New Fuzzy Modeling of Nonlinear Dynamic System Using the Relevance Vector Machine

Jongcheol Kim*, Taewon Kim*, and Yasuo Suga**†

* School of Integrated Design Engineering, Graduate School of Science and Technology, Keio University

** Department of Mechanical Engineering of Science and Technology, Keio University

3-14-1, Hiyoshi, Kouhoku-ku, Yokohama, 223-8522, Japan.

†Email: suga@mech.keio.ac.jp

***Abstract*— In the system modeling from training data with measurement noise, it is an important problem to select the appropriate structure of fuzzy model that can present good generalization. To solve this problem, this paper presents a new fuzzy inference system for modeling nonlinear dynamic system based on input and output data measurement noise. In the proposed fuzzy system, the number of fuzzy rules and parameter values of membership functions are automatically determined using the relevance vector machine (RVM) which does not have a bias term. The RVM has a probabilistic Bayesian learning framework. The RVM is described by the sum of times of weight and kernel function. Kernel function projects input space into high dimensional feature space. The structure of proposed fuzzy system is same as that of the Takagi-Sugeno fuzzy model. Especially, the number of fuzzy rules can be reduced under the process of optimizing a marginal likelihood by adjusting parameter values of kernel functions using the gradient ascent method. Once a structure is selected, coefficients in consequent part are determined by the least square method. Examples illustrate the effectiveness of the proposed new fuzzy inference system.**

I. INTRODUCTION

The Fuzzy Inference System (FIS) has shown powerful capability for the modeling of nonlinear systems [1], [2]. The FIS based on only human expertness may not lead to sufficient accuracy for complex and uncertain systems. Because of this reason, neuro-fuzzy modeling which acquires knowledge from a set of input-output data has been actively investigated [3]. If training data set for modeling has measurement noise and (or) available data size is too small in the real system modeling, neural network can bring out over-fitting problem which is a factor of poor generalization.

Therefore it is an important problem to select the appropriate structure of neuro-fuzzy model that can perform good generalization. Some researches have been progressed in order to solve this problem. Branco *et al.* [4] investigated how and why fuzzy modeling systems are affected when learning data is corrupted by noise. Holmstrom *et al.* [5] made an effort to improve the generalization capability of a neural network by introducing additive noise to the training samples. Karystinos *et al.* [6] addressed K-mean clustering algorithm which results from the least entropic Gaussian mixture upon equal-likelihood cross-validated shaping for improving milti-layer perceptrons (MLP) generalization ability. Lee *et al.* [7] described a general regression neural network with fuzzy ART clustering (GRNNFA), as hybrid neural network model, based on the fusion of fuzzy adaptive resonance theory (Fuzzy ART) and the general regression neural network (GRNN) for data regression. However, many researches have usually dealt system optimization [6], [7] and generalization problem [5] independently.

Recently, statistical approach methods have popularly developed in nonlinear system modeling based on input and output data with measured noise [8], [9], [10], [11]. Statistical techniques generally deal with trade-off between fitting the training data and simplifying model capacity. In statistical method, kernel function offers an alternative solution by mapping the data into high dimensional feature space to increase the computational power. The state-of-the-art Support Vector Machine (SVM)[12] has been used in order to automatically find the number of network nodes or fuzzy rules based on given error bound [11], [13]. The Support Vector Neural Network (SVNN) is proposed to select the best structure of radial based function network for the given precision [11]. The Support Vector Fuzzy Inference System is proposed to find the reduced number of rules using gradient descent method updating kernel parameters [13]. SVMs have delivered good performance in various application. However, Tipping [14] pointed out the following disadvantages of the support vector learning methodology and proposed the Relevance Vector Machine (RVM) method. In the SVM, predictions are not probabilistic and the kernel function $K(\mathrm{x}, \mathrm{x}_i)$ must satisfy Mercer's condition. It is also necessary to estimate the error/margin trade-off parameter *C*. Above all, the SVM is relatively less sparse better than the RVM. Particularly, the RVM base on a kernel-based Bayesian estimation method. The RVM has shown a comparable generalization performance with fewer kernel function than the SVM in [14].

In this paper, to select the appropriate structure of fuzzy model that can present good generalization, we propose a new fuzzy inference system for modeling nonlinear dynamic system based on measured noisy input and output data. In the suggested fuzzy system, the number of fuzzy rules and parameter value of membership functions are automatically found by using the RVM [14] which does not have a bias term. The structure of proposed fuzzy system is same as that of the Takagi-Sugeno (TS) fuzzy model. Especially, the number of fuzzy rules can be reduced under the process of optimizing a marginal likelihood by adjusting parameter values of kernel functions using the gradient ascent method. Once a structure is selected, coefficients in consequent part are determined by the least square method.

The rest of this paper is organized as follows. The RVM is introduced in Section II. The structure and learning algorithm of the new FIS using the RVM are given in Section III. The effectiveness of the proposed FIS are illustrated by examples involving nonlinear dynamic systems in Section IV. Conclusion is given in Section V.

