ELSEVIER

Fuzzy Sets and Systems 65 (1994) 237–253

FUZZY sets and systems

Construction of fuzzy classification systems with rectangular fuzzy rules using genetic algorithms

Hisao Ishibuchi*, Ken Nozaki, Naohisa Yamamoto, Hideo Tanaka

Department of Industrial Engineering, University of Osaka Prefecture, Gakuen-cho 1-1, Sakai, Osaka 593, Japan

Received January 1994; revised February 1994

Abstract

This paper proposes a genetic-algorithm-based approach to the construction of fuzzy classification systems with rectangular fuzzy rules. In the proposed approach, compact fuzzy classification systems are automatically constructed from numerical data by selecting a small number of significant fuzzy rules using genetic algorithms. Since significant fuzzy rules are selected and unnecessary fuzzy rules are removed, the proposed approach can be viewed as a knowledge acquisition tool for classification problems. In this paper, we first describe a generation method of rectangular fuzzy rules from numerical data for classification problems. We next formulate a rule selection problem for constructing a compact fuzzy classification system as a combinatorial optimization problem with two objectives: to minimize the number of selected fuzzy rules and to maximize the number of correctly classified patterns. We then show how genetic algorithms are applied to the rule selection problem. Last, we illustrate the proposed approach by computer simulations on numerical examples and the iris data of Fisher.

Key words: Pattern recognition; Data analysis methods; Fuzzy classification; Rule selection; Genetic algorithms

1. Introduction

Fuzzy-rule-based control systems have been applied to various problem (for example, see [14, 18]). Fuzzy rules in those control systems were usually derived from human experts. Recently several approaches have been proposed for automatically generating fuzzy rules from numerical data without domain experts [19, 21, 24]. Tuning techniques for the membership functions of antecedent and consequent fuzzy sets have been also proposed in many studies. For example, Ichihashi and Watanabe [6] and Nomura et al. [16] proposed tuning techniques based on descent methods. Horikawa et al. [5], Jang [11] and Lin and Lee [15] combined the learning ability of neural networks with fuzzy control systems to form self-learning fuzzy controllers. Berenji and Khedkar [1] proposed a reinforcement learning technique for fuzzy control systems.

Genetic algorithms [3, 4] have been also employed for the learning of fuzzy rules. For example, the membership functions of antecedent and consequent fuzzy sets of fuzzy rules were adjusted in [12, 13], the

* Corresponding author: Fax: +81-722-59-3340.

SSDI 0165-0114(94)00064-E

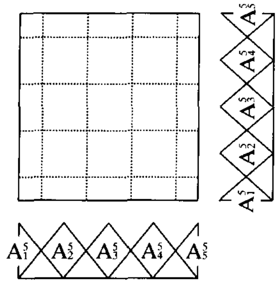


Fig. 1. An example of the fuzzy partition of a two-dimensional pattern space by a simple fuzzy grid.

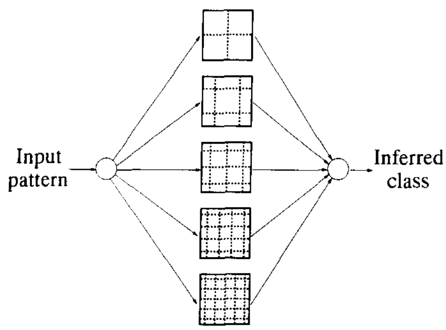


Fig. 2. A fuzzy classification system based on 90 fuzzy rules in five fuzzy partitions.

fuzzy partition of an input space was determined in [17], and an appropriate fuzzy set in the consequent part of each fuzzy rule was selected in [22]. Appropriate fuzzy sets in the antecedent and consequent parts of each fuzzy rule were also selected by the fuzzy classifier system in [23].

In the above-mentioned studies, fuzzy-rule-based systems were mainly applied to control problems. Since control problems and classification problems are different from each other, we cannot apply fuzzy control methods to fuzzy-rule-based classification systems. For example, in fuzzy control the inference step consists of composing the outputs of all rules then applying a defuzzification procedure while in fuzzy-rule-based classification the outcome of each rule is independent and a method is provided for determining which rule outcome to accept. Few methods for generating fuzzy rules have been proposed for classification problems [7–10]. Ishibuchi et al. [7] proposed a rule generation method from numerical data based on fuzzy partitions by simple fuzzy grids. Fig. 1 shows an example of the fuzzy partition of a two-dimensional pattern space by a simple fuzzy grid. The fuzzy classification method in [7] simultaneously employed all the fuzzy rules generated for several fuzzy partitions of different sizes. In Fig. 2, we show a fuzzy classification system based on 90 ($= 2^2 + 3^2 + 4^2 + 5^2 + 6^2$) fuzzy rules generated for five fuzzy partitions. The main drawback of this approach is that the number of fuzzy rules is enormous especially for classification problems in high-dimensional pattern spaces. In order to remove unnecessary fuzzy rules from fuzzy classification systems, Ishibuchi et al. [9, 10] proposed a genetic-algorithm-based approach that can reduce the number of fuzzy rules in fuzzy classification systems. Genetic algorithms were employed for selecting significant rules from the set of generated fuzzy rules.

Since the above-mentioned fuzzy classification methods were based on fuzzy partitions by simple fuzzy grids as shown in Figs. 1 and 2, all the fuzzy rules in those methods had square fuzzy subspaces. Fuzzy rules

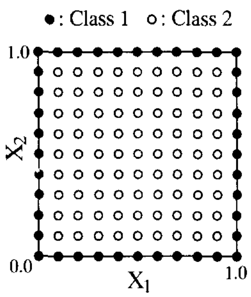


Fig. 3. A two-class classification problem. Closed circles and open circles represent the given patterns from Class 1 and Class 2, respectively.

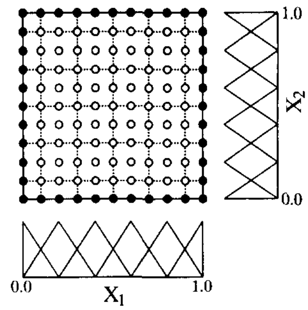


Fig. 4. A fuzzy partition with 36 fuzzy subspaces.

with rectangular fuzzy subspaces, however, may be more appropriate than square fuzzy rules in many classification problems. As an example, let us consider a two-class classification problem in Fig. 3 where closed circles and open circles denote the given patterns in Classes 1 and 2, respectively. If we try to classify all the given patterns by fuzzy rules based on a simple fuzzy grid, a fine fuzzy partition and a large number of fuzzy rules are required (see Fig. 4 where the pattern space is partitioned into 6×6 fuzzy subspaces). On the contrary, if we use fuzzy rules with rectangular fuzzy subspaces, all the given patterns in Fig. 3 may be correctly classified by the following five fuzzy rules:

Rule 1: If x_1 is very small then Class 1,

Rule 2: If x_1 is very large then Class 1,

Rule 3: If x_2 is very small then Class 1,

Rule 4: If x_2 is very large then Class 1,

Rule 5: If x_1 is not very small and x_1 is not very large and x_2 is not very small and x_2 is not very large then Class 2.

