A Novel Approach to Design of Takagi–Sugeno Fuzzy Classifier

Moon Hwan Kim
Department of Electrical
and Electronic Engineering
Yonsei University, Seodaemun-gu, Seoul,
120-749, Korea
email: jmacs@control.yonsei.ac.kr

Young Hoon Joo
School of Electronic
and Information Engineering
Kunsan National University,
Kunsan, Korea
email: yhjoo@kunsan.ac.kr

Jin Bae Park
Department of Electrical
and Electronic Engineering,
Yonsei University, Seodaemun-gu, Seoul,
120-749, Korea
e-mail: jbpark@control.yonsei.ac.kr

***Abstract*— In this paper, we present a new linear matrix inequality (LMI) approach to design of Takagi–Sugeno (T–S) fuzzy classifier. In this paper, we present a novel approach to design of Takagi–Sugeno (T–S) fuzzy classifier. In order to design the T–S fuzzy classifier, we set up two sub design problems: the antecedent and consequent part design problem. The main idea is that the sub design problem is viewed as a LMI optimization problem. Finally, the performance of the T–S fuzzy classifier is evaluated by computer simulations and compared to other fuzzy classifier.**

I. Introduction

We focus on the problems of designing a fuzzy rule-based classifier form observation data. Frequently, in data-driven fuzzy classification approaches, the T–S type fuzzy model is used [1]. The typical identification of the T–S fuzzy classifier is done in two steps. First, the fuzzy rule antecedents are determined; then, consequent parameters are identified by using optimization techniques.

Many methods have been proposed to determine antecedent part and consequent part of the T–S fuzzy classifier: genetic algorithm (GA), neural network (NN), and statistical approaches [2]- [9], [12]. The conventional design method has been successfully applied to T–S fuzzy classifier, however, there are some weak points. GA can be used widely to design the T–S fuzzy classifier, but, GA can guarantee the optimality and unique of obtained T–S fuzzy classifier and spend much time to get solution. In NN case, because NN require a lot of data to get good confidence in the result, NN is not suitable to problems which have small size data set. Differ form GA, statistical approach gives uniquely optimal solution for given problems, however, it need prior probability information.

In order to solve these disadvantages, we use LMI optimization technique. LMI optimization technique has following main advantages: LMI optimization technique has uniquely optimal solution, and is derivative-free optimization method, and can be formulated by using only inequality relationship, and is solved efficiently [18]. These advantages help to get over the disadvantages from conventional methods.

In this paper, we present a new LMI approach to T–S fuzzy classifier design. Similar to typical identification method, two-step procedure is proposed. First, antecedent part of T–S fuzzy classifier is determined. By formulating and solving LMI optimization problem, the center and width of membership function are identified. In the second, consequent part of the T–S fuzzy classifier is determined. The consequent parameters of the T–S fuzzy classifier are also determined by formulating LMI optimization problems and solving it.

The organization of this paper is as follows. After the introduction, the basic definitions and the structure of fuzzy classifier are presented in Section 2. Section 3 presents the basic approach to classifier design by LMI including the classifier problem statement. Simulation results testify to the classifier's performances and the utilities of the proposed method are discussed in Section 4 including two design examples: iris, Wisconsin breast cancer diagnostic (WBCD).

II. Preliminaries

Pattern classification problems are summarized by assigning a class C_i from a predefined class category set $\mathcal{C} = \{C_1, C_2, .., C_n\}$ to an object described as a point in a certain feature space $x \in \mathbb{R}^m$.

The fuzzy classifier implements these proper mappings by using fuzzy sets and IF-THEN rules. One of them, the T–S fuzzy classifier is considered in this paper. The T–S fuzzy classifier has following fuzzy rules [12]:

$$\begin{aligned} R_i \ &: \text{IF } x_1 \text{ is } M_{i1} \text{ AND ... AND } x_m \text{ is } M_{im} \\ &\quad \text{THEN } y_i(x) \ = \ a_{i1}x_1 + \ldots + a_{im}x_m + b_i \end{aligned} \tag{1}$$

Here $x_i \in \mathbb{R}$ is the ith feature input, $M_{i1}, \ldots, M_{im}$ are the antecedent fuzzy sets, $y_i(x)$ is the consequent output of the ith rule, $x = [x_1 \ldots x_m]^T \in F \subset \mathbb{R}^m$ is input feature vector, and F is the feature vector set containing all feature vector, and a_{ij} and b_i are consequent parameters. In this paper, we consider only gaussian membership function as the membership function of the T–S fuzzy classifier.