II. Relevance Vector Machine

The RVM has an exploited probabilistic Bayesian learning framework. It acquires relevance vectors and weights by maximizing a marginal likelihood. The structure of the RVM is described by the sum of times of weights and kernel functions. A kernel function means a set of basis function projecting the input data into a high dimensional feature space to increase the computational power.

Given a data set of input-target pairs $\{\mathbf{x}_n, t_n\}_{n=1}^N$, and assuming that the targets are independent and contaminated with mean-zero Gaussian noise ϵ_n with variance σ^2:

$$t_n = y(\mathbf{x}_n; \mathbf{w}) + \epsilon_n. \tag{1}$$

The RVM without a bias can be represented as follows [14]:

$$y(\mathbf{x}; \mathbf{w}) = \sum_{i=1}^{N} w_i K(\mathbf{x}, \mathbf{x}_i) = \Phi \mathbf{w}, \tag{2}$$

where N is the length of the data, weight vector $\mathbf{w} = (w_1, ..., w_N)^T$ and $(N \times N)$ design matrix $\Phi = [\phi(\mathbf{x}_1), \phi(\mathbf{x}), ..., \phi(\mathbf{x}_N)]^T$, wherein $\phi(\mathbf{x}_n) = [K(\mathbf{x}_n, \mathbf{x}_1), K(\mathbf{x}_n, \mathbf{x}_2), ..., K(\mathbf{x}_n, \mathbf{x}_N)]^T$ and $K(\mathbf{x}, \mathbf{x}_i)$ is a kernel function.

The likelihood of the measured training data set is written as:

$$p(\mathbf{t}|\mathbf{w}, \sigma^2) = (2\pi\sigma^2)^{-N/2} \exp\left\{-\frac{1}{2\sigma^2}\|\mathbf{t} - \Phi\mathbf{w}\|^2\right\} \tag{3}$$

where target vector $\mathbf{t} = (t_1, ..., t_N)^T$. Maximizing likelihood estimation of $\mathbf{w}$ and σ^2 from Eq. (3) leads to over-fitting. To avoid this, a zero-mean Gaussian prior distribution over $\mathbf{w}$ with variance $\boldsymbol{\alpha}^{-1}$ is added as:

$$p(\mathbf{w}|\boldsymbol{\alpha}) = \prod_{i=0}^{N} \sqrt{\frac{\boldsymbol{\alpha}}{2\pi}} \exp\left(-\frac{\boldsymbol{\alpha}}{2} w_i^2\right), \tag{4}$$

where $\boldsymbol{\alpha} = (\alpha_1, \alpha_2, ..., \alpha_N)^T$.

The posterior distribution over the weight from Bayes rule is thus given by:

$$\begin{aligned} p(\mathbf{w}|\mathbf{t}, \boldsymbol{\alpha}, \sigma^2) &= \frac{\textit{likelihood} \times \textit{prior}}{\textit{normalizing factor}}, \\ &= \frac{p(\mathbf{t}|\mathbf{w}, \sigma^2) p(\mathbf{w}|\sigma^2)}{p(\mathbf{t}|\boldsymbol{\alpha}, \sigma^2)}, \\ &= (2\pi)^{-(N+1)/2} |\boldsymbol{\Sigma}|^{-1/2} \cdot \\ &\quad \exp\left\{-\frac{1}{2}(\mathbf{w} - \boldsymbol{\mu})^T \boldsymbol{\Sigma}^{-1} (\mathbf{w} - \boldsymbol{\mu})\right\} \end{aligned} \tag{5}$$

where the posterior mean $\boldsymbol{\mu}$ and covariance $\boldsymbol{\Sigma}$ are as follows:

$$\boldsymbol{\mu} = \sigma^{-2} \boldsymbol{\Sigma} \Phi^{\mathrm{T}} \mathbf{t}, \tag{6}$$

$$\boldsymbol{\Sigma} = (\sigma^{-2} \Phi^{\mathrm{T}} \Phi + \mathbf{A})^{-1}, \tag{7}$$

with $\mathbf{A} = \mathrm{diag}(\alpha_1, \alpha_2, ..., \alpha_N)$.

The likelihood distribution over the training targets Eq. (3) can be marginalized with respect to the weights to obtain the *marginal likelihood*, which is also a Gaussian distribution

$$\begin{aligned} p(\mathbf{t}|\boldsymbol{\alpha}, \sigma^2) &= \int p(\mathbf{t}|\mathbf{w}, \sigma^2) p(\mathbf{w}|\boldsymbol{\alpha}) d\mathbf{w}, \\ &= (2\pi)^{-N/2} |\mathbf{C}|^{-1/2} \exp\left\{-\frac{1}{2}\mathbf{t}^T \mathbf{C}^{-1} \mathbf{t}\right\} \end{aligned} \tag{8}$$

with covariance $\mathbf{C} = \sigma^2 \mathbf{I} + \Phi\ \mathbf{A}^{-1} \Phi^{\mathrm{T}}$.