Five fuzzy subspaces corresponding to these fuzzy rules are shown in Fig. 5. That is, all the given patterns may be classified by the five fuzzy rules with the rectangular fuzzy subspaces in Fig. 5. From the comparison between Figs. 4 and 5, we can see that the introduction of rectangular fuzzy rules has a large effect on the reduction of the number of fuzzy rules. This discussion motivates us to propose a genetic-algorithm-based approach to the construction of compact fuzzy classification systems with rectangular fuzzy rules.

In this paper, we use the term "rectangular fuzzy rules" for referring to fuzzy rules that fire in rectangular (or hyper-rectangular) subspaces of a pattern space. This means that the antecedent fuzzy sets of a rectangular fuzzy rule compose a fuzzy subspace whose support set is a rectangle (or hyper-rectangle). Examples of fuzzy partition of a two-dimensional pattern space are shown in Fig. 6.

The outlines of the genetic-algorithm-based rule selection method proposed in this paper can be written as follows:

Step 1. Divide each axis of a pattern space by using triangular fuzzy sets. In this fuzzy partition, we use different triangular fuzzy sets with various sizes as shown in Fig. 6. The shape and the location of each fuzzy set are fixed.

Step 2. Divide the pattern space into rectangular fuzzy subspaces by using the triangular fuzzy sets in each axis. In this fuzzy partition, the pattern space is divided in various manners as shown in Fig. 6.

Step 3. Generate a fuzzy rule for each fuzzy subspace. The generated fuzzy rules are fixed.

Step 4. Select a small number of significant rules from the set of the generated fuzzy rules.

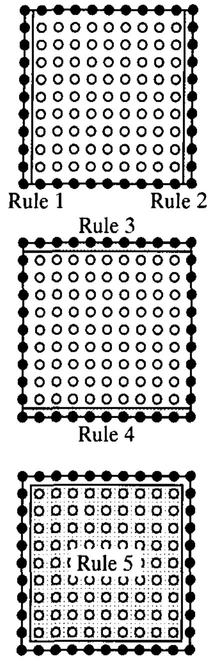


Fig. 5. Fuzzy subspaces corresponding to the five rectangular fuzzy rules.

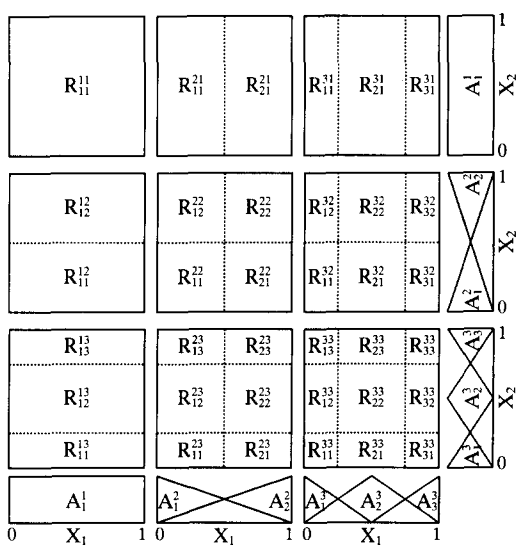


Fig. 6. Fuzzy subspaces generated by specifying $I_{max} = 3$ and $J_{max} = 3$ and the labels of corresponding fuzzy rules.

Genetic algorithms are applied only to the rule selection problem, and the membership function of each antecedent fuzzy set is fixed and not adjusted. Therefore, we do not always obtain the optimal (or minimum) number of rules. In order to compensate this inflexibility of each fuzzy rule, we first generate a large number of fuzzy rules with various membership functions, then select a small number of significant rules from them.

This paper is organized as follows. In Section 2, we describe a generation method of rectangular fuzzy rules from numerical data for classification problems. A fuzzy reasoning method based on the generated fuzzy rules is also described in the same section for classifying new patterns. In Section 3, we formulate a combinatorial optimization problem to construct compact fuzzy classification systems by selecting significant rules from the generated fuzzy rules. This optimization problem has two objectives: to minimize the number of selected fuzzy rules and to maximize the number of correctly classified patterns. We then show how genetic algorithms are applied to this problem. In Section 4, simulation results for the numerical example in Fig. 3 and the classification problem of the iris data [2] are shown to illustrate the proposed approach. In the application to the iris data, it is shown that 149 patterns (99.33% of the given patterns) can be correctly classified by only five fuzzy rules. Finally, Section 5 concludes this paper.

2. Fuzzy-rule-based classification systems

In this section, we describe a generation method of rectangular fuzzy rules from numerical data. Basically, this method is the same as the generation method of square fuzzy rules in [7] except for fuzzy partitions.

2.1. Classification problems

Let us consider a classification problem in the two-dimensional pattern space $[0, 1] \times [0, 1]$ for enhancing graphical illustration. It is assumed that m patterns $\boldsymbol{x}_p = (x_{p1}, x_{p2})$, $p = 1, 2, \ldots, m$ are given as training data from M classes ($C1$: Class 1, $C2$: Class 2, ..., CM: Class M). Fig. 3 shows an example of the classification problem of this kind (In Fig. 3, $M = 2$ and $m = 121$). Our aim is to construct a fuzzy classification system with rectangular fuzzy rules from these numerical data.

2.2. Fuzzy partition

Let us divide each axis of the pattern space into K ($K \geqslant 2$) fuzzy subsets $\{A_1^K, A_2^K, \ldots, A_K^K\}$. We can use any type of membership functions (e.g., triangular, trapezoid and exponential) for these fuzzy subsets. In this paper, the following symmetric triangular membership function is employed for A_i^K.

$$\mu_i^K(x) = \max\{1 - |x - a_i^K|/b^K, 0\}, \quad i = 1, 2, \ldots, K \ (K \geqslant 2), \tag{1}$$

where $\mu_i^K(x)$ is the membership function of A_i^K and

$$a_i^K = (i-1)/(K-1), \quad i = 1, 2, \ldots, K, \tag{2}$$

$$b^K = 1/(K-1). \tag{3}$$

For the case of $K = 1$, let us define the membership function of A_1^1 as

$$\mu_1^1(x) = \begin{cases} 1 & \text{if } 0 \leqslant x \leqslant 1, \\ 0 & \text{otherwise.} \end{cases} \tag{4}$$

In this case, the fuzzy subset A_1^1 is the unit interval $[0, 1]$.

The pattern space is partitioned by Cartesian product of the fuzzy partition of each axis. In our former work [7–10], we employed the same fuzzy partition for each axis. This led to the fuzzy partition of the pattern space by square fuzzy subspaces as shown in Fig. 1. In this paper, we relax this restriction on the fuzzy partition to generate rectangular fuzzy subspaces.

