Approximation Capabilities of Interpretable Fuzzy Inference Systems

Hirofumi Miyajima†[1], Noritaka Shigei†[2], and Hiromi Miyajima†[3]

Abstract—**Many studies on modeling of fuzzy inference systems have been made. The issue of these studies is to construct automatically fuzzy inference systems with interpretability and accuracy from learning data based on metaheuristic methods. Since accuracy and interpretability are contradicting issues, there are some disadvantages for self-tuning method by metaheuristic methods. Obvious drawbacks of the method are lack of interpretability and getting stuck in a shallow local minimum. Therefore, the conventional learning methods with multi-objective fuzzy modeling and fuzzy modeling with constrained parameters of the ranges have become popular. However, there are little studies on effective learning methods of fuzzy inference systems dealing with interpretability and accuracy. In this paper, we will propose a fuzzy inference system with interpretability. Firstly, it is proved theoretically that the proposed method is a universal approximator of continuous functions. Further, the capability of the proposed model learned by the steepest descend method is compared with the conventional models using function approximation and pattern classification problems in numerical simulation. Lastly, the proposed model is applied to obstacle avoidance and the capability of interpretability is shown.**

Index Terms—**approximation capability, fuzzy inference, interpretability, universal approximator, obstacle avoidance.**

I. Introduction

FUZZY inference systems are widely used in system modeling for the fields of classification, regression, decision support system and control [1], [2]. Therefore, many studies on modeling of fuzzy inference systems have been made. The issue of these studies is to construct automatically fuzzy systems with interpretability and accuracy from learning data based on metaheuristic methods [3]–[7]. Since accuracy and interpretability are contradicting issues, there are some disadvantages for self-tuning method by metaheuristic methods. Obvious drawbacks of the method are lack of interpretability and getting stuck in a shallow local minimum [8]. As metaheuristic methods, some novel methods have been developed which 1) use GA and PSO to determine the structure of fuzzy systems [7], [9], 2) use fuzzy inference systems composed of small number of input rule modules, such as SIRMs and DIRMs methods [10], [11], 3) use a self-organization or a vector quantization technique to determine the initial assignment and 4) use combined methods of them [4], [7], [9]. Since accuracy and interpretability are conflicting goals, the conventional learning methods with multi-objective fuzzy modeling and fuzzy modeling with constrained parameters of the ranges have become popular [7], [12]. However, there are little studies on effective learning methods of fuzzy inference systems satisfying with interpretability and accuracy. It is desired to develop effective fuzzy systems with interpretability to realize the linguistic processing of information which is the original issue of fuzzy logic. Shi has proposed a fuzzy inference system with interpretability and accuracy [13]. Further, Miyajima has proposed a generalized model of it [14]. However, there are no studies on the detailed capability of this type of model.

In this paper, we will propose a fuzzy inference system with interpretability and accuracy. Firstly, it is proved theoretically that the proposed method is a universal approximator of continuous functions. Further, the capability of the proposed model learned by the steepest descend method is compared with the conventional models using function approximation and pattern classification problems in numerical simulation. Lastly, the proposed model is applied to obstacle avoidance and the capability of interpretability is shown.

Affiliation: Graduate School of Science and Engineering, Kagoshima University, 1-21-40 Korimoto, Kagoshima 890-0065, Japan
corresponding auther to provide email: miya@eee.kagoshima-u-ac.jp
†[1] email: k3768085@kadai.jp
†[2] email: shigei@eee.kagoshima-u-ac.jp
†[3] email: miya@eee.kagoshima-u-ac.jp

II. Preliminaries

A. The conventional fuzzy inference model

The conventional fuzzy inference model is described [1]. Let $Z_j = \{1, \cdots, j\}$ for the positive integer j. Let $\boldsymbol{R}$ be the set of real numbers. Let $\boldsymbol{x} = (x_1, \cdots, x_m)$ and y^r be input and output data, respectively, where $x_j \in \boldsymbol{R}$ for $j \in Z_m$ and $y^r \in \boldsymbol{R}$. Then the rule of fuzzy inference model is expressed as

$$R_i \ : \ \text{if } x_1 \text{ is } M_{i1} \text{ and } \cdots \text{ and } x_m \text{ is } M_{im} \quad \text{then } y \text{ is } f_i(x_1, \cdots, x_m) \tag{1}$$

, where $i \in Z_n$ is a rule number, $j \in Z_m$ is a variable number, M_{ij} is a membership function of the antecedent part, and $f_i(x_1, \cdots, x_m)$ is the function of the consequent part.

A membership value of the antecedent part μ_j for input $\boldsymbol{x}$ is expressed as

$$\mu_i = \prod_{j=1}^{m} M_{ij}(x_j). \tag{2}$$

If Gaussian membership function is used, then M_{ij} is expressed as follow (See Fig.1):

$$M_{ij} = a_{ij} \exp\left(-\frac{1}{2}\left(\frac{x_j - c_{ij}}{b_{ij}}\right)^2\right). \tag{3}$$

, where a_{ij}, c_{ij} and b_{ij} are the amplitude, the center and the width values of M_{ij}, respectively.

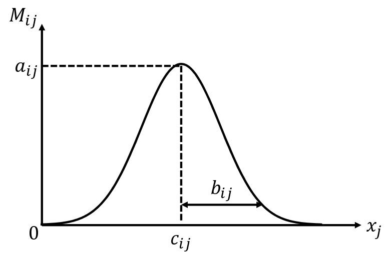


Fig. 1. The Gaussian membership function

The output y^* of fuzzy inference is calculated by the following equation:

$$y^* = \frac{\sum_{i=1}^{n} \mu_i \cdot f_i}{\sum_{i=1}^{n} \mu_i}. \quad (4)$$

Specifically, simplified fuzzy inference model is known as one with $f_i(x_1, \cdots, x_m) = w_i$ for $i \in Z_n$, where $w_i \in \boldsymbol{R}$ is a real number. The simplified fuzzy inference model is called Model 1.

