H. Ishibuchi and T. Nakashima, "Effect of rule weights in fuzzy rule-based classification systems," IEEE Trans. on Fuzzy Systems, Vol. 9, No. 4, pp. 506-515, August 2001.

Effect of Rule Weights in Fuzzy Rule-Based Classification Systems

Hisao Ishibuchi, *Member, IEEE*, and Tomoharu Nakashima, *Member, IEEE*

Department of Industrial Engineering, Osaka Prefecture University

Gakuen-cho 1-1, Sakai, Osaka 599-8531, Japan

Abstract - This paper examines the effect of rule weights in fuzzy rule-based classification systems. Each fuzzy if-then rule in our classification system has antecedent linguistic values and a single consequent class. We use a fuzzy reasoning method based on a single winner rule in the classification phase. The winner rule for a new pattern is the fuzzy if-then rule that has the maximum compatibility grade with the new pattern. When we use fuzzy if-then rules with certainty grades (i.e., rule weights), the winner is determined as the rule with the maximum product of the compatibility grade and the certainty grade. In this paper, the effect of rule weights is illustrated by drawing classification boundaries using fuzzy if-then rules with/without certainty grades. It is also shown that certainty grades play an important role when a fuzzy rule-based classification system is a mixture of general rules and specific rules. Through computer simulations, we show that comprehensible fuzzy rule-based systems with high classification performance can be designed without modifying the membership functions of antecedent linguistic values when we use fuzzy if-then rules with certainty grades.

Index Terms - Pattern classification, fuzzy rule-based systems, fuzzy reasoning, rule extraction.

I. INTRODUCTION

The main application area of fuzzy rule-based systems has been control problems [1]-[4]. Fuzzy rule-based systems for control problems can be viewed as approximators of nonlinear mappings from non-fuzzy input vectors to non-fuzzy output values. Recently fuzzy rule-based systems have often been applied to classification problems where non-fuzzy input vectors are to be assigned to one of a given set of classes. Many approaches have been proposed for generating and learning fuzzy if-then rules from numerical data for classification problems. For example, fuzzy rule-based classification systems are created by simple heuristic procedures [5],[6], neuro-fuzzy techniques [7]-[9], clustering methods [10], fuzzy nearest neighbor methods [11], and genetic algorithms [12]-[15].

Fuzzy if-then rules for a c-class pattern classification problem with n attributes can be written as

$$\text{Rule } R_j : \text{If } x_1 \text{ is } A_{j1} \text{ and ... and } x_n \text{ is } A_{jn} \text{ then Class } C_j,\ j = 1,2,...,N, \tag{1}$$

where $\mathbf{x} = (x_1, ..., x_n)$ is an n-dimensional pattern vector, A_{ji} is an antecedent linguistic value such as *small* and *large* $(i = 1,2,...,n)$, C_j is a consequent class (i.e., one of the given c classes), and N is the number of fuzzy if-then rules. When we use a grid-type fuzzy partition (e.g., Ishibuchi et al.[12]), the antecedent part of each fuzzy if-then rule is specified by a combination of linguistic values. The total number of possible combinations is K^n when each attribute x_i has K linguistic values $(i = 1,2,...,n)$. Thus each linguistic value is shared by a number of fuzzy if-then rules. On the other hand, each fuzzy if-then rule may have its own n antecedent fuzzy sets (or a single n-dimensional antecedent fuzzy set [9]). In this paper, we use a grid-type fuzzy partition for generating fuzzy if-then rules.

The following fuzzy if-then rules with certainty grades are also used for our classification problem:

$$\text{Rule } R_j : \text{If } x_1 \text{ is } A_{j1} \text{ and ... and } x_n \text{ is } A_{jn} \text{ then Class } C_j \text{ with } CF_j,\ j = 1, 2, ..., N, \qquad (2)$$

where CF_j is the certainty grade of the fuzzy if-then rule R_j. Usually CF_j is a real number in the unit interval [0, 1] (i.e., $0 \le CF_j \le 1$).

The aim of this paper is to examine the effect of certainty grades on the performance of fuzzy rule-based classification systems. Nauck & Kruse [16] discussed the effect of rule weights in fuzzy rule-based systems for function approximation problems. They showed how the learning of rule weights can be equivalently replaced by the modification of the membership functions of antecedent or consequent fuzzy sets. Based on this observation, they also showed that it is not necessary to use rule weights for the learning in fuzzy rule-based systems. In this paper, we show a similar relation between rule weights and membership functions in a totally different viewpoint from [16]. We show that compact fuzzy rule-based classification systems can be designed without adjusting membership functions when we use fuzzy if-then rules with certainty grades. That is, the learning of membership functions can be partially replaced by the adjustment of certainty grades.

In this paper, we assume that a set of antecedent linguistic values is given by domain experts for each attribute of our *n*-dimensional classification problem. This means that a fuzzy partition of the *n*-dimensional pattern space is given. We also assume that *m* training (i.e., labeled) patterns $\mathbf{x}_p = (x_{p1}, ..., x_{pn}),\ p = 1, 2, ..., m$ are given from *c* classes. Our task is to design a comprehensible fuzzy rule-based classification system from the given numerical data using the given linguistic values. We implicitly assume that modifying the membership functions of the given linguistic values deteriorates the comprehensibility of fuzzy if-then rules. This is because the modification is likely to cause a gap between modified membership functions and an experts' understanding of linguistic values. Based on this implicit assumption, we try to design a fuzzy rule-based classification system without modifying the given membership functions. Of course, the learning of membership functions may be necessary when classification performance is our main criterion. Learning is also necessary when we have to construct a membership function of each antecedent fuzzy set from numerical data (i.e., when linguistic values are not given by domain experts).

As shown in (2), each rule has its own rule weight (i.e., certainty grade). Since the rule weight is a single real number, its adjustment is much easier than the learning of antecedent fuzzy sets (i.e., the learning of a number of parameter values of each membership function). The simplicity of adjustment is one advantage of the use of rule weights. Another advantage is that the classification performance can be improved without modifying the membership function of each linguistic value. This means that the comprehensibility of fuzzy rule-based systems is not deteriorated. The rule weight can be interpreted as the strength of each rule. As shown in this paper, the larger the rule weight is, the large the decision area of each rule is. As pointed out by Nauck & Kruse [16], the use of the rule weight for each fuzzy if-then rule in (2) has the same effect on fuzzy reasoning as the modification of its antecedent fuzzy sets. Even when we interpret the use of rule weights as the modification of antecedent fuzzy sets, the position of each fuzzy set is not changed as shown in this paper (also see [16]).

This paper is organized as follows. Before discussing fuzzy if-then rules in (1) and (2) for classification problems, we discuss the effect of rule weights on a simplified fuzzy reasoning method for function approximation problems in Section II. It is shown that the effect of rule weights is replaced by the learning of consequent real numbers. Discussions in Section II are based on Nauck & Kruse [16]. Then we describe fuzzy rule-based classification systems with no certainty grades in Section III. In Section IV, we discuss the effect of certainty grades on classification results from fuzzy if-then rules. The effect is clearly illustrated by drawing classification boundaries obtained by fuzzy if-then rules with/without certainty grades. In Section V, we show that certainty grades are necessary for simultaneously handling fuzzy if-then rules with different specificity levels (i.e., general rules and specific rules). While general rules have only a few antecedent conditions, specific rules have many antecedent conditions. Specific rules are used for describing complicated classification boundaries. Sometimes they work as exceptions to general rules. In Section VI, the effect of certainty grades is examined through computer simulations on commonly-used data sets in the literature. Simulation results show that compact fuzzy rule-based systems with high classification performance can be designed without adjusting membership functions when we use certainty grades. Section VII concludes this paper.