Let us divide the axis of the first attribute value (i.e., the horizontal axis: x_1) into I fuzzy subsets $\{A_1^I, A_2^I, \ldots, A_I^I\}$ and the axis of the second attribute value (i.e., the vertical axis: x_2) into J fuzzy subsets

$\{A_1^J, A_2^J, \dots, A_J^J\}$. The membership functions of these fuzzy subsets are given by (1)–(3) or (4). In this case, the pattern space is partitioned into $I \times J$ fuzzy subspaces: $\{A_i^I \times A_j^J$: $i = 1, 2, \dots, I$; $j = 1, 2, \dots, J\}$. Various fuzzy partitions of the pattern space can be constructed by combining different fuzzy partitions of the two axes. Let us assume that the possible values of I and J are $I = 1, 2, \dots, I_{\max}$ and $J = 1, 2, \dots, J_{\max}$. Then the total number of possible fuzzy partitions of the pattern space constructed by the combination of fuzzy partitions of the two axes is $I_{\max} \times J_{\max}$. If we use all the $I_{\max} \times J_{\max}$ fuzzy partitions, $(1 + 2 + \cdots + I_{\max}) \times (1 + 2 + \cdots + J_{\max})$ fuzzy subspaces: $\{A_i^I \times A_j^J$: $i = 1, 2, \dots, I$; $I = 1, 2, \dots, I_{\max}$; $j = 1, 2, \dots, J$; $J = 1, 2, \dots, J_{\max}\}$ are generated. All the fuzzy subspaces for the case of $I_{\max} = 3$ and $J_{\max} = 3$ are shown in Fig. 6 (The label R_{ij}^{IJ} of each fuzzy subspace is explained in the next subsection).

2.3. Rule generation

Let R_{ij}^{IJ} be the label of the fuzzy rule corresponding to the fuzzy subspace $A_i^I \times A_j^J$ (see Fig. 6). As in our former studies [7–10], the fuzzy rule R_{ij}^{IJ} for the two-dimensional classification problem can be written as follows.

Rule R_{ij}^{IJ}: If x_{p1} is A_i^I and x_{p2} is A_j^J then (x_{p1}, x_{p2}) belongs to Class C_{ij}^{IJ} with $CF = CF_{ij}^{IJ}$,

$$i = 1, 2, \dots, I;\ I = 1, 2, \dots, I_{\max};\ j = 1, 2, \dots, J;\ J = 1, 2, \dots, J_{\max}, \tag{5}$$

where the consequent C_{ij}^{IJ} is one of the M classes and CF_{ij}^{IJ} is the certainty of the fuzzy rule R_{ij}^{IJ}. Since one fuzzy rule corresponds to one fuzzy subspace, the total number of the fuzzy rules in (5) is $(1 + 2 + \cdots + I_{\max}) \times (1 + 2 + \cdots + J_{\max})$.

The consequent C_{ij}^{IJ} and the certainty CF_{ij}^{IJ} of each rule can be determined by the following procedure.

[Procedure 1: Generation of fuzzy rules]

(i) Calculate β_{CT} for $T = 1, 2, \dots, M$ as

$$\beta_{CT} = \sum_{x_p \in CT} \mu_i^I(x_{p1}) \times \mu_j^J(x_{p2}). \tag{6}$$

(ii) Find Class X (CX) such that

$$\beta_{CX} = \max\{\beta_{C1}, \beta_{C2}, \dots, \beta_{CM}\}. \tag{7}$$

If two or more classes take the maximum value in (7), the consequent C_{ij}^{IJ} of the fuzzy rule R_{ij}^{IJ} corresponding to the fuzzy subspace $A_i^I \times A_j^J$ cannot be determined uniquely. In this case, let C_{ij}^{IJ} be ϕ to denote that R_{ij}^{IJ} is a dummy rule. If a single class takes the maximum value in (7), C_{ij}^{IJ} is determined as Class X (CX) in (7).

(iii) If a single class takes the maximum value in (7), CF_{ij}^{IJ} is determined as

$$CF_{ij}^{IJ} = (\beta_{CX} - \beta) \Big/ \sum_{T=1}^{M} \beta_{CT}, \tag{8}$$

where

$$\beta = \sum_{\substack{T=1 \\ CT \neq CX}}^{M} \beta_{CT}/(M - 1). \tag{9}$$

In this procedure, the consequent C_{ij}^{IJ} is determined as Class X (CX) which has the largest sum of $\mu_i^I(x_{p1}) \times \mu_j^J(x_{p2})$ among the M classes in (7). Fuzzy rules with ϕ in the consequent part are dummy rules that have no effect on fuzzy inference for classifying new patterns. If there is no pattern in the fuzzy subspace $A_i^I \times A_j^J$, a dummy rule is generated by this procedure at that fuzzy subspace.

The definition of the certainty CF_{ij}^{IJ} in (8) and (9) is clear if we consider a two-class problem. Let us assume that $\beta_{C1} > \beta_{C2}$ for the fuzzy rule R_{ij}^{IJ} in a two-class problem. In this case, the consequent class C_{ij}^{IJ} is Class 1 and the certainty CF_{ij}^{IJ} is $(\beta_{C1} - \beta_{C2})/(\beta_{C1} + \beta_{C2})$. If there are no Class 2 patterns in the fuzzy subspace $A_i^I \times A_j^J$, it follows that $\beta_{C1} > \beta_{C2} = 0$ and $CF_{ij}^{IJ} = 1$ (Maximal certainty). On the other hand, if the numbers of Class 1 patterns and Class 2 patterns in the fuzzy subspace are similar to each other, it follows that $\beta_{C1} \approx \beta_{C2}$ and $CF_{ij}^{IJ} \approx 0$ (Minimal certainty).

Let us denote the set of all the generated $(1 + 2 + \cdots + I_{\max}) \times (1 + 2 + \cdots + J_{\max})$ fuzzy rules by S_{ALL}:

$$S_{\mathrm{ALL}} = \{R_{ij}^{IJ}: i = 1, 2, \ldots, I;\ I = 1, 2, \ldots, I_{\max};\ j = 1, 2, \ldots, J;\ J = 1, 2, \ldots, J_{\max}\}. \tag{10}$$

We also denote the set of the fuzzy rules in each fuzzy partition by S^{IJ}:

$$S^{IJ} = \{R_{ij}^{IJ}: i = 1, 2, \ldots, I;\ j = 1, 2, \ldots, J\}, \quad I = 1, 2, \ldots, I_{\max};\ J = 1, 2, \ldots, J_{\max}. \tag{11}$$

That is, S^{IJ} is the set of the fuzzy rules in the fuzzy partition by the $I \times J$ fuzzy grid. The rule set S_{ALL} of all the fuzzy rules can be written as

$$S_{\mathrm{ALL}} = \bigcup_{I=1}^{I_{\max}} \bigcup_{J=1}^{J_{\max}} S^{IJ}. \tag{12}$$

The rule selection problem in this paper is to select significant fuzzy rules and to remove unnecessary fuzzy rules from the rule set S_{ALL}. This problem will be discussed in Section 3.

2.4. *Classification of new patterns*

Let us assume that a subset S of the rule set S_{ALL} is given to form a fuzzy classification system. Using the fuzzy rules in S, a new pattern $\boldsymbol{x}_p = (x_{p1}, x_{p2})$ is classified by the following procedure.

[Procedure 2: Classification of a new pattern]

(i) Calculate α_{CT} for $T = 1, 2, \ldots, M$ as

$$\alpha_{CT} = \max\{\mu_i^I(x_{p1}) \times \mu_j^J(x_{p2}) \times CF_{ij}^{IJ}: C_{ij}^{IJ} = \text{Class } T \text{ and } R_{ij}^{IJ} \in S\}. \tag{13}$$

(ii) Find Class X (CX) such that

$$\alpha_{CX} = \max\{\alpha_{C1}, \alpha_{C2}, \ldots, \alpha_{CM}\}. \tag{14}$$

If two or more classes take the maximum value in (14) then the classification of $\boldsymbol{x}_p$ is rejected (i.e., $\boldsymbol{x}_p$ is left as an unclassifiable pattern), else assign $\boldsymbol{x}_p$ to Class X (CX) determined by (14).