B. Learning algorithm for the conventional model

In order to construct the effective model, the conventional learning method is introduced. The objective function E is defined to evaluate the inference error between the desirable output y^r and the inference output y^*. In this section, we describe the conventional learning algorithm. Let $\boldsymbol{D} = \{(x_1^p, \cdots, x_m^p, y_p^r) | p \in Z_P\}$ be the set of learning data. The objective of learning is to minimize the following mean square error(MSE):

$$E = \frac{1}{P} \sum_{p=1}^{P} (y_p^* - y_p^r)^2. \quad (5)$$

In order to minimize the objective function E, each parameter $\alpha \in \{a_{ij}, c_{ij}, b_{ij}, w_i\}$ is updated based on the descent method as follows [1]:

$$\alpha(t+1) = \alpha(t) - K_\alpha \frac{\partial E}{\partial \alpha} \quad (6)$$

where t is iteration time and K_α is a constant. When Gaussian membership function with $a_{ij} = 1$ for $i \in Z_n$ and $j \in Z_m$ are used, the following relation holds [6].

$$\frac{\partial E}{\partial w_i} = \frac{\mu_i}{\sum_{i=1}^{n} \mu_i} \cdot (y^* - y^r) \quad (7)$$

$$\frac{\partial E}{\partial c_{ij}} = \frac{\mu_j}{\sum_{i=1}^{n} \mu_i} \cdot (y^* - y^r) \cdot (w_i - y^*) \cdot \frac{x_j - c_{ij}}{b_{ij}^2} \quad (8)$$

$$\frac{\partial E}{\partial b_{ij}} = \frac{\mu_i}{\sum_{i=1}^{n} \mu_i} \cdot (y^* - y^r) \cdot (w_i - y^*) \cdot \frac{(x_j - c_{ij})^2}{b_{ij}^3} \quad (9)$$

Then, the conventional learning algorithm is shown as below [1], [2], [6].

Learning Algorithm A

Step A1 : The threshold θ of inference error and the maximum number of learning time T_{max} are given. The initial assignment of fuzzy rules is set to equally intervals. Let n be the number of rules and $n = d^m$ for an integer d. Let $t = 1$.

Step A2 : The parameters b_{ij}, c_{ij} and w_i are set to the initial values.

Step A3 : Let $p = 1$.

Step A4 : A data $(x_1^p, \cdots, x_m^p, y_p^r) \in \boldsymbol{D}$ is given.

Step A5 : From Eqs.(2) and (4), μ_i and y^* are computed.

Step A6 : Parameters c_{ij}, b_{ij} and w_i are updated by Eqs.(8), (9) and (7).

Step A7: If $p = P$ then go to Step A8 and if $p < P$ then go to Step A4 with $p \leftarrow p + 1$.

Step A8: Let $E(t)$ be inference error at step t calculated by Eq.(5). If $E(t) > \theta$ and $t < T_{max}$ then go to Step A3 with $t \leftarrow t+1$ else if $E(t) \leq \theta$ and $t \leq T_{max}$ then the algorithm terminates.

Step A9: If $t > T_{max}$ and $E(t) > \theta$ then go to Step A2 with $n = d^m$ as $d \leftarrow d + 1$ and $t = 1$.

C. The proposed model

It is known that Model 1 is effective, because all the parameters are adjusted by learning. On the other hand, all the parameters move freely, so interpretability capability is low. Therefore, we propose the following model.

$$R^{i_1 \cdots i_m} : \text{if } x_1 \text{ is } M_{i_1 1} \text{ and } \cdots \text{ and } x_m \text{ is } M_{i_m m} \text{ then } y \text{ is } f_{i_1 \cdots i_m}(x_1, \cdots, x_m) \quad (10)$$

, where $1 \leq i_j \leq l_j$, $M_{i_j j}(x_j)$ is the membership function for x_j and $j \in Z_m$.

A membership value of the antecedent part $\mu_{i_1 \cdots i_m}$ for input $\boldsymbol{x}$ is expressed as

$$\mu_{i_1 \cdots i_m} = \prod_{j=1}^{m} M_{i_j j}(x_j) = M_{i_1 1}(x_1) \cdots \cdot M_{i_m m}(x_m) \quad (11)$$

Then, the output y^* is calculated by the following equation.

$$y^* = \frac{\sum_{i_1} \cdots \sum_{i_m} \mu_{i_1 \cdots i_m} f_{i_1 \cdots i_m}(x_1, \cdots, x_m)}{\sum_{i_1} \cdots \sum_{i_m} \mu_{i_1 \cdots i_m}} \quad (12)$$

In this case, the model with $f_{i_1, \cdots, i_m}(x_1 \cdots, x_m) = w_{i_1, \cdots, i_m}$ and triangular membership function with $a_{i_j j} = 1$ has already proposed in the Ref. [13].

We will consider a model with $f_{i_1 \cdots i_m}(x_1, \cdots, x_m) = w_{i_1 \cdots i_m}$ and Gaussian membership functions. The model is called Model 2.

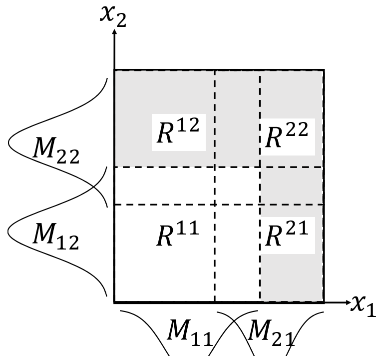
[Example 1]

Let us show an example of the proposed model with $m = 2$ and $1 \leq i_1, i_2 \leq 2$ as follows:

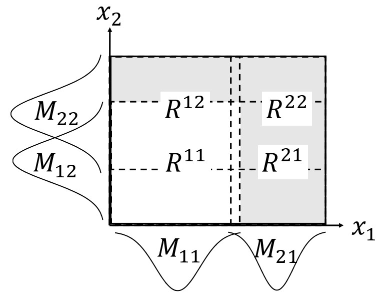
$$R^{11} : if\ x_1\ is\ M_{11}\ and\ x_2\ is\ M_{12}\ then\ y\ is\ w_{11}$$
$$R^{12} : if\ x_1\ is\ M_{11}\ and\ x_2\ is\ M_{12}\ then\ y\ is\ w_{12}$$
$$R^{21} : if\ x_1\ is\ M_{21}\ and\ x_2\ is\ M_{22}\ then\ y\ is\ w_{21}$$
$$R^{22} : if\ x_1\ is\ M_{21}\ and\ x_2\ is\ M_{22}\ then\ y\ is\ w_{22}$$

Fig.2 show figures for Example 1. □

Remark that Model 2 is one that the parameters of membership function for each input are adjusted by learning.

