to these output sets in the interests of generating a type-1 set by combining these output sets and performing a centroid calculation. Finally, the type reduced type-1 set is defuzzified to produce crisp outputs.

Specifically speaking, consider a type-2 Mamdani FLS having n inputs $x_1 \in X_1, \cdots, x_n \in X_n$ and one output $y \in Y$, the rule base contains M type-2 fuzzy rules expressed in the following form:

$$R^l : \ if \ x_1 \ is \ \tilde{F}_1^l \ and \ \cdots and \ x_n \ is \ \tilde{F}_n^l, \ then \ y \ is \ \tilde{G}^l \quad (1)$$

where $l = 1, \cdots, M$, $\tilde{F}_i^l$ and $\tilde{G}^l$ are type-2 fuzzy sets. These rules represents fuzzy relations between the multiple

Shang-Ming Zhou, Robert John, and Francisco Chiclana are with the Centre for Computational Intelligence, School of Computing, De Montfort University, Leicester, LE1 9BH, UK (email: smzhou@dmu.ac.uk, smzhou@ieee.org; rij@dmu.ac.uk; chiclana@dmu.ac.uk).

Jonathan M. Garibaldi is with the School of Computer Science and IT, University of Nottingham, Wollaton Road, Nottingham, NG8 1BB, UK (email: jmg@cs.nott.ac.uk).

Fig. 1. The general structure of a type-2 Mamdani FLS

dimensional input space $X \triangleq X_1 \times \cdots \times X_n$ and the output space Y. Given an input x, the singleton or non-singleton fuzzifier [2] can be used to map x into a type-2 fuzzy set $\tilde{A}_x$. The inference engine combines the above rules and maps $\tilde{A}_x$ into an output type-2 set $\tilde{B}^l$ by using the extended sup-star composition principle [2],

$$\mu_{\tilde{B}^l}(y) = \mu_{\tilde{A}_x \circ R^l}(y) = \sqcup_{x \in X} \left[\mu_{\tilde{A}_x}(x) \sqcap \mu_{R^l}(x, y) \right] \quad (2)$$

where $\sqcup$ and $\sqcap$ are join and meet operators respectively, and

$$\mu_{R^l}(x, y) = \left[\sqcap_{i=1}^n \mu_{\tilde{F}_i^l}(x) \right] \sqcap \mu_{\tilde{G}^l}(y) \quad (3)$$

The centroid type-reducer combines all the rule output type-2 fuzzy sets $\tilde{B}^l$ to find their union. Whereas the union of type-2 fuzzy sets corresponds to computing the join of their secondary membership functions

$$\mu_{\tilde{B}}(y) = \sqcup_{l=1}^M \mu_{\tilde{B}^l}(y) \quad (4)$$

The centroid type-reducer carries on calculating the centroid of $\tilde{B}$, $Y_c(x)$, which is a type-1 fuzzy set, in the following,

$$Y_c(x) = \int_{\theta_1 \in J_{y_1}} \cdots \int_{\theta_N \in J_{y_N}} f_{y_1}(\theta_1) * \cdots * \\ f_{y_N}(\theta_N) \Bigg/ \frac{\sum_{i=1}^N y_i \theta_i}{\sum_{i=1}^N \theta_i} \quad (5)$$

where the y-domain is discretised into N points $\{y_1, \cdots, y_N\}$, J_{y_i} are the primary memberships of y_i, and $f_{y_i}(\theta_i) = \mu_{\tilde{B}}(y_i, \theta_i)$ representing the secondary membership grade of y_i on $\theta_i \in J_{y_i} \subseteq [0, 1]$.

