This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TII.2018.2813977, IEEE Transactions on Industrial Informatics

Bayesian Takagi-Sugeno-Kang Fuzzy Model and Its Joint Learning of Structure Identification and Parameter Estimation

Xiaoqing Gu, Shitong Wang

***Abstract*—In this paper, a novel Bayesian Takagi-Sugeno- Kang (TSK) fuzzy model and its joint learning method BTSK-JL of structure identification and parameter estimation are proposed for regression tasks from the perspective of Bayesian inference framework with a prior assumption about the number of fuzzy rules. Unlike most of existing TSK fuzzy systems where both their structure identification and parameter estimation of each fuzzy rule involved in them are learnt in a separate manner, BTSK-JL can determine the number of fuzzy rules and antecedent/consequent parameters of rules simultaneously in the proposed model. In order to guarantee their optimal solutions, BTSK-JL adopts a particle filter method to find the maximum-a-posterior value of the parameters. Due to taking into account the subtle interaction between input and output spaces, BTSK-JL can obtain good predictive performance and a set of compact fuzzy rules. Fuzziness and probability can work complementarily rather than competitively for such a TSK fuzzy system modeling. Experimental results on four time-series datasets and a glutamic acid fermentation process dataset have shown the validity and effectiveness of the proposed model.**

***Index Terms*—Takagi-Sugeno-Kang (TSK) fuzzy model, regression tasks, Bayesian model, structure identification, parameter estimation**

I. Introduction

As one of popular regression approaches, fuzzy systems, which are motivated by imitating human knowledge and reasoning processes, have been developed to represent the input-output relationships of system modeling tasks in the form of fuzzy rules [1]. Fuzzy system modeling for input-output data collected from real application scenarios has been obtaining great success in industrial control, fault diagnosis, pattern recognition, time-varying data analysis and so on [2-7]. In contrast to other regression approaches, the greatest advantage of fuzzy systems is their linguistically interpretability with a set of fuzzy rules. Due to their simple structure, high interpretability and strong approximation capability, Takagi-Sugeno-Kang (TSK) fuzzy systems [8-9] have been earning more and more attractions and practical applications. A Takagi-Sugeno-Kang (TSK) fuzzy system consists of "IF-THEN" fuzzy rules. For the *d*-dimension input vector $\mathbf{x} = [\mathbf{x}_1, \mathbf{x}_2, ..., \mathbf{x}_d]^T$, the *k*th fuzzy rule, which is denoted as R^k, in a TSK fuzzy system is represented as

Rule R^k: IF $x_1(k)$ is A_{k1}, $x_2(k)$ is A_{k2}, …, $x_d(k)$ is A_{kd},

$$\text{THEN}\quad f_k(\mathbf{x}) = v_{k0} + \sum_{j=1}^{d} v_{kj} x_j = \mathbf{v}_k{}^T \tilde{\mathbf{x}}_n\,, \tag{1}$$

where $k = 1,\ldots,K$, $A_{ki}(i = 1,2,....,d)$ is a fuzzy subset, $\tilde{\mathbf{x}}_n = [1, \mathbf{x}_n^T]^T$ and $\mathbf{v}_k = [v_{k0}, v_{k1}, ..., v_{kd}]^T$ indicates the vector of consequent parameters of the *k*th rule. In this study, the Gaussian membership function is employed to represent $A_{ki}(x_i)$, which can be expressed by

$$\mu_{A_{ki}}(x_i) = \exp(-\left\|x_i - c_{ki}\right\|^2 / \delta_{ki})\,, \tag{2}$$

where the parameters c_{ki} and δ_{ki} are the center and width of the corresponding Gaussian membership function, respectively. When a simple weighted sum is used for the defuzzification operation, the output of the TSK fuzzy system can be written as

$$\hat{y} = \sum_{k=1}^{K} \frac{\mu_k(\mathbf{x})}{\sum_{k'=1}^{K} \mu_k^{'}(\mathbf{x})} f_k(\mathbf{x}) = \sum_{k=1}^{K} \tilde{\mu}_k(\mathbf{x}) f_k(\mathbf{x})\,, \tag{3}$$