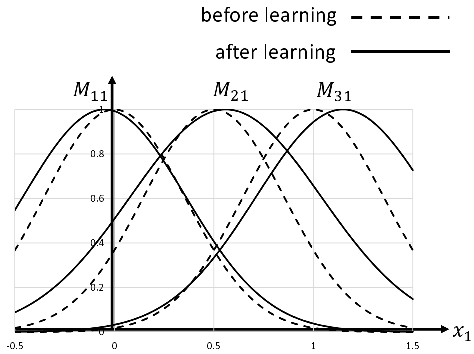


(a)Before learning.

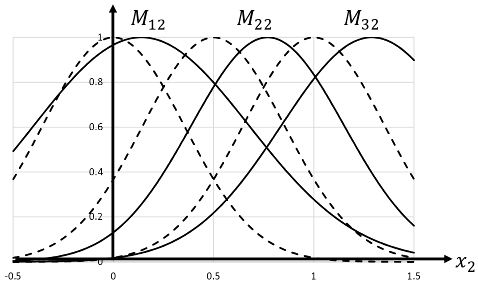


(b)After learning.

Fig. 2. The figure to explain Model 2 with $m = 2$ and $i_1 = i_2 = 2$. The assignment (a) of fuzzy rules for Model 2 is transformed into the assignment (b) after learning.



(a)Membership functions of the variable x_1



(b)Membership functions of the variable x_2

Fig. 3. The assignments of fuzzy rules of before and after learning for Eq.(19).

Learning equation for Model 2 is obtained as follows:

$$\frac{\partial E}{\partial M_{i_j j}} = \sum_{i_1} \cdots \sum_{i_{j-1}} \sum_{i_{j+1}} \cdots \sum_{i_m} \frac{\mu_{i_1 \cdots i_m}}{\sum_{i_1} \cdots \sum_{i_m} \mu_{i_1 \cdots i_m}} \cdot (y^* - y^r) \cdot (y^* - w_{i_1 \cdots i_m}) \tag{13}$$

$$\frac{\partial E}{\partial w_{i_1 \cdots i_m}} = \frac{\mu_{i_1 \cdots i_m}}{\sum_{i_1} \cdots \sum_{i_m} \mu_{i_1 \cdots i_m}} \cdot (y^* - y^r) \tag{14}$$

When Gaussian membership function of Eq.(15) is used, the following equations for $a_{i_j j}$, $c_{i_j j}$ and $b_{i_j j}$ are obtained;

$$M_{i_j j}(x_j) = a_{i_j j} \exp\left(-\frac{1}{2}\sum_{j=1}^{m}\frac{(x_j - c_{i_j j})^2}{b_{i_j j}^2}\right) \tag{15}$$

$$\frac{\partial M_{i_j j}}{\partial a_{i_j j}} = \exp\left(-\frac{1}{2}\sum_{j=1}^{m}\frac{(x_j - c_{i_j j})^2}{b_{i_j j}^2}\right) \tag{16}$$

$$\frac{\partial M_{i_j j}}{\partial c_{i_j j}} = \frac{(x_j - c_{i_j j})}{b_{i_j j}^2} \exp\left(-\frac{1}{2}\sum_{j=1}^{m}\frac{(x_j - c_{i_j j})^2}{b_{i_j j}^2}\right) \tag{17}$$

$$\frac{\partial M_{i_j j}}{\partial b_{i_j j}} = \frac{(x_j - c_{i_j j})^2}{b_{i_j j}^3} \exp\left(-\frac{1}{2}\sum_{j=1}^{m}\frac{(x_j - c_{i_j j})^2}{b_{i_j j}^2}\right) \tag{18}$$

Learning algorithm for the proposed model with $a_{i_j j} = 1$ is also defined using Eqs.(14), (17) and (18). Fig.2 explains the assignments of the antecedent parts of membership functions for before and after learning.

Let us show an example for the practical movement of membership functions.

[Example 2]

Let us consider the following function:

$$y = \frac{(2x_1 + 4x_2^2)(2x_1 + 4x_2^2 + 0.2)}{37.2} \tag{19}$$

, where $0 \leq x_1, x_2 \leq 1$.

In order to approximate Eq.(19), learning of the proposed model with $m = 2$ and $1 \leq i_1, i_2 \leq 3$ is performed, where the number of learning data is 100.

Fig.3 shows the assignments of membership functions after learning, where waving and solid lines mean the assignments of membership functions before and after learning. □

Mamdani type model is special case of Model 2 [7]. It is the model with the fixed parameters of antecedent part of fuzzy rule and membership function assigned to equally intervals. The model is called Model 3. It has good interpretable capability, but the accuracy capability is law. Therefore, TSK model with the weight of linear function $f_i(x_1, \cdots, x_m)$ is introduced as a generalized model of Model 3 [13]. The model is called Model 4.

III. Fuzzy inference system as Universal Approximator

In this section, the universal approximation capabilities of Model 1, 2, 3 and 4 are discussed using the well-known Stone-Weierstrass Theorem. See Ref. [2] about the mathematical terms.

[Stone-Weierstrass Theorem] [2]

Let S be a compact set with m dimensions, and $C(S)$ be

a set of all continuous real-valued functions on S. Let Φ be the set of continuous real-valued functions satisfying the conditions:
(i) Identity function : The constant function $f(\boldsymbol{x})=1$ is in Φ.
(ii) Separability : For any two points $\boldsymbol{x}_1, \boldsymbol{x}_2 \in S$ and $\boldsymbol{x}_1 \neq \boldsymbol{x}_2$, there exists a $f \in \Phi$ such that $f(\boldsymbol{x}_1) \neq f(\boldsymbol{x}_2)$.
(iii) Algebraic closure : For any $f, g \in \Phi$ and $\alpha, \beta \in \boldsymbol{R}$, the functions $f \cdot g$ and $\alpha f + \beta g$ are in Φ.