The final output of the T–S fuzzy classifier is inferred by following equations.

$$Y(x) = \frac{\sum_{i=1}^{l} h_i(x)(\sum_{j=1}^{m} a_{ij}x_j + b_i)}{\sum_{i=1}^{l} h_i(x)} \tag{2}$$

$$h_i(x) = \prod_{j=1}^{m} \mu_{M_{ij}}(x_j) \tag{3}$$

where $h_i(x)$ is the firing strength of the ith rule, $\mu_{M_{ij}}(x_j) \in \mathbb{R}[0,1]$ is the membership degree of the jth feature of the ith rule. For computational convenience, the final output of the T–S fuzzy classifier can be represented as following matrix equation:

$$Y(x) = H^T(Ax + B) \tag{4}$$

where

$$H = \begin{bmatrix} h_1(x) \\ \cdot \\ \cdot \\ h_i(x) \\ \cdot \\ \cdot \\ h_l(x) \end{bmatrix}, A = \begin{bmatrix} a_{11} & \ldots & a_{1m} \\ \cdot & & \cdot \\ \cdot & & \cdot \\ a_{i1} & \ldots & a_{im} \\ \cdot & & \cdot \\ \cdot & & \cdot \\ a_{l1} & \ldots & a_{lm} \end{bmatrix} = \begin{bmatrix} \mathbf{a}_1 \\ \cdot \\ \cdot \\ \mathbf{a}_i \\ \cdot \\ \cdot \\ \mathbf{a}_l \end{bmatrix}, B = \begin{bmatrix} b_1 \\ \cdot \\ \cdot \\ b_i \\ \cdot \\ \cdot \\ b_l \end{bmatrix}. \tag{5}$$

After the final results of the T–S fuzzy classifier is calculated, the class label of x is determined. The classification mapping result should have one of the class label corresponding to x. To determine the class label, the T–S fuzzy classifier is designed to have the final output which is equal to corresponding class label. For example, if x belongs to class C_i, the final output should be equivalent to C_i. For this reason, the class label is determined by computing the errors between $Y(x)$ and C_i. The class label that has the smallest error becomes the final class C,

$$C = \arg_i \min\{|i - Y(x)|\},\ i \in [1, \ldots, n]. \tag{6}$$

III. Fuzzy Classifier Design via Linear Matrix Inequalities

In this section, we shall show how to design the T–S fuzzy classifier by using convex optimization technique. Designing of the T–S fuzzy classifier is achieved by solving two sub design problems: the membership function identification problem and the consequent parameter identification problem. . To solving two sub design problems, it needs to define design problem specifically. Therefore, we formulate the following T–S fuzzy classifier design problems:

Problem 1: When x belongs to class C_i, the antecedent membership functions and the A and B for (4) should satisfy the following design objectives:

1) $h_i(x)$ is maximized, and the $h_{j,j\neq i}(x)$ is minimized.
2) The membership function $h_i(x)$ of the fuzzy classifier should be equivalent to i.

A. Antecedent part design

Let us discuss the design process of the antecedent part in full detail. Consider the first objective of Problem 1. To convert the identification problem to the convex optimization problem, the norm distance between the linear functions is calculated. Then, the LMI optimization problem is formulated by using Schur's complement law. If x belongs to class C_i, h_i should be maximized. So, if antecedent part is design to satisfy the first objective of Problem 1, $h_i(x)$ satisfies following equation,

$$h_i(x) = 1, x \in C_i. \tag{7}$$

Notice that the firing strength can be formulated as following forms,

$$\begin{aligned} h_i(x) &= \mathrm{e}^{-\frac{(c_1^i - x_1)^2}{v_1^i}} \times \mathrm{e}^{-\frac{(c_2^i - x_2)^2}{v_2^i}} \times \ldots \times \mathrm{e}^{-\frac{(c_n^i - x_m)^2}{v_m^i}} \\ &= \mathrm{e}^{-\sum_{j=1}^{m} \frac{(c_j^i - x_i)^2}{v_j^i}} \\ &= \mathrm{e}^{-(x-c_i)^T V_i^T V_i (x-c_i)} \end{aligned} \tag{8}$$