II. FUZZY RULES FOR FUNCTION APPROXIMATION

For function approximation problems, the following fuzzy if-then rules with consequent real numbers have often been used:

$$\text{Rule } R_j : \text{If } x_1 \text{ is } A_{j1} \text{ and ... and } x_n \text{ is } A_{jn} \text{ then } y \text{ is } r_j ,\ j = 1, 2, ..., N , \tag{3}$$

where y is an output variable, and r_j is a consequent real number. These fuzzy if-then rules can be viewed as a simplified version of the well-known Takagi-Sugeno fuzzy rules with linear functions in the consequent part [17]. The estimated output value $\hat{y}(\mathbf{x})$ for an input vector $\mathbf{x} = (x_1, ..., x_n)$ is calculated as the weighted average of consequent real numbers:

$$\hat{y}(\mathbf{x}) = \frac{\sum_{j=1}^{N} \mu_j(\mathbf{x}) \cdot r_j}{\sum_{j=1}^{N} \mu_j(\mathbf{x})} , \tag{4}$$

where $\mu_j(\mathbf{x})$ is the compatibility grade of the fuzzy if-then rule R_j with the input vector $\mathbf{x}$. The compatibility grade $\mu_j(\mathbf{x})$ is usually calculated by the product operator as:

$$\mu_j(\mathbf{x}) = \mu_{j1}(x_1) \times \cdots \times \mu_{jn}(x_n) , \tag{5}$$

where $\mu_{ji}(\cdot)$ is the membership function of the antecedent linguistic value A_{ji} ($i = 1, 2, ..., n$).

We have the following fuzzy if-then rules by attaching a rule weight w_j to each rule in (3):

$$\text{Rule } R_j : \text{If } x_1 \text{ is } A_{j1} \text{ and ... and } x_n \text{ is } A_{jn} \text{ then } y \text{ is } r_j \text{ with } w_j ,\ j = 1, 2, ..., N . \tag{6}$$

For handling these fuzzy if-then rules, the simplified fuzzy reasoning method in (4) is modified as:

$$\hat{y}(\mathbf{x}) = \frac{\sum_{j=1}^{N} \mu_j(\mathbf{x}) \cdot r_j \cdot w_j}{\sum_{j=1}^{N} \mu_j(\mathbf{x})} , \tag{7}$$

where we assume that the weight w_j affects only the consequent part (i.e., the consequent real number r_j). If we handle the product $r_j \cdot w_j$ as a new real number (say $s_j := r_j \cdot w_j$), the formulation in (7) is reduced to (4) with the consequent real number s_j. This means that the learning of w_j in (7) is totally replaced by the learning of r_j. As a result, we can conclude that the rule weight w_j in (6) is not necessary for function approximation problems. This discussion is based on Nauck & Kruse [16] where they discussed various ways for handling rule weights. For example, they discussed the handling of rule weights in Mamdani-type fuzzy systems as well as Sugeno-type fuzzy systems. They discussed the case where rule weights were applied to the antecedent part of fuzzy if-then rules as well as the case where rule weights were applied to the consequent part. They concluded that the learning of rule weights can be equivalently replaced by the modification of the membership functions of antecedent or consequent fuzzy sets.

III. CLASSIFICATION WITHOUT CERTAINTY GRADES

In our fuzzy rule-based classification system, we use a fuzzy reasoning method based on a single winner rule in the classification phase. Other fuzzy reasoning methods for classification problems were studied in [18]-[20]. We use the single winner method because its classification mechanism is simple and intuitive for human users.

When fuzzy if-then rules have no certainty grades as in (1), a new pattern $\mathbf{x}_p = (x_{p1}, ..., x_{pn})$ is classified by the single winner rule R_{j*} defined by

$$\mu_{j*}(\mathbf{x}_p) = \max\{\mu_j(\mathbf{x}_p) : j = 1, 2, ..., N\}, \tag{8}$$

where $\mu_j(\mathbf{x}_p)$ is the compatibility grade of the fuzzy if-then rule R_j with the new pattern $\mathbf{x}_p$, which is usually defined by the product operator as in (5).

From (8), we can see that each fuzzy if-then rule has its own decision area in which new patterns are classified by that rule. The decision area of each rule is illustrated in Fig. 1 where we have nine fuzzy if-then rules generated by three antecedent linguistic values (i.e., *S*: *small*, *M*: *medium*, and *L*: *large*) on each axis of the two-dimensional pattern space $[0,1] \times [0,1]$. Each of the nine cells (or patches) in Fig. 1 corresponds to the decision area of each fuzzy if-then rule. Kuncheva [21], [22] proved that fuzzy if-then rules with no certainty grades have rectangular or hyper-rectangular decision areas when no fuzzy if-then rules are missing in fuzzy rule tables.

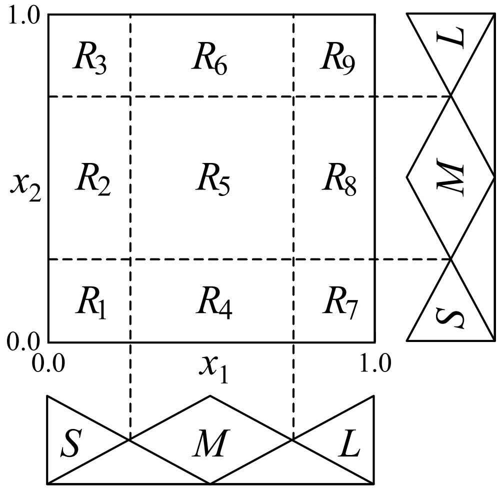


Fig. 1. Decision area of each fuzzy if-then rule.

Examples of classification boundaries by the nine fuzzy if-then rules in Fig. 1 are shown in Fig. 2. As shown in Fig. 2, if no rules are missing in fuzzy rule tables, classification boundaries are always parallel to the axes of the pattern space in the case of fuzzy if-then rules without certainty grades. This is because the decision area of each fuzzy if-then rule is a rectangular or hyper-rectangular cell as shown in Fig. 1. Classification boundaries consist of the borders between the decision areas of fuzzy if-then rules with different consequent classes. In Fig. 2, we specified the consequent class C_j of each fuzzy if-then rule R_j as follows:

$$\sum_{p \in \text{Class } C_j} \mu_j(\mathbf{x}_p) = \max\{ \sum_{p \in \text{Class } k} \mu_j(\mathbf{x}_p) : k = 1, 2, ..., c \}\,, \tag{9}$$

where c is the number of given classes. In (9), the consequent class C_j is specified as the dominant class in the fuzzy subspace corresponding to the antecedent part of each fuzzy if-then rule.

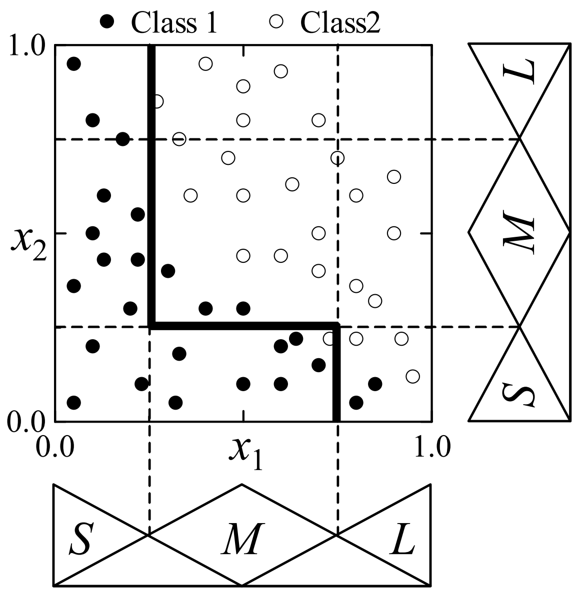


Fig. 2. Classification boundary by the nine fuzzy if-then rules.