Then, Φ is dense in $C(S)$. In other words, for any $\varepsilon > 0$ and any function $g \in C(S)$, there is a function $f \in \Phi$ such that

$$|g(\boldsymbol{x}) - f(\boldsymbol{x})| < \varepsilon$$

for all $\boldsymbol{x} \in S$.

It means that the set Φ satisfying the above conditions can approximate any continuous function with any accuracy.
Since the sets of RBF and Model 1 are satisfied with the conditions of Stone-Weierstrass Theorem, they hold for universal approximation capabilities [2], [16], [17]. Further, we can show the result about Model 2 in the following.
[Theorem]

Let Φ be the set of all functions that can be computed by Model 2 on a compact set $S \subset \boldsymbol{R}^m$ as follows:
Let

$$\Phi_{l_1 \cdots l_m} = \left\{ f(\boldsymbol{x}) = \frac{\sum_{i_1} \cdots \sum_{i_m} \prod_j M_{i_j j}(x_j) w_{i_1 \cdots i_m}}{\sum_{i_1} \cdots \sum_{i_m} M_{i_j j}(x_j)} , w_{i_1 \cdots w_m}, a_{i_j j}, c_{i_j j}, b_{i_j j} \in \boldsymbol{R}, \boldsymbol{x} \in S \right\}$$

for $M_{i_j j}(x_j) = a_{i_j j} \exp\left(-\frac{1}{2}\left(\frac{x_j - c_{i_j j}}{b_{i_j j}}\right)^2\right)$ and

$$\Phi = \bigcup_{l_1=1}^{\infty} \cdots \bigcup_{l_m=1}^{\infty} \Phi_{l_1 \cdots l_m} \tag{20}$$

Then Φ is dense in $C(S)$.
[Proof]

In order to prove the result, it is shown that three conditions of Stone-Weierstrass Theorem are satisfied.
(i)The function $f(\boldsymbol{x}) = 1$ for $\boldsymbol{x} \in S$ is in Φ, because it can be considered as a Gaussian function with infinite variable b.
(ii)The set Φ satisfies with the separability since the exponential function is monotonic.
(iii)Let f and g be two functions in Φ and be represented as

$$f(\boldsymbol{x}) = \frac{\sum_{i_1} \cdots \sum_{i_m} \prod_j M^f_{i_j j}(x_j) w^f_{i_1 \cdots i_m}}{\sum_{i_1} \cdots \sum_{i_m} \prod_j M^f_{i_j j}(x_j)} \tag{21}$$

$$g(\boldsymbol{x}) = \frac{\sum_{l_1} \cdots \sum_{l_m} \prod_j M^g_{l_j j}(x_j) w^g_{l_1 \cdots l_m}}{\sum_{l_1} \cdots \sum_{l_m} \prod_j M^g_{l_j j}(x_j)} \tag{22}$$

for

$$M^f_{i_j j}(x_j) = a^f_{i_j j} \exp\left(-\frac{1}{2}\left(\frac{x_j - c^f_{i_j j}}{b^f_{i_j j}}\right)^2\right) \tag{23}$$

$$M^g_{l_j j}(x_j) = a^g_{l_j j} \exp\left(-\frac{1}{2}\left(\frac{x_j - c^g_{l_j j}}{b^g_{l_j j}}\right)^2\right) \tag{24}$$

Then, we show $\alpha f + \beta g \in \Phi$ and $f \cdot g \in \Phi$ for $\alpha, \beta \in \boldsymbol{R}$.

We define

$$\begin{aligned} M^{fg}_{i_j l_j j}(x_j) &= M^f_{i_j j}(x_j) \cdot M^g_{l_j}(x_j) \\ &= a^{fg}_{i_j l_j j} \exp\left(-\frac{1}{2}\left(\frac{x_j - c^{fg}_{i_j l_j j}}{b^{fg}_{i_j l_j j}}\right)^2\right) \end{aligned} \tag{25}$$

$$w^{fg1}_{i_1 \cdots i_m l_1 \cdots l_m} = w^f_{i_1 \cdots i_m} + w^g_{l_1 \cdots l_m} \tag{26}$$

$$w^{fg2}_{i_1 \cdots i_m l_1 \cdots l_m} = w^f_{i_1 \cdots i_m} \cdot w^g_{l_1 \cdot l_m} \tag{27}$$

, where $a^{fg}_{i_j l_j j}, c^{fg}_{i_j l_j j}, b^{fg}_{i_j l_j j}, w^{fg1}_{i_1 \cdots w_{i_m} l_1 \cdots l_m}, w^{fg2}_{i_1 \cdots i_m l_1 \cdots l_m} \in \boldsymbol{R}$. By using these values,

$$\alpha f + \beta g = \frac{\sum_{i_1} \cdots \sum_{i_m} \sum_{l_1} \cdots \sum_{l_m} \prod_j M^{fg}_{i_j l_j j}(x_j) w^{fg1}_{i_1 \cdots i_m l_1 \cdots l_m}}{\sum_{i_1} \cdots \sum_{i_m} \sum_{l_1} \cdots \sum_{l_m} \prod_j M^{fg}_{i_j l_j j}(x_j)} \tag{28}$$

$$f \cdot g = \frac{\sum_{i_1} \cdots \sum_{i_m} \sum_{l_1} \cdots \sum_{l_m} \prod_j M^{fg}_{i_j l_j j}(x_j) w^{fg2}_{i_1 \cdots i_m l_1 \cdots l_m}}{\sum_{i_1} \cdots \sum_{i_m} \sum_{l_1} \cdots \sum_{l_m} \prod_j M^{fg}_{i_j l_j j}(x_j)} \tag{29}$$

Further, Eqs.(28) and (29) have the same form as Model 2. Therefore, $(\alpha f + \beta g)$ and $f \cdot g \in \Phi$ hold.□

[Remarks]
Remark that the results using Stone-Weierstrass Theorem hold only for Model 1 and 2 with $f_i(x_1 \cdots x_m) = w_i$ and Gaussian membership function. On the other hand, Stone-Weierstrass Theorem does not always hold for Model 3, 4 and the models with triangular membership function, because the multiplicative condition fails. Further, it is an existence theorem and there is another problem whether we can get the system with high accuracy. Therefore, we need effective learning algorithm. Learning Algorithm A is a learning algorithm based on the steepest descend method.