Values of $\boldsymbol{\alpha}$ and σ^2 maximizing the *magrinal likelihood* cannot be obtained in closed form, and an iterative re-estimation method is required [14]. The following the approach of MacKay [15] gives:

$$\alpha_i^{new} = \frac{\gamma_i}{\mu_i^2}, \tag{9}$$

$$(\sigma^2)^{new} = \frac{\|\mathbf{t} - \boldsymbol{\Sigma}\boldsymbol{\mu}\|^2}{N - \sum_i \gamma_i}, \tag{10}$$

where μ_i is the i-th posterior mean weight Eq. (6) and the quantities $\gamma_i \equiv 1 - \alpha_i \sum_{ii}$ with the i-th diagonal element $\sum_{ii}$ of the posterior weight covariance Eq. (7).

In the process of solving this optimization problem, the vector from the training set that associates with nonzero hyperparameter is called the *relevance vector*.

III. New Fuzzy Inference System Using The Relevance Vector Machine

This section describes the structure of the new fuzzy inference system based on the TS fuzzy model and the learning algorithm.

A. The Structure of the FIS Using the Relevance Vector Machine

Suppose we have given input and target data

$$(\mathbf{x}_1, t_1),\ ...,\ (\mathbf{x}_N, t_N) \tag{11}$$

where $\mathbf{x}_i$=$[x_1^i, x_1^i, ..., x_D^i]$ $(i = 1, 2, ..., N)$ is a input variable and $\mathbf{t}$=$[t_1, ..., t_N]$ is a target variable. The proposed TS fuzzy model with fuzzy if-then rules can be represented by Eq. (12).

$$\begin{aligned} R_1 &: \text{If } x_1 \text{ is } K(x_1, x_{11}^*) \text{ and ... } x_D \text{ is } K(x_D, x_{1D}^*), \\ &\quad \text{Then } f_1 = a_{10} + a_{11}x_1 + ... + a_{1D}x_D \\ R_2 &: \text{If } x_1 \text{ is } K(x_1, x_{21}^*) \text{ and ... } x_D \text{ is } K(x_D, x_{2D}^*), \\ &\quad \text{Then } f_2 = a_{20} + a_{21}x_1 + ... + a_{2D}x_D \\ &\quad \vdots \qquad\qquad \vdots \qquad\qquad \vdots \\ R_n &: \text{If } x_1 \text{ is } K(x_1, x_{n1}^*) \text{ and ... } x_D \text{ is } K(x_D, x_{nD}^*), \\ &\quad \text{Then } f_n = a_{n0} + a_{n1}x_1 + ... + a_{nD}x_D, \end{aligned} \tag{12}$$

where n is the number of fuzzy rules, D is the dimension of input variables, $x_j (j = 1, 2, ..., D)$ is a input variable, f_i is

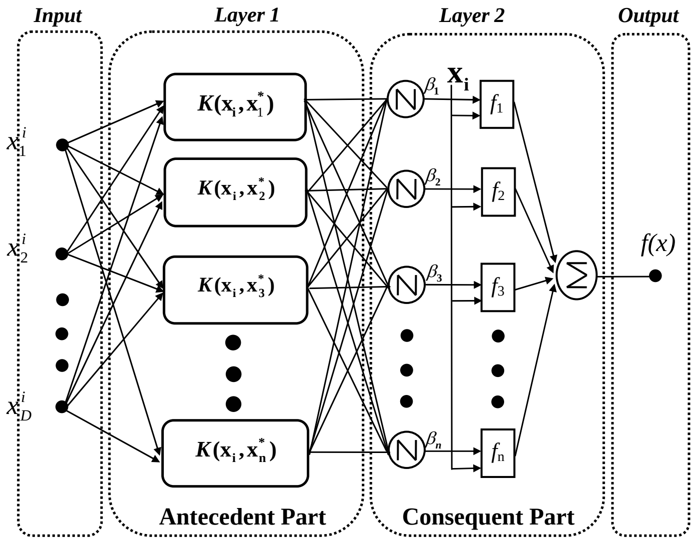

Fig. 1. Structure of the proposed fuzzy inference system

the i-th local output variable, $K(x_j, x_{ij}^*)(i = 1, 2, ..., n, j = 1, 2, ..., D)$ is a fuzzy set and $a_{ij}(i = 1, 2, ..., n, j = 0, 1, ..., D)$ is a consequent parameter.

Now, we describe the structure of FIS using the RVM. It consists of two layers as shown in Fig. 1. The two layers involved in the proposed FIS are presented as follows:

Layer 1: Input space is nonlinearly projected into feature space using kernel functions. Relevance Vectors are obtained from the RVM learning algorithm. The Gaussian kernel function with variance θ_i is used as follows:

$$K(\mathbf{x}, \mathbf{x}_i^*) = \exp\Big(-\frac{(\mathbf{x}-\mathbf{x}_i^*)^2}{2\theta_i^2}\Big), i = 1, 2, ..., n \quad (13)$$

where $\mathbf{x}_i^*$ is a RV, θ_i is called a kernel parameter and n is the number of RVs. After all, this kernel function becomes a Gaussian membership function in the proposed FIS. $\mathbf{x}_i^*$ and θ_i is respectively the center and variance of the i-th Gaussian membership function. The RVM algorithm is a fuzzy inference engine determining the number of fuzzy rules in FIS. The Layer 1 is related to the antecedent part of the FIS.