At first sight, the expressions of the above general T2FLS are elegant and simple, however, the computing load involved is huge in practice, which has become a major factor in curtailing applications of T2FLS. For an IT2FLS in which the fuzzy sets $\tilde{F}_i^l$ and $\tilde{G}^l$ are the interval fuzzy sets, the computing of T2FLS can be greatly simplified. The membership grades of interval fuzzy sets can be fully characterised by their lower and upper membership grades of the footprint of uncertainty (FOU) separately [2], without

loss of generality, let $\mu_{\tilde{F}_i^l}(x) = \left[\underline{\mu}_{\tilde{F}_i^l}(x), \overline{\mu}_{\tilde{F}_i^l}(x) \right]$ and $\mu_{\tilde{G}^l}(y) = \left[\underline{\mu}_{\tilde{G}^l}(y), \overline{\mu}_{\tilde{G}^l}(y) \right]$ for each sample (x, y). The firing set of an IT2FLS $\mu_{\tilde{F}^l}(x) \triangleq \sqcap_{i=1}^n \mu_{\tilde{F}_i^l}(x)$ is an interval set [2], i.e.,

$$\mu_{\tilde{F}^l}(x) = \left[\underline{f}^l(x), \overline{f}^l(x) \right] \quad (6)$$

where

$$\underline{f}^l(x) = \underline{\mu}_{\tilde{F}_1^l}(x) * \cdots * \underline{\mu}_{\tilde{F}_n^l}(x) \quad (7)$$

$$\overline{f}^l(x) = \overline{\mu}_{\tilde{F}_1^l}(x) * \cdots * \overline{\mu}_{\tilde{F}_n^l}(x) \quad (8)$$

and $*$ is the t-norm operator like minimum or product. In this paper, the singleton fuzzifier is used in the type-2 fuzzy inference process. The centroid of the type-2 interval consequent set $\tilde{G}^i$ is an interval set calculated as follows,

$$C_{\tilde{G}^l} \triangleq [\underline{y}^l, \overline{y}^l] = \int_{\theta_1 \in \left[\underline{\mu}_{\tilde{G}^i}(y_1), \overline{\mu}_{\tilde{G}^i}(y_1) \right]} \cdots \\ \int_{\theta_N \in \left[\underline{\mu}_{\tilde{G}^i}(y_N), \overline{\mu}_{\tilde{G}^i}(y_N) \right]} 1 \Bigg/ \frac{\sum_{i=1}^N y_i \theta_i}{\sum_{i=1}^N \theta_i} \quad (9)$$

for the discretised y-domain $\{y_1, \cdots, y_N\}$. The IT2FLS output set via type-reduction, $Y_c(x)$, is also an interval set having the following structure [2]:

$$Y_c(x) \triangleq [y_l, y_r] = \int_{y^1 \in [\underline{y}^1, \overline{y}^1]} \cdots \int_{y^M \in [\underline{y}^M, \overline{y}^M]} \int_{f^1 \in [\underline{f}^1, \overline{f}^1]} \cdots \\ \int_{f^M \in [\underline{f}^M, \overline{f}^M]} 1 \Bigg/ \frac{\sum_{i=1}^M y^i f^i}{\sum_{i=1}^M f^i} \quad (10)$$

Then the defuzzified output of the IT2FLS is

$$y(x) = \frac{y_l + y_r}{2} \quad (11)$$

However, special attention should be paid to the calculations of the end points of $Y_c(x)$, y_l and y_r. From (10), we see that y_l and y_r can be expressed separately as follows

$$y_l = \frac{\sum_{i=1}^M \underline{y}^i f_l^i}{\sum_{i=1}^M f_l^i} = \sum_{i=1}^M \underline{y}^i p_l^i \quad (12)$$

$$y_r = \frac{\sum_{i=1}^M \overline{y}^i f_r^i}{\sum_{i=1}^M f_r^i} = \sum_{i=1}^M \overline{y}^i p_r^i \quad (13)$$

where $f_l^i = \underline{f}^i$ or $\overline{f}^i$ contributing to y_l, $f_r^i = \underline{f}^i$ or $\overline{f}^i$ contributing to y_r, $p_l^i = f_l^i / \sum_{i=1}^M f_l^i$ and $p_r^i = f_r^i / \sum_{j=1}^M f_r^i$. Hence it is needed to determine which of $\left\{ \underline{f}^i \right\}_{i=1}^M$ and $\left\{ \overline{f}^i \right\}_{i=1}^M$ contribute to y_l and which of $\left\{ \underline{f}^i \right\}_{i=1}^M$ and $\left\{ \overline{f}^i \right\}_{i=1}^M$ contribute to y_r. This is can be done by a procedure developed in [2], Table 1 shows this procedure that is to calculate the right end point y_r.