IV. Numerical simulations

In this section, three kinds of simulations are performed to compare the capability of the proposed model with the conventional model for learning method based on steepest descend method. In the simulations, let $a_{ij} = 1$ and $a_{i_j j} = 1$ for $i \in Z_n$ and $j \in Z_m$.

A. Function approximation

This simulation uses four systems specified by the following functions with $[0, 1] \times [0, 1]$.

$$y = \sin(\pi x_1^3) \cdot x_2 \tag{30}$$

$$y = \frac{\sin(2\pi x_1^3) \cdot \cos(\pi x_2) + 1}{2} \tag{31}$$

$$y = \frac{1.9(1.35 + e^{x_1} \sin(13(x_1 - 0.6)^2) \cdot e^{-x_2} \sin(7x_2))}{7} \tag{32}$$

$$y = \frac{\sin(10(x_1 - 0.5)^2 + 10(x_2 - 0.5)^2) + 1.0}{2} \tag{33}$$

The condition of the simulation is shown in Table I. The value θ is 1.0×10^{-5} and the numbers of learning and test

TABLE I
INITIAL CONDITION FOR SIMULATION OF FUNCTION APPROXIMATION.

	Model 1	Model 2	Model 3	Model 4
T_{max}	50000	50000	50000	50000
K_c	0.01	0.01	0.0	0.0
K_b	0.01	0.01	0.0	0.0
K_w	0.1			
d	3	7	4	6
Initial c_{ij}	equal intervals			
Initial b_{ij}	$\frac{1}{2(d-1)}\times$(the domain of input)			
Initial w_{ij}	random on $[0,1]$			

TABLE II
RESULTS FOR SIMULATION OF FUNCTION APPROXIMATION.

		Eq.(30)	Eq.(31)	Eq.(32)	Eq.(33)
Model 1	learning	0.10	0.61	0.93	0.22
	test	0.36	1.48	2.65	0.80
	♯parameter	45	45	45	45
Model 2	learning	0.10	0.70	0.11	1.17
	test	0.44	5.97	0.26	3.76
	♯parameter	45	45	45	45
Model 3	learning	1.17	12.37	1.28	4.44
	test	4.22	38.69	4.12	12.94
	♯parameter	49	49	49	49
Model 4	learning	0.24	2.49	0.70	3.01
	test	0.92	10.89	2.04	11.10
	♯parameter	48	48	48	48

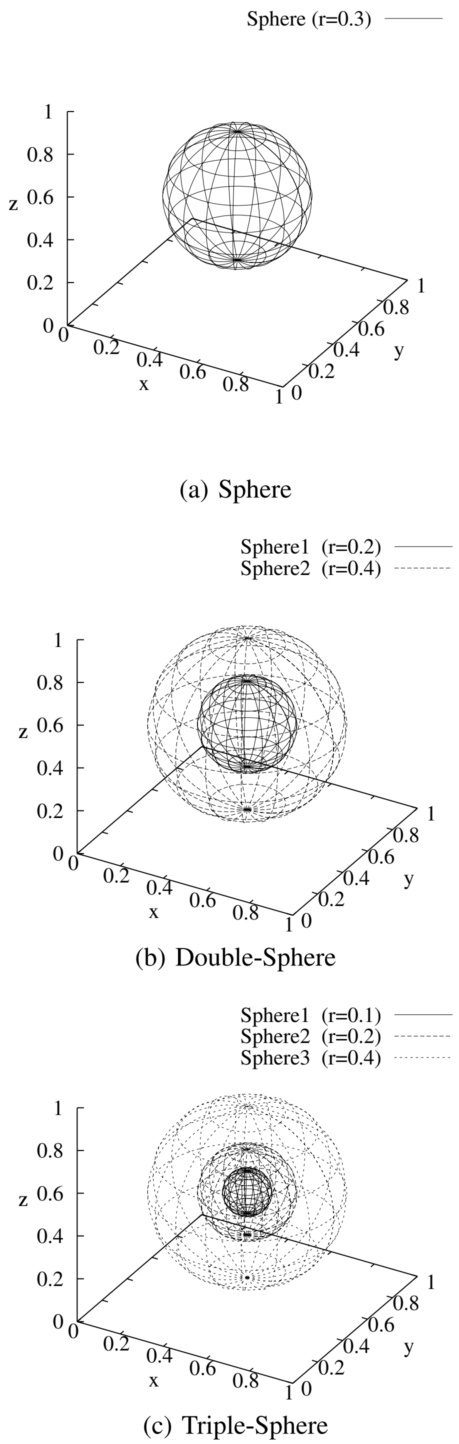
data selected randomly are 200 and 2500, respectively. Table II shows the results on comparison among four models. In Table II, Mean Square Error(MSE) of learning($\times 10^{-4}$) and MSE of test($\times 10^{-4}$) are shown. The result of simulation is the average value from twenty trials. Table II shows that Model 1 and 2 have almost the same capability in this simulation, where ♯parameter means the number of parameters.

B. Pattern classification

In the next, we perform two-category classification problems as in Fig. 4 to investigate another approximation capability of Model 1 and 2. In the classification problems, points on $[0,1]\times[0,1]\times[0,1]$ are classified into two classes: class 0 and class 1. The class boundaries are given as spheres centered at $(0.5,0.5,0.5)$. For Sphere, the inside of sphere is associated with class 1 and the outside with class 0. For Double-Sphere, the area between Spheres 1 and 2 is associated with class 1 and the other area with class 0. For triple-Sphere, the inside of Sphere 1 and the area between Sphere 2 and Sphere 3 is associated with class 1 and the other area with class 0. The desired output y_p^r is set as follows: if $\boldsymbol{x_p}$ belongs to class 0, then $y_p^r = 0.0$. Otherwise $y_p^r = 1.0$. The numbers of learning and test data selected randomly are 512 and 6400, respectively. The simulation condition is shown in Table III. The results on the rate of misclassification are shown in Table IV. The result of simulation is the average from twenty trials. Table IV shows that Model 1 and 2 have almost the same capability.

