GA-Based Construction of Fuzzy Classifiers Using Information Granules

Do Wan Kim, Ho Jae Lee, Jin Bae Park*, and Young Hoon Joo

Abstract: A new GA-based methodology using information granules is suggested for the construction of fuzzy classifiers. The proposed scheme consists of three steps: selection of information granules, construction of the associated fuzzy sets, and tuning of the fuzzy rules. First, the genetic algorithm (GA) is applied to the development of the adequate information granules. The fuzzy sets are then constructed from the analysis of the developed information granules. An interpretable fuzzy classifier is designed by using the constructed fuzzy sets. Finally, the GA is utilized for tuning of the fuzzy rules, which can enhance the classification performance on the misclassified data (e.g., data with the strange pattern or on the boundaries of the classes). To show the effectiveness of the proposed method, an example, the classification of the Iris data, is provided.

Keywords: Fuzzy classifier, fuzzy set, information granules, genetic algorithm.

1. INTRODUCTION

The classification technique plays an important role in various engineering application fields including medical diagnosis [3], signal processing [4], power system [5], and recognition [6]. There are two fundamental areas in the classification problem: feature selection and pattern classification. In dealing with the feature selection and/or the pattern classification, it is quite promising to utilize the knowledge of the domain experts. Since Zadeh suggested the fuzzy set in 1965 [8], the fuzzy theory has been regarded as a resolution of a mathematical modeling of the intuitive human knowledge [1,2]. With this technical point of view, various construction methods of fuzzy classifiers have been proposed [7,9-13,25,26]. A table look up scheme was studied to generate the fuzzy rules for the fuzzy classifier directly [9], or by using the genetic algorithm (GA) [10]. Abe *et al.* designed several fuzzy classifiers from the various information granules to express the characteristics of the fuzzy classifiers by considering the shapes of the information granules [7,12].

Generally, the construction process of the fuzzy classifier consists of the following three stages: (i) selecting the information granules, (ii) constructing the fuzzy sets associated with the information granules, and (iii) tuning the fuzzy rules. In Stage (i), it is important to optimally select the information granules from the feature spaces because the performance of the fuzzy classifier highly depends on the choice of the information granules. Unfortunately, dividing the feature space into the optimal information granules is generally known to be a complex and mutually associated problem. An approach to resolve this difficulty is to generate the activation and the inhibition hyperboxes, recursively [12]. However, it generates too many ones to analyze information granules easily as well as to show the pattern visually. In Stage (ii), the fuzzy sets produced by information granules should fulfill two requirements: the fuzzy sets should exhibit a large degree of overlapping, and the fuzzy sets should describe the context of the information granule. Wu *et al.* proposed a construction technique of the fuzzy sets via the α-cut and the similarity degree [14]. However, when all the fuzzy sets for a feature are overlapped, some of the fuzzy sets are merged into one. Consequently, they may be irrelevant to the pattern classification because all classes are represented by only one fuzzy set. In Stage (iii), the tuning of the fuzzy rules constructed from the previous stages is necessary to appropriately deal with the misclassified data (to improve the

Manuscript received December 23, 2004; revised July 29, 2005; accepted December 22, 2005. Recommended by Editor Jin Young Choi under the direction of past Editor-in-Chief Myung Jin Chung.

Do Wan Kim and Jin Bae Park are with the Department of Electrical and Electronic Engineering, Yonsei University, 134 Shinchon-dong, Seodaemoon-gu, Seoul 120-749, Korea (e-mails: dwkim@control.yonsei.ac.kr, jbpark@yonsei.ac.kr).

Ho Jae Lee is with the Department of Electronic Engineering, Inha University, 253 Yonghyun-dong, Nam-gu, Incheon 402-751, Korea (e-mail: mylchi@inha.ac.kr).

Young Hoon Joo is with the School of Electronic and Information Engineering, Kunsan National University, San 68, Miryong-dong, Kunsan, Jeonbuk 573-701, Korea (e-mail: yhjoo@kunsan.ac.kr).

* Corresponding author.

classification performance). Although Abe *et al.* proposed a tuning method of the slope of the membership function [7], their techniques neither consider the overfitting nor the outlier problem.

Motivated by the above observations, this paper aims at improving the performance of the fuzzy classifier by resolving the above-mentioned problems. To this end, we propose a GA-based construction method of the fuzzy classifier. In order to determine optimal information granules, the GA, which has been shown to be a flexible and robust optimization tool [15], are used in Stage (i). An efficient fuzzy sets constructing technique based on the α-cut and the similarity degree is then proposed in Stage (ii) to extract, merge, and reduce the fuzzy sets from the developed information granules, which produces a set of human interpretable classification rules in the form of the multi-inputs and multi-outputs (MIMO) Takagi-Sugeno (T-S) fuzzy model. In Stage (iii), to decrease the number of the misclassified data, the GA is also used to tune parameters of the obtained fuzzy sets and to newly generate additional fuzzy rules.

The layout of this paper is organized as follows: Section 2 introduces the fuzzy classifier model and reviews the information granules. Section 3 contains the proposed approach to the classification based on information granules. An example is provided in Section 4 to show the effectiveness of the proposed method. Section 5 concludes this paper.

2. PRELIMINARIES

This section describes the structure of the fuzzy classifier (i.e., MIMO T-S fuzzy system), and the concept of the information granules.

2.1. Fuzzy classifier model

To perform the classification, the MIMO T-S fuzzy system is designed by

$$\begin{aligned}&R_i: \text{IF } x_1 \text{ is } A_{i1} \text{ and } \ldots \text{ and } x_N \text{ is } A_{iN}\\ &\text{THEN the class is } i: (y_{i1}=\xi_{i1},\ldots,y_{iL}=\xi_{iL}),\end{aligned} \tag{1}$$

where $x_j, j \in I_N = \{1,2,...,N\}$, is the j th feature; $R_i, i \in I_L = \{1,...,L\}$, is the i th fuzzy rule; $y_{ih},(i,h) \in I_L \times I_L$, is the h th singleton output in the i th rule; $\xi_{ih} \in [0,1]$ is a real constant value; and $A_{ij},(i,j) \in I_L \times I_N$, is the fuzzy set to be determined from the information granules.

Remark 1: Here, we initially equate the number of the classes to the one of the fuzzy rules, and set the i th output of the i th fuzzy rule R_i, y_{ii} as a maximum so that the fuzzy rule R_i can well describe the i th class.