TABLE I
PROCEDURE OF CALCULATING y_r [2]

Assuming $\left\{\overline{y}^i\right\}_{i=1}^M$ are arranged in ascending order: $\overline{y}^1 \leq \cdots \leq \overline{y}^M$.

Step 1. Compute y_r in (13) by initially setting $f_r^i = \left(\overline{f}^i + \underline{f}^i\right)/2$ for $i = 1, \cdots, M$, where $\overline{f}^i$ and $\underline{f}^i$ are pre-computed in (7) and (8) separately. Let $y_r' \overset{\Delta}{=} y_r$.

Step 2. Find K $(1 \leq K \leq M-1)$ such that $y_r^K \leq y_r' \leq y_r^{K+1}$.

Step 3. Compute y_r in (13) with $f_r^i = \underline{f}^i$ for $i \leq K$ and $f_r^i = \overline{f}^i$ for $i > K$. Let $y_r'' \overset{\Delta}{=} y_r$.

Step 4. If $y_r'' \neq y_r'$, then go to Step 5. If $y_r'' = y_r'$, then stop, and set $y_r \overset{\Delta}{=} y_r''$.

Step 5. Set y_r' equal to y_r'', and go to step 2.

Then given the data set $\left\{x^{(i)}, y^{(i)}\right\}_{i=1}^S$ for designing an IT2FLS, we now wish to train an IT2FLS such that the following error is minimised:

$$e = \frac{1}{2}\sum_{t=1}^S \left(y(x^{(t)}) - y^{(t)}\right)^2 \qquad (14)$$

In this paper, the membership functions of the type-2 fuzzy sets $\tilde{F}_i^l$ and $\tilde{G}^l$ used are the Gaussian primary functions with uncertain means, i.e.,

$$\mu_{\tilde{F}_i^l}(x_i) = \exp\left(-\frac{1}{2}\left(\frac{x_i - m_i^l}{\sigma_i^l}\right)^2\right) \quad m_i^l \in \left[m_{i1}^l, m_{i2}^l\right] \qquad (15)$$

and

$$\mu_{\tilde{G}^l}(y) = \exp\left(-\frac{1}{2}\left(\frac{y - m^l}{\sigma^l}\right)^2\right) \quad m^l \in \left[m_1^l, m_2^l\right] \qquad (16)$$

Then the back-propagation method can be used to tune the antecedent and consequent parameters in (15) and (16) so as to mimimise the mean-square error (14), details of how this training method works can be found in [2].

III. New Rule Ranking Indices for Rule Reduction

A. R-values of fuzzy rules considering rule base structure

Liang and Mendel applied the SVD-QR with column pivoting algorithm to IT2FLS for generating a compact type-2 rule base by reducing the redundant rules [18]. But it is required to estimate an efficient rank in the SVD-QR with column pivoting algorithm. In order to avoid the estimation of rank, in the following we propose to apply pivoted QR decomposition method to the construction of parsimonious IT2FLS. The idea behind this method is to assign a rule significance index to each fuzzy rule, then rank and select the influential fuzzy rules in terms of this index. Hence the

rule order is important, but in designing T2FLS, the rule orders have been changed for calculating y_l and y_r.

First for y_r, let the original rule order be $I = [1, 2, \cdots, M]^T$. After re-ordering $\left\{\overline{y}^i\right\}_{i=1}^M$ in ascending order, the rule order is $I' = QI$, where Q is a permutation matrix, then re-number the order-changed rules I' as $1, 2, \cdots, M$, which is used in the procedure of calculating y_r in Table I.