where $V_i = \mathrm{diag}(\frac{1}{\sqrt{v_1^i}}, \ldots, \frac{1}{\sqrt{v_m^i}})$ is the diagonal matrix containing the width of the gaussian membership functions of the ith rule, and $c_i = [c_1^i, \ldots, c_m^i]$ represents vector that has center values of the membership function of the ith rule. c_j^i is the center of gaussian membership function of the jth feature of the ith rule.

Then (7) can be reformulated as following linear equations:

$$(x - c_i)^T V_i^T V_i (x - c_i) = 0, \quad x \in C_i. \tag{9}$$

To design the antecedent part, we should determine c_i and V_i in (9). By using (9), we can formulate the following the LMI optimization problem.

Problem 2: When x labeled as class C_i is given, determine V_i and c_i for (4) such as the following constraint is satisfied:

$$\underset{c_i, V_i}{Minimize} \quad \gamma \quad \text{subject to}$$

$$0 < V_i \tag{10}$$

$$V_i < W\gamma \tag{11}$$

$$\|V_i(x - c_i)\|_2^2 < \gamma, \qquad \forall x \in C_i \tag{12}$$

where W is weight matrix which is defined as follows:

$$W = diag(w_1, \dots, w_m) = \begin{bmatrix} w_1 & 0 & \dots & 0 \\ 0 & w_2 & \dots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \dots & w_m \end{bmatrix}. \quad (13)$$

Remark 1: To consider the influence of variances of features, we use weight matrix W. By adding inequality condition (11), V_i is influenced by variance of feature i.

Theorem 1: If x labeled as C_i exists, V_i and c_i are determined by solving the following general eigenvalue problem (GEVP),

$$\underset{c_i, V_i}{Minimize} \quad \gamma \quad \text{subject to}$$

$$\gamma W > V_i > 0 \quad (14)$$

$$\forall x \in C_i, \quad L_a(x) = \begin{bmatrix} \gamma & \star \\ V_i x - \mathbf{q}_i & \gamma \end{bmatrix} > 0 \quad (15)$$

where $q_i = V_i c_i$ and $\star$ denotes the transposed element matrix for the symmetric position.

Proof: The convex optimization constraint (12) can be rewritten as follows:

$$(V_i(x - c_i))^T (V_i(x - c_i)) \leq \gamma^2.$$

By using the Schur's complement, the convex optimization constraint can be recasted in the LMI form as follows:

$$\begin{bmatrix} \gamma & \star \\ V_i x - V_i c_i & \gamma \end{bmatrix} \geq 0$$

Because $V_i c_i$ is jointly convex form, we introduce new LMI variable $\mathbf{q}_i$ instead of $V_i c_i$. Therefore, the following LMI constraints can be hold,

$$\gamma W > V_i > 0 \quad (16)$$

$$\begin{bmatrix} \gamma & \star \\ V_i x - \mathbf{q}_i & \gamma \end{bmatrix} > 0, \qquad \forall x \in C_i \quad (17)$$

■

Theorem 1 shows that way to identify the membership function of the ith rule. By using Theorem 1, we can construct algorithm for the design of antecedent part. The algorithm is shown in Algorithm 1.

Theorem 1 shows the method for identification of membership function. By using Theorem 1, we can develop the design procedure for antecedent part. The algorithm is shown in Algorithm 1.

B. Consequent part design

Consider the second objective of Problem 1. In the second objective, we suppose that the firing strength matrix H is predetermined. To satisfy the second objective of Problem 1, it is necessary to determine the A and B so that the following matrix equality constraint should be satisfied.

$$Y_d = H^T (Ax + B), \quad \forall x \in F \quad (18)$$

where Y_d is the desired output of the classifier. Equation (18) is formulated by using (6). Therefore, if the input feature x belonged to class C_i, the desired output Y_d becomes i.