By using product inference engine, singleton fuzzifier, and center average defuzzifier, the global output of the h th class inferred from the T-S fuzzy system (1) is represented by

$$y_h = \sum_{i=1}^{L} \theta_{A_i}(x) y_{ih},$$

where $\theta_{A_i}(x) = \dfrac{\prod_{j=1}^{N} A_{ij}(x_j)}{\sum_{i=1}^{L}\prod_{j=1}^{N} A_{ij}(x_j)}$ and $A_{ij}(x_j) \in [0,1]$ is the membership function value of the linguistic variable x_j on the fuzzy set A_{ij}. The global output y_h implies the confidence measure of each class. The predicted class is computed as

$$y = \max_{h \in I_L} y_h.$$

2.2. Information granules

Information granules are an efficient tool which can describe the input-output pattern from the given data. Fig. 1 shows an example of information granule in 2 dimensional case, where '$*$' denotes given data and the rectangle means the information granule or the hyperbox. Most of the existing clustering techniques operate on the numeric objects and produce the representatives that are again entirely numeric, whereas the information granules technique operates on the visual objects and produces the representatives that are again visual [16].

Remark 2: To determine the optimal fuzzy region in the supervised learning, the information granules should satisfy the following requirements as much as possible:

(i) The information granules should cover as many data as possible so that the density of the data within the information granules is made as high as possible [16].

(ii) There should be the minimum overlaps among the information granules because each

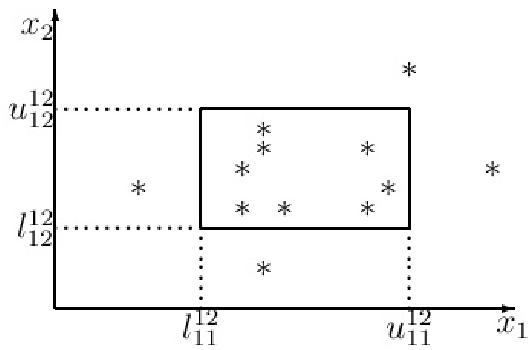


Fig. 1. Example of information granules in 2 dimensional case. *: the given data; the rectangle: the information granule.

information granule should describe different characteristics in the given data.

(iii) One information granule should cover one class within the admissible limits so that the fuzzy sets extracted from an information granule describe a class on several feature spaces.

3. CLASSIFICATION USING INFORMATION GRANULES

This section proposes the design procedure of a fuzzy classifier: the GA-based development of the information granules, the construction method of fuzzy sets by analyzing the information granules, and the GA-based management scheme for the misclassification.

3.1. Optimization of information granules using the GA

This section discusses the parameters optimization of information granules using the GA. In order to obtain the fuzzy classifier with low complexity, we concentrate on determining the information granules subjected to the number of the class regions without generating the additional information granules.

Obviously according to Remark 2, the information granules should be developed such that the following objectives are fulfilled:

$$\begin{aligned}
&\underset{1\le m<n\le N}{Maximize}\ J_1^{mn} = \frac{\Lambda^{mn}}{D},\\
&\underset{1\le m<n\le N}{Minimize}\ J_2^{mn} = \frac{\sum_{i=1}^{L}\Omega_i^{mn}}{\Lambda^{mn}},\\
&\underset{1\le m<n\le N}{Maximize}\ J_3^{mn} = \prod_{i=1}^{L}\frac{\Psi_i^{mn}}{\Omega_i^{mn}},\\
&\underset{1\le m<n\le N}{Maximize}\ J_4^{mn} = \sum_{i=1}^{L}\frac{\Omega_i^{mn}}{area(l_{ij}^{mn},u_{ij}^{mn})},
\end{aligned} \tag{2}$$

where the superscript '*mn*' denotes a 2 dimensional feature space with x_m and x_n. Throughout this paper, it fulfills $1 \le m < n \le N$. J_1^{mn} is the covering rate; J_2^{mn} is the overlapping rate; J_3^{mn} is the classification performance; J_4^{mn} is the density rate, and D is the number of the given data; Λ^{mn} is the number of the data covered by all information granules in the mn th feature space; Ω_i^{mn} is the number of the data covered by the i th information granule in the mn th feature space; Ψ_i^{mn} is the number of the data labelled as the i th class among the data covered by the i th information granule; and $\mathrm{area}(l_{ij}^{mn},u_{ij}^{mn})$ is the area of the i th information

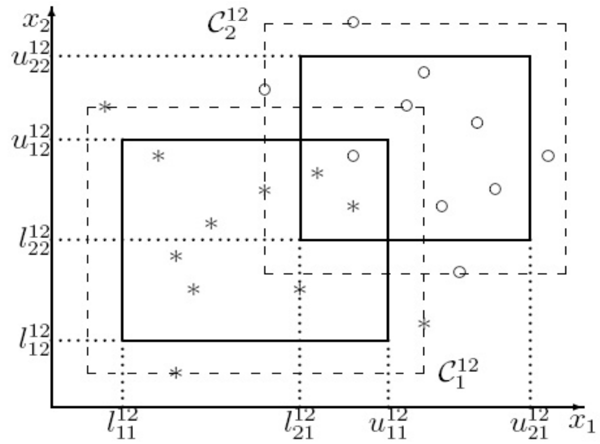


Fig. 2. Example of two information granules (solid boxes) corresponding to two class regions C_1^{12}, C_2^{12} (dashed boxes) with $D = 21$, $\Lambda^{12} = 14$, $\Omega_1^{12} = 9$, $\Omega_2^{12} = 8$, $\Psi_1^{12} = 8$, and $\Psi_1^{12} = 6$. *: the data included in class 1, o: the data included in class 2.

granule to be searched, where l_{ij}^{mn} and u_{ij}^{mn} are the left-lower and the right-upper vertex of the i th hyperbox on x_j -axis in the mn th feature space, respectively. Fig. 2 shows a simple example of two information granules (solid boxes) corresponding to two class region C_1^{12}, C_2^{12} (dashed boxes) with D = 21, Λ^{12} = 14, Ω_1^{12} = 9, Ω_2^{12} = 8, Ψ_1^{12} = 8, and Ψ_1^{12} = 6. These multiple objective functions (2) can be combined into one overall objective function J^{mn}:

$$\underset{1\le m<n\le N}{Minimize}\ J^{mn} = \frac{J_2^{mn}}{J_1^{mn}J_3^{mn}} + \frac{\nu_1}{J_4^{mn}}, \tag{3}$$

where ν_1 is a positive scalar to adjust the weight between the two terms in (3).