where $\mu_k(\mathbf{x})$ and $\tilde{\mu}_k(\mathbf{x})$ are the fuzzy membership function and the normalized fuzzy membership function of the antecedent parameters of the *k*th fuzzy rule, respectively, which are defined as

$$\mu_k(\mathbf{x}) = \prod_{i=1}^{d} \mu_{A_{ki}}(x_i)\,, \quad \tilde{\mu}_k(\mathbf{x}) = \mu_k(\mathbf{x}) / \sum_{k'=1}^{K} \mu_k^{'}(\mathbf{x})\,. \tag{4}$$

The construction of a TSK fuzzy system requires two main stages: its structure identification and parameter estimation of each fuzzy rule therein. The work of structure identification relates to find a proper partition of input space. Parameter estimation deals with finding the optimal values of all the parameters of rules involved in a TSK fuzzy system, including

This work was supported in part by the Hong Kong Polytechnic University under Grant G-UA68, in part by the National Natural Science Foundation of China under Grants 61572236, 61772241 and 61702225, by the Fundamental Research Funds for the Central Universities under Grant JUSRP51614A, and in part by the Natural Science Foundation of Jiangsu Province under Grant BK 20160187.

X. Q. Gu is with the School of Digital Media, Jiangnan University, Wuxi 214122, China, and is also with the School of Information Science and Engineering, Changzhou University, Changzhou 213164, China (e-mail: czxqgu@163.com).

S. T. Wang (corresponding author) is with the School of Digital Media, Jiangnan University, Wuxi 214122, China (e-mail: wxwangst@aliyun.com).

the parameters of fuzzy membership functions used in antecedent part of each rule, and the linear coefficients in the corresponding consequent part of each rule [10]. In particular, determining the number of fuzzy rules is one crucial issue in structure identification. Too many redundant fuzzy rules may perhaps lead to a complex fuzzy model and overfitting problem, while too few fuzzy rules may be insufficient to achieve the good performance for a fuzzy system [11, 12].

The simplest strategy for determining the number of rules is to partition input space into a fixed grid-type fuzzy partition. Many successful realizations of this strategy are achieved by using genetic algorithm (GA), evolutionary multiobjective optimization (EMO), or evolutionary algorithm (EA) [13, 14]. The second commonly used strategy for determining the number of fuzzy rules is to take clustering techniques, such as *k*-means and FCM [15, 16] and one-pass clustering [17]. In this strategy, the number of clusters is equal to the number of fuzzy rules. Clustering algorithms are naturally leveraged to partition input space for fuzzy rules. However, the cluster strategy of structure identification and parameter estimation has two shortcomings. One is the number of rules generally has to be decided manually. The other is that the subtle interaction between input and output spaces cannot be captured and leveraged in the corresponding TSK fuzzy system.

To tackle with the first problem, the most frequently used strategy is to select the number of fuzzy rules with a given search grid in the cross-validation strategy [15]. Obviously, such a cross-validation is time consuming, and selecting a reasonable search grid is not an easy work. What is more, we should consider the non-uniqueness of the representations of fuzzy rules for practical fuzzy modeling. For example, one expert believes that 50 rules are appropriate for a fuzzy system, while the other one specifies 70 rules for the same system. Which is better? Also, even if we build the corresponding two fuzzy systems, the learning will be time consuming and even inconsistent rules may have different interpretations. Although some cluster validity indices like the Xie-Beni index, Mountain potential index and so on are taken for fuzzy rule identification [18], these validity indices are naturally designed for cluster validation, they may not be fully appropriate for the rule-based system identification problem [16]. In order to overcome the second challenge, a few joint learning methods [16, 19-21] for simultaneous antecedent and consequent parameter learning have been exploited for recent years. For example, in [22], an iterative linear support vector regression (SVR) is taken to jointly determine antecedent/consequent parts of fuzzy rules. Unfortunately, these joint learning strategies still require setting the number of rules manually.