C. Obstacle avoidance and arriving at designated point

In order to show interpretability, let us perform simulation of control problem for Model 1 and Model 2 [11]. As shown in Fig.5, the distance r_1 and the angle θ_1 between mobile object and obstacle and the distance r_2 and the angle θ_2



(a) Sphere

(b) Double-Sphere

(c) Triple-Sphere

Fig. 4. Two-category Classification Problems [11]

TABLE III
INITIAL CONDITION FOR SIMULATION OF TWO-CATEGORY CLASSIFICATION PROBLEMS.

	Model 1	Model 2
T_{max}	50000	50000
K_w	0.01	0.01
K_c	0.001	0.001
K_b	0.001	0.001
d	3	5
Initial c_{ij}	equal intervals	
Initial b_{ij}	$\frac{1}{2(d-1)}\times$(the domain of input)	
Initial w_{ij}	random on $[0,1]$	

TABLE IV
SIMULATION RESULT FOR TWO-CATEGORY CLASSIFICATION PROBLEM.

		Sphere	Double-Sphere	Triple-Sphere
Model 1	learning	0.007	0.033	0.037
	test	0.027	0.078	0.078
	♯parameter	189	189	189
Model 2	learning	0.009	0.018	0.020
	test	0.036	0.057	0.061
	♯parameter	155	155	155

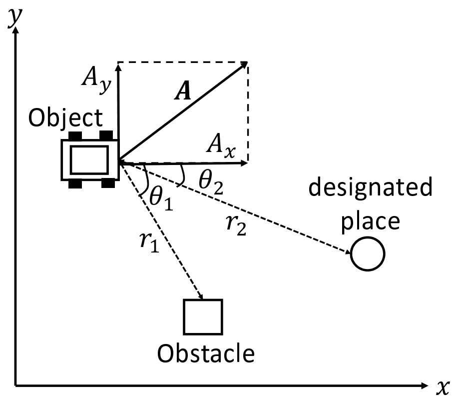


Fig. 5. Simulation on obstacle avoidance and arriving at the designated place.

between mobile object and the designated place are selected as input variables, where θ_1 and θ_2 are normalized.

The problem is to construct fuzzy inference system that mobile object avoids obstacle and arrives at the designated place. Fuzzy inference rules for Model 1 and Model 2 are constructed from learning for data of 400 points shown in Fig. 6. An obstacle is placed at $(0.5, 0.5)$ and a designated place is placed at $(1.0, 0.5)$. The number of rules for each model is 81 and the number of attributes is 3. Then, the numbers of parameters for Model 1 and 2 are 729 and 93, respectively. The mobile object moves with the vector $\boldsymbol{A}$ at each step, where A_x of $\boldsymbol{A}$ is constant and A_y of $\boldsymbol{A}$ is output variable. Learning for two models are successful and the following tests are performed.

(1)Test 1 is simulation for obstacle avoidance and arriving at the designated place when the mobile object stars from various places (See Fig.7). Fig.7 shows the results of moves of mobile object for starting places

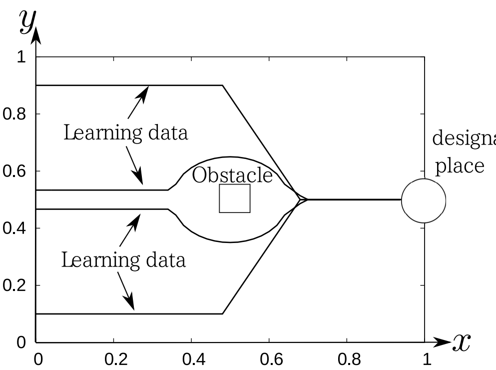
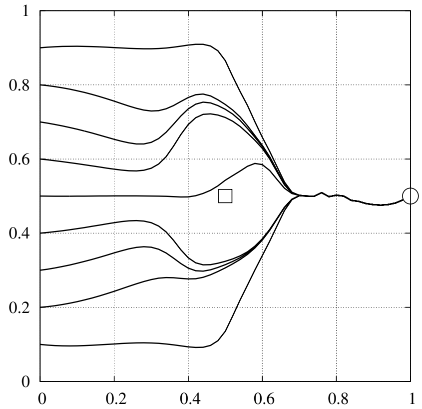


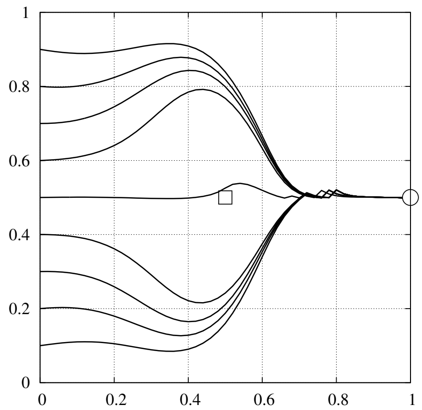
Fig. 6. Learning data to avoid obstacle and arrive at the designated place (1.0, 0.5).

TABLE V
INITIAL CONDITION FOR SIMULATION OF OBSTACLE AVOIDANCE.

	Model 1	Model 2
T_{max}	50000	50000
K_c	0.001	0.001
K_b	0.001	0.001
K_w	0.01	
d	3	3
Initial c_{ij}	equal intervals	
Initial b_{ij}	$\frac{1}{2(d-1)}\times$(the domain of input)	
Initial w_{ij}	0.0	
♯parameters	729	105



(a)Model 1



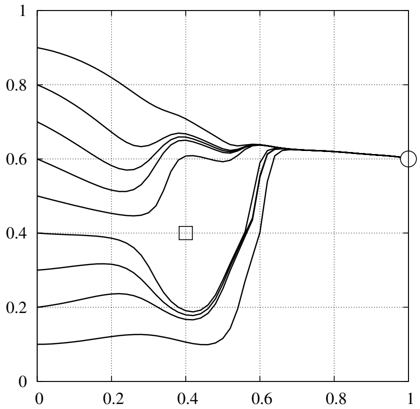
(b)Model 2

Fig. 7. Simulation for obstacle avoidance and arriving at the designated place starting from various places after learning.