The GA represents the searching variables of the given optimization problem (3) as the chromosome that contains one or more sub-strings. In this case, the searching variables are l_{ij}^{mn} and u_{ij}^{mn}. A convenient way to convey the searching variables into a chromosome is to gather all searching variables associated with all information granules in the mn th feature space, and to concatenate the strings as follows:

$$\begin{aligned}
&G^{mn} = \left\{\left(l_1^{mn},u_1^{mn}\right),\ldots,\left(l_N^{mn},u_N^{mn}\right)\right\},\\
&G = \{G^{12},G^{13},G^{23},\ldots,G^{(n-2)(n-1)},G^{1n},\\
&\quad \ldots,G^{(n-1)n},G^{1(n+1)},\ldots,G^{(N-1)N}\},
\end{aligned} \tag{4}$$

where G^{mm} is the parameter substring of the *mn* th feature space in a chromosome and G denotes a chromosome.

Initial population is made up with initial individuals to the extent of the population size. To efficiently perform the optimization, the search spaces in GA should be reduced as much as possible. In this case, we utilize the following constraints on the search spaces.

$$S(G^{mn}) \subset F^{mn}, \tag{5}$$

$$S(l_{ij}^{mn}, u_{ij}^{mn}) \subset C_i^{mn}, \tag{6}$$

where $S(G^{mn})$ and $S(l_{ij}^{mn}, u_{ij}^{mn})$ are the search space for the sub-string and the i th information granule, respectively, F^{mm} implies the *mn* th feature space, and C_i^{mn} denotes the i th class region.

Since the GA originally searches the optimal solution so that the fitness function value is maximized, it is necessary to map the objective function (3) to the fitness function formed by

$$f(J^{mn}) = \frac{J_1^{mn} J_3^{mn}}{J_2^{mn}} + v_1 J_4^{mn}. \tag{7}$$

The best chromosome $\widehat{G}_t$ is composed of the best sub-chromosomes $\widehat{G}_t^{mn}$ with the highest fitness values $\widehat{f}_t^{mn}$ at the t th generation as follows:

$$\begin{aligned}\widehat{G}_t = \{&\widehat{G}_t^{12}, \widehat{G}_t^{13}, \widehat{G}_t^{23}, \dots, \widehat{G}_t^{(n-2)(n-1)}, \widehat{G}_t^{1n},\\ &\dots, \widehat{G}_t^{(n-1)n}, \widehat{G}_t^{1(n+1)}, \dots, \widehat{G}_t^{(N-1)N}\},\end{aligned}$$

where $t \in I_T = \{1, 2, \quad, T\}$ is the genetic process generation number. Then, the best overall fitness value in the m th generation can be written as

$$\widehat{f}_t = \sum_{1 \le m < n \le N} \widehat{f}_t^{mn}.$$

Remark 3: It is noted that each sub-chromosome is independent of the others because (5) is evaluated for each feature space and the searching variables are independent of the feature space.

Remark 4: Since the proposed method establishes the best chromosome $\widehat{G}$ as the best and independent sub-chromosomes $\widehat{G}^{mn}$, it has following advantages: The chromosome with low fitness and best sub-chromosomes may be removed by the elitism, which may degrade the optimization performance, whereas, the proposed method can preserve the best sub-chromosome, and thus improve the optimization performance.

Remark 5: In [19], it is shown an evolutionary optimization technique of the information granules with unlabelled samples in the unsupervised learning for clustering. The basic idea is somewhat similar to [19] but the main contents are completely new and different: the information granules are applied to the fuzzy classifier construction and the GA is applied to optimizing the information granules with the labelled samples in the supervised learning.

3.2. Construction of fuzzy sets

This subsection describes how to construct the fuzzy sets A_{ij} in (1) from the information granules developed in the previous subsection. The α -cut operation and the similarity measurement are utilized for extracting, merging, and removing the fuzzy sets.

Let A_{ij}^{mn} be the fuzzy sets extracted from the i th information granule in the *mn* th feature space, and their membership functions are represented by:

$$A_{ij}^{mn}(x_j) = \begin{cases} \dfrac{x_j - a_{ij}^{mn}}{\left|b_{ij}^{mn} - a_{ij}^{mn}\right|} & \text{if } a_{ij}^{mn} \le x_j < b_{ij}^{mn} \\ \dfrac{c_{ij}^{mn} - x_j}{\left|c_{ij}^{mn} - b_{ij}^{mn}\right|} & \text{if } b_{ij}^{mn} \le x_j < c_{ij}^{mn} \\ 0 & \text{otherwise,} \end{cases} \tag{8}$$

where l_{ij}^{mn} and u_{ij}^{mn} denote the left-lower and the right-upper vertices of the i th information granule on the horizontal x_j -axis, and $b_{ij}^{mn} = \frac{l_{ij}^{mn} + u_{ij}^{mn}}{2}$, respectively. The parameters a_{ij}^{mn} and c_{ij}^{mn} in (8) are computed from the α-cut operation $\left(A_{ij}^{mn}\right)_\alpha = \left\{x_j \in U_j \mid A_{ij}^{mn}(x_j) \ge \alpha\right\}$, $U_j = \left[x_{j_{\min}}, x_{j_{\max}}\right]$, $\alpha \in [0,1]$ of a fuzzy set A_{ij}^{mn} as follows:

$$a_{ij}^{mn} = b_{ij}^{mn} - \frac{b_{ij}^{mn} - l_{ij}^{mn}}{1-\alpha},$$

$$c_{ij}^{mn} = b_{ij}^{mn} + \frac{u_{ij}^{mn} - b_{ij}^{mn}}{1-\alpha}.$$

Fig. 3 shows an example of extracting the fuzzy sets with $L = 3$ and $N = 2$. The membership functions $A_{ij}^{mn}(x_j)$ extracted from the *mn* th feature spaces are merged into one membership function $A_{ij}(x_j)$ with the following parameters:

$$a_{ij} = b_{ij} - \frac{b_{ij} - l_{ij}}{1-\alpha},$$

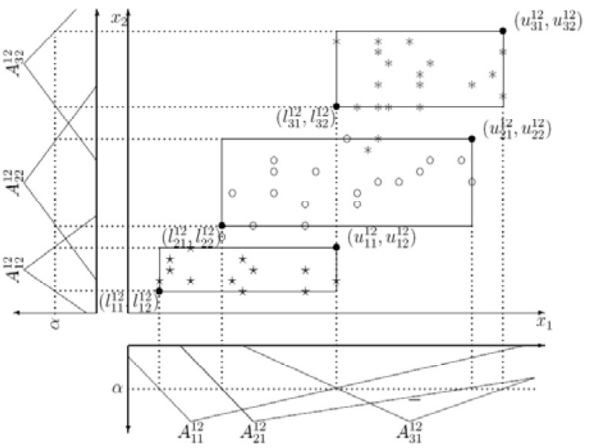


Fig. 3. Example of extracting the fuzzy sets with $L = 3$ and $N = 2$. ★: data included in class 1, o: data included in class 2, and *: data included in class 3.