Several works on fuzzy systems have been witnessing Zadeh’s assertion that *probabilistic techniques and fuzziness are complementary rather than competitive* [23]. For example, in [24], probability theory and fuzzy logic are combined to derive a nonstationary fuzzy model for handling stochastic uncertainties. In [25], the equivalence between Gaussian mixture model and additive fuzzy systems is revealed and accordingly a novel training algorithm about additive fuzzy systems is developed. In [21, 26], probabilistic techniques such as the Metropolis Hastings (MH) sampling method and Kalman filter algorithm are adopted to train fuzzy systems. Motivated on these works, a new Bayesian TSK fuzzy model and its joint learning method BTSK-JL of structure identification and parameter estimation of fuzzy rules are developed in this paper to obtain a TSK fuzzy model with satisfactory prediction performance and compact fuzzy rules for regression tasks. The core concept of the proposed TSK fuzzy model is to use a joint likelihood probability to characterize fuzzy regression under Bayesian inference framework with a prior framework about the number of fuzzy rules. With the maximum a posteriori (MAP) principle, BTSK-JL adopts a particle filter method to simultaneously identify the structure and estimate antecedent/ consequent parts of fuzzy rules.

The contributions of this work can be summarized as the following aspects.

(1) This work explores how a TSK fuzzy system can be casted into a Bayesian inference framework. To the best of our knowledge, it is the first attempt in establishing the Bayesian model for simultaneous structure identification and parameter estimation of fuzzy rules for a TSK fuzzy system. This model indeed leverages the strengths of both statistics and fuzzy logic to solve nonlinear regression problems effectively.

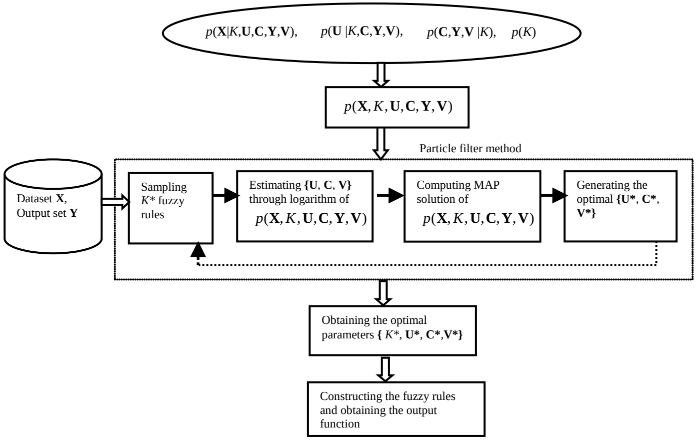
(2) Different from the classical modeling methods of a TSK fuzzy system which tune unknown parameters by using a "black box" strategy such as the grid search, the proposed Bayesian TSK fuzzy model can jointly learn all parameters of fuzzy rules, especially identify the number of fuzzy rules without any prior expert experience. Since fuzzy rule extraction is in a joint learning way, the proposed leaning method can exploit the subtle interaction between input and output spaces.

(3) Experimental results on four time-series datasets and a glutamic acid fermentation process dataset have shown the effectiveness of the proposed Bayesian TSK fuzzy model. The proposed Bayesian TSK fuzzy model indeed results in good predictive performance with a smaller number of fuzzy rules, in contrast to the comparison algorithms.

The remainder of this paper is organized as follows. Section II presents in detail the new TSK fuzzy model based on a Bayesian probabilistic model, as well as its structure and parameter learning method BTSK-JL. The experimental results are reported in Section III and Section IV concludes the whole article.