In this procedure, the inferred class is the consequent of the fuzzy rule that has the maximum value of $\mu_i^I(x_{p1}) \times \mu_j^J(x_{p2}) \times CF_{ij}^{IJ}$ among all the fuzzy rules in S. If there are no fuzzy rules such that $\mu_i^I(x_{p1}) \times \mu_j^J(x_{p2}) \times CF_{ij}^{IJ} > 0$ at $\boldsymbol{x}_p$, that pattern $\boldsymbol{x}_p$ cannot be classified.

Examples of generated fuzzy rules and the corresponding classification result are shown in Fig. 7. Fig. 7(a) shows the 5×5 fuzzy rules in S^{55}:

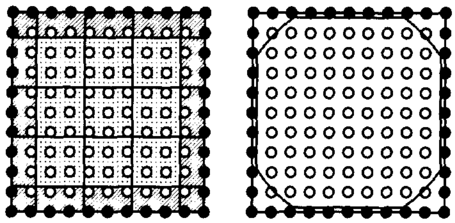
$$S^{55} = \{R_{ij}^{55}: i = 1, 2, 3, 4, 5;\ j = 1, 2, 3, 4, 5\}. \tag{15}$$

In Fig. 7(a), hatched areas and dotted areas represent the following:

(i) Hatched area: The consequent class of the generated fuzzy rule in this area is Class 1 (closed circles).

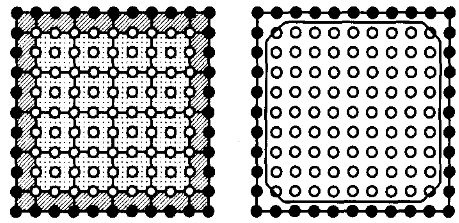
(ii) Dotted area: The consequent class of the generated fuzzy rule in this area is Class 2 (opened circles).

Fig. 7(b) shows the classification boundary obtained by specifying $S = S^{55}$ in Procedure 2. In Fig. 7(b), four patterns in Class 2 (open circles) are misclassified.



(a) Fuzzy rules in S^{55} (b) Classification boundary

Fig. 7. Fuzzy rules in the rule set S^{55} and the corresponding classification boundary.



(a) Fuzzy rules in S^{66} (b) Classification boundary

Fig. 8. Fuzzy rules in the rule set S^{66} and the corresponding classification boundary.

By using a finer fuzzy partition, we can increase the classification rate. For example, all the given patterns can be correctly classified by specifying $S = S^{66}$:

$$S^{66} = \{R_{ij}^{66}: i = 1, 2, 3, 4, 5, 6; j = 1, 2, 3, 4, 5, 6\}. \tag{16}$$

The 6×6 fuzzy rules in S^{66} and the classification boundary obtained by these rules are shown in Fig. 8(a) and (b), respectively. From Fig. 8(b), we can see that all the given patterns are correctly classified.

3. Rule selection by genetic algorithms

While all the given patterns are correctly classified in Fig. 8(b), the number of the fuzzy rules in Fig. 8(a) is not small. As shown in Fig. 5, the number of fuzzy rules can be reduced by selecting significant fuzzy rules with rectangular fuzzy subspaces. In this section, we show how genetic algorithms can be employed to select significant fuzzy rules for constructing a compact fuzzy classification system with a small number of rectangular fuzzy rules.

The main aim of this section is to show the feasibility of applying genetic algorithms to our rule selection problem. Therefore, the structure of genetic algorithms used in this section is very simple. It may not be entirely consistent with the emerging trends in recent research of genetic algorithms. This is because our aim is not to investigate genetic algorithms themselves but to show the applicability of genetic algorithms to our rule selection problem in a simple manner. By the same reason, we do not carefully adjust the parameter specifications for genetic algorithms. Therefore, simulation results in this paper will be improved if we use more sophisticated algorithms with carefully adjusted parameter values.

3.1. Formulation of a rule selection problem

Our rule selection problem is to find a compact rule set S that has high classification power. Therefore, our problem has the following two objectives:

(i) The first objective is to maximize the number of correctly classified patterns by the fuzzy rules in S.

(ii) The second objective is to minimize the number of the fuzzy rules in S.

By combining these two objectives, we formulate the following problem.

$$\text{maximize} \quad W_{\text{NCP}} \times \text{NCP}(S) - W_S \times |S|, \tag{17}$$

$$\text{subject to} \quad S \subseteq S_{\text{ALL}}, \tag{18}$$

where W_{NCP} and W_S are positive weights such that $W_S \ll W_{\mathrm{NCP}}$, $\mathrm{NCP}(S)$ is the number of correctly classified patterns by S, and $|S|$ is the number of the fuzzy rules in S. While the same problem was formulated in our former studies [9, 10], the definition of S_{ALL} is different. S_{ALL} in our former studies consisted of only square fuzzy rules, but S_{ALL} in this paper includes rectangular fuzzy rules.

In general, a fuzzy rule in a coarse fuzzy partition can classify more patterns than that in a fine fuzzy partition. Therefore, the former is more desirable for constructing a compact fuzzy classification system than the latter. If we try to construct a fuzzy classification system of fuzzy rules in a fine fuzzy partition, a large number of fuzzy rules may be required (see Fig. 8(a)). A fuzzy rule in a coarse fuzzy partition is also desirable from a point of view of knowledge acquisition because it is a general rule that can be valid in a large subspace of the pattern space. Therefore, we modify the rule selection problem (17) and (18) by assigning a different weight to each fuzzy rule.

Let us define an index of the fineness of each fuzzy rule as

$$\mathrm{Fineness}(R_{ij}^{IJ}) = I + J. \tag{19}$$

This index can be viewed as the fineness of the fuzzy partition where the fuzzy rule is generated. That is, the finer a fuzzy partition is, the larger the fineness of fuzzy rules in that fuzzy partition is. In order to select fuzzy rules in coarse fuzzy partitions (i.e., those with small fineness values), we modify the objective function of the rule selection problem as follows.

$$\text{maximize} \quad W_{\mathrm{NCP}} \times \mathrm{NCP}(S) - W_S \times \sum_{R_{ij}^{IJ} \in S} \mathrm{Fineness}(R_{ij}^{IJ}). \tag{20}$$

3.2. Genetic operations

In genetic algorithms in this paper, a rule set S is treated as an individual. The value of the objective function in (20) is the fitness value of each individual. That is, the fitness function $f(S)$ is defined as

$$f(S) = W_{\mathrm{NCP}} \times \mathrm{NCP}(S) - W_S \times \sum_{R_{ij}^{IJ} \in S} \mathrm{Fineness}(R_{ij}^{IJ}). \tag{21}$$

In genetic algorithms, each individual should be represented as a string. In this paper, let us represent a rule set S as $S = s_1 s_2 \ldots s_N$ where $N = (1 + 2 + \cdots + I_{\max}) \times (1 + 2 + \cdots + J_{\max})$ is the number of the fuzzy rules in S_{ALL}, and $s_r = 1$, -1 or 0 denotes the following:

$s_r = 1$ means that the rth rule is included in the rule set S,

$s_r = -1$ means that the rth rule is not included in the rule set S,

$s_r = 0$ means that the rth rule is a dummy rule.