It is noted that the number K determined in Table 1 is very important. For $i \leq K$, $f_r^i = \underline{f}^i$, and for $i > K$, $f_r^i = \overline{f}^i$. Thus,

$$y_r = \frac{\sum_{i=1}^K \overline{y}^i \underline{f}^i + \sum_{i=K+1}^M \overline{y}^i \overline{f}^i}{\sum_{i=1}^K \underline{f}^i + \sum_{i=K+1}^M \overline{f}^i} \qquad (17)$$

Then, for $i \leq K$,

$$p_r^i = \begin{cases} \underline{f}^i \Big/ \left(\sum_{i=1}^K \underline{f}^i + \sum_{i=K+1}^M \overline{f}^i\right) & i \leq K \\ \overline{f}^i \Big/ \left(\sum_{i=1}^K \underline{f}^i + \sum_{i=K+1}^M \overline{f}^i\right) & i > K \end{cases} \qquad (18)$$

So a firing strength vector given an input x is obtained by restoring the original rule order,

$$p(x) = Q^{-1}\left[p_r^1, \cdots, p_r^K, p_r^{K+1}, \cdots, p_r^M\right]^T \qquad (19)$$

Then the S training samples $\left\{x^{(i)}, y^{(i)}\right\}_{i=1}^S$ lead to S firing strength vectors composing a firing strength matrix P_r,

$$P_r = [p_1, \cdots, p_s]^T \qquad (20)$$

Finally, the QR with column pivoting algorithm addressed in Table II is applied to P_r, in which each rule is assigned a R-value as its significance index value. Because the R-values, i.e., the absolute values of the diagonal elements of decomposition matrix R tend to approach to the singular values, so they can be used as rule ranking index in designing a T2FLS with compact rule base. In term of the R-values, assume $\hat{s}_1$ most influential fuzzy rules be selected for calculating y_r, let T_1 denote the selected rule set, $1 \leq \hat{s}_1 \leq M$.

Similar procedure is applied to the corresponding firing strength matrix P_l for y_l leading to $\hat{s}_2$ rules T_2, $1 \leq \hat{s}_2 \leq M$.

Then, the finally reduced-rule IT2FLS combines the rules in T_1 and T_2 as $T_1 \cup T_2$, which is the union of the rule sets T_1 and T_2.

B. c-values of fuzzy rules considering effects of rule consequents

Both the SVD-QR with column pivoting method and pivoted QR method only take into account the rule base structure focusing on the rule antecedent parts when applied to rule reduction of IT2FLS. Another way of ranking type-2 fuzzy rules is based on the effects of rule consequents $\tilde{G}^i$.

As a matter of fact, it can be seen from the procedure of designing IT2FLS described in the above section that for each type-2 fuzzy rule, its left and right end points of the centroid of consequent set $\tilde{G}^i$, $\underline{y}^i$ and $\overline{y}^i$, separately determine the depths of the effects of the rule consequent

Step 1. Calculate the QR decomposition of P and get the permutation matrix Π via $G\Pi = QR$, where Q is an unitary matrix, R is an upper triangular matrix. The absolute values of the diagonal elements of R, denoted as $|R_{ii}|$, decrease as i increases and are named as R-values.

Step 2. Rank fuzzy rules in terms of the R-values and the permutation matrix Π. Each column of Π has one element taking value 1 and all the other elements taking value 0. Each column of Π corresponds to a fuzzy rule. The numbering of the jth most important rule in the original rule base is the same as the numbering of the row where the "1" element of the jth column is located. For example, if the "1" of the 1st column is in the 4th row, then the 4th rule is the most important one and its importance is measured as $|R_{11}|$. The rule corresponding to the first column is the most important, and in descending order the rule corresponding to the last column is the least important.

---

on the output end points y_l and y_r. Hence $\underline{y}^i$ and $\overline{y}^i$ are very useful indices for measuring the output contributions of the type-2 fuzzy rules. These $\underline{y}^i$ and $\overline{y}^i$, the left and right end points of the centroid of consequent, are called c-values of type-2 fuzzy rules in this paper.

For calculating y_r, the c-values $\overline{y}^i$ are used as rule ranking index to select the most influential fuzzy rules. Assume $\hat{s}_1$ most influential fuzzy rules be selected for y_r, let T_1 denote the selected rule set, $1 \leq \hat{s}_1 \leq M$.

For calculating y_l, the c-values $\underline{y}^i$ are used as rule ranking index to select the most influential fuzzy rules. Assume $\hat{s}_2$ most influential fuzzy rules be selected for y_l, and let T_2 denote the selected rule set, $1 \leq \hat{s}_2 \leq M$.