However, It is hard to get consequent matrices A and B which satisfy equality condition (18) for all input features. To overcome this difficulty, convex optimization technique is considered. By taking norm distance between Y_d and $Y(x)$, we can formulate following LMI optimization problem:

Problem 3: When x and H are given, determine A and B for (4) so that the following constraint is satisfied:

$$Minimize \quad \gamma \quad \text{subject to}$$

$$\|Y_d - H^T (Ax + B)\|_2^2 < \gamma, \quad \text{for} \quad \forall x \in F. \quad (19)$$

Theorem 2: If x, Y_d, and H exist, A and B of the proposed T–S fuzzy classifier are determined by solving the following GEVP

$$\underset{A, B}{Minimize} \quad \gamma \quad \text{subject to}$$

$$\forall x \in \mathbf{F}, \quad L_c(x) = \begin{bmatrix} \gamma & \star \\ Y_d - H^T (Ax + B) & I \end{bmatrix} > 0. \quad (20)$$

Proof: The constraint (19) can be reformulated to LMI term by using Schur's complement rule as following step:

$$(Y_d - H^T (Ax + B))^T (Y_d - H^T (Ax + B)) < \gamma \quad (21)$$

$$\gamma - \left[Y_d - Y(x)\right]^T \times \left[Y_d - Y(x)\right] > 0 \quad (22)$$

$$\begin{bmatrix} \gamma & \star \\ Y_d - H^T (Ax - B)) & I \end{bmatrix} > 0. \quad (23)$$

■

Theorem 2 shows that way to identify consequent parameters. By using Theorem 2, we can construct algorithm for the design of consequent part. The algorithm is shown in Algorithm 2.

IV. Simulation

In the following subsections, we consider two different classification problems: iris and WBCD data. Table I shows the summary of data sets used in simulation. Two kinds of simulation are performed to evaluate various performances of classifier: The first simulation, testing simulation, performs training with all data set and testing with all data set; the second simulation, training simulation, performs training with random half of whole data sets and testing with another half of whole data set. At the fist simulation, we obtain the design results of T–S fuzzy classifier: membership function and

TABLE I

DATA SETS WITH NUMERICAL ATTRIBUTE VALUES.

Data sets	No. of Samples	No. of Features	No. of Classes
Iris	150	4	3
WBCD	683	9	2

TABLE II

CLASSFICATION RESULTS ON VARIOUS DATA SETS: TRAINING CASE.

Data sets	No. of Rules	Accuracy(%)
Iris	2	98.67
WBCD	2	96.93

consequent parameter and compare to other classifier. Then the comparisons of generalizing capability between the proposed T–S fuzzy classifier and other classifier are shown by using the second simulation.

A. Real World Data

In order to evaluate the performance of the proposed approach, we perform two simulations. Table II summaries the results of the training simulation on various data set. In addition, Table III summaries the results of testing simulation on various data set. From Table II and Table III, we find that there are a little difference between results on testing simulation and training simulation. This means the propose method has good generalization capability. In the following subsection, the detail descriptions for data sets and obtained membership function and consequent parameter are shown.

1) Iris Data: The iris database created by Fisher is a common benchmark in the classification and the pattern recognition studies [17]. It has four feature variables: sepal length, sepal width, petal length, and petal width and consists of 150 feature vectors: 50 for each iris sestosa, iris versicolor, iris virinica. In the first simulation, 150 patterns are used to train T–S fuzzy classifier. Then, same number of sample data are used to evaluate the performance of T–S fuzzy classifier. Training procedure of T–S fuzzy classifier is comprised by two Algorithm stated in Section 3. we can get the identified membership function via Algorithm 1. Figure 1 show the membership functions. By using Algorithm 2, we can determine consequent parameter that described as

TABLE III

CLASSIFICATION RESULTS ON VARIOUS DATA SETS: TESTING CASE.