The index r of each rule is specified as shown in Fig. 9 (see also Fig. 6 for the label of each rule). In general, the index r of the fuzzy rule R_{ij}^{IJ} is calculated as

$$r = (1 + 2 + \cdots + J - 1 + j - 1) \times (1 + 2 + \cdots + I_{\max}) + (1 + 2 + \cdots + I - 1 + i). \tag{22}$$

Since dummy rules have no effect on the fuzzy inference for classifying new patterns in Procedure 2, they should be excluded from a rule set S. Therefore, they are represented as $s_r = 0$ in the coding process in order to prevent S from including them. Dummy rules are preserved only to maintain the rule ordering consistency. This consistency makes the programming much easier if compared with a coding that excludes dummy rules from strings. While dummy rules make the representation of an organism (i.e., string) larger, they do not increase the search space because they are given a special encoding. If there were a great many of them, they would introduce some unpredictable biases in the genetic algorithm search.

A string $S = s_1 s_2 \ldots s_N$ is decoded as

$$S = \{R_{ij}^{IJ} \colon s_r = 1;\ r = 1, 2, \ldots, N\}. \tag{23}$$

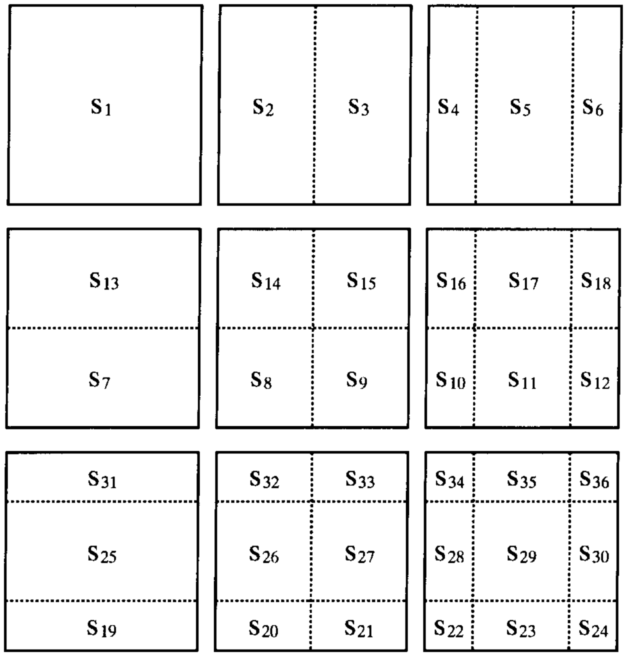


Fig. 9. The index of each fuzzy rule in the case of $I_{max} = 3$ and $J_{max} = 3$.

The following genetic operations are employed to generate and handle a set of strings (i.e., a population) in genetic algorithms of this paper (Steps 1–5 are iterated until a prespecified stopping condition is satisfied).

[Genetic algorithm for the rule selection problem]

Step 0 (Initialization). Generate an initial population containing N_{pop} strings where N_{pop} is the number of strings in each population. In this operation, each string is generated by assigning 0 to dummy rules and randomly assigning 1 or -1 to the other rules.

Step 1 (Selection). Select $N_{pop}/2$ pairs of strings from the current population. The selection probability $P(S)$ of string S in a population Ψ is specified as

$$P(S) = \frac{f(S) - f_{\min}(\Psi)}{\sum_{S' \in \Psi}\{f(S') - f_{\min}(\Psi)\}}, \tag{24}$$

where

$$f_{\min}(\Psi) = \min\{f(S) : S \in \Psi\}. \tag{25}$$

Step 2 (Cross-over). For each selected pair, randomly choose bit positions. Each bit position is chosen with the probability of 0.5. Interchange the bit values at the chosen positions in the selected pair.

Step 3 (Mutation). For each bit value of the generated strings by the cross-over operation, apply the following mutation operation:

$$s_r = 1 \to s_r = -1 \text{ with the mutation probability } P_m(1 \to -1), \tag{26}$$

$$s_r = -1 \to s_r = 1 \text{ with the mutation probability } P_m(-1 \to 1). \tag{27}$$

Parent 1:	s_1	s_2	s_3	s_4	s_5	s_6	s_7	...	s_N
Parent 2:	$\underline{s}_1$	$\underline{s}_2$	$\underline{s}_3$	$\underline{s}_4$	$\underline{s}_5$	$\underline{s}_6$	$\underline{s}_7$	...	$\underline{s}_N$
Positions:	*		*			*			
Child 1:	$\underline{s}_1$	s_2	$\underline{s}_3$	s_4	s_5	$\underline{s}_6$	s_7	...	s_N
Child 2:	s_1	$\underline{s}_2$	s_3	$\underline{s}_4$	$\underline{s}_5$	s_6	$\underline{s}_7$	...	$\underline{s}_N$

Fig. 10. Illustration of the crossover operation. Bit values at the marked positions are interchanged in the two parents to form two children.

Step 4 (Elitist strategy). Randomly remove one string from N_{pop} strings generated by the above operations, and add the string with the maximum fitness value in the previous population to the current one.

Step 5 (Termination test). If a prespecified stopping condition is not satisfied, return to Step 1.

The cross-over operation in Step 2 is illustrated in Fig. 10. This type of cross-over was called as the uniform cross-over in [20]. In Step 3, different mutation probabilities $P_m(1 \to -1)$ and $P_m(-1 \to 1)$ are assigned to the mutations from 1 to -1 and from -1 to 1, respectively. A large probability is usually assigned to $P_m(1 \to -1)$ than to $P_m(-1 \to 1)$ in order to reduce the number of fuzzy rules in each individual. The effect of these biased mutation probabilities was investigated in [10]. The total number of generations is used as a stopping condition in this paper.

4. Simulation results

4.1. Simulation results for numerical examples

The genetic algorithm described in the last section was applied to the classification problem in Fig. 3. The rule set S_{ALL} was generated by specifying $I_{max} = 6$ and $J_{max} = 6$ because all the patterns were correctly classified by the fuzzy rules in the fuzzy partition by the 6×6 fuzzy grid in Fig. 8(a). The total number of the fuzzy rules in S_{ALL} is $(1 + 2 + 3 + 4 + 5 + 6) \times (1 + 2 + 3 + 4 + 5 + 6) = 441$. These fuzzy rules were generated as candidate rules to form a fuzzy classification system. The length of each string in the genetic algorithm was also 441. The positive weights W_{NCP} and W_S in the fitness function (21) and the population size N_{pop} were specified as $W_{NCP} = 1000$, $W_S = 1$ and $N_{pop} = 50$. The mutation probabilities were specified as $P_m(1 \to -1) = 0.01$ and $P_m(-1 \to 1) = 0.001$. The algorithm was terminated after 2000 populations were generated.