$$c_{ij} = b_{ij} + \frac{u_{ij} - b_{ij}}{1-\alpha}, \tag{9}$$

where $b_{ij} = \frac{1}{n-1}\sum_{1\le m<n\le N} b_{ij}^{mn}$, $l_{ij} = \frac{1}{n-1}\sum_{1\le m<n\le N} l_{ij}^{mn}$, and $u_{ij} = \frac{1}{n-1}\sum_{1\le m<n\le N} u_{ij}^{mn}$.

There are a lot of methods to measure the similarity between two distinct fuzzy sets [17,18]. In this study, the set-theoretic operation-based similarity measurement [14,25]

$$S((\cdot)_\alpha,(\cdot\cdot)_\alpha) = \frac{|(\cdot)_\alpha \cap (\cdot\cdot)_\alpha|}{|(\cdot)_\alpha \cup (\cdot\cdot)_\alpha|}$$

is applied to removing the similar fuzzy sets, where $|\cdot|$ denotes the cardinality of a set.

Therefore, if the fuzzy set $A_{\tilde{i}j}$ satisfies the following condition inequality, it is removed.

$$\sum_{i=1, i\neq \tilde{i}}^{N} S\left(\left(A_{ij}\right)_\alpha, \left(A_{\tilde{i}j}\right)_\alpha\right) \ge \beta, \tag{10}$$

where $\beta \in [0,1]$ denotes a real constant value.

Remark 6: The key point in this method is that the j th feature is ignored in the fuzzy classifier if all the fuzzy sets for the j th feature are removed.

3.3. GA-based management of misclassification

It is necessary to tune the fuzzy classifiers constructed in the previous subsection for the classification performance improvement on the data that lie into the overlaps of the classes, and the strange pattern unlike the well classified data such as outliers. This subsection proposes a GA-based management technique for misclassification, which consists of the tuning process of the obtained fuzzy rules and the generation process of additional fuzzy rules.

To decrease risk due to overfitting, the following constraints are imported:

$$A_{ij}(x_j) = \begin{cases} A_{ij}(x_j; \Delta a_{ij}, b_{ij}, \Delta c_{ij}) & \text{if overalps exist in } C^i \\ A_{ij}(x_j; a_{ij}, b_{ij}, c_{ij}) & \text{otherwise} \end{cases}$$
$$y_{ih} = \begin{cases} \Delta y_{ih} & \text{if overalps exist in } C^i \\ y_{ih} & \text{otherwise} \end{cases} \tag{11}$$
$$L' \in \left[0, \bar{L}\right],$$

where $\bar{L} \in Z^+$ and L' means the number of fuzzy rules to be newly generated, C^i denotes the i th class region, and $(\cdot)$ symbolizes the tuning of $(\cdot)$ to be determined from the GA-based technique that will be discussed in this subsection. Considering (8) and (11), and tuning its cutting level α_{ij} for A_{ij} produce

$$A_{ij}(x_j) = \begin{cases} \frac{x_j - \Delta a_{ij}}{|b_{ij} - \Delta a_{ij}|} & \text{if } \Delta a_{ij} \le x_j < b_{ij} \\ \frac{\Delta c_{ij} - x_j}{|\Delta c_{ij} - b_{ij}|} & \text{if } b_{ij} \le x_j < \Delta c_{ij} \\ 0 & \text{otherwise,} \end{cases} \tag{12}$$

where $\Delta a_{ij} = b_{ij} - \frac{b_{ij} - l_{ij}}{1-\Delta\alpha_{ij}}$, $\Delta c_{ij} = b_{ij} + \frac{u_{ij} - b_{ij}}{1-\Delta\alpha_{ij}}$. Fig. 4 illustrates an example of the tuned fuzzy set ΔA_{ij}.

Now, we can design the final fuzzy system by adding the following supplementary fuzzy rules to the fuzzy classifier (1).

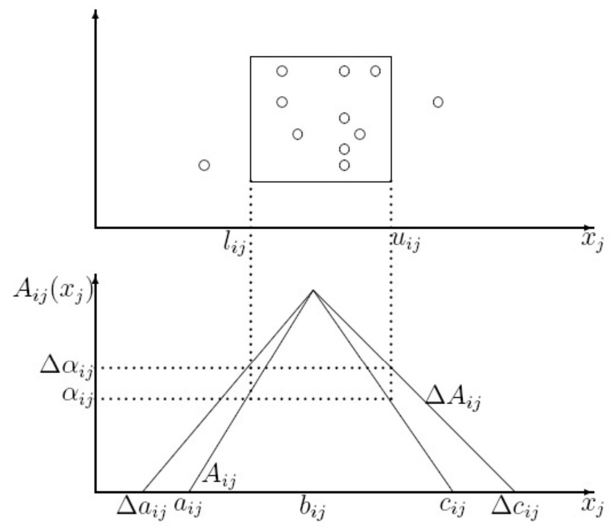


Fig. 4. The fuzzy set ΔA_{ij} tuned by changing $\Delta\alpha_{ij}$.