Then, the final reduced-rule IT2FLS combines the rules in T_1 and T_2 as $T_1 \cup T_2$, which is the union of the rule sets T_1 and T_2.

IV. EXPERIMENTAL RESULTS

In this section, we evaluate the proposed rule reduction methods for constructing parsimonious IT2FLSs to recover an original signal from data highly contaminated by noise. In the experiments, the noisy signal is generated by

$$v(t) = \tilde{v}(t) + \tilde{\theta}(t) \tag{21}$$

$$\tilde{v}(t) = \sin\left(40/(x(t) + 0.03)\right) + x(t-1)/10 \tag{22}$$

where $\tilde{v}$ is the original signal, $\tilde{\theta}$ is an interference signal generated from another Gaussian noise source $\tilde{n}$ with zero mean and standard deviation one via a certain unknown nonlinear process:

$$\tilde{\theta}(t) = 4\sin\left(\tilde{n}(t)\right)\tilde{n}(t-1)/\left(1 + \tilde{n}(t-1)^2\right) \tag{23}$$

The measured signal v is the sum of the original information signal $\tilde{v}$ and the interference $\tilde{\theta}$, however, we do not know the interference signal $\tilde{\theta}$. The only signals available to us are the noise $\tilde{n}$ and the measured signal v. Our task is to recover the original information signal $\tilde{v}$ from the measured signal v. T2FLS is suitable for signal processing [8] due to its strong capability of characterising higher uncertainty exhibiting within noisy data. In the following, an initial IT2FLS model with two inputs $x(t), x(t-1)$ and one output $y(t)$ is constructed in which the antecedent and consequent parameters in (15) and (16) are optimised by back-propagation algorithm [2]. In order to train the interval type-2 fuzzy model to grasp the nonlinearity and higher uncertainty of the system, the data generation process (21), (22) and (23) run 10 times, in each run 100 samples $\left\{x^{(i)}, y_k^{(i)}\right\}_{i=1}^{S}$ (S=100, $k = 1, \cdots, 10$) are generated with $x^{(i)} \in [2, 5]$ and $y_k^{(i)}$ obtained by (21). Then the data set $\left\{x^{(i)}, \min_k y_k^{(i)}\right\}_{i=1}^{100}$ are used to generate the antecedent and consequent means $\{m_{i1}\}$ and $\{m_1\}$ in (15) and (16) by the fuzzy c-means (FCM) unsupervised clustering algorithm [21], whist the data set $\left\{x^{(i)}, \max_k y_k^{(i)}\right\}_{i=1}^{100}$ are used to generate the means $\{m_{i2}\}$ and $\{m_2\}$. The width parameters in (15) and (16) are determined using the nearest neighbor heuristic suggested by Moody and Daken [22] based on the corresponding data sets. Hence all the initial antecedent and consequent parameters are determined from given date sets, rather than manually set up.

Then four initial type-2 fuzzy sets are generated for each input variable, which leads to 16 rules in the initial interval type-2 fuzzy model. After training process, this interval type-2 fuzzy model shows its ability of recovering the original signal well with root-mean-square error (RMSE) 0.27395 as shown in Figure 2.
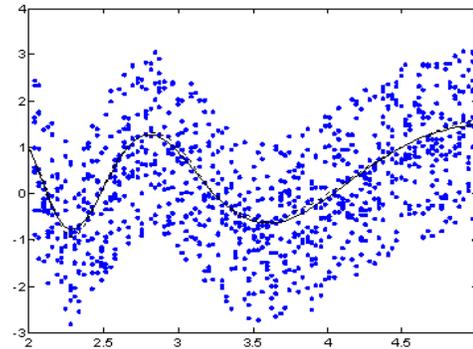
Fig. 2. Signal recovering by IT2FLS model: solid line (SL) represents the original signal $\tilde{v}$ and dotted line (DL) represents the recovered signal.