We applied the genetic algorithm with these parameters specifications to the classification problem in Fig. 3. Computer simulations were performed 10 times with different initial populations. In nine trials out of the 10 simulations, the same rule set with five fuzzy rules was selected. The selected five fuzzy rules are shown tin Fig. 11 (a)–(c) where hatched areas and dotted areas denote the selected fuzzy rules. Fig. 11(d) shows the classification boundary obtained by the selected fuzzy rules. From Fig. 11(d), we can see that all the given patterns are correctly classified by the selected fuzzy rules. The selected fuzzy rules are also shown in Fig. 12 where meshed rectangles represent the antecedent fuzzy set A_1^1 (i.e., unit interval). This A_1^1 plays a special role in fuzzy rules. In fuzzy rules with A_1^1 for one feature and a different fuzzy set (e.g., A_1^6) for the other feature, the feature with A_1^1 does not matter. The fuzzy rule R_{11}^{11} with A_1^1 for both features is used only when none of the other rules have a sufficiently large compatibility to an input pattern. Therefore, each fuzzy rule in Fig. 12 can be interpreted as the following linguistic rules:

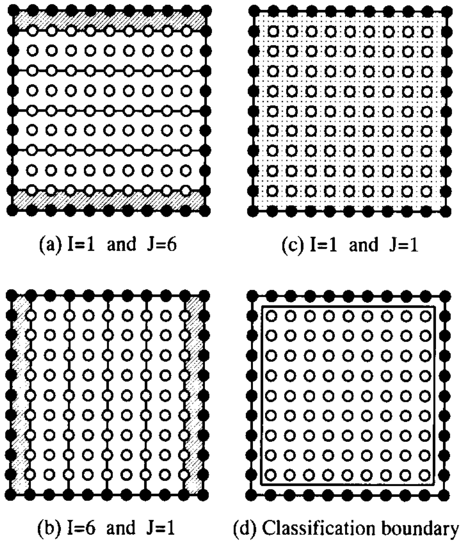


Fig. 11. Selected five fuzzy rules in nine trials and the corresponding classification boundary.

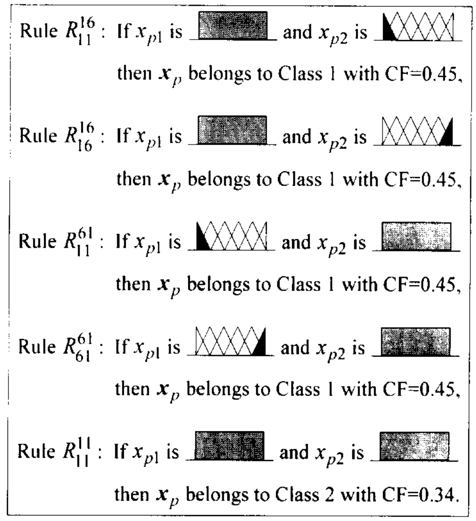


Fig. 12. Selected five fuzzy rules in nine trials.

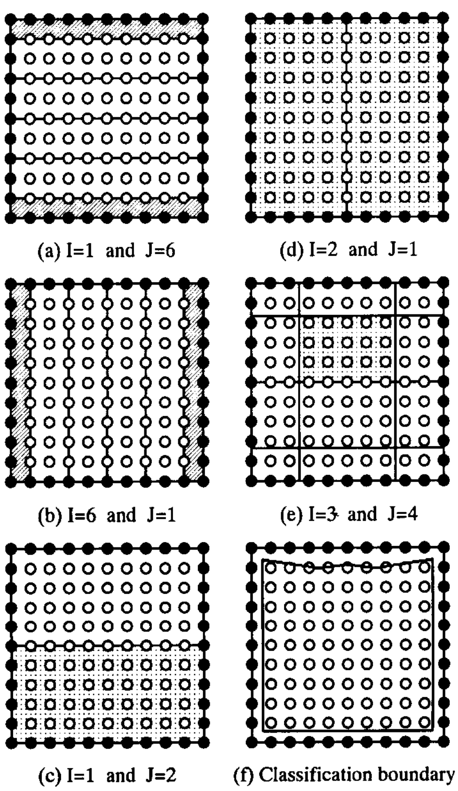


Fig. 13. Selected eight fuzzy rules in the other trial and the corresponding classification boundary.

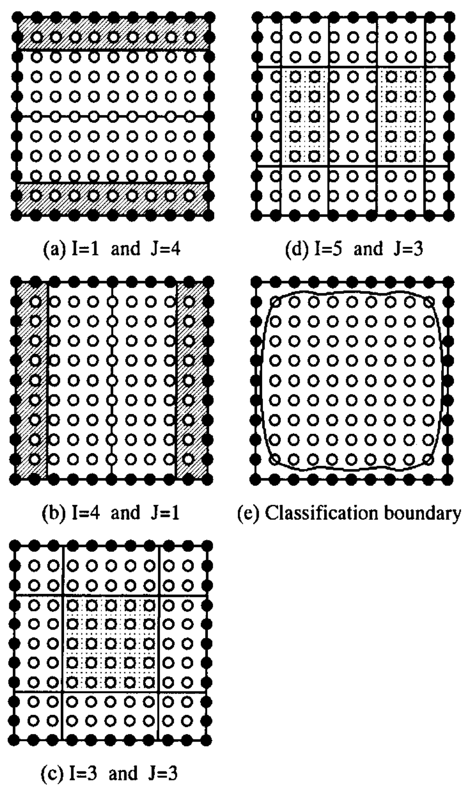


Fig. 14. Selected fuzzy rules under the parameter specifications of $I_{max} = 5$ and $J_{max} = 5$ and the corresponding classification boundary.

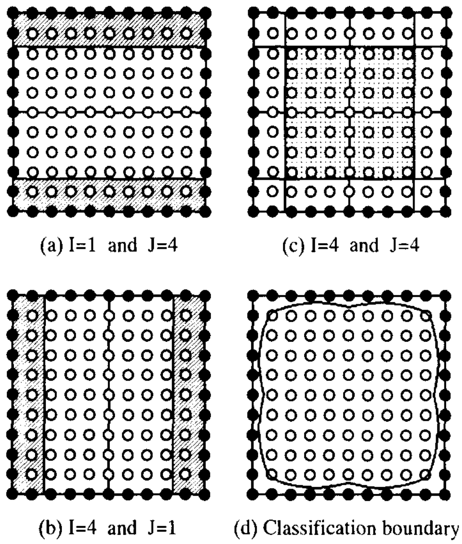


Fig. 15. Selected fuzzy rules under the parameter specifications of $I_{max} = 4$ and $J_{max} = 4$ and the corresponding classification boundary.

(1) Rule R_{11}^{16} (the bottom rule in Fig. 11(a)): If x_{p2} is very small then $\boldsymbol{x}_p$ belongs to Class 1 (medium certainty).
(2) Rule R_{16}^{16} (the top rule in Fig. 11(a)): If x_{p2} is very large then $\boldsymbol{x}_p$ belongs to Class 1 (medium certainty).
(3) Rule R_{11}^{61} (the left rule in Fig. 11(b)): If x_{p1} is very small then $\boldsymbol{x}_p$ belongs to Class 1 (medium certainty).
(4) Rule R_{61}^{61} (the right rule in Fig. 11(b)): If x_{p1} is very large then $\boldsymbol{x}_p$ belongs to Class 1 (medium certainty).
(5) Rule R_{11}^{11} (the rule in Fig. 11(c)): $\boldsymbol{x}_p$ belongs to Class 2 (small certainty).