Data sets	Rules	Worst Acc.(%)	Avrage Acc. (%)	Best Acc. (%)
Iris	2	94.67	97.91	100
WBCD	2	95.02	96.30	97.66

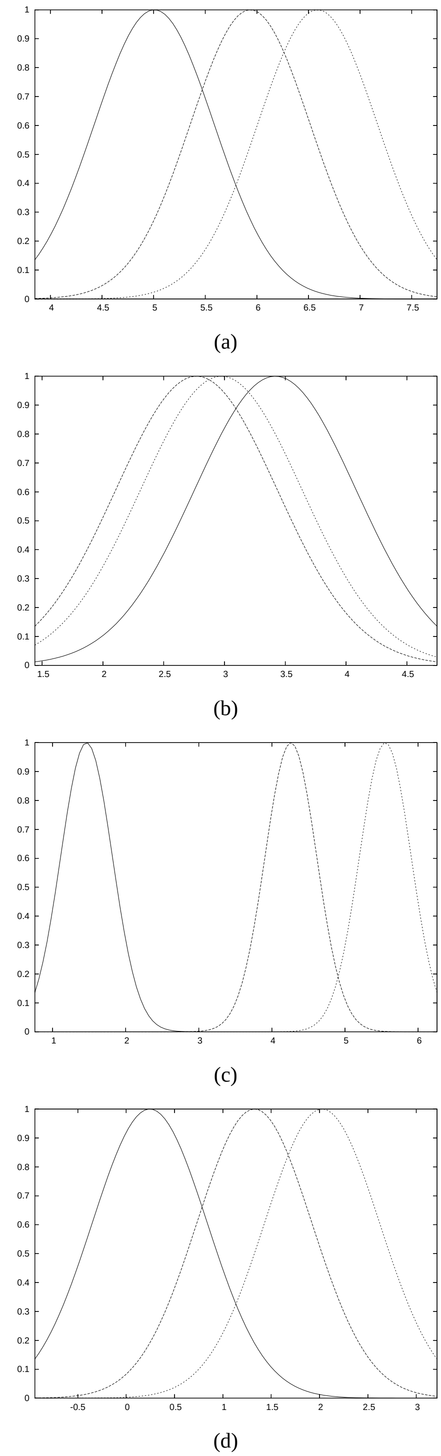

Fig. 1. Membership functions for iris data: training simulation. (a) feature x_1. (b) feature x_2. (c) feature x_3. (d) feature x_4.

TABLE IV

CLASSFICATION RESULTS ON VARIOUS DATA SET: TESTING DATA SET

Data set	Classfiication accuracy (%)	
	Ours	C4.5
Iris	97.91	94.00
WBCD	96.30	93.84

$$A = \begin{bmatrix} -0.0000 & 0.0000 & -0.0001 \\ -0.1121 & -0.2234 & 0.0029 \\ -0.1020 & -0.0624 & 0.1276 \end{bmatrix}, \quad B = \begin{bmatrix} 0.6667 \\ 1.7547 \\ 1.8412 \end{bmatrix}.$$

In the second simulation, we use 76 patterns to train the T–S fuzzy classifier and use 75 patterns to test the generalizing capability.

2) Wisconsin Breast Cancer Diagnostic Data: The WBCD was obtained from the university of Wisconsin Hospitals, Madison from Dr. W.H. Wolberg [14]. This data contains 699 patterns and has two classes; 468 patterns belong to the 'benign' and other 241 patterns belong to the 'malignant'. Since 16 patterns have missing values, we use 683 patterns to evaluate the performance of the proposed classifier.

To evaluate the training performance, 683 patterns are used to identify the T–S fuzzy classifier. In similar to iris data simulation, the proposed two algorithms are used. The identified membership functions are shown in Figure 2. The consequent parameters can be denoted as

$$A = \begin{bmatrix} 0.0155 & 0.0125 \\ 0.0263 & 0.0021 \\ 0.0314 & 0.0030 \\ 0.0109 & 0.0051 \\ 0.0056 & -0.0012 \\ 0.0504 & 0.0052 \\ 0.0163 & 0.0096 \\ 0.0288 & 0.0015 \\ 0.0548 & 0.0028 \end{bmatrix}^T, \quad B = \begin{bmatrix} 0.6857 \\ 1.6990 \end{bmatrix}.$$

In the testing simulation, the proposed classifier is trained by using 242 random data. Another 241 data are used to evaluate the general capability of classifier.