In Fig. 2, six patterns are misclassified. Classification boundaries can be adjusted by modifying the membership functions of the linguistic values. Fig. 3 is an example of an adjusted classification boundary where almost all patterns are correctly classified. From Fig. 3, we can see that the classification boundary after the modification of the membership functions is still parallel to the axes of the pattern space because the decision area of each fuzzy if-then rule is a rectangular cell. As shown in Fig. 3, the decision area can be adjusted by modifying the membership functions.

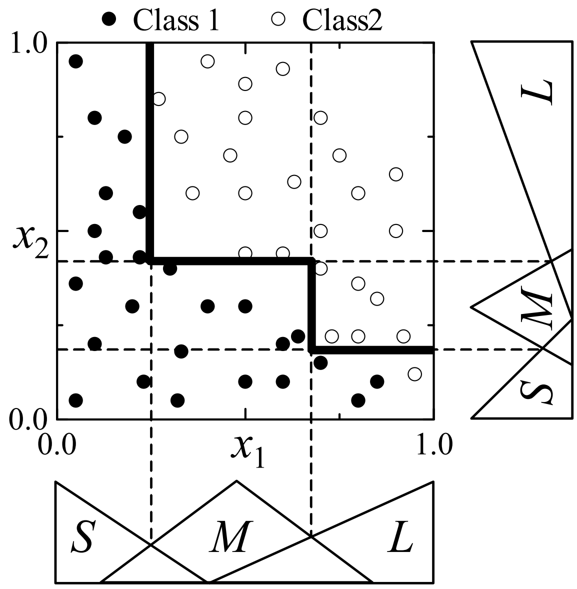


Fig. 3. Classification boundary after the modification of the membership functions.

When fuzzy rule tables are incomplete (i.e., some fuzzy if-then rules are missing), the decision area of each fuzzy if-then rule is not always rectangular. In this case, classification boundaries are not always parallel to the axes of the pattern space. This is illustrated in Fig. 4 where the decision area of each fuzzy if-then rule is drawn using incomplete 3×3 fuzzy rule tables. Fig. 4 corresponds to the situations where R_5 is missing in Fig. 1 and Fig. 3.

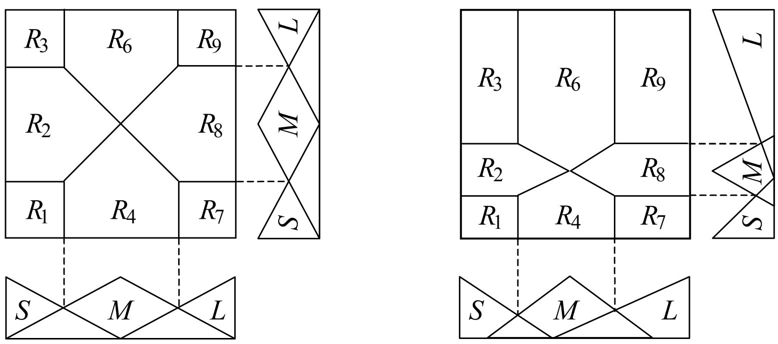


Fig. 4. Decision area of each fuzzy if-then rule in the case of incomplete fuzzy rule tables.

Fig. 5 shows the difference between non-fuzzy if-then rules and fuzzy if-then rules with no certainty grades. When rule tables are complete as in Fig. 1 ~ Fig. 3, these two kinds of if-then rules have the same rectangular decision areas. For example, each non-fuzzy if-then rule in Fig. 5 has the same decision area as the corresponding fuzzy if-then rule in Fig. 2. Even in this case, the classification boundary in Fig. 5 is not exactly the same as that in Fig. 2. In Fig. 5, the upper-right rule (i.e., R_9 in Fig. 1) can not be generated because there is no training pattern compatible with its antecedent part. Thus the classification of any pattern in its decision area (i.e., shaded rectangular cell in Fig.5) is rejected.

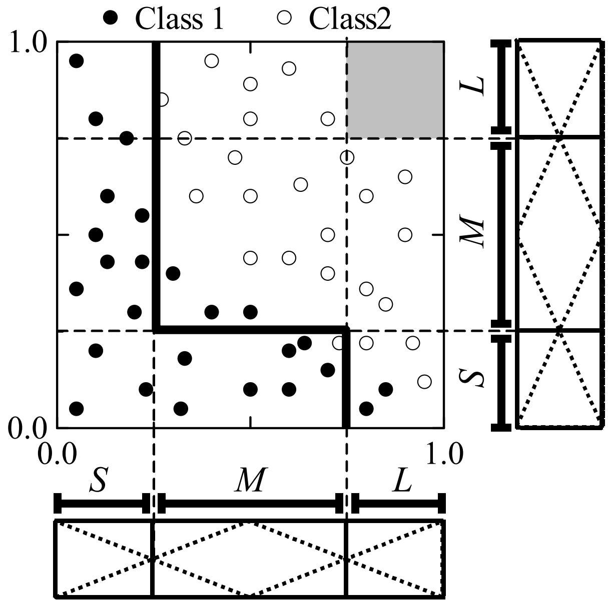
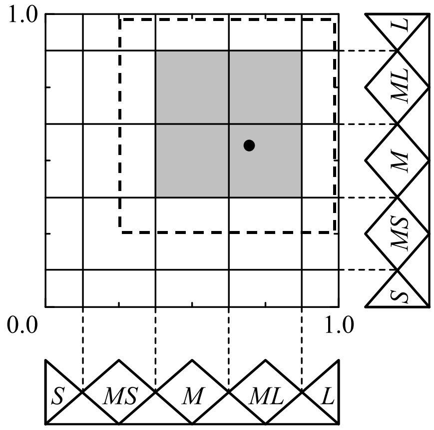
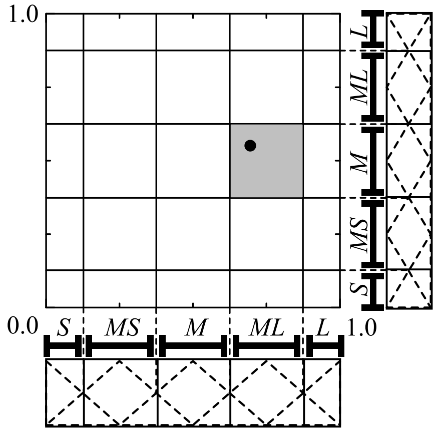


Fig. 5. Classification boundary by non-fuzzy if-then rules.

The difference between fuzzy and non-fuzzy partitions becomes clearer if we consider how many rules can be generated from a single training pattern (i.e., how many rules can be activated by a single training pattern). In the case of the crisp partition in Fig. 6 (b), only a single non-fuzzy if-then rule can be generated from a single training pattern. That is, the non-fuzzy if-then rule in the shaded area including the training pattern is generated. This is because there is no overlap between neighboring subspaces (i.e., no overlap between neighboring crisp intervals on each axis). On the other hand, if we use the fuzzy partition in Fig. 6 (a), four fuzzy if-then rules in the shaded area can be generated from the single training pattern. The generated four rules cover the larger square region denoted by dashed lines. In general, 2^n fuzzy if-then rules can be generated from a single training pattern for an *n*-dimensional pattern classification problem when we use a fuzzy partition such as Fig. 6 (a).



(a) A fuzzy partition



(b) A crisp partition

Fig. 6. Generated rules by a single training pattern.

From the comparison between Fig. 2 and Fig. 5, we can see that the same classification boundary is obtained from the fuzzy if-then rules in Fig. 2 and the non-fuzzy if-then rules in Fig. 5. Since the decision area of each fuzzy if-then rule is a rectangle or hyper-rectangle when there is no missing rule in a fuzzy rule table (see Kuncheva [21], [22]), classification boundaries by complete fuzzy rule tables can be also realized by non-fuzzy rule tables. This is not the case when each fuzzy if-then rule has a rule weight. That is, decision areas by fuzzy if-then rules with rule weights are not always rectangles or hyper-rectangles as shown in the next section.