If we try to intuitively derive classification rules from the given patterns in this numerical example, we will have similar linguistic rules. Therefore, we can conclude that the selected fuzzy rules coincide with our intuitive pattern recognition.

One trial out of the 10 simulations could not find the rule set in Fig. 11. The simulation results of this trial are shown in Fig. 13. In Fig. 13, all the given patterns are correctly classified by the selected eight fuzzy rules. From the comparison between Fig. 11(a) and (b) and Fig. 13(a) and (b), we can see that the same four fuzzy rules with Class 1 in the consequent part were selected in all the 10 trials.

In order to examine the effect of the specification of the rule set S_{ALL} on the result by the genetic algorithm, we also performed computer simulations with different two specifications of S_{ALL}. One rule set was generated by specifying $I_{max} = 5$ and $J_{max} = 5$. In this case, the total number of the fuzzy rules in S_{ALL} is $(1 + 2 + 3 + 4 + 5) \times (1 + 2 + 3 + 4 + 5) = 225$. The other rule set was generated by specifying $I_{max} = 4$ and $J_{max} = 4$ (100 fuzzy rules were generated). The best result in 10 trials with each specification of S_{ALL} is shown in Figs. 14 and 15 where all the given patterns are correctly classified by the selected seven and eight fuzzy rules, respectively. From Figs. 11, 13–15, we can see that the selected rule sets are similar to each other.

We also applied the genetic algorithm to the classification problem in Fig. 16. It was shown in [8] that this classification problem cannot be handled by a single fuzzy rule table with square fuzzy rules such as Fig. 1. The selected fuzzy rules and the classification boundary are shown in Fig. 17. From this figure, we can see that all the given patterns are correctly classified by the selected four fuzzy rules.

4.2. Simulation results for the iris data

In order to examine the ability of the proposed approach in high-dimensional classification problems, we applied the genetic algorithm to the iris data of Fisher (three-class classification problem in a

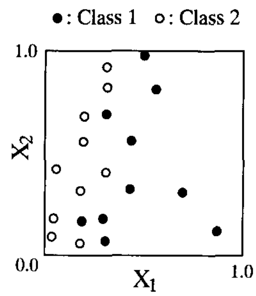


Fig. 16. Classification problem.

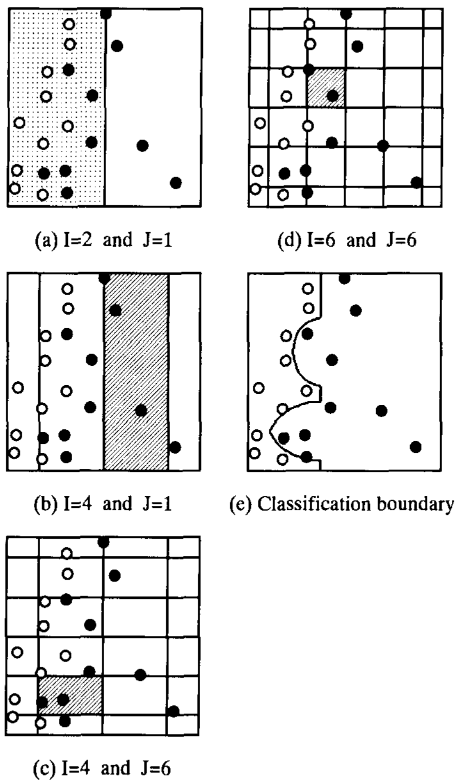


Fig. 17. Selected four fuzzy rules and the corresponding classification boundary.

four-dimensional pattern space [2]). Our approach described in the previous sections for two-dimensional problems can be easily extended to the case of high-dimensional classification problems. For example, the fuzzy rules in (5) are modified for handling the four-dimensional classification problem as

$$\text{Rule } R_{ijkl}^{IJKL}\text{: If } x_{p1} \text{ is } A_i^I \text{ and } x_{p2} \text{ is } A_j^J \text{ and } x_{p3} \text{ is } A_k^K \text{ and } x_{p4} \text{ is } A_l^L$$

$$\text{then } x_p \text{ belongs to Class } C_{ijkl}^{IJKL} \text{ with } CF = CF_{ijkl}^{IJKL}, \qquad (28)$$

where $\boldsymbol{x}_p = (x_{p1}, x_{p2}, x_{p3}, x_{p4})$. The definitions of β_{CT} in (6) and α_{CT} in (13) are modified as

$$\beta_{CT} = \sum_{x_p \in CT} \mu_i^I(x_{p1}) \times \mu_j^J(x_{p2}) \times \mu_k^K(x_{p3}) \times \mu_l^L(x_{p4}), \tag{29}$$

$$\alpha_{CT} = \max\{\mu_i^I(x_{p1}) \times \mu_j^J(x_{p2}) \times \mu_k^K(x_{p3}) \times \mu_l^L(x_{p4}) \times CF_{ijkl}^{IJKL}\colon CF_{ijkl}^{IJKL} = \text{Class } T \text{ and } R_{ijkl}^{IJKL} \in S\}. \tag{30}$$

After these modifications, we applied the genetic algorithm to the iris data. We employed the same parameter specifications as in Section 4.1 except for the mutation probabilities and the rule set S_{ALL}. The mutation probabilities were specified as $P_{\text{m}}(1 \to -1) = 0.1$ and $P_{\text{m}}(-1 \to 1) = 0.0001$. The rule set S_{ALL} was specified as follows:

$$S_{\text{ALL}} = \{R_{ijkl}^{IJKL}\colon i = 1, \dots, I;\ I = 1, 2, 3, 4;\ j = 1, \dots, J;\ J = 1, 2, 3, 4;\ k = 1, \dots, K;\ K = 1, 2, 3, 4;$$
$$l = 1, \dots, L;\ L = 1, 2, 3, 4\}. \tag{31}$$

The total number of fuzzy rules in S_{ALL} is 10000 ($= (1 + 2 + 3 + 4) \times (1 + 2 + 3 + 4) \times (1 + 2 + 3 + 4) \times (1 + 2 + 3 + 4)$). Therefore, our problem is to select significant rules from S_{ALL} with 10000 fuzzy rules (including 1918 dummy rules).

The genetic algorithm with these parameter specifications selected only five fuzzy rules that can correctly classify 149 patterns (99.33% of the given 150 patterns). From the viewpoint of the number of fuzzy rules, this result outperforms our previous work [10] where 149 patterns were correctly classified by 10 square fuzzy rules.

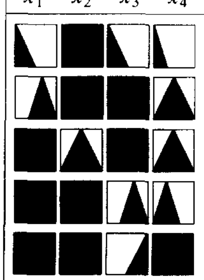
Each of the selected five fuzzy rules are shown in Fig. 18. From this figure, we can see that all the patterns in Class 1 can be correctly classified by the first rule and those in Class 3 by the last rule. From these two fuzzy rules, we can acquire the following linguistic knowledge:

(i) If the first, third and fourth attribute values of a pattern are small, then that pattern belongs to Class 1 (maximum certainty).

(ii) If the third attribute value of a pattern is large, then that pattern belongs to Class 3 (large certainty).

As these results on the iris data show, the proposed approach can be viewed as a knowledge acquisition tool from numerical data. Since a small number of fuzzy rules with high classification power are selected, we can carefully examine each of the selected rules. This is almost impossible if hundreds of fuzzy rules are selected to form a fuzzy classification system.