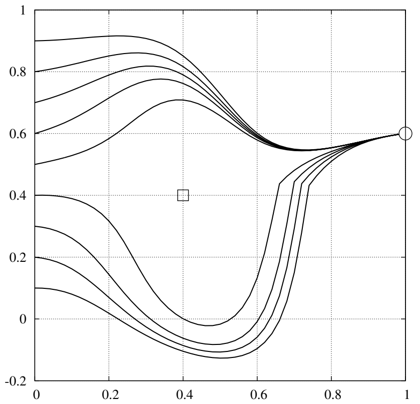
at $(0.1, 0), (0.2, 0), \cdots, (0.8, 0), (0.9, 0)$ after learning. As shown in Fig.7, the test simulations are successful for both models.

(2)Test 2 is simulation for the case where the mobile object avoids obstacle placed at different place and arrives at the different designated place. Simulations with obstacle placed at the place $(0.4, 0.4)$ and arriving at the designated place $(1, 0.6)$ are performed for two models. The results are successful as shown in Fig.8.

(3)Test 3 is simulation for the case where obstacle moves with the fixed speed. Simulations with obstacle moving with the speed $(0.01, 0.02)$ from the place $(0.3, 1.0)$ to the place $(0.8, 0.0)$ and object arriving at the place $(1, 0.6)$ are



(a)Model 1



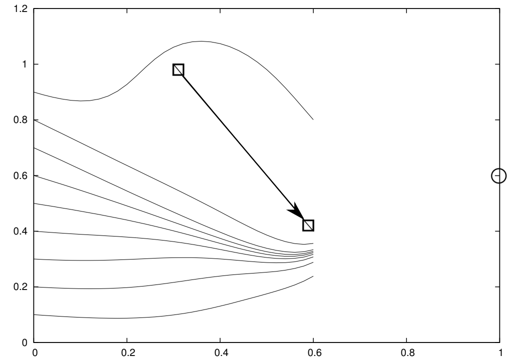
(b)Model 2

Fig. 8. Simulation for obstacle placed at the different place $(0.4, 0.4)$ and arriving at the different place $(1.0, 0.6)$.

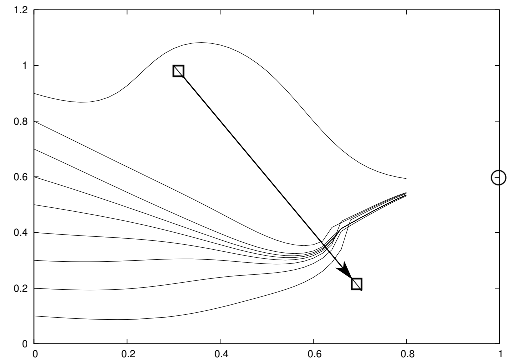
performed. Fig.9 shows the results of the steps 30 and 40 for test simulation. Since the object does not collide in the obstacle at $t = 30$, thereafter it does not collide in the obstacle. The final results for both models are successful as shown in Fig.10.

(4)Test 4 is simulation for the case where obstacle moves with the speed $\boldsymbol{B}$ randomly as shown in Fig.11, where $|\boldsymbol{B}|$ is constant, and the angle θ_b is determined randomly at each step. Simulations with obstacle starting from the place $(0.5, 1.0)$ are performed for two models. The results for both models are successful as shown in Fig.12.

Lastly, let us consider interpretability for the proposed model. Let us consider fuzzy rules constructed for Model 2 by learning. Assume that three attributes are short, middle and long for r_1 and r_2, minus, central and plus for θ_1 and θ_2 and left, center and right for the direction of A_y, respectively. Then, main fuzzy rules for Model 2 are constructed as shown in TableVI. From TableVI, we can get the rules : "If the object approaches to the obstacle, move in the direction away from the object." and "If the object approach to the goal, move toward to the goal." They are similar to human activity to solve the problem. On the other hand, fuzzy rules for Model 1 are not so clear.

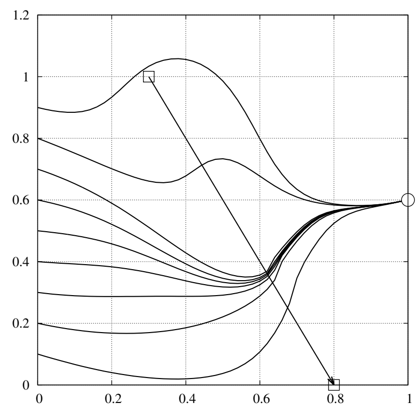


(i)Model 2(step $t = 30$)

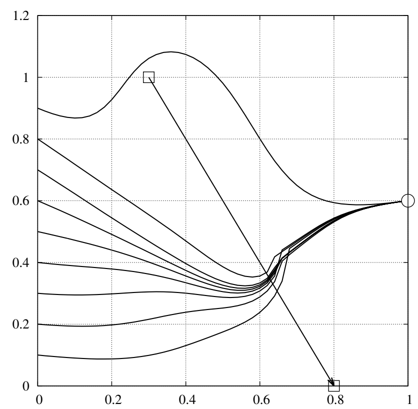


(ii)Model 2(step $t = 40$)

Fig. 9. Simulation for moving obstacle avoidance with fixed speed and the different designated place $(1.0, 0.6)$.



(a)Model 1



(b)Model 2

Fig. 10. Simulation for moving obstacle avoidance with fixed speed and the different designated place $(1.0, 0.6)$.

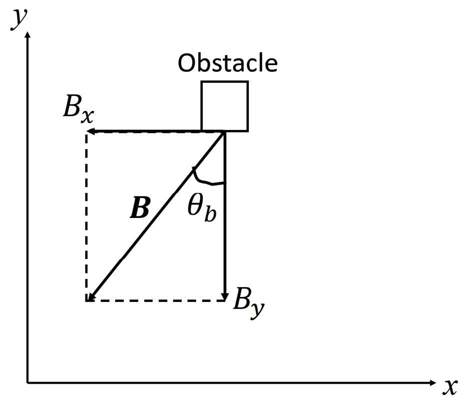
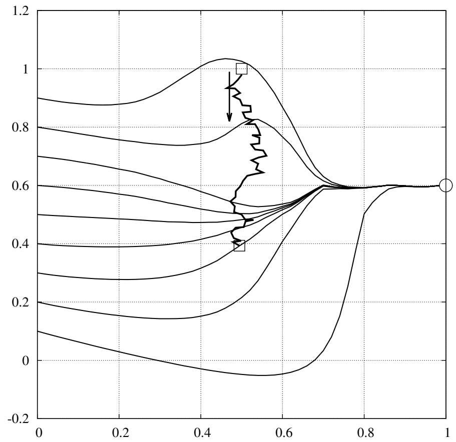
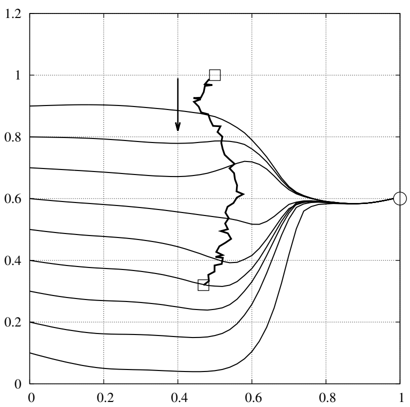