IV. CLASSIFICATION WITH CERTAINTY GRADES

When we use the fuzzy if-then rules with certainty grades in (2), the winner rule R_{j*} for a new pattern $\mathbf{x}_p = (x_{p1}, ..., x_{pn})$ is defined by

$$\mu_{j*}(\mathbf{x}_p) \cdot CF_{j*} = \max\{\mu_j(\mathbf{x}_p) \cdot CF_j : j = 1, 2, ..., N\}. \tag{10}$$

As in the previous section, each fuzzy if-then rule has its own decision area. The size of the decision area of each rule is determined by its certainty grade and the membership functions of its antecedent linguistic values. That is, the decision area can be adjusted by modifying the certainty grade even if we do not change the membership functions. Some examples of decision areas are shown in Fig. 7. Those decision areas correspond to the nine fuzzy if-then rules in Fig. 1. It should be noted that we do not modify the membership functions of the three linguistic values *"S*: *small"*, *"M*: *medium"* and *"L*: *large"* in Fig. 1. We only change the certainty grade of each fuzzy if-then rule as shown in Table 1. Fig. 7 (a) corresponds to the case where all the nine fuzzy if-then rules have the same certainty grade. In this case, the decision area of each fuzzy if-then rule is the same as the case with no certainty grades (see Fig. 1). In Fig. 7 (b)-(f), certainty grades are not the same. In general, the larger the certainty grade of a fuzzy if-then rule is, the larger its decision area is. From Fig. 7, we can see that the decision area of each fuzzy if-then rule is not always rectangular even in the case of complete fuzzy rule tables (i.e., Fig. 7 (c), (d) and (f)).

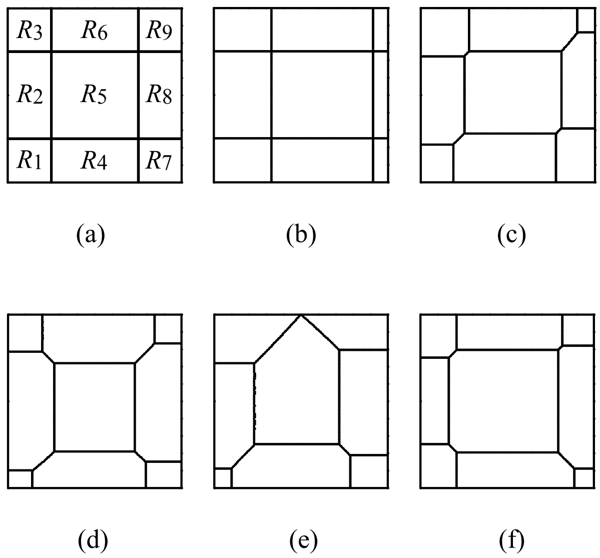


Fig. 7. Decision area of each fuzzy if-then rule with a different certainty grade.

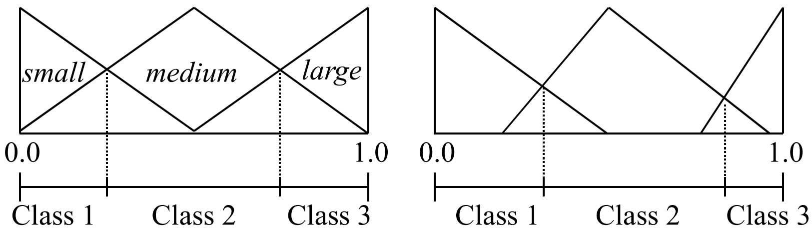
Table 1. Certainty grade of each of the nine fuzzy if-then rules in Fig. 7. The nine rules are labeled as in Fig.7 (a).

Figure	CF_1	CF_2	CF_3	CF_4	CF_5	CF_6	CF_7	CF_8	CF_9
Fig. 6 (a)	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0
Fig. 6 (b)	1.0	1.0	1.0	0.5	0.5	0.5	0.1	0.1	0.1
Fig. 6 (c)	0.6	0.8	1.0	1.0	0.8	0.8	0.8	0.5	0.2
Fig. 6 (d)	0.2	0.8	0.6	0.5	0.7	0.9	0.35	0.8	0.4
Fig. 6 (e)	0.2	0.7	0.9	0.8	0.8	0.0	0.6	1.0	0.7
Fig. 6 (f)	0.5	0.5	0.5	0.7	1.0	0.7	0.2	0.7	0.4

For illustrating the adjustment of classification boundaries, let us assume that we have the following three fuzzy if-then rules for a single-dimensional pattern classification problem on the unit interval [0,1]:

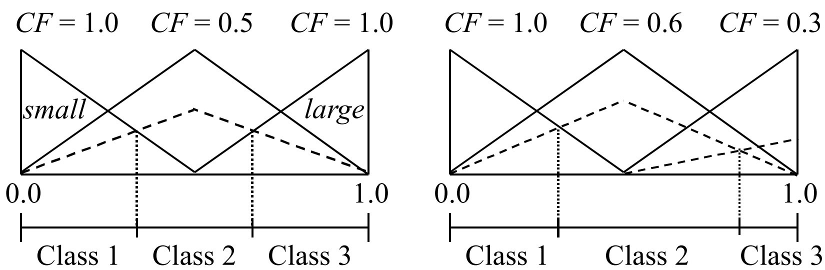
If x is *small* then Class 1,
If x is *medium* then Class 2,
If x is *large* then Class 3.

The unit interval is classified by these three rules as Fig. 8 (a). When we do not use certainty grades, the adjustment of classification boundaries is performed by modifying the membership function of each antecedent linguistic value as shown in Fig. 8 (b). On the other hand, when we use certainty grades, the adjustment of classification boundaries is performed by modifying the certainty grade of each fuzzy if-then rule. This is illustrated in Fig. 9 where dashed lines show the product of the certainty grade and the compatibility grade for each fuzzy if-then rule.



(a) (b)

Fig. 8. Adjustment of classification boundaries using fuzzy if-then rules without certainty grades.



(a) (b)

Fig. 9. Adjustment of classification boundaries using fuzzy if-then rules with certainty grades.

As in the previous section, the consequent class C_j of each fuzzy if-then rule R_j can be determined from the given training patterns by (9). That is, the consequent class C_j is specified as the dominant class in the fuzzy subspace corresponding to the antecedent part (see Fig. 10). The certainty grade can be viewed as the grade of the dominance of the consequent class. For example, the certainty grade is specified as follows for a two-class pattern classification problem [5]:

(i) When the consequent class is Class 1,

$$CF_j = \frac{\beta_{\text{Class 1}}(R_j) - \beta_{\text{Class 2}}(R_j)}{\beta_{\text{Class 1}}(R_j) + \beta_{\text{Class 2}}(R_j)}, \tag{11}$$

where

$$\beta_{\text{Class }k}(R_j) = \sum_{\mathbf{x}_p \in \text{Class }k} \mu_j(\mathbf{x}_p)\,,\ k = 1, 2\,. \tag{12}$$

(ii) When the consequent class is Class 2,

$$CF_j = \frac{\beta_{\text{Class 2}}(R_j) - \beta_{\text{Class 1}}(R_j)}{\beta_{\text{Class 1}}(R_j) + \beta_{\text{Class 2}}(R_j)}\,. \tag{13}$$

From (11) and (13), we can see that $0 \le CF_j \le 1$. When all compatible patterns with the fuzzy if-then rule R_j (i.e., such patterns that $\mu_j(\mathbf{x}_p) > 0$) belong to the same class, the certainty grade CF_j takes its maximum value (i.e., $CF_j = 1$). On the other hand, if no class is clearly dominant, the certainty grade is almost the same as its minimum value (i.e., $CF_j \cong 0$ in the case of $\beta_{\text{Class 1}}(R_j) \cong \beta_{\text{Class 2}}(R_j)$). The latter characteristic feature is the main motivation to use the formulations in (11)-(13). In Fig. 10, we illustrate the determination of the consequent class and the certainty grade. For the purpose of illustration, we use four different sets of training patterns in Fig. 10.