Layer 2: For the overall output of the fuzzy model constructed, defuzzification using the Center Of Gravity (COG) method is performed.

$$\begin{aligned} f(\mathbf{x}) &= \frac{\sum_{i=1}^n K(\mathbf{x}, \mathbf{x}_i^*)\, f_i}{\sum_{j=1}^n K(\mathbf{x}, \mathbf{x}_j^*)}, \\ &= \sum_{i=1}^n \beta_i\,(a_{i0} + a_{i1}x_1 + \cdots + a_{iD}x_D) \quad (14) \end{aligned}$$

where $\beta_i = \frac{K(\mathbf{x}, \mathbf{x}_i^*)}{\sum_{j=1}^n K(\mathbf{x}, \mathbf{x}_j^*)}$ and $f_i = a_{i0} + a_{i1}x_1 + \cdots + a_{iD}x_D$. It is assumed that $K(\mathbf{x}, \mathbf{x}_i^*) \geq 0$, $\sum_{j=1}^n K(\mathbf{x}, \mathbf{x}_j^*) > 0$. Therefore, $0 \leq \beta_i \leq 1$, $(i = 1, 2, ..., n)$.

The Layer 2 connects with the consequent part of the FIS.

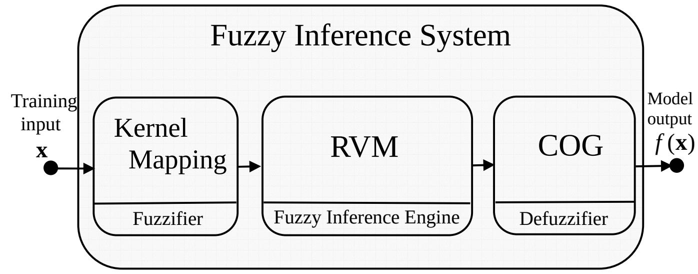

Fig. 2. Learning algorithm of the proposed FIS

B. The Learning Algorithm of the FIS Using the Relevance Vector Machine

The learning algorithm of the FIS using the RVM is shown in Fig. 2. It can be summarized by the following iterative procedure.

Step 1: Assign the initial hyperparameter $\boldsymbol{\alpha}$ and kernel parameter θ_i.

Step 2: Using the following extended RVM algorithm based on kernel mapping [14], find RVs $\mathbf{x}_i^*$ being the centers $\mathbf{c}_i$ of Gaussian membership function and weight $\mathbf{w}$. Particularly, using the Gradient Ascent Method (GAM), kernel parameter θ_i is adjusted in order to select the appropriate type of kernel function related to the nonlinear dynamic system. Assume that the log of the marginal likelihood Eq. (8) is the objective function L,

$$\begin{aligned} L = -\frac{1}{2}\ &[\log|\sigma^2\mathbf{I} + \Phi\ \mathbf{A}^{-1}\Phi^{\mathrm{T}}| \\ &+ \mathbf{t}^{\mathrm{T}}(\sigma^2\mathbf{I} + \Phi\ \mathbf{A}^{-1}\Phi^{\mathrm{T}})^{-1}\mathbf{t}]. \quad (15) \end{aligned}$$

From the GAM, the kernel parameter θ_i is updated such that the objective function L is maximized as:

$$\begin{aligned} \Delta\theta_k &= \eta_\theta \nabla_{\theta_k} L = \eta_\theta \frac{\partial L}{\partial \theta_k}, \\ &= \eta_\theta \frac{\partial L}{\partial \phi_{nm}} \frac{\partial \phi_{nm}}{\partial \theta_k}, \\ &= \eta_\theta \theta_k^{-3} \left[\sum_{n=1}^N \sum_{m=1}^N D_{nm}\Phi_{nm}(x_{nk} - x_{mk})^2\right] (16) \end{aligned}$$

where $D_{nm} = \partial L / \partial \phi_{nm}$ wherein matrix $D = \sigma^{-2}[(\mathbf{t}-\mathbf{y})\boldsymbol{\mu}^{\mathrm{T}} - \Phi\boldsymbol{\Sigma}]$, a set of Gaussian kernel function $\phi_{nm} = \exp\{-\sum_{k=1}^n (x_{nk} - x_{mk})^2 / 2\theta_k^2\}$ and η_θ is the learning late of θ_i.

Step 3: Using the following Least Square Estimation (LSE) method, estimate the parameter a_{ij} of the linear equation f_i in Eq. (14).