B. Performance Comparisons

To check the superiority of the proposed classifier, we compare the performance of the proposed classifier with other classifier. The C4.5 algorithm is compared with the proposed method for all data sets. The C4.5 algorithm is well-known and frequently used as common bench mark. Table IV gives the results of comparison and shows the superiority of the proposed method for all data set.

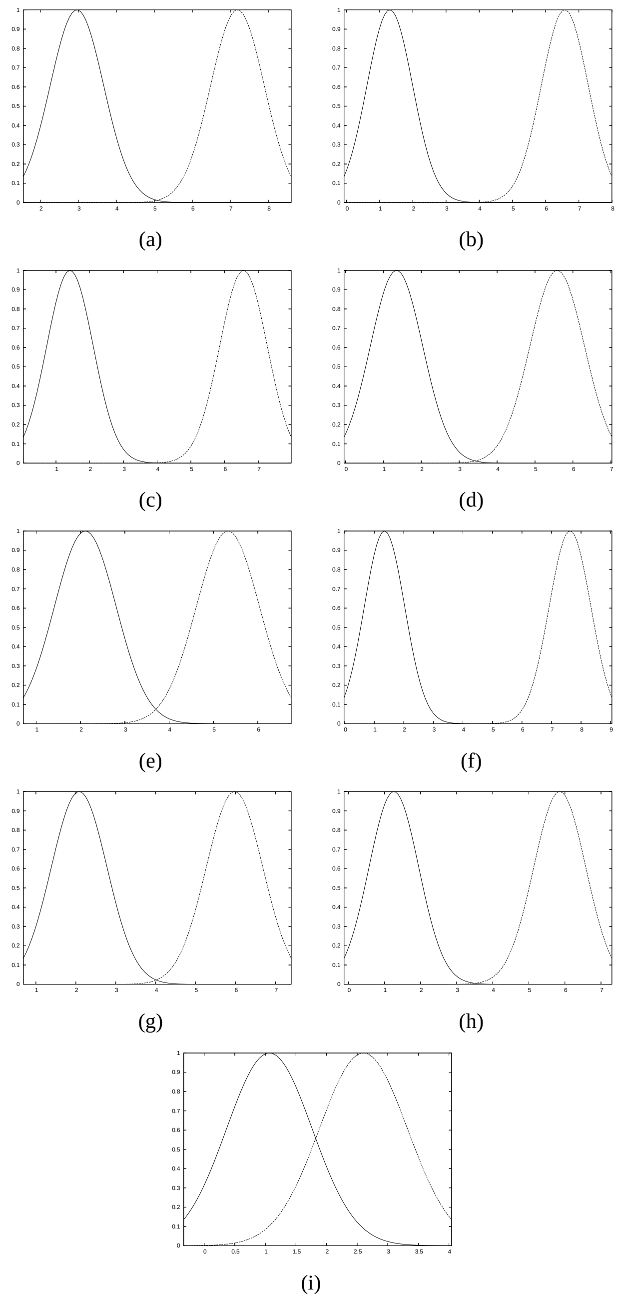

Fig. 2. Membership functions for WBCD data: training simulation. (a)-(i) show the membership function of features from x_1 to x_9.

TABLE V

COMPARISON OF CLASSIFICATION RESULTS ON WBCDD DATA: TESTING DATA SET

Reference	Average of classification accuracy (%)
[14]	95.14
[15]	95.57
[16]	95.60
Ours	96.30

TABLE VI

COMPARISON OF CLASSIFICATION RESULTS ON IRIS DATA: TESTING DATA SET

Reference	Number of rules	Classification accuracy (%)
[10]	17	97.33
[10]	5	92.00
[13]	48	97.33
Ours	3	97.66

In addition, to compare the proposed classifier with another classifier that except C.45 algorithm, WBCD data set is used for comparison. Table V shows the results of the comparison for WBCD data set. From Table V, we can confirm that the proposed classifier has better performance than conventional classifies.

Next, in order to show the superiority of the proposed design method, we compare the proposed classifier with conventional fuzzy classifier. To give the confidence of the performance comparison, we use iris data set which is used in common bench mark. Table IV show the comparison results on iris data set. We can figure out that the proposed method has higher performance than other classifier.