$R_{i'}$: IF x_1 is $A_{i'1}$ and … and x_N is $A_{i'N}$
THEN the class is misclassification:

$$(y_{i'1} = \xi_{i'1}, \ldots, y_{i'L} = \xi_{i'L}), \tag{13}$$

where $\xi_{i'h} \in [-1,1]$. By using product inference engine, singleton fuzzifier, and center average defuzzifier, the hth global output of the final fuzzy classifier (1) and (13) is represented as

$$y_h = \underbrace{\sum_{i=1}^{L} \theta_{A_i}(x) y_{ih}}_{(1)} + \underbrace{\sum_{i'=1}^{L'} \theta_{A_{i'}}(x) y_{i'h}}_{(13)}, \tag{14}$$

where $\theta_{A_i}(x) = \dfrac{\prod_{j=1}^{N} A_{ij}(x_j)}{\sum_{i=1}^{L}\prod_{j=1}^{N} A_{ij}(x_j) + \sum_{i'=1}^{L'}\prod_{j=1}^{N} A_{i'j}(x_j)}$,

$$\theta_{A_{i'}}(x) = \frac{\prod_{j=1}^{N} A_{i'j}(x_j)}{\sum_{i=1}^{L}\prod_{j=1}^{N} A_{ij}(x_j) + \sum_{i'=1}^{L'}\prod_{j=1}^{N} A_{i'j}(x_j)}.$$

The predicted class $\hat{y}$ is decided by the maximum of (14).

Naturally, the objective of the resulting fuzzy classifier (14) is that the classification performance should be as high as possible. Furthermore, it is desired to reduce the number of the additional fuzzy rules from the viewpoints of hardware implementation and computation resource. The following objective functions should be maximized and minimized, respectively:

$$\textit{Maximize} \quad J_5 = \frac{1}{D}\sum_{d=1}^{D} \delta(\hat{y}(x_d) - (\text{class of } x_d)),$$

$$\textit{Minimize} \quad J_6 = L',$$

where $\delta(\cdot)$ is the Dirac delta function. The fitness function to be maximized via the GA is then

$$f(J_5, J_6) = J_5 + \nu_2 \frac{1}{J_6 + \lambda},$$

where λ is the coefficient to prevent the large fitness function value. The GA is again applied to optimally tune (α_{ij} , ξ_{ih}) and the parameters introduced in (13). At the same time, to identify the number of the supplementary fuzzy rules, h is also tuned. The chromosome $\widetilde{G}$ is organized as

$$\begin{aligned}\widetilde{G} = \{&\alpha_{11}, \ldots, \alpha_{1N}, \ldots, \alpha_{L1}, \ldots, \alpha_{LN}, \xi_{11}, \ldots, \xi_{1L}, \ldots,\\ &\xi_{(L-1)1}, \ldots, \xi_{LL}, a_{11}, b_{11}, c_{11}, \ldots, a_{1N}, b_{1N}, c_{1N}, \ldots,\\ &a_{L'1}, b_{L'1}, c_{L'1}, \ldots, a_{L'N}, b_{L'N}, c_{L'N}, \xi_{11}, \ldots, \xi_{1L}, \ldots,\\ &\xi_{(L'-1)1}, \ldots, \xi_{L'L}, L'\}.\end{aligned} \tag{15}$$

3.4. Construction procedure of fuzzy classifier

The fuzzy classifier algorithm proposed in the previous subsections is summarized as follows:

Step 1: Set the number of the training data D from the given data: N features and L classes.

Step 2: Choose the GA parameters: the generation number T, the weight factors ν_1, and the population size P_s, crossover rate P_c, and mutation rate P_m. Encode the chromosome (4). Generate the initial population G_0 in a random manner such that all searching variables exist within the search spaces (6). Execute the GA process to develop the optimal information granules.

Step 3: Decode the chromosome of each individual and determine the fuzzy classifiers (1). Evaluate them by (3) and give a fitness value to each sub-chromosome G^{mn}.

Step 4: Construct a new individual by concatenating each best sub-chromosome. Evolve a new population. Increase the generation number t by one.

Step 5: Repeat from Step 2 to 4 until one of the followings is satisfied:

A. the satisfactory population shows up.
B. the generation number reaches T.
C. the fitness function value is not increased for the predetermined number of generations.

Step 6: Make the antecedent fuzzy sets using the information granules developed in Step 5 through the following process: extract fuzzy sets A_{ij}^{mn} with the triangular membership function (8). Merge A_{ij}^{mn} obtained from several feature spaces into A_{ij} by (14). Choose the fuzzy sets to be removed by (15) and remove them.

Step 7: Design the fuzzy classifier (1) with A_{ij}.

Step 8: Evaluate the classification performance of (1) via the training data.

Step 9: If the designed fuzzy classifier does not classify the training data satisfactorily, apply the GA-based management scheme for misclassification discussed in Section 3.3. To execute genetic process, determine the generation number T', the population size P_s', crossover rate P_c', and mutation rate P_m', the maximum number of the fuzzy rules L', and the weight factors ν_2, λ. Encode a chromosome (15) as well.

Step 10: Execute the genetic process. Establish the additional fuzzy rules (13) and the final fuzzy classifier (14).

Step 11: Compute the classification performance of (14) for the given data.

Remark 7: Compared with the previous fuzzy classification methods, the proposed method has the following advantages:

A. In the previous research [12], many hyper boxes should be recursively generated to represent the class regions, whereas only the information granules as many the number of classes are needed to represent them in our proposed method. Therefore, our approach is more effective than that of [12] in the sense of the class region representation.

B. The previous research has the the fuzzy set merging process with a large degree of overlapping [14], while the proposed method, in addition, has the removal process of fuzzy sets. Compared with [14], the proposed method has two advantages: the number of the fuzzy sets can be decreased, and a fuzzy set covering several class regions does not occur.
C. In the previous method [7], all the fuzzy sets are tuned, while only the fuzzy sets selected by (11) are tuned in the proposed method, hence can effectively decrease the overfitting risk.

4. AN EXAMPLE: IRIS DATA

The Iris data [20] is a common benchmark in the classification and the pattern recognition studies [21-24,26]. It has four continuous features: x_1-sepal length, x_2-sepal width, x_3-petal length, and x_4-petal width and consists of 150 instances: 50 for each class (Iris sestosa, Iris vericolor, and Iris virginica). Table 1 shows the initial parameters for the classification. To examine the effectiveness of the proposed method, we provide two simulations for Iris data-Simulation 1: all 150 instances are selected as training data, and Simulation 2: one half of 150 instances are randomly selected as the training data and the other half are used as the test data.

Step 1-Step 5 (GA-based development of information granules)**:** The GA-based method is applied to search the optimal information granules in all $6 = {}_4C_2$ feature spaces. We consider real-coded GA [27]. The selection function is used to create evolutionary pressure, i.e., well-performing chromosomes have a higher chance to survive. The roulette wheel selection method [27] is used to select chromosomes for operation. Two classical operators, simple arithmetic crossover and uniform mutation are used in the GA. These operators have been successfully applied in [27]. Fig. 5 shows the information granules determined via the proposed scheme. Using the fitness function (7), the best chromosome evaluates to $f(J^{12}) = 0.6721$, $f(J^{13}) = 1.0078$, $f(J^{14}) = 1.1656$, $f(J^{23}) = 1.0133$, $f(J^{24}) = 1.24052$, and $f(J^{24}) = 1.2999$.