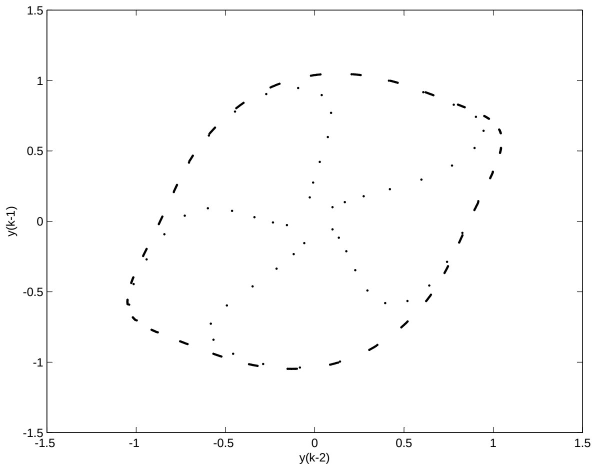

Fig. 3. Output data of dynamic system for $e(k) \equiv 0$

Let

$$Q = \left[\; a_{10} \; a_{11} \; ... \; a_{1D} \; , ..., \; a_{n0} \; a_{n1} \; ... \; a_{nD} \; \right]^{\mathrm{T}}, \tag{17}$$

$$W = \begin{bmatrix} \beta_1^1 \; \beta_1^1 x_1^1 \; ... \; \beta_1^1 x_D^1, ..., \; \beta_n^1 \; \beta_n^1 x_1^1 \; ... \; \beta_n^1 x_D^1 \\ \beta_1^2 \; \beta_1^2 x_1^2 \; ... \; \beta_1^2 x_D^2, ..., \; \beta_n^2 \; \beta_n^2 x_1^2 \; ... \; \beta_n^2 x_D^2 \\ \vdots \quad \vdots \quad ... \quad \vdots \; , ..., \; \vdots \quad \vdots \quad ... \quad \vdots \\ \beta_1^n \; \beta_1^n x_1^l \; ... \; \beta_1^n x_D^n, ..., \; \beta_n^n \; \beta_n^n x_1^n \; ... \; \beta_n^n x_D^n \end{bmatrix} \tag{18}$$

where $\beta_i^j = \frac{K(\mathbf{x}_j, \mathbf{x}_i^*)}{\sum_{k=1}^{n} K(\mathbf{x}, \mathbf{x}_k^*)}$. Thus fuzzy model output is $f(\mathbf{x}) = WQ$.

If $(W^{\mathrm{T}}W)$ is nonsingular, the parameter vector Q is calculated by

$$Q \;\; = \;\; (W^{\mathrm{T}}W)^{-1}W^{\mathrm{T}}\mathbf{y}. \tag{19}$$

IV. Examples

In this section, we show two simulation results of the proposed FIS for the modeling of the nonlinear dynamic systems.

A. *Example 1 : Modeling of 2-Input Nonlinear dynamic system*

Consider the nonlinear dynamic system [11],

$$\begin{aligned} y(k) \;\; = \;\; & (0.8 - 0.5 \exp(-y^2(k-1)))y(k-1) \\ & - (0.3 + 0.9 \exp(-y^2(k-1)))y(k-2) \\ & + 0.1 \sin(\pi y(k-1)) + e(k) \end{aligned} \tag{20}$$

where $e(k)$ is a white noise, $e(k) \sim N(0, 0.1^2)$. The training input of the model is $X(k) = [y(k-1), y(k-2)]$. For $e(k) \equiv 0$, this nonlinear dynamic system is unstable at the origin. Training data of dynamic system input with 300 data points is shown in Figure 3. This data points are generated from an initial condition of $X(1) = [0.1, 0.1]$. But, the training input data of 300 point pairs are generated from initial condition of $X(1) = [0, 0]$. After the simulation, the proposed FIS generates the 7 RVs ($\mathbf{x}_i^*$), so that it has 7 rules as follows,

$$\begin{aligned} R_i \;\; : \;\; & \text{If } x_1 \text{ is } K(x_1, x_{i1}^*) \text{ and } x_2 \text{ is } K(x_2, x_{i2}^*), \\ & \text{Then} f_i = a_{i0} + a_{i1}x_1 + a_{i2}x_2, i = 1, ..., 7. \end{aligned} \tag{21}$$

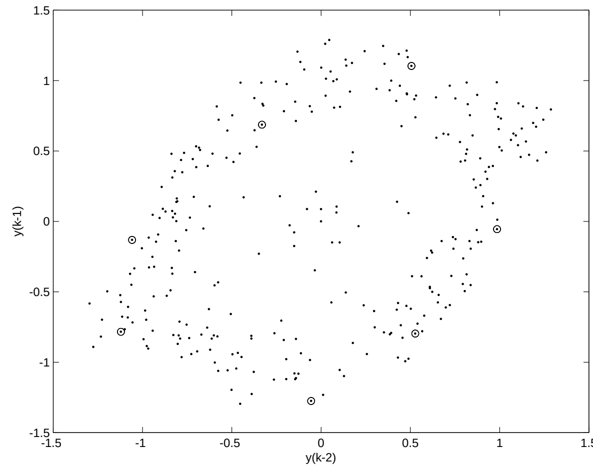

Fig. 4. Training data of dynamic system with noise and selected RVs (O)