V. CONCLUSIONS

In this paper, the LMI based design method for T–S fuzzy classifier is proposed. In order to apply LMI optimization technique to T–S fuzzy classifier, we relax design problem of T–S fuzzy classifier to LMI optimization problem. The main advantage of the proposed T–S fuzzy classifier is that the T–S fuzzy classifier can be obtained, very reliably and efficiently, using interior-point method or other special methods for LMI optimization. In order to check the performance of the proposed T–S fuzzy classifier, we use four real world data for simulation. From the simulation result, we can confirm that the membership functions and consequent parameter is well identified and the performance is higher than conventional classifier.

REFERENCES

[1] T. Takagi and M. Sugeno, "Fuzzy identification of systems and its application to modeling and control, IEEE Trans. Syst., Man, Cybern., vol. 15, pp. 116-132, 1985.

[2] 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 Syst., vol. 89, pp. 135-150, 1997.

[3] C. -H. Wang, T. -P. Hong, and S. -S. Tseng, "Integrating fuzzy knowledge by genetic algorithms, Fuzzy Sets Syst., vol. 2, no. 4, pp. 138.149, 1998.

[4] H. Ishibuchi, T. Nakashima, and T. Murata, "Performance evaluation of fuzzy classifier systems for multidimensional pattern classification problems, IEEE Trans. Syst., Man, Cybern., pt. B, vol. 29, pp. 601-618, Oct. 1999.

[5] L. O. Hall, I. B. Ozyurt, and J. C. Bezdek, "Clustering with genetically optimized approach, IEEE Trans. Evolut. Computing, vol. 3, pp. 103-112, July 1999.

[6] H. -S. Hwang, "Control strategy for optimal compromise between trip time and energy consumption in a high-speed railway, IEEE Trans. Syst., Man, Cybern., pt. A, vol. 28, pp. 791-802, Nov. 1998.

[7] I. Jagielska, C. Matthews, and T. Whitfort,"An investigation into the application of neural networks, fuzzy logic, genetic algorithms, and rough sets to automated knowledge acquisition for classification problems, Neurocomputing, vol. 24, pp. 37-54, 1999.

[8] M. Russo, "FuGeNeSys.A fuzzy genetic neural system for fuzzy modeling, IEEE Trans. Fuzzy Syst., vol. 6, pp. 373-388, Aug. 1998.

[9] L. Wang and J. Yen, "Extracting fuzzy rules for system modeling using a hybrid of genetic algorithms and Kalman filter, Fuzzy Sets Syst., vol. 101, pp. 353-362, 1999.

[10] 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 Syst.*, vol. 3, pp. 18-28, 1995.

[11] O. Cordon, M. J. del Jesus, and F. Herrera, "A proposal on reasoning methods in fuzzy rule-based classification system," *Int. J. Approx. Reasoning*, vol. 20, pp. 21-45, 1999.

[12] M. Setnes and H. Roubos, "GA-fuzzy modeling and classification: complexity and performance," *IEEE Trans. Fuzzy Syst.*, vol. 8, pp. 509-522, 2000.

[13] V. Uebele, S. Abe, and M. -S. Lan, "A neural-network-based fuzzy classifier,"*IEEE Trans. Syst., Man, Cybern.*, vol. 25, pp. 353-361, 1995.

[14] H.M. Lee, C. M. Chen, J. M. Chen, and Y.L. Jou,"An efficient fuzzy classifier with feature selection based on fuzzy entropy," *IEEE Trans. Syst., Man, Cybern. B*, vol. 31, pp. 426-432, 2001.

[15] H. M. Lee, C. M. Chen, and Y. F. Lu, "A self-organizing HCMAC neural netowkr classifier," *IEEE Trans. Neural Network*, vol. 14, pp. 15-27, 2003.

[16] M. Muselli and D. Liberati, "Binary rule generation via hamming clustering," *IEEE Trans. Knowdlege and Data Engineering*, vol. 14, pp. 1258-1268, 2002.

[17] R. A. Fisher, The use of multiple measurements in taxonomic problem, *Ann. Eugenics*, vol. 7, no. 2, pp. 179.188, 1936.

[18] S. Boyd and L. Vandenberghe, *Convex Optimization*, Cambridge University Press, 2003.