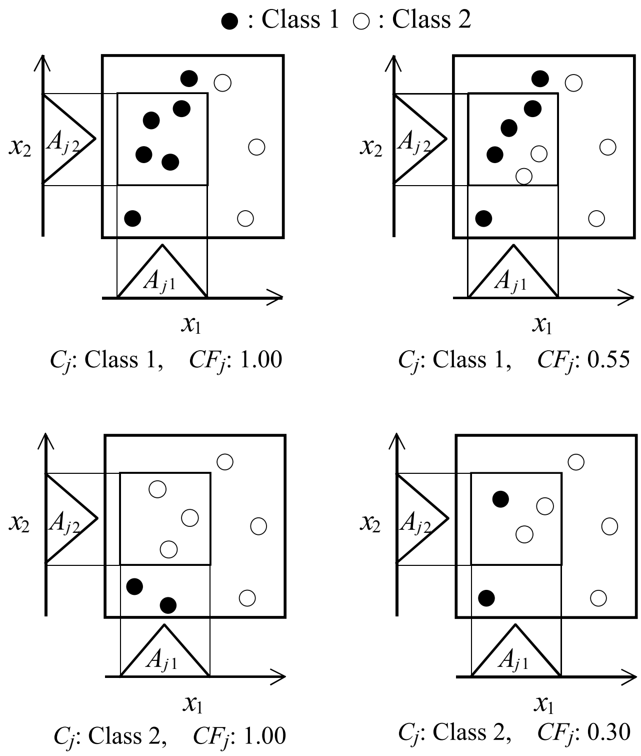


Fig. 10. Determination of the consequent class and the certainty grade.

From the given training patterns in Fig. 2, we specified the consequent class and the certainty grade of each fuzzy if-then rule in a 3×3 fuzzy rule table. We used the three linguistic values in Fig. 1 with no modification of their membership functions. Generated fuzzy if-then rules are shown in Fig. 11. One may think from Fig. 2 and Fig. 11 that the certainty grade of the upper-left fuzzy if-then rule (i.e., 0.41) is too small. When we use the non-fuzzy partition in Fig. 5, the certainty grade of the corresponding non-fuzzy rule is 1.0. Since the upper-left fuzzy if-then rule in Fig. 11 is compatible with some patterns from Class 2 outside the corresponding rectangular cell in Fig. 11 (see Fig. 2), the certainty grade 0.41 is much smaller than 1.0 (i.e., the certainty grade in the case of the non-fuzzy partition). The classification boundary by the fuzzy if-then rules in Fig. 11 is shown in Fig. 12. While six patterns were misclassified in Fig. 2 without certainty grades, two patterns are misclassified in Fig. 12 with certainty grades. From the comparison between Fig. 2 and Fig. 12, we can see that the use of certainty grades can improve the classification ability of fuzzy if-then rules. We can also see the difference between the learning of membership functions and the use of certainty grades from the comparison between Fig. 3 and Fig. 12.

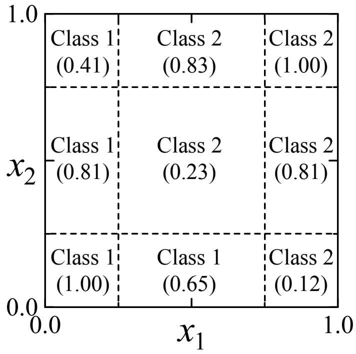


Fig. 11. Generated fuzzy if-then rules.

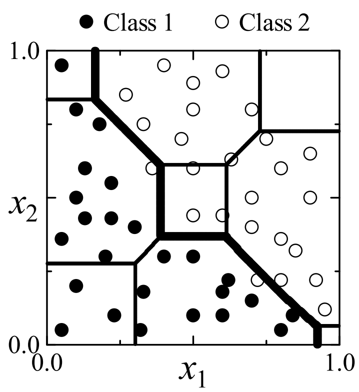


Fig. 12. Decision area of each rule and classification boundary.

The formulation for determining the certainty grade in (11)-(13) is extended to the case of c-class pattern classification problems as follows [5]:

$$CF_j = \frac{\beta_{\text{Class } C_j}(R_j) - \overline{\beta}}{\sum_{k=1}^{c} \beta_{\text{Class } k}(R_j)}, \tag{14}$$

where C_j is the consequent class, and

$$\overline{\beta} = \sum_{k \neq C_j} \beta_{\text{Class } k}(R_j) / (c-1). \tag{15}$$

V. HANDLING OF GENERAL AND SPECIFIC RULES

In this section, we illustrate the necessity of certainty grades when a fuzzy rule-based classification system is a mixture of general and specific fuzzy if-then rules. Let us assume that we have the following two fuzzy if-then rules:

$$R_1: \text{If } x_1 \text{ is } small \text{ and } x_2 \text{ is } small \text{ and } x_3 \text{ is } small \text{ then Class 2,} \tag{16}$$

$$R_2: \text{If } x_3 \text{ is } small \text{ then Class 1.} \tag{17}$$

The fuzzy if-then rule R_1 is specific while R_2 is general. From the definition of the compatibility grade in (5), we can see that R_1 can not have a larger compatibility grade than R_2 for any pattern:

$$\mu_{R_1}(\mathbf{x}) \le \mu_{R_2}(\mathbf{x}) \quad \text{for } \forall \mathbf{x} \tag{18}$$

This means that the specific fuzzy if-then rule R_1 is never selected as the winner rule in the classification phase for classifying new patterns.

When the above two fuzzy if-then rules are given, we intuitively think that R_1 may be used as an exceptional rule to R_2. That is, new patterns with *small* x_1, *small* x_2 and *small* x_3 are intuitively classified as Class 2 by the fuzzy if-then rule R_1 while R_1 does not have larger compatibility grades with those patterns than R_2. We can realize such intuitive reasoning as a fuzzy reasoning method by assigning a larger certainty grade to R_1 than R_2 (i.e., $CF_1 > CF_2$). In this case, the specific fuzzy if-then rule R_1 is selected as the winner rule for a new pattern $\mathbf{x}_p$ when the following inequality holds:

$$\mu_{R_1}(\mathbf{x}_p) \cdot CF_1 > \mu_{R_2}(\mathbf{x}_p) \cdot CF_2 . \tag{19}$$

It is intuitively acceptable that specific rules have larger certainty grades than general rules. Since general rules cover large areas of the pattern space, they may include several exceptions. On the other hand, specific rules usually include no exceptions because they cover very small areas. Therefore it is natural to assign larger certainty grades to specific rules than general rules. Large certainty grades assigned to specific rules in turn give higher priority to those rules than general rules in the classification phase.

As an example, let us consider a pattern classification problem in Fig. 13. All the given training patterns can be correctly classified by 25 fuzzy if-then rules without certainty grades corresponding to the 5×5 fuzzy partition in Fig. 14. If we use certainty grades, all the given training patterns can be correctly classified by the following three fuzzy if-then rules:

$$\text{If } x_1 \text{ is } medium \text{ then Class 2 with 0.75,} \tag{20}$$

$$\text{If } x_2 \text{ is } large \text{ then Class 2 with 0.84,} \tag{21}$$

$$\mathbf{x} \text{ is Class 1 with 0.19,} \tag{22}$$

where *medium* and *large* correspond to the triangular membership functions M and L in Fig. 14, respectively. The certainty grades of these rules were determined by the procedure in the previous section. The fuzzy if-then rule in (22) with no antecedent conditions can be viewed as having "*don't care*" conditions on both of the two attributes x_1 and x_2. A rectangular membership function that covers the entire domain [0, 1] of each attribute can be used as the antecedent fuzzy set corresponding to the "*don't care*" condition. The fuzzy if-then rules in (20) and (21) have a "*don't care*" condition on x_2 and x_1, respectively.