Fig. 11. The obstacle moves with the vector $\boldsymbol{B}$, where $|\boldsymbol{B}|$ is constant and θ_b is selected randomly.



(a)Model 1



(b)Model 2

Fig. 12. Simulation for moving obstacle randomly and the different designated place $(1.0, 0.6)$.

TABLE VI
MAIN FUZZY RULES OBTAINED BY LEARNING FOR MODEL 2.

	r_1	r_2	θ_1	θ_2	A_y
Rule 1	short	long	plus	center	right
Rule 2			minus		left
Rule 3		middle		middle	right
Rule 4			plus	plus	left
Rule 5	middle				left
Rule 6			minus	minus	right

V. CONCLUSION

In this paper, a theoretical result and some numerical simulations including obstacle avoidance are presented in order to compare the capability of the proposed model with the conventional models. It is shown that Model 1 and Model 2 with Gaussian membership function and $f_i(x_1, \cdots, x_m) = w_i$ are satisfied with the conditions of Stone-Weierstrass Theorem, so both models are universal approximators of continuous functions. Further, in order to compare the capability of learning algorithms for models, numerical simulation of function approximation and pattern classification problems are performed. Lastly, some simulations on obstacle avoidance are performed. In the simulations, it is shown that both models are successful in all trials. Specifically it is shown that Model 2 with the small number of parameters and interpretability is constructed.

In the future work, we will find an effective learning method for Model 2.

REFERENCES

[1] C. Lin and C. Lee, Neural Fuzzy Systems, Prentice Hall, PTR, 1996.
[2] M.M. Gupta, L. Jin and N. Homma, Static and Dynamic Neural Networks, IEEE Press, 2003.
[3] J. Casillas, O. Cordon and F. Herrera, L. Magdalena, Accuracy Improvements in Linguistic Fuzzy Modeling, Studies in Fuzziness and Soft Computing, Vol. 129, Springer, 2003.
[4] S. M. Zhoua, J. Q. Ganb, Low-level Interpretability and High-Level Interpretability: A Unified View of Data-driven Interpretable Fuzzy System Modeling, Fuzzy Sets and Systems 159, pp.3091-3131, 2008.
[5] J. M. Alonso, L. Magdalena and G. Gonza, Looking for a Good Fuzzy System Interpretability Index: An Experimental Approach, Journal of Approximate Reasoning 51, pp.115-134, 2009.
[6] H. Nomura, I. Hayashi and N. Wakami, A Learning Method of Simplified Fuzzy Reasoning by Genetic Algorithm, Proc. of the Int. Fuzzy Systems and Intelligent Control Conference, pp.236-245, 1992.
[7] O. Cordon, A Historical Review of Evolutionary Learning Methods for Mamdani-type Fuzzy Rule-based Systems, Designing interpretable genetic fuzzy systems, Journal of Approximate Reasoning, 52, pp.894-913, 2011.
[8] S. Fukumoto and H. Miyajima, Learning Algorithms with Regularization Criteria for Fuzzy Reasoning Model, Journal of Innovative Computing Information and Control, 1,1, pp.249-163, 2006.
[9] K. Kishida, H. Miya jima, M. Maeda and S. Murashima, A Self-tuning Method of Fuzzy Modeling using Vector Quantization, Proceedings of FUZZ-IEEE'97, pp397-402, 1997.
[10] N. Yubazaki, J. Yi and K. Hirota, SIRMS(Single Input Rule Modules) Connected Fuzzy Inference Model, J. Advanced Computational Intelligence, 1, 1, pp.23-30, 1997.
[11] H. Miyajima, N. Shigei and H. Miyajima, Fuzzy Inference Systems Composed of Double-Input Rule Modules for Obstacle Avoidance Problems, IAENG International Journal of Computer Science, Vol. 41, Issue 4, pp.222-230, 2014.
[12] M. J. Gacto, R. Alcala and F. Herrera, Interpretability of Linguistic Fuzzy Rule-based Systems:An Overview of Interpretability Measures, Inf. Sciences 181, pp.4340-4360, 2011.
[13] Y. Shi, M. Mizumoto, N. Yubazaki and M. Otani, A Self-Tuning Method of Fuzzy Rules Based on Gradient Descent Method, Japan Journal of Fuzzy Set and System 8, 4, pp.757-747, 1996.
[14] H. Miyajima, N. Shigei and H. Miyajima, On the Capability of a Fuzzy Inference System with Improved Interpretability, Lecture Notes in Engineering and Computer Science, Proceedings of The International MultiConference of Engineers and Computer Scientists 2015, IMECS 2015, 18-20 March, 2015, Hong Kong, pp.51-56.
[15] W. L. Tung and C. Quek, A Mamdani-takagi-sugeno based Linguistic Neural-fuzzy Inference System for Improved Interpretability-accuracy Representation, Proceedings of the 18th International Conference on Fuzzy Systems(FUZZ-IEEE'09), pp. 367-372, 2009.
[16] L. X. Wang and J. M. Mendel, Fuzzy Basis Functions, Universal Approximation and Orthogonal Least Square Learning, IEEE Trans. Neural Networks, Vol.3, No.3, pp.807-814, 1992.
[17] D. Tikk, L. T. Koczy and T. D. Gedeon, A Survey on Universal Approximation and its Limits in Soft Computing Techniques, Journal of Approximate Reasoning 33, pp.185-202, 2003.