Step 6-Step 8 (Fuzzy sets construction)**:** From the parameters obtained by the information granules, the fuzzy sets are constructed by the proposed procedures: extracting, merging, and removing the fuzzy sets. Because all fuzzy sets for sepal width are removed, three features (i.e., sepal length, petal length, and petal width) are selected for the pattern classification. The

Table 1. Initial parameters for classification.

Procedure	Symbol	Value
Step 1 - Step 5	P_s	100
	P_c	0.9
	P_m	0.3
	T	10000
	ν_1	0.002
Step 6 - Step 8	α	0.5
	β	0.4
Step 9 - Step 11	P_s'	50
	P_c'	0.9
	P_m'	0.2
	T'	500
	L'	2
	ν_2	0.001
	λ	0.1

Table 2. Fuzzy set parameters.

Fuzzy sets	Fuzzy classifier (16)	Tuned fuzzy classifier (17)
Short	(3.60, 5.00, 6.40)	(3.60, 5.00, 6.40)
Long	(4.85, 6.88, 8.92)	Δ (4.39, 6.88, 8.13)
Very short	∅	Δ (4.62, 4.66, 6.89)
Short	(0.55, 1.45, 2.35)	(0.5500, 1.4500, 2.3500)
Middle	(2.12, 3.85, 5.65)	Δ (2.53, 3.85, 5.23)
Long	(3.95, 5.92, 7.88)	Δ (2.08, 5.92, 9.75)
A little long	∅	(1.90, 4.57, 6.47)
Narrow	(−0.15, 0.35, 0.85)	(−0.15, 0.35, 0.85)
Middle	(0.68, 1.32, 1.95)	(0.27, 1.32, 2.37)
Wide	(1.45, 2.15, 2.85)	(0.33, 2.15, 3.97)
Nearly wide	∅	(0.48, 2.22, 2.41)

fuzzy set A_{21} for the sepal length is also removed. Based on the constructed fuzzy sets, the obtained fuzzy classifier is interpretable as follows:

R^1: IF x_1 is short and x_3 is short and x_4 is narrow
THEN the class is Iris sestosa ($y_1^1 = 1$, $y_2^1 = 0.5$, $y_3^1 = 0.5$)
R^2: IF x_3 is middle and x_4 is middle
THEN the class is Iris vericolor ($y_1^2 = 0.5$, $y_2^2 = 1$, $y_3^2 = 0.5$)
R^3: IF x_1 is long and x_3 is long and x_4 is wide
THEN the class is Iris virginica ($y_1^3 = 0.5$, $y_2^3 = 0.5$, $y_3^3 = 1$). (16)

The associated parameters of the antecedent fuzzy sets in (16) are given in Table 2.

Step 9-Step 11 (GA-based management of misclassification)**:** The GA-based method for the

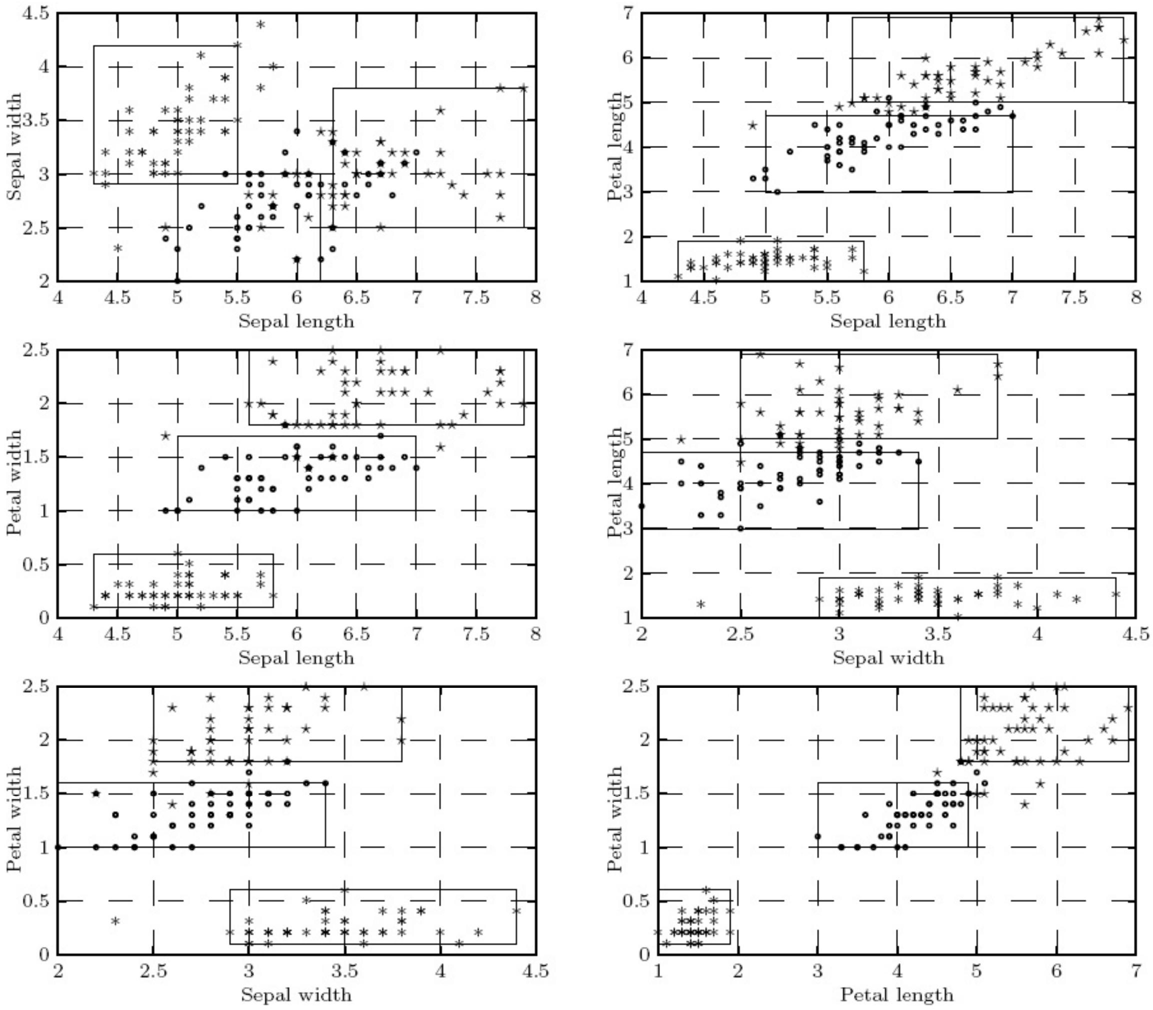


Fig. 5. Information granules for 150 training data. *: Iris sestosa; o: Iris vericolor; and ★: Iris virginica.