Since the certainty grade of the fuzzy if-then rule in (22) is very small, this rule is selected as the winner rule only when a new pattern does not have high compatibility grades with the other rules. In this manner, general and specific fuzzy if-then rules with certainty grades are efficiently utilized in our fuzzy rule-based classification system. If we remove the certainty grades from the above three rules, no pattern is classified as Class 2 because the most general rule (22) with the Class 1 consequence covers the entire pattern space.

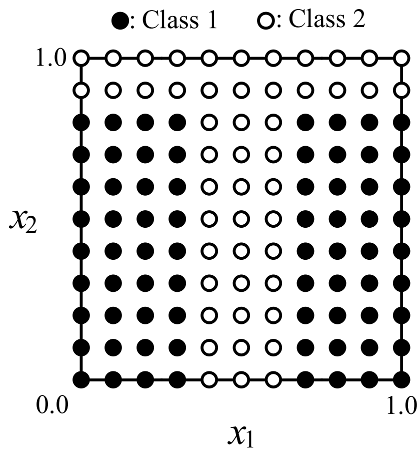


Fig. 13. An example of a pattern classification problem.

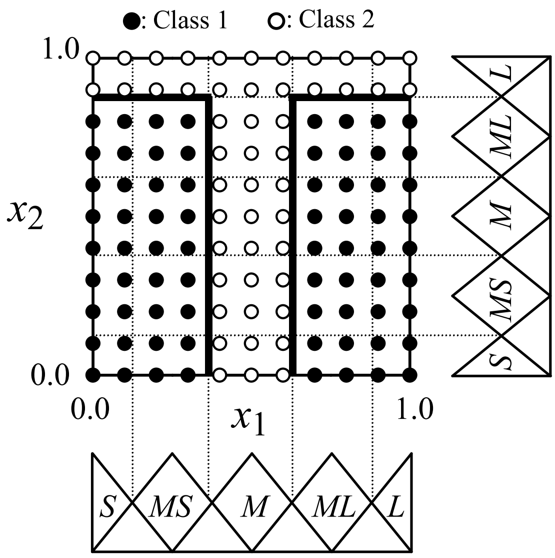


Fig. 14. Classification boundary by 25 fuzzy if-then rules without certainty grades.

As shown in the above example, certainty grades are necessary when our fuzzy rule-based classification system is a mixture of specific and general fuzzy if-then rules. General fuzzy if-then rules, however, are not always necessary in the case of low-dimensional problems with a few input variables. For example, the two-dimensional problem in Fig. 13 can be handled by 25 fuzzy if-then rules without certainty grades in Fig. 14. Of course, the above three rules are more concise and intuitive than those 25 fuzzy if-then rules in Fig. 14. On the other hand, general fuzzy if-then rules are necessary for handling high-dimensional problems with many input variables. Let K_i be the number of antecedent linguistic values for the i-th input variable. In this case, an n-dimensional fuzzy rule table consists of $K_1 \times \cdots \times K_n$ fuzzy if-then rules. The number of fuzzy if-then rules exponentially increases with the number of input variables. It is practically impossible to use all fuzzy if-then rules for a high-dimensional problem (i.e., when n is large). Thus we may use only a part of such a huge number of fuzzy if-then rules. The entire input space, however, can not be covered by a small number of fuzzy if-then rules because each rule covers only a tiny portion of the input space. Thus general fuzzy if-then rules with a few antecedent conditions are necessary for handling high-dimensional problems because each general rule covers a large fuzzy subspace. That is, it is possible to construct compact fuzzy rule-based systems using a small number of general fuzzy if-then rules for high-dimensional problems.

From the above discussions, we can see that general fuzzy if-then rules are necessary for handling high-dimensional problems. We also showed that certainty grades are necessary for handling fuzzy if-then rules with different specificity levels. As a result, we conclude that certainty grades are necessary for handling high-dimensional problems. The necessity of certainty grades is examined by computer simulations on wine data with 13 input variables in the next section.

VI. PERFORMANCE EVALUATION

We have already demonstrated that classification boundaries can be adjusted by modifying the certainty grade of each fuzzy if-then rule even when we use antecedent linguistic values with fixed membership functions. In this section, we examine the effect of certainty grades on the performance of fuzzy rule-based classification systems through computer simulations on the well-known iris data. We also examine the necessity of certainty grades for handling high-dimensional problems through computer simulations on wine data.

The iris data consist of 150 samples with four continuous attributes from three classes. We used the iris data after normalizing each attribute value to a real number in the unit interval [0,1]. That is, the iris data set was handled as a three-class pattern classification problem in the four-dimensional unit hypercube $[0,1]^4$. Since the generalization ability of fuzzy if-then rules with certainty grades had already been examined for the iris data by Nozaki et al.[23], we only examined the performance of fuzzy if-then rules without certainty grades. As in [23], we used the two-fold cross-validation (2CV) and the leaving-one-out (LV1) for evaluating the generalization ability (see Weiss and Kulikowski [24] for the 2CV and the LV1). In the 2CV, the iris data are divided into two subsets of the same size. That is, each subset consists of 75 samples. One subset is used as training data for generating fuzzy if-then rules. The other subset is used as test data for evaluating the generated fuzzy if-then rules. The same training-and-testing procedure is also performed after exchanging the role of each subset. Since the error rate on test data in the 2CV depends on the partition of the entire data set into the two subsets, we iterated the 2CV ten times using different partitions of the iris data set. On the other hand, the LV1 uses only a single sample as test data. The other 149 samples are used as training data. This procedure is iterated 150 times until all the given 150 samples are used as test data. In general, the average classification rate on test data in the

LV1 is higher than that in the 2CV because the size of training data in the LV1 is much larger than that in the 2CV. The average classification rate in the 2CV can be viewed as indicating the generalization ability when the size of training data is small.

In our computer simulations on the iris data, we homogeneously partitioned the pattern space $[0,1]^4$ by triangular fuzzy sets as in Fig. 14. We examined the performance of fuzzy if-then rules without certainty grades for fine fuzzy partitions as well as coarse fuzzy partitions. We used five different fuzzy partitions where each axis of the pattern space $[0,1]^4$ was uniformly divided into K triangular fuzzy sets ($K = 2,3,4,5,6$) in the same manner as Fig. 14 and other figures in this paper. For example, Fig. 6 (a) and Fig. 14 correspond to the cases of $K = 5$. Fig. 1 and Fig. 2 correspond to the case of $K = 3$. Each of the five fuzzy partitions was used for generating a fuzzy rule-based system from training data. That is, a four-dimensional $K \times K \times K \times K$ fuzzy rule table was generated from each fuzzy partition. For example, when $K = 2$, a fuzzy rule-based system is a $2 \times 2 \times 2 \times 2$ fuzzy rule table with 16 fuzzy if-then rules. A single fuzzy if-then rule was generated in each cell (i.e., each fuzzy subspace) of each fuzzy partition. Of course, all the $K \times K \times K \times K$ fuzzy if-then rules cannot be always generated. In general, many fuzzy if-then rules cannot be generated when we use fine fuzzy partitions. This is because each cell in fine fuzzy partitions is very small and likely to include no training patterns. When no training patterns are compatible with the antecedent part of a fuzzy if-then rule, we cannot specify its consequent part (thus we cannot generate the fuzzy if-then rule).

Simulation results are summarized in Table 2 and Table 3 where we also cite the results reported by Nozaki et al.[23]. In these tables, heuristic CF means that the certainty grade of each fuzzy if-then rule was determined by the heuristic procedure described in Section IV. Adjusted CF means that the certainty grade was adjusted by a reward-and-punishment scheme in Nozaki et al.[23]. Classification performance of various non-fuzzy classification methods on the iris data was examined using the LV1 in Weiss & Kulikowski [24] (e.g., 98.0% by the linear discriminant, 96.0% by the nearest neighbor method, 96.7% by the back-propagation algorithm, etc.).