TABLE I

Parameter values of the FIS for modeling of $X(k) = [y(k-1), y(k-2)]$

Rule	Antecedent part		Consequent part
	c_{ij}	θ_{ij}	$a_{i0},\ a_{i1},\ a_{i2}$
1	1.1036 0.5064	1.2486 1.2462	176.34 -29.17 -17.59
2	-0.1316 -1.0581	1.2761 1.2836	256.04 -60.45 21.09
3	-0.7969 0.5276	1.2731 1.2719	284.20 -50.14 -79.64
4	-0.7838 -1.1208	1.2730 1.2779	-262.79 -30.91 -13.15
5	-0.0553 0.9856	1.2817 1.3074	-43.97 -20.68 22.49
6	-1.2754 -0.0553	1.2788 1.2754	-374.52 -52.24 -46.78
7	0.6863 -0.3308	1.2720 1.2582	-116.25 0.02 -37.16

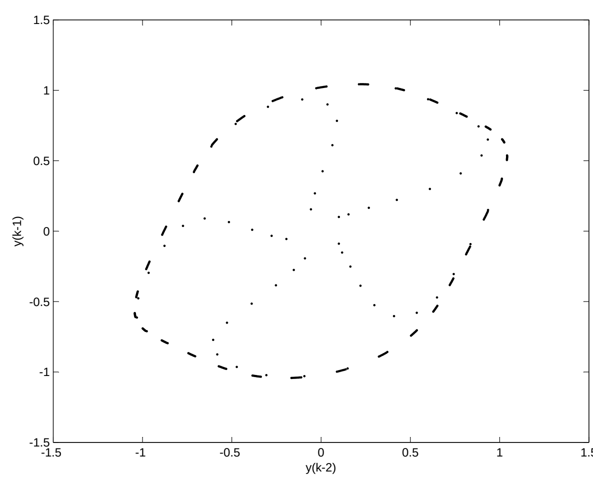

Fig. 5. Estimated dynamic system output of $X(k) = [y(k-1), y(k-2)]$

TABLE II

Compared results of nonlinear dynamic function

Type	Rules(or SVs)	Model error
Chan *et al.* [11]	10	0.099
Proposed FIS	7	0.017

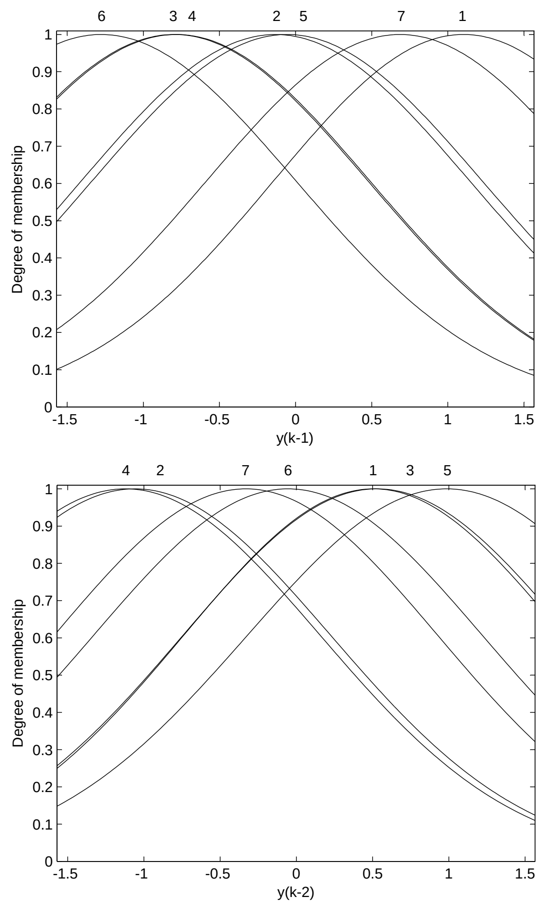

Fig. 6. Membership function of modeling the dynamic system

Figure 4 shows training data with noises and selected 7 relevance vectors. The parameter values of premise and consequence parts are listed in Table I. Figure 5 shows the modeling result of estimated dynamic system output of $X(k) = [y(k-1), y(k-2)]$. Membership functions show in Figure 6. The method in the literature applied to the same dynamic system, and the results listed on the Table II. The modeling error is the standard deviation of test errors. Compared with the number of rules and modeling error, the proposed method gives the smaller number of rules and modeling error rather than other.