Now we apply the proposed rule reduction methods to the trained interval type-2 fuzzy model. First, the QR with column pivoting algorithm is applied to the firing strength matrices P_r and P_l, in which the R-values of fuzzy rules are generated for selecting influential rules in calculating

y_r and y_l separately. Figure 3 and Figure 4 depict the corresponding R-values and singular values of fuzzy rules based on the firing strength matrices P_r and P_l individually, which indicates that the R-values track the singular values well, so the R-values of P_r and P_l can be used to rank the fuzzy rules. These R-values in the original rule order are illustrated in Figure 5 and Figure 6, then the rule ranking results in terms of R-values of fuzzy rules are obtained as shown in Table III.
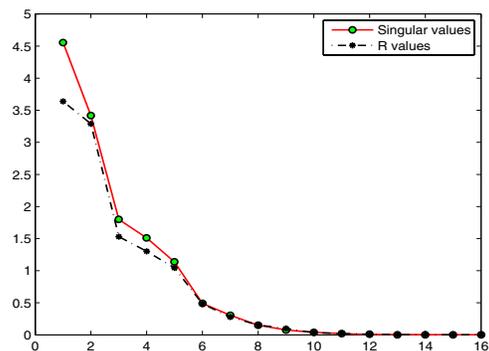
Fig. 3. R-values and singular values of firing strength matrix P_r
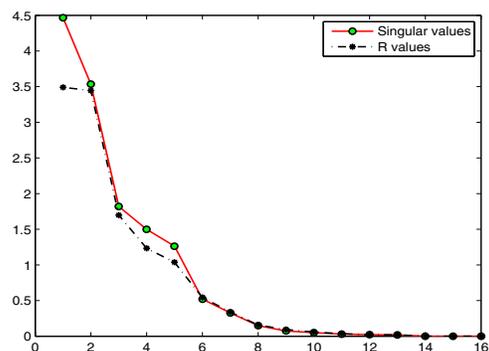
Fig. 4. R-values and singular values of firing strength matrix P_l

TABLE III
RULE RANKING RESULTS IN TERMS OF R-VALUES

Firing strength matrix	Rule ranking order
P_r	16 11 15 1 7 10 6 12 5 9 2 14 3 13 8 4
P_l	11 16 15 1 6 10 2 12 7 5 3 14 9 8 13 4

By applying the proposed procedure as addressed in subsection III-A for selecting significant rules in terms of the R-values, the rule selection results are delineated in Table IV. The RMSE threshold is set to be 0.3, then a parsimonious interval type-2 fuzzy model can be constructed by 12 fuzzy rules identified in terms of the R-values of fuzzy rules.
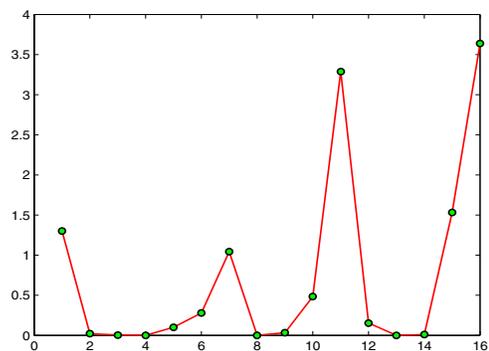
Fig. 5. R-values of fuzzy rules based on firing strength matrix P_r
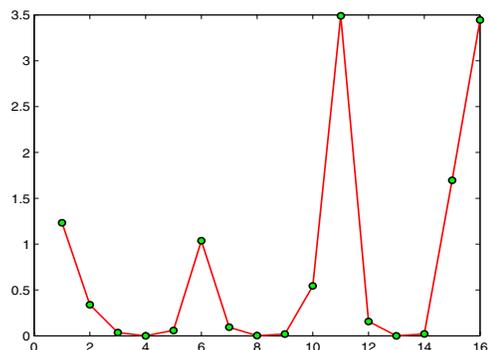
Fig. 6. R-values of fuzzy rules based on firing strength matrix P_l

With the consideration of the consequent effects of trained fuzzy rules on overall system output, the c-values $\overline{y}^i$ and $\underline{y}^i$ of fuzzy rules depicted in Figure 7 and Figure 8 separately are used to identify the important fuzzy rules, which leads to a parsimonious interval fuzzy model with 11 rules as delineated in Table V given the model RMSE threshold 0.3.