From Table 2 and Table 3, we can observe the following:

(i) The performance of fuzzy if-then rules without certainty grades is poor especially when they are generated from the coarsest fuzzy partition (i.e., $K = 2$). In this case, the performance of fuzzy if-then rules with heuristic certainty grades is also poor.

(ii) By adjusting the certainty grade of each fuzzy if-then rule, we can improve the classification performance of fuzzy rule-based systems. This is prominent in the case of coarse fuzzy partitions.

(iii) In the case of fine fuzzy partitions with $K = 5$ and $K = 6$, the effect of the adjustment of certainty grades is not clear.

(iv) The performance of fuzzy if-then rules without certainty grades is very sensitive to the choice of a fuzzy partition. This sensitivity is remedied by introducing certainty grades and adjusting them.

Table 2. Average classification rates on test data evaluated by the 2CV method.

	$K=2$	$K=3$	$K=4$	$K=5$	$K=6$
Without CF	70.53	90.47	78.20	92.93	95.60
Heuristic CF	69.27*	92.43*	90.03*	95.27*	95.57*
Adjusted CF	91.73*	94.80*	94.53*	94.80*	95.37*

* Cited from Nozaki et al.[23]

Table 3. Average classification rates on test data evaluated by the LV1 method.

	$K=2$	$K=3$	$K=4$	$K=5$	$K=6$
Without CF	71.33	92.00	81.33	94.67	97.33
Heuristic CF	67.33*	93.33*	89.33*	95.33*	96.67*
Adjusted CF	92.00*	95.33*	98.00*	94.67*	96.67*

* Cited from Nozaki et al.[23]

The necessity of certainty grades for handling high-dimensional problems was examined through computer simulations on wine data. The wine data set is a three-class pattern classification problem with 13 continuous attributes. It is impractical to design fuzzy rule-based classification systems using 13-dimensional fuzzy rule-tables due to the exponential increase in the number of fuzzy if-then rules. We used a Michigan-style fuzzy GBML (genetics-based machine learning) algorithm [25] for designing compact fuzzy rule-based classification systems for the wine data. In our fuzzy GBML algorithm for the wine data, each fuzzy if-then rule is denoted by a string of the length 13 as "$A_{j1}A_{j2}\cdots A_{j13}$". Since the consequent class and the certainty grade of each fuzzy if-then rule can be easily specified by the heuristic procedure in Section IV, only the antecedent part is coded. Fuzzy if-then rules without certainty grades are also coded in the same manner. In this coding, "*don't care*" is handled as an additional antecedent linguistic value. Thus A_{ji} is one of the given linguistic values or "*don't care*". We use this coding for generating fuzzy if-then rules with different specificity levels (i.e., with a different number of "*don't care*" conditions in the antecedent part) by genetic operations in our fuzzy GBML algorithm (for details, see [25]).

In computer simulations on the wine data using our fuzzy GBML algorithm, the population size (i.e., the number of fuzzy if-then rules) was specified as 60. Classification performance was evaluated by using all the 178 samples in the wine data as training data. Because our fuzzy GBML algorithm is a stochastic search algorithm, the average classification rate was calculated by its 20 iterations. On the other hand, classification performance on test data was evaluated by five iterations of the ten-fold cross-validation (10CV) using different subdivisions of the wine data set into ten subsets (see [24] for the 10CV). In the 10CV, the wine data are divided into ten subsets of almost the same size. Nine subsets are used as training data for generating fuzzy if-then rules. The remaining single subset is used as test data for evaluating the generated fuzzy if-then rules. In the 10CV, this procedure is performed ten times after exchanging the role of each subset so that every subset is used as test data. Since the error rate on test data in the 10CV depends on the partition of the entire data set, we evaluated the generalization performance by five trials of the 10CV with different subdivisions of the wine data set into ten subsets.

Simulation results are summarized in Table 4 for training data and Table 5 for test data. From these tables, we can see that the search ability of our fuzzy GBML algorithm was severely deteriorated by the use of fuzzy if-then rules without certainty grades. These results support the discussions in Section V where we concluded that certainty grades are necessary for handling high-dimensional problems.

Table 4. Average classification rates on training data (wine data).

	$K = 3$	$K = 4$	$K = 5$	$K = 6$
Without CF	76.5	70.8	69.6	66.5
Heuristic CF	97.2	98.1	98.5	97.3

Table 5. Average classification rates on test data (wine data).

	$K = 3$	$K = 4$	$K = 5$	$K = 6$
Without CF	72.4	71.0	61.7	63.8
Heuristic CF	94.9	94.8	94.2	93.1

VII. CONCLUDING REMARKS

In this paper, we examined the effect of certainty grades on the performance of fuzzy if-then rules for pattern classification problems. First we showed that the decision area of each fuzzy if-then rule with no certainty grade is always rectangular (or hyper-rectangular) in the case of complete fuzzy rule tables. This means that classification boundaries are always parallel to the axes of the pattern space. Non axis-parallel classification boundaries can be generated only when some rules are missing (i.e., fuzzy rule tables are incomplete). This suggests the possibility to adjust classification boundaries by selecting fuzzy if-then rules when we do not use certainty grades. On the other hand, fuzzy if-then rules with certainty grades have decision areas of various shapes even when fuzzy rule tables are complete (i.e., when no rules are missing). We showed that classification boundaries are not always parallel to the axes of the pattern space in the case of fuzzy if-then rules with certainty grades. We also showed that classification boundaries can be adjusted by modifying the certainty grade of each fuzzy if-then rule even when we do not change the membership functions of antecedent fuzzy sets. Then we showed the necessity of certainty grades for handling fuzzy if-then rules of different specificity levels. This leads to the necessity of certainty grades when we apply fuzzy rule-based systems to high-dimensional problems. Finally we examined the effect of certainty grades through computer simulations on the iris data and the wine data. Simulation results on the iris data showed that the use of certainty grades and their adjustment can improve the classification performance of fuzzy if-then rules. This effect was prominent especially when we used coarse fuzzy partitions. When the fuzzy partition was fine, the effect of certainty grades was not significant. In the case of fine fuzzy partitions, the pattern space was divided into small fuzzy subspaces (i.e., small cells or patches). Thus we obtained high classification rates without adjusting certainty grades (i.e., without adjusting classification boundaries). On the other hand, simulation results on the wine data demonstrated the necessity of certainty grades for handling high-dimensional problems. We designed fuzzy rule-based classification systems by a fuzzy GBML algorithm. When we used fuzzy if-then rules without certainty grades, average classification rates decreased by more than 20 percent.

In this paper, we suggested that comprehensible fuzzy rule-based systems with high classification performance can be designed using linguistic values with fixed membership functions (also see Ishibuchi et al.[12], [25]). The certainty grade of each fuzzy if-then rule can be interpreted as the strength of that rule. In general, the larger the certainty grade is, the larger the decision area of the fuzzy if-then rule is. As shown in Fig. 8 and Fig. 9, the adjustment of certainty grades has a similar effect on classification boundaries to the modification of membership functions. When a set of linguistic values is given by domain experts for each attribute of a particular pattern classification problem, the modification of the membership function of each linguistic value alters their original linguistic meaning. On the contrary, the modification of certainty grades does not change the meaning of each linguistic value. It just affects the strength of each fuzzy if-then rule in the classification phase. One problem of the use of certainty grades is their interpretation. In this paper, we interpreted the certainty grade of each fuzzy if-then rule as the rule strength because the certainty grade has a direct effect on the size of the decision area. Nauck & Kruse [16] showed that the use of rule weights (i.e., certainty grades) can be viewed as the modification of membership functions in fuzzy reasoning. As shown in (10) in Section IV, the compatibility grade of each fuzzy if-then rule is multiplied by its certainty grade in our fuzzy reasoning method. Thus the use of the certainty grade can be interpreted as modifying the membership function of each antecedent fuzzy set as shown by the dashed lines in Fig. 9 in Section IV. According to [16], the certainty grade CF_i can be viewed as modifying the membership function of each antecedent fuzzy set A_{ji} as follows for an n-dimensional problem when we use the production operation for the calculation of the compatibility grade:

$$\mu_{A_{ji}}(x_i) \Rightarrow \sqrt[n]{CF_j} \cdot \mu_{A_{ji}}(x_i) \ . \tag{23}$$

Even in this interpretation, the certainty grade does not change the position of the antecedent fuzzy set (while the modified antecedent fuzzy set is not normal anymore). Whatever interpretation we may accept, the use of certainty grades introduces a new dimension of complexity to fuzzy if-then rules. Thus the choice between the learning of membership functions and the use of certainty grades may depend on problems and user's preference. As shown in this paper, these two schemes have similar (but not the same) effects on the adjustment of classification boundaries.