For the comparison of different cross-over operations, in Table 1 we show simulation results by the genetic algorithm with the uniform cross-over in Section 3 and that with a one-point cross-over. In Table 1, we



x_1 x_2 x_3 x_4	Consequent	CF	Patterns*
	Class 1	1.00	50
	Class 2	0.41	8
	Class 2	0.41	18
	Class 2	0.92	23
	Class 3	0.71	50

Patterns* : the number of patterns correctly classified by each fuzzy if-then rule

Fig. 18. Selected five fuzzy rules for the iris data of Fisher.

Table 1
Simulation results with different cross-over operations

Trial number	Uniform cross-over		One-point cross-over	
	No. of patterns	No. of rules	No. of patterns	No. of rules
1	149	5	149	5
2	149	6	149	5
3	148	6	150	7
4	149	7	149	6
5	149	7	149	6
Average	148.8	6.2	149.2	5.8

No. of patterns: the number of correctly classified patterns.
No. of rules: the number of selected rules.

cannot observe significant difference between the two cross-over operations. In Section 3, we used the uniform cross-over to remove unpredictable biases in the genetic algorithm search caused by our coding mechanism of strings (i.e., ordering of fuzzy rules). From Table 1, we can conclude that our coding mechanism shown in Fig. 9 had not bad biases in the genetic algorithm search even if we used a one-point cross-over.

5. Conclusion

In this paper, we proposed a genetic-algorithm-based approach to the construction of compact fuzzy classification systems with rectangular fuzzy rules. In the proposed approach, first a large number of rectangular fuzzy rules were generated from numerical data. Then significant rules were selected from the generated fuzzy rules by genetic algorithms to form a compact fuzzy classification system. By computer simulations on numerical examples, we demonstrated that the proposed approach can select a small number of fuzzy rules that coincide with our intuitive pattern recognition. The ability of the proposed approach was also demonstrated by the application to the iris data. That is, we showed that 149 patterns in the iris data can be correctly classified by only five rectangular fuzzy rules. From the point of view of the number of selected fuzzy rules, this result outperforms our former study [10] where 10 square fuzzy rules were selected for classifying 149 patterns.

Due to the probabilistic nature of the genetic algorithm search, the same fuzzy rules are not always selected by the proposed approach. Therefore, it is recommended to apply genetic algorithms several times to a classification problem at hand.

References

[1] H.R. Berenji and P. Khedkar, Learning and tuning fuzzy logic controllers through reinforcements, *IEEE Trans. Neural Networks* 3 (1992) 724–740.
[2] R.A. Fisher, The use of multiple measurements in taxonomic problems, *Ann. Eugenics* 7 (1936) 179–188.
[3] D.E. Goldberg, *Genetic Algorithms in Search, Optimization, and Machine Learning* (Addison-Wesley, Reading, MA, 1989).
[4] J.H. Holland, *Adaptation in Natural and Artificial Systems* (University of Michigan Press, Ann Arbor, MI, 1975).
[5] S. Horikawa, T. Furuhashi and Y. Uchikawa, On fuzzy modeling using fuzzy neural networks with the back-propagation algorithm, *IEEE Trans. Neural Networks* 3 (1992) 801–806.

[6] H. Ichihashi and T. Watanabe, Learning control by fuzzy models using a simplified fuzzy reasoning, *J. Japan Society for Fuzzy Theory and Systems* **2** (1990) 429–437 (in Japanese).

[7] H. Ishibuchi, K. Nozaki and H. Tanaka, Distributed representation of fuzzy rules and its application to pattern classification, *Fuzzy Sets and Systems* **52** (1992) 21–32.

[8] H. Ishibuchi, K. Nozaki and H. Tanaka, Efficient fuzzy partition of pattern space for classification problems, *Fuzzy Sets and Systems* **59** (1993) 295–304.

[9] H. Ishibuchi, K. Nozaki and N. Yamamoto, Selecting fuzzy rules by genetic algorithm for classification problems, *Proc. 2nd IEEE Internat. Conf. on Fuzzy Systems* (San Francisco, CA, March 28–April 1, 1993) 1119–1124.

[10] H. Ishibuchi, K. Nozaki, N. Yamamoto and H. Tanaka, Genetic operations for rule selection in fuzzy classification systems, *Proc. 5th IFSA World Congr.* (Seoul, Korea, July 4–9, 1993) 15–18.

[11] J.S.R. Jang, Self-learning fuzzy controllers based on temporal back propagation, *IEEE Trans. Neural Networks* **3** (1992) 714–723.

[12] C.L. Karr, Design of an adaptive fuzzy logic controller using a genetic algorithm, *Proc. 4th Internat. Conf. on Genetic Algorithms* (San Diego, CA, July 13–16, 1991) 450–457.

[13] C.L. Karr and E.J. Gentry, Fuzzy control of pH using genetic algorithms, *IEEE Trans. Fuzzy Systems* **1** (1993) 46–53.

[14] C.C. Lee, Fuzzy logic in control systems: fuzzy logic controller – Parts I and II, *IEEE Trans. on Systems, Man and Cybernetics* **20** (1990) 404–435.

[15] C.T. Lin and C.S.G. Lee, Neural-network-based fuzzy logic control and decision system, *IEEE Trans. on Computers* **40** (1991) 1320–1336.

[16] H. Nomura, I. Hayashi and N. Wakami, A learning method of fuzzy inference rules by descent method, *Proc. of the IEEE Internat. Conf. on Fuzzy Systems* (San Diego, CA, March 8–12) 203–210.

[17] H. Nomura, I. Hayashi and N. Wakami, A self-tuning method of fuzzy reasoning by genetic algorithm, *Proc. of the 1992 Internat. Fuzzy Systems and Intelligent Control Conf.* (Louisville, Kentucky, March 16–18, 1992) 236–245.

[18] M. Sugeno, An introductory survey of fuzzy control, *Information Sciences* **36** (1985) 59–83.

[19] M. Sugeno and T. Yasukawa, A fuzzy-logic-based approach to qualitative modeling, *IEEE Trans. Fuzzy Systems* **1** (1993) 7–31.

[20] G. Syswerda, Uniform crossover in genetic algorithms, *Proc. 3rd Internat. Conf. on Genetic Algorithms* (George Mason University, June 4–7, Morgan Kaufmann, San Mateo, CA, 1989) 2–9.

[21] T. Takagi and M. Sugeno, Fuzzy identification of systems and its applications to modeling and control, *IEEE Trans. Systems, Man and Cybernetics* **15** (1985) 116–132.

[22] P. Thrift, Fuzzy logic synthesis with genetic algorithms, *Proc 4th Internat. Conf. on Genetic Algorithms* (San Diego, CA, July 13–16, 1991) 509–513.

[23] M. Valenzuela-Rendon, The fuzzy classifier system: A classifier system for continuously varying variables, *Proc. 4th Internat. Conf. on Genetic Algorithms* (San Diego, CA, July 13–16, 1991) 346–353.

[24] L.X. Wang and J.M. Mendel, Generating fuzzy rules by learning from examples, *IEEE Trans. Systems, Man and Cybernetics* **22** (1992) 1414–1427.