V. CONCLUSIONS

In this paper, the named R-values and c-values of type-2 fuzzy rules are suggested to identify the most influential fuzzy rules and reduce the redundant ones. The R-values of type-2 fuzzy rules obtained by QR decomposition with column pivoting algorithm pay attention to the rule base structure, the definition of the c-values of fuzzy rules focus on contributions of rule consequents. The experimental results have shown that parsimonious type-2 fuzzy system models can be effectively constructed by the fuzzy rules selected in terms of the proposed indices.

TABLE IV

RULE REDUCTION RESULTS BY QR WITH COLUMN PIVOTING METHOD

Rule No.	Rules Selected	Model RMSE
8	11 16 15 1 7 10 6 2	1.50692
9	11 16 15 1 7 10 6 2 12	1.50628
10	11 16 15 1 7 10 6 2 12 5	1. 20580
11	11 16 15 1 7 10 6 2 12 5 9	0.53817
12	**11 16 15 1 7 10 6 2 12 5 9 3**	**0.27472**
13	11 16 15 1 7 10 6 2 12 5 9 3 14	0.27398
14	11 16 15 1 7 10 6 2 12 5 9 3 14 13	0. 27398
15	11 16 15 1 7 10 6 2 12 5 9 3 14 13 8	0. 27394
16	11 16 15 1 7 10 6 2 12 5 9 3 14 13 8 4	0.27395
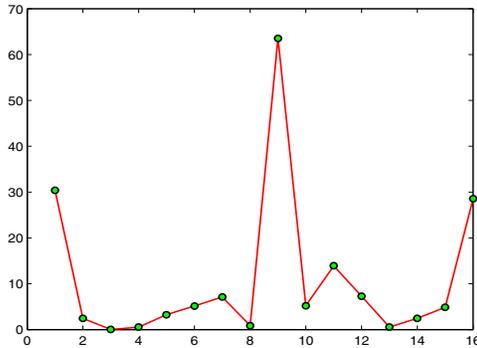
Fig. 7. c-values of fuzzy rules: $\overline{y}^i$
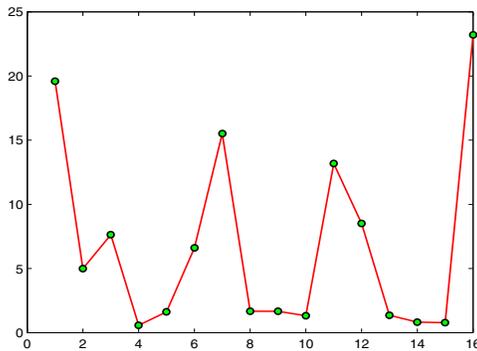
Fig. 8. c-values of fuzzy rules: $\underline{y}^i$

TABLE V

RULE REDUCTION RESULTS BY c-VALUES

Rule No.	Rules Selected	Model RMSE
9	9 16 1 7 11 12 3 10 6	1. 14829
10	9 16 1 7 11 12 3 10 6 2	1. 14404
11	**9 16 1 7 11 12 3 10 6 2 15**	**0.28751**
12	9 16 1 7 11 12 3 10 6 2 15 8	0. 28699
13	9 16 1 7 11 12 3 10 6 2 15 8 5	0. 27438
14	9 16 1 7 11 12 3 10 6 2 15 8 5 14	0. 27394
15	9 16 1 7 11 12 3 10 6 2 15 8 5 14 13	0. 27394
16	9 16 1 7 11 12 3 10 6 2 15 8 5 14 13 4	0.27395

REFERENCES

[1] L. A. Zadeh, "The concept of a linguistic variable and its application to approximate reasoning-1," Information Science, vol. 8, pp. 199-249, 1975.

[2] J. M. Mendel, *Uncertainty Rule-based Fuzzy Logic Systems: Introduction and New Directions*, Prentice Hall, Inc., 2001.

[3] J. M. Mendel, and R. I. John, "Type-2 fuzzy sets made simple," *IEEE Trans. on Fuzzy Systems*, vol.10, no.2, pp.117-127, 2002.