REFERENCES

[1] M. Sugeno, "An introductory survey of fuzzy control," *Information Sciences*, vol. 36, no. 1/2, pp. 59-83, 1985.

[2] C. C. Lee, "Fuzzy logic in control systems: fuzzy logic controller Part I and Part II," *IEEE Trans. on Systems, Man, and Cybernetics*, vol. 20, no. 2, pp. 404-435, March/April, 1990.

[3] D. Driankov, H. Hellendoorn, and M. Reinfrank, *An Introduction to Fuzzy Control*, Springer-Verlag, Berlin, 1993.

[4] C. T. Leondes (ed.), *Fuzzy Theory Systems: Techniques and Applications* (vol. 1 - vol. 4), Academic Press, San Diego, 1999.

[5] H. Ishibuchi, K. Nozaki, and H. Tanaka, "Distributed representation of fuzzy rules and its application to pattern classification," *Fuzzy Sets Systems*, vol. 52, no. 1, pp. 21-32, November, 1992.

[6] S. Abe and M. -S. Lan, "A method for fuzzy rules extraction directly from numerical data and its application to pattern classification", *IEEE Trans. on Fuzzy Systems*, vol. 3, no. 1, pp. 18-28, February, 1995.

[7]V. Uebele, S. Abe, and M. -S. Lan, "A neural-network-based fuzzy classifier," *IEEE Trans. on Systems, Man, and Cybernetics*, vol. 25, no. 2, pp. 353-361, February, 1995.

[8] S. Mitra and L. I. Kuncheva, "Improving classification performance using fuzzy MLP and two-level selective partitioning of the feature space," *Fuzzy Sets and Systems*, vol. 70, no. 1, pp. 1-13, February, 1995.

[9] D. Nauck and R. Kruse, "A neuro-fuzzy method to learn fuzzy classification rules from data," *Fuzzy Sets and Systems*, vol. 89, no. 3, pp. 277-288, August, 1997.

[10] S. Abe and R. Thawonmas, "A fuzzy classifier with ellipsoidal regions," *IEEE Trans. on Fuzzy Systems*, vol. 5, no. 3, pp. 358-368, August, 1997.

[11] L. I. Kuncheva and J. C. Bezdek, "A fuzzy generalized nearest prototype classifier," *Proc. of 7th IFSA World Congress*, vol. III, pp. 217-222 (Prague, June 25-29, 1997).

[12] H. Ishibuchi, T. Murata, and I. B. Turksen, "Single-objective and two-objective genetic algorithms for selecting linguistic rules for pattern classification problems," *Fuzzy Sets and Systems*, vol. 89, no. 2, pp. 135-149, July, 1997.

[13] O. Cordon, M. J. del Jesus, and F. Herrera, "Genetic learning of fuzzy rule-based classification systems cooperating with fuzzy reasoning methods," *International Journal of Intelligent Systems,* vol. 13, pp. 1025-1053, 1998.

[14] T. Murata, H. Ishibuchi, T. Nakashima, and M. Gen, "Fuzzy partition and input selection by genetic algorithms for designing fuzzy rule-based classification systems," in V. M. Porto, N. Saravanan, D. Waagen, and A. E. Eiben (eds.), *Springer-Verlag Lecture Notes in Computer Science Vol. 1447: Evolutionary Programming IV*, pp. 407-416, Springer-Verlag, Berlin, 1998.

[15] A. Gonzalez and R. Perez, "SLAVE: A genetic learning system based on an iterative approach," *IEEE Trans. on Fuzzy Systems*, vol. 7, no. 2, pp. 176-191, April, 1999.

[16] D. Nauck and R. Kruse, "How the learning of rule weights affects the interpretability of fuzzy systems," *Proc. of 7th IEEE International Conference on Fuzzy Systems*, pp. 1235-1240 (Anchorage, May 4-9, 1998).

[17] T. Takagi and M. Sugeno, "Fuzzy identification of systems and its applications to modeling and control," *IEEE Trans. on Systems, Man, and Cybernetics*, vol. 15, no. 1, pp. 116-132, January/February, 1985.

[18] J. C. Bezdek, "Pattern Analysis", in E. H. Ruspini, P. Bonissone and W. Pedrycz (eds.), *Handbook of Fuzzy Computation*, Chapter F6, Institute of Physics Publishing, London, 1998.

[19] O. Cordon, M. J. del Jesus, and F. Herrera, "A proposal on reasoning methods in fuzzy rule-based classification systems," *International Journal of Approximate Reasoning*, vol. 20, no. 1, pp. 21-45, January, 1999.

[20] H. Ishibuchi, T. Nakashima, and T. Morisawa, "Voting in fuzzy rule-based systems for pattern classification problems," *Fuzzy Sets and Systems*, vol. 103, no. 2, pp. 223-238, April, 1999.

[21] L. I. Kuncheva, *Fuzzy Classifier Design*, Physica-Verlag, Heidelberg, 2000.

[22] L. I. Kuncheva, “How good are fuzzy if-then classifiers,” *IEEE Trans. on Systems, Man, and Cybernetics - Part B: Cybernetics*, vol. 30, no. 4, pp. 501-509, August, 2000.

[23] K. Nozaki, H. Ishibuchi, and H. Tanaka, “Adaptive fuzzy rule-based classification systems,” *IEEE Trans. on Fuzzy Systems*, vol. 4, no. 3, pp. 238-250, August 1996.

[24] S. M. Weiss and C. A. Kulikowski, *Computer Systems That Learn*, Morgan Kaufmann, San Mateo, 1991.

[25] H. Ishibuchi, T. Nakashima, and T. Murata, “Performance evaluation of fuzzy classifier systems for multi-dimensional pattern classification problems,” *IEEE Trans. on Systems, Man, and Cybernetics - Part B: Cybernetics*, vol. 29, no. 5, pp. 601-618, October 1999.

Hisao Ishibuchi (M'93) received the B.S. and M.S. Degrees in precision mechanics from Kyoto University, Kyoto, Japan, in 1985 and 1987, respectively, and the Ph.D. degree from Osaka Prefecture University, Osaka, Japan, in 1992.

Since 1987, he has been with Department of Industrial Engineering, Osaka Prefecture University, where he is currently a Professor. He was a Visiting Research Associate at the University of Toronto from August 1994 to March 1995 and from July 1997 to March 1998. His research interests include fuzzy systems, genetic algorithms, neural networks, and evolutionary games.

Tomoharu Nakashima (S'99-M'01) received the B.S., M.S., and Ph.D. degrees in industrial engineering from Osaka Prefecture University, Osaka Japan, in 1995, 1997, 2000, respectively.

He is currently an Assistant Professor, Department of Industrial Engineering, Osaka Prefecture University. From June 1998 to September 1998, he was a Postgraduate Scholar at the Knowledge-Based Intelligent Engineering Systems Centre, University of South Australia. His research interests include fuzzy systems, machine learning, genetic algorithms, and multi-agent systems.