B. Example 2 : Modeling of Robot Arm Data

The training robot arm data are obtained from the relationship between input variables (x_1, x_2) being joint angles and target variables (y_1, y_2) being positions,

$$y_1 = 2.0\cos x_1 + 1.3\cos(x_1 + x_2) + \delta, \tag{22}$$

$$y_2 = 2.0\sin x_1 + 1.3\sin(x_1 + x_2) + \delta, \tag{23}$$

where δ is a Gaussian noise, $\delta \sim N(0, 0.05^2)$. We use the 400 input-target pairs of robot arm which was used by MacKay [16] and Chu *et al.* [17]. In this data set, the first 200 data and the second 200 data are respectively used as training and test data set. After the simulation, the proposed FIS respectively

TABLE III

PARAMETER VALUES OF THE FIS FOR MODELING OF y_1

Rule	Antecedent part		Consequent part
	c_{ij}	θ_{ij}	$(a_{i0},\ a_{i1},\ a_{i2})(10^4)$
1	-0.9451 3.0913	1.6588 1.6775	-0.0130 0.0009 0.0103
2	1.2031 2.7042	2.0207 1.8010	-1.2575 0.2623 0.0405
3	1.1397 0.5988	1.8474 1.8775	-0.0953 -0.0161 0.0598
4	0.5122 1.4056	1.7968 1.7698	-0.9086 -0.0897 0.0128
5	-1.8941 0.8151	1.8014 1.7803	0.3043 0.0316 -0.0023
6	1.6345 1.7778	2.1581 1.8619	7.0789 -0.1506 -0.5634
7	-0.9796 1.5137	1.8311 1.8411	-0.7581 0.0134 0.1401
8	1.7438 1.1445	2.0774 1.8797	-3.8325 -0.1710 -0.5539
9	0.8388 3.1328	1.8193 1.8737	1.0665 -0.0248 -0.0979

TABLE IV

PARAMETER VALUES OF THE FIS FOR MODELING OF y_2

Rule	Antecedent part		Consequent part
	c_{ij}	θ_{ij}	$(a_{i0},\ a_{i1},\ a_{i2})(10^4)$
1	-0.9362 1.3506	1.7683 1.7693	0.2474 0.0291 0.0046
2	0.9283 2.6690	1.7913 1.7589	0.1231 0.0648 0.0328
3	1.1615 0.6477	1.7438 1.8362	0.0178 -0.0015 0.0047
4	-1.0987 3.1254	1.8134 1.8089	0.2060 -0.0208 -0.0094
5	1.8904 1.6948	1.8718 1.7891	-0.0174 0.0027 -0.0017
6	-1.8941 0.8151	1.9341 1.8339	1.2382 0.0751 0.1493
7	-1.7287 2.5087	1.8177 1.7801	0.2529 -0.0354 -0.0872
8	-1.0644 0.5615	1.8274 1.8352	-0.5924 -0.0096 -0.0887
9	1.5707 3.0659	1.7598 1.7811	-0.4055 0.0153 0.0208
10	-1.7653 1.4311	1.8860 1.7826	-0.9732 0.0560 -0.0287

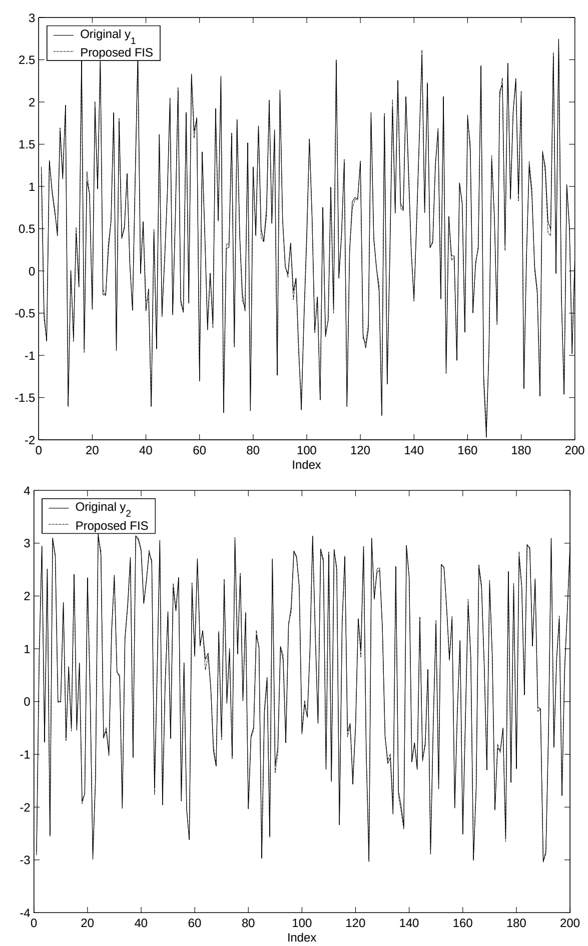

Fig. 7. Comparison of test data of y_1, y_2 and outputs of the proposed FIS

TABLE V

COMPARED RESULTS OF THE MODELING ROBOT ARM DATA y_1 AND y_2

Type		Rules(or SVs)	ASE (10^{-3})
Chu *et al.* [17]	y_1	21	2.491
	y_2	42	3.184
Proposed FIS	y_1	9	2.465
	y_2	10	3.046

generates the 9 and 10 RVs $(\mathbf{x}_i^*)$ for y_1 and y_2, so that it has 9 and 10 rules as follows,

$$\begin{aligned} R_i \quad : \quad & \text{If } x_1 \text{ is } K(x_1, x_{i1}^*) \text{ and } x_2 \text{ is } K(x_2, x_{i2}^*), \\ & \text{Then } \; f_i = a_{i0} + a_{i1}x_1 + a_{i2}x_2. \end{aligned} \tag{24}$$

The parameter values of premise and consequence parts are listed in Table III and IV. Figure 7 shows the modeling results of test robot arm data of y_1 and y_2.