misclassification is applied to (16). We consider real-coded GA [27]. The selection function is used to create evolutionary pressure, i.e., well-performing chromosomes have a higher chance to survive. The roulette wheel selection method [27] is used to select chromosomes for operation. Two classical operators, simple arithmetic crossover and uniform mutation are used in the GA. Because there are the overlaps among the class region 2 (Iris vericolor) and the class region 3 (Iris virginica), the antecedent fuzzy sets of R^2 and R^3 and the consequent parameters y_2^2 and y_3^3 are tuned by using the GA method. The best chromosome evaluates to $f(J_5, J_6) = 0.9867$. As a result, the final fuzzy classifier is given as follows:

R^1: IF x_1 is short and x_3 is short and x_4 is narrow
THEN the class is Iris sestosa ($y_1^1 = 1$, $y_2^1 = 0.5$, $y_3^1 = 0.5$)

R^2: IF x_3 is middle and x_4 is middle
THEN the class is Iris vericolor ($y_1^2 = 0.5$, $y_2^2 = 1$, $y_3^2 = 0.5$)

R^3: IF x_1 is long and x_3 is long and x_4 is wide
THEN the class is Iris virginica ($y_1^3 = 0.5$, $y_2^3 = 0.5$, $y_3^3 = 1$)

R^4: IF x_1 is very short and x_3 is a little long and x_4 is nearly wide
THEN the class is Iris virginica ($y_1^4 = 0.2373$, $y_2^4 = 0.6690$, $y_3^4 = 0.7894$), (17)

where R^4 is generated for the management of misclassification. The parameters of the antecedent fuzzy sets are shown in Table 2. The classification performance of the fuzzy classifier (16) is the 96.67% accuracy rate with 5 misclassifications on 150 training data. On the other hand, by using the GA-based management method for the misclassification, the classification performance of the final fuzzy classifier (17) is improved to the classification performance 98.67% with 2 misclassification on 150 training data. In Table 3, the simulation result is compared with the other methods in terms of the rule number and the classification performance. Moreover, we tried the

Table 3. Classification rate of Iris data (150 training data).

Ref.	Rules	Accuracy rate (%)
[26]	4	98%
[21]	9	95.3%
[22]	17	95.3%
[23]	7	96%
Ours	3	96.67%
	4	98.67%

Table 4. Classification results of Iris data (75 test data).

Ref.	Rules	Accuracy rate (%)
[12]	5	92%
[24]	6.21	95.57%
Ours	3.5	97.33%

other simulation for the Iris data: one half of 150 instances are randomly selected as the training data and the other half are used as the test data and vice versa. The average accuracy rate of the final fuzzy classifier is 99.33% on the training data and 97.33% on the test data. Finally we examine the average performance of two simulations: training with one half of the given data and testing with the other half, vise versa. The result is in Table 4, which also dictates our approach is superior to the compared methods.

5. CONCLUSIONS

In this paper, the GA-based method for constructing the fuzzy classifier has been proposed. The advantages of the proposed method are threefold: First, although the number of the information granules equals to the number of the classes, the information granules developed by the GA accomplish the satisfactory fuzzy region. Second, the procedure of constructing the fuzzy sets from the information granules provides an effective tool for the feature selection and the pattern classification. Finally, as we additionally generate the fuzzy rules for the misclassification management, the final fuzzy classifier can describe the misclassified data. Therefore, the proposed method provides the selection of the information granules as well as the solution to the two major problems: the feature selection and the pattern classification. The simulation results have highly visualized that the proposed method has the effectiveness for the classification. It indicates the great potential for reliable application of the pattern recognition.

REFERENCES

[1] Y. H. Joo, H. S. Hwang, K. B. Kim, and K. B. Woo, "Linguistic model identification for fuzzy system," *Electron. Letter*, vol. 31, no. 4, pp. 330-331, 1995.

[2] Y. H. Joo, H. S. Hwang, K. B. Kim, and K. B. Woo, "Fuzzy system modeling by fuzzy partition and GA hybrid schemes," *Fuzzy Sets Syst.*, vol. 86, no. 3, pp. 279-288, 1997.

[3] J. S. Baras and S. Dey, "Combined compression and classification with learning vector quantization," *IEEE Trans. on Information Theroy*, vol. 45, no. 6, pp. 1911-1920, 1999.

[4] G. Tzanetakis and P. Cook, "Musical genre classification of audio signals," *IEEE Trans. on Speech and Audio Processing*, vol. 10, no. 5, pp. 293-302, 2002.

[5] J. Huang, M. Negnevitsky, and D. T. Nguyen, "A neural-fuzzy classifier for recognition of power quality disturbances," *IEEE Trans. on Power Delivery*, vol. 17, no. 2, pp. 609-616, 2002.

[6] M. A. Greenspan, "Geometric probing of dense range data," *IEEE Trans. on Pattern Analysis and Machine Intelligence*, vol. 24, no. 4, pp. 495-508, 2002.

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

[8] L. A. Zadeh, "Fuzzy sets," *Informat. Control*, vol. 8, pp. 338-353, 1965.

[9] L. X. Wang and J. M. Mendel, "Generating fuzzy rules by learning from examples," *IEEE Trans. Syst., Man, Cybern. B.*, vol. 22, no. 6, pp. 1414-1427, 1992.

[10] H. Ishibuchi, K. Nozaki, N. Yamamoto, and H. Tanaka, "Selecting fuzzy if-then rules for classification problems using genetic algoritms," *IEEE Trans. on Fuzzy Systems*, vol. 3, no. 3, pp. 260-270, 1995.

[11] R. Thawonmas and S. Abe, "A novel approach to feature selection based on analysis of class regions," *IEEE Trans. Syst., Man, Cybern. B.*, vol. 27, no. 2, pp. 196-207, 1997.