[4] J. L. Chaneau, M. Gunaratne, and A. G. Altschaeffl, "An application of type-2 sets to decision making in engineering," in J. C. Bezdek, Ed., *Analysis of Fuzzy Information-Vol. II: Artificial Intelligence and Decision Systems*, Boca Raton, FL: CRC, 1987.

[5] R. R. Yager, "Fuzzy subsets of type II in decisions," *Journal of Cybernetics*, vol.10, pp. 137-159, 1980.

[6] R. I. John, and P. R. Innocent, "Modeling uncertainty in clinical diagnosis using fuzzy logic," *IEEE Trans. on Systems, Man and Cybernetics-part B*, vol.35, no.6, pp.1340-1350, 2005.

[7] N. N. Karnik, J. M. Mendel, and Q. Liang, "Type-2 fuzzy logic systems," *IEEE Trans. on Fuzzy Systems*, vol.7, no.6, pp.643-658, 1999.

[8] J. M. Mendel, "Uncertainty, fuzzy logic, and signal processing," Signal Processing, vol.80, no.6, pp. 913-933, 2000.

[9] L. Li, W.-H. Lin and H. Liu, "Type-2 fuzzy logic approach for short-term traffic forecasting," *IEE Proc. Intelligent Transportation System*, vol.153, no.1, pp.33-40, 2006.

[10] H. A. Hagras, "A hierarchical type-2 fuzzy logic control architecture for autonomous mobile robots," *IEEE Trans. on Fuzzy Systems*, vol.12, no.4, pp.524-539, 2004.

[11] H. B. Mitchell, "Pattern recognition using type-II fuzzy sets," *Information Sciences*, vol.170, pp.409-418, 2005.

[12] J. Zeng and Z. Q. Liu, "Type-2 fuzzy sets for handling uncertainty in pattern recognition," *Proc. FUZZ-IEEE*, July, Vancouver, Canada, pp.6597-6602, 2006.

[13] R. Sepulveda, O. Castillo, P. Melin, A. Rodriguez-Diaz and O. Montiel, "Handling uncertainty in controllers using type-2 fuzzy logic," *Proc. FUZZ-IEEE*, May, Reno, NV, pp.248-253, 2005.

[14] R. Sepulveda, O. Castillo, P. Melin, A. Rodriguez-Diaz and O. Montiel, "Integrated development platform for intelligent control based on type-2 fuzzy logic," *Proc. North American Fuzzy Info. Processing Society (NAFIPS)*, pp.607-610, 2005.

[15] K. M. Bossley, D. J. Mills, M. Brown, and C. J. Harris, "Construction and design of parsimonious neurofuzzy systems," in K. J. Hunt, G. R. Irwin, and K. Warwick (eds), Proceedings of Neural Network Engineering in Control Systems, pp.153-177, Springer-Verlag, 1995.

[16] M. J. Er, and S. Wu, "A fast learning algorithm for parsimonious fuzzy neural systems," *Fuzzy Sets and Systems*, vol.126, no.3, pp.337-351, 2002.

[17] H. Roubos nd M. Setnes, "Compact fuzzy models through complexity reduction and evolutionary optimization," *Proc. of FUZZ-IEEE*, San Antonio, USA, May, pp.762-767, 2000.

[18] Q. Liang, and J. M. Mendel, "Designing interval type-2 fuzzy logic systems using an SVD-QR method: rule reduction," *International Journal of Intelligent Systems*, vol.15, pp.939-957, 2000.

[19] L. X. Wang and J. M. Mendel, "Fuzzy basis functions, universal approximation, and orthogonal least squares learning," *IEEE Trans. on Neural Networks,* vol.3, no.5, pp.807–814, 1992.

[20] S.-M. Zhou, and J. Q. Gan, "Constructing parsimonious fuzzy lassifiers based on L2-SVM in high-dimensional space with automatic mdel selection and fuzzy rule ranking," *IEEE Trans. on Fuzzy Systems* (in press), 2007.

[21] J. C. Bezdek, *Pattern Recognition with Fuzzy Objective Function Algorithms*, Plenum Press, New York, 1981.

[22] M. J. Moody and C. J. Darken, "Fast learning in networks of locally-tuned processing units," *Neural Computation*, vol.1, no.2, pp.281-294, 1989.