To analyze the performance of the proposed FIS, the modeling error is defined by as following Average Square Error (ASE)

$$\text{ASE} = \frac{\sum_{k=1}^{N}(y_k - f(x_k))^2}{N} \tag{25}$$

where N is the number of data, y_k and $f(x_k)$ are respectively the original system and fuzzy modeling output.

The method in the literature applied to the same dynamic system, and the results listed on the Table V. Compared with the number of rules and modeling error, the proposed method gives the smaller number of rules regard to similar modeling error rather than other.

V. CONCLUSION

In this paper, we have introduced a new approach to fuzzy modeling using the relevance vector machine. Our main concern is to determine the best structure of the TS fuzzy model for modeling nonlinear dynamic systems with measurement error. The number of rules and the parameter values of membership functions in the proposed FIS can be decided using maximizing the marginal likelihood of the RVM. Parameter values of kernel functions were adjusted using the gradient ascent method. Coefficients in consequent part of the TS fuzzy model were determined by the least square method. Simulation of examples showed the effectiveness of the proposed FIS for modeling of nonlinear dynamic systems with noise.

REFERENCES

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

[2] M. Sugeno and T. Yasukawa, ”A Fuzzy-logic-based Approach to Qualitative Modeling,” *IEEE Trans. on Fuzzy Sys.*, vol.1, no.1, pp.7-31, 1993.

[3] J. R. Jang, ”ANFIS: Adaptive-Network-based Fuzzy Inference Systems,” IEEE Trans. Syst., Man. Cybern., vol.23, pp.665-685, 1993.

[4] P. J. Costa Branco and J. A. Dente, ”Noise Effects in Fuzzy Modelling Systems: Three Case Studies,” *In: Computational Intelligence and Applications*, pp.103-108, World Scientific and Engineering Society Press, Danvers, USA, 1999.

[5] L. Holmstrom and P. Koistine, ”Using Additive Noise in BackPropagation Training,” *IEEE Trans. Neural Networks*, vol.3, no.1, pp.24-38, 1992.

[6] G. N. Karystinos and D. A. Pados, ”On Overfitting, Generalization, and Randomly Expanded Training Sets,” *IEEE Trans. Neural Networks*, vol.13, no.5, pp.1050-1057, 2000.

[7] E. W. M. Lee, C. P. Lim, R. K. K. Yuen, and S. M. Lo, ”A Hybrid Neural Network Model for Noisy Data Regression,” *IEEE Trans. Syst., Man. Cybern. Part B:Cybern.*, vol.34, no.2, pp.951-960, 2004.

[8] J. Yen and L. Wang, ”Application of Statistical Information Criteria for Optimal Fuzzy Model Construction,” *IEEE Trans. Fuzzy Systems*, vol.6, no.3, pp.362-372, 1998.

[9] T. Matsumoto, Y. Nakajima, J. Sugi, and Hl. Hamagishi, ”Reconstructions and Predictions of Nonlinear Dynamical Systems: A Hierarchical Bayesian Approach,” *IEEE Trans. Signal Processing*, vol.49, no.9, pp.2138-2155, 2001.

[10] I. Rivals and L. Personnaz, ”Neural-Network Construction and Selection in Nonlinear Modeling,” *IEEE Trans. Neural Networks*, vol.14, no.4, pp.804-819, 2003.

[11] W. C. Chan, W. C. Chan, K. C. Cheung and C. J. Hariss, ”On the Modeling of Nonlinear Dynamic Systems Using Support Vector Neural Networks,” *Engineering Application of Artifical Intelligence*, vol.14, pp.105-113, 2001.

[12] V. Vapnik, *The Nature of Statistical Learning Theory*, New York: Spring-Verlag, 1995.

[13] Jongcheol Kim and Sangchul Won, ”New Fuzzy Inference System using a Support Vector Machine,” *Proc. 41th IEEE conference of decision and control*, Las Vegas, USA, pp1349-1354, 2002.

[14] M. E. Tipping, ”Sparse Bayesian Learning and the Relevance Vector Machine,” *Journal of Machine Learning Research*, vol.1, pp.211-244, 2001.

[15] D. J. C. MacKay, ”Bayeseian interpolation,” *Neural Computation*, vol.4, no.3, pp.415-445, 1992.

[16] D. J. C. MacKay ”A Practical Baysian Framework for Backpropagation Network,” *Neural Computation*, vol.4, no.3, pp.448-472, 1992.

[17] W. Chu, S.S. Keerthi and C.J. Ong, ”Bayesian Support Vector Regression Using a Unified Loss Function,” *IEEE Trans. Neural Networks*, vol.15, no.1, pp.29-44, 2004.