[12] S. Abe and M. S. Lan, "A method for fuzzy rules extraction directly from numerical data and its application to pattern classification," *IEEE Trans. Fuzzy Systems*, vol. 5, no. 1, pp. 358-368, 1995.

[13] H. M. Lee, C. M. Chen, J. M. Chen, and Y. L. selection based on fuzzy entropy," *IEEE Trans. Syst., Man, Cybern. B.*, vol. 3, no. 3, pp. 426-432, 1997.

[14] T. P. Wu and S. M. Chen, "A new method for constructing membership functions and fuzzy rules from training examples," *IEEE Trans. Syst., Man, Cybern. B.*, vol. 29, no. 1, pp. 25-40, 1999.

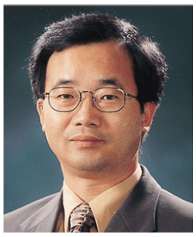
[15] D. E. Goldberg, *Genetic Algorithms in Search, Optimization, and Machine Learning.* Addison-Wesley publishing company, Inc., 1989.

[16] W. Pedrycz and A. Bargiela, "Granular clustering: A granular signature of data," *IEEE Trans. Syst., Man, Cybern. B.*, vol. 32, no. 2, pp.

212-224, 2002.
[17] S. M. Chen, M. S. Yeh, and P. Y. Hsiao, "A comparison of similarity measures of fuzzy values," *Fuzzy Sets Syst.*, vol. 72, no. 1, pp. 79-89, 1995.
[18] V. R. Young, "Fuzzy subsethood," *Fuzzy Sets Syst.*, vol. 77, no. 3, pp. 371-384, 1996.
[19] M. Reformat and W. Pedrycz, "Evolutionary optimization of information granules," *Proc. IFSA World Congress, 20th NAFIPS Int. Conf., Joint 9th*, vol. 4, pp. 2035-2040, 2001.
[20] C. J. Merz and P. M. Murphy, "UCI repository of machine learning databases," http://www. ics. uci.edu/mlearn/MLRepository.html, Irvine, Dept. of Information and Computer Science, Univ. of California, Irvine, 1996.
[21] N. Kasabov, "Learning fuzzy rules and approximate reasoning in fuzzy neural networks and hybrid systems," *Fuzzy Sets Syst.*, vol. 82, no. 2, pp. 135-149, 1996.
[22] N. Kasabov and B. Woodford, "Rule insertion and rule extraction from evolving fuzzy neural networks: Algorithms and applications for building adaptive, intelligent expert systems," *Proc. IEEE Int. Conf. Fuzzy Syst. FUZZ-IEEE'99*, vol. 3, Seoul, Korea, pp. 1406-1411, Aug. 1999.
[23] S. Halgamuge and M. Glesner, "Neural networks in designing fuzzy systems for real world applications," *Fuzzy Sets Syst.*, vol. 65, pp. 1-12, 1994.
[24] T. P. Hong and C. Y. Lee, "Induction of fuzzy rules and membership functions from training examples," *Fuzzy Sets Syst.*, vol. 84, no. 3, pp. 33-47, 1996.
[25] H. Roubos and M. Setnes "Compact transparent fuzzy models and classifiers through iterative complexity reduction," *IEEE Trans. Fuzzy Systems*, vol. 9, no. 4, pp. 516-524, 2001.
[26] Y. Shi, R. Eberhart, and Y. Chen, "Implementation of evolutionary fuzzy systems," *IEEE Trans. on Fuzzy Systems*, vol. 7, no. 2, pp. 109-119, 1999.
[27] Z. Michalewicz, *Genetic Algorithms+Data Structures=Evolution Programs*, 2nd edition, Springer-Verlag, New York, 1994.

Do Wan Kim received the B.S. and M.S. degrees in Electrical and Electronic Engineering from Yonsei University, Seoul, Korea, in 2002 and 2004, respectively. He is currently working toward the Ph.D. degree in the Department of Electrical and Electronic Engineering, Yonsei University, Seoul, Korea. His current research interests include stability analysis in fuzzy control systems and sampled-data systems, and digital redesign.

Ho Jae Lee received the B.S., M.S., and Ph.D. degrees in Electrical and Electronic Engineering from Yonsei University, Seoul, Korea, in 1998, 2000, and 2004, respectively. He was a Lecturer in the Department of Electrical and Electronic Engineering, the Graduate Scholl of Yonsei University, Seoul, Korea in 2004. He was with the University of Houston, Houston, TX, USA as a Visiting Assistant Professor in the Department of Electrical and Computer Engineering. He is currently a Full-time Instructor in Department of Electronic Engineering, Inha University, Incheon, Korea. His current research interests include stability analysis in digital control systems, fuzzy control systems, hybrid dynamical systems, and digital redesign.



Jin Bae Park received the B.E. degree in Electrical Engineering from Yonsei University, Seoul, Korea, and the M.S. and Ph.D. degrees in Electrical Engineering from Kansas State University, Manhattan, in 1977, 1985, and 1990, respectively. Since 1992, he has been with the Department of Electrical and Electronic Engineering, Yonsei University, Seoul, Korea, where he is currently a Professor. His research interests include robust control and filtering, nonlinear control, mobile robot, fuzzy logic control, neural networks, genetic algorithms, and Hadamard transform spectroscopy. He served as the Director for the Transactions of the Korean Institute of Electrical Engineers (KIEE) (2001–2006) and as the Editor-in-Chief for International Journal of Control, Automation, and Systems (IJCAS) (2006).

Young Hoon Joo received the B.S., M.S., and Ph.D. degrees in Electrical Engineering from Yonsei University, Seoul, Korea, in 1982, 1984, and 1995, respectively. He worked with Samsung Electronics Company, Seoul, Korea, from 1986 to 1995, as a Project Manager. He was with the University of Houston, Houston, TX, from 1998 to 1999, as a Visiting Professor in the Department of Electrical and Computer Engineering. He is currently an Associate Professor in the School of Electronic and Information Engineering, Kunsan National University, Korea. His major interest is mainly in the field of intelligent control, fuzzy modeling and control, genetic algorithms, intelligent robot, and nonlinear systems control. He is serving as Vice-President for the Journal of Fuzzy Logic and Intelligent Systems (KFIS) (2005-2006) and Director for the Transactions of the Korean Institute of Electrical Engineers (KIEE) (2005-2006) and for the Institute of Control, Automation and Systems Engineers (ICASE) (2006).