II. Bayesian Takagi-Sugeno-Kang Fuzzy System and its Joint Learning of Structure Identification and Parameter Estimation

In this section, we will reveal that TSK fuzzy model can be understood from the perspective of a Bayesian inference framework. As an attempt in this aspect, we develop a new Bayesian TSK fuzzy model to learn jointly all parameters of the rules, especially the number of fuzzy rules can be identified without any prior expert experience. Let $\mathbf{X}$ be the input dataset and $\mathbf{Y}$ be the corresponding output set. The joint likelihood of both the data and all parameters of proposed model is $p(\mathbf{X}, K, \mathbf{U}, \mathbf{C}, \mathbf{Y}, \mathbf{V})$ which consists of four unknown parameters: the number of fuzzy rules K, the cluster center matrix $\mathbf{C}$, the fuzzy partition matrix $\mathbf{U}$ and the consequent parameter matrix $\mathbf{V}$. As the beginning point of this study, four factors in the joint likelihood $p(\mathbf{X},K,\mathbf{U},\mathbf{C},\mathbf{Y},\mathbf{V})$ are discussed, and then the objective function of the proposed fuzzy model is given. Based on the particle filter method, the joint learning method BTSK-JL uses the maximum-a-posterior (MAP) principle to find the optimal parameters by a set of random particles associated with weights. Fig.1 shows the joint learning framework of BTSK-JL.



$p(\mathbf{X}|K,\mathbf{U},\mathbf{C},\mathbf{Y},\mathbf{V})$, $p(\mathbf{U}|K,\mathbf{C},\mathbf{Y},\mathbf{V})$, $p(\mathbf{C},\mathbf{Y},\mathbf{V}|K)$, $p(K)$
$p(\mathbf{X},K,\mathbf{U},\mathbf{C},\mathbf{Y},\mathbf{V})$
Particle filter method
Dataset **X**, Output set **Y**
Sampling K^* fuzzy rules
Estimating **{U, C, V}** through logarithm of $p(\mathbf{X},K,\mathbf{U},\mathbf{C},\mathbf{Y},\mathbf{V})$
Computing MAP solution of $p(\mathbf{X},K,\mathbf{U},\mathbf{C},\mathbf{Y},\mathbf{V})$
Generating the optimal **{U*, C*, V*}**
Obtaining the optimal parameters **{** K*, **U*, C*,V*}**
Constructing the fuzzy rules and obtaining the output function

Fig.1. The proposed Bayesian TSK fuzzy model and its learning framework of BTSK-JL

A. The Proposed Bayesian TSK Fuzzy Model

The objective here is to consider both structure identification and parameter estimation in a joint learning model, which can establish a link between input and output space to improve performance and interpretability for a TSK fuzzy model. To achieve this, a joint likelihood of the data and all unknown parameters of the proposed Bayesian TSK fuzzy model is described as

$$p(\mathbf{X},K,\mathbf{U},\mathbf{C},\mathbf{Y},\mathbf{V}) = p(\mathbf{X}\,|\,K,\mathbf{U},\mathbf{C},\mathbf{Y},\mathbf{V})\,p(\mathbf{U}\,|\,K,\mathbf{C},\mathbf{Y},\mathbf{V})\,p(\mathbf{C},\mathbf{Y},\mathbf{V}\,|\,K)\,p(K)\ . \tag{5}$$

Obviously, four factors: $p(\mathbf{X}\,|\,K,\mathbf{U},\mathbf{C},\mathbf{Y},\mathbf{V})$, $p(\mathbf{U}\,|\,K,\mathbf{C},\mathbf{Y},\mathbf{V})$, $p(\mathbf{C},\mathbf{Y},\mathbf{V}\,|\,K)$ and $p(K)$ are involved in this joint likelihood. They will be discussed respectively, from the perspective of TSK fuzzy model.

Firstly, let us discuss how to appropriately define the conditional likelihood $p(\mathbf{X}\,|\,K,\mathbf{U},\mathbf{C},\mathbf{Y},\mathbf{V})$. The proposed Bayesian TSK fuzzy model adopts two ideas. Likewise in most of the existing fuzzy systems, the first idea is that a fuzzy rule geometrically corresponds to a cluster on input space. Thus $p(\mathbf{X}\,|\,K,\mathbf{U},\mathbf{C},\mathbf{Y},\mathbf{V})$ is independent of the label set $\mathbf{Y}$ and the consequent parameter matrix $\mathbf{V}$. Likewise in most of the existing probabilistic models, the second idea is that each sample $\mathbf{x}_n$ in $\mathbf{X}$ follows the independent and identical distribution, hence the conditional likelihood of $\mathbf{X}$ becomes $p(\mathbf{X}\,|\,K,\mathbf{U},\mathbf{C},\mathbf{Y},\mathbf{V}) = \prod_{n=1}^{N} p(\mathbf{x}_n\,|\,K,\mathbf{U},\mathbf{C})$. Since we can simply think that the data sample is generated simultaneously by the likelihood of K distributions which characterize fuzzy clustering, we can consider a product of the following K normal distributions on the sample $\mathbf{x}_n$:

$$\begin{aligned} p(\mathbf{x}_n\,|\,K,\mathbf{U},\mathbf{C}) &= \prod_{k=1}^{K} \mathcal{N}(\mathbf{x}_n\,|\,\boldsymbol{\mu}=\mathbf{c}_k,\ \boldsymbol{\Lambda}=\frac{K\mathbf{I}}{u_{nk}^{m}}) \\ &= \frac{1}{2\pi^{d/2}}\prod_{k=1}^{K}(\frac{u_{nk}^{m}}{K})^{d/2} exp(-\frac{1}{2K}u_{nk}^{m}\left\|\mathbf{x}_n-\mathbf{c}_k\right\|^2), \end{aligned} \tag{6}$$

where m is the fuzzy index. u_{nk} is the fuzzy membership of the nth sample $\mathbf{x}_n$ belonging to the kth cluster. c_{ki} is the mean of the ith component of the data samples in the kth cluster.

In summary, we can define the conditional likelihood $p(\mathbf{X}\,|\,K,\mathbf{U},\mathbf{C},\mathbf{Y},\mathbf{V})$ as

$$\begin{aligned} p(\mathbf{X}\,|\,K,\mathbf{U},\mathbf{C},\mathbf{Y},\mathbf{V}) &= \prod_{n=1}^{N} p(\mathbf{x}_n\,|\,K,\mathbf{U},\mathbf{C}) \\ &\propto \prod_{n=1}^{N}\prod_{k=1}^{K}(\frac{u_{nk}^{m}}{K})^{d/2}\times exp(-\frac{1}{2K}u_{nk}^{m}\left\|\mathbf{x}_n-\mathbf{c}_k\right\|^2). \end{aligned} \tag{7}$$

Secondly, let us observe the conditional likelihood $p(\mathbf{U}\,|\,K,\mathbf{C},\mathbf{Y},\mathbf{V})$. Each fuzzy membership vector $\mathbf{u}_n$ in the fuzzy partition matrix $\mathbf{U}$ is unique to each sample $\mathbf{x}_n$. Since fuzzy rules are generated by clustering on the input space, the fuzzy partition matrix $\mathbf{U}$ can be assumed to be only dependent on the cluster center matrix $\mathbf{C}$ and the number of clustering K. Thus we can reasonably express the conditional likelihood of $\mathbf{U}$ as $p(\mathbf{U}\,|\,K,\mathbf{C},\mathbf{Y},\mathbf{V}) = \prod_{n=1}^{N} p(\mathbf{u}_n\,|\,K,\mathbf{C})$. What is more, let us recall that Dirichlet distribution about membership can automatically express the positivity and sum-to-one constraint on the memberships [27]. Laplace distribution has high excess kurtosis at the mean point, which expects to obtain high value of u_{nk} for achieving compact fuzzy rules. Thus the Laplace distribution about membership can be used to promote the model sparsity. Therefore, we take both Dirichlet and Laplace distributions into the conditional likelihood of $p(\mathbf{u}_n\,|\,K,\mathbf{C})$ in the proposed Bayesian TSK fuzzy model $p(\mathbf{u}_n\,|\,K,\mathbf{C})$, i.e.,

$$p(\mathbf{u}_n\,|\,K,\mathbf{C}) = (\prod_{k=1}^{K}(\frac{u_{nk}^{m}}{K})^{-d/2})Dirichlet(\mathbf{u}_n\,|\,\boldsymbol{\alpha})^{1/K}\times\prod_{k=1}^{K}Laplace(u_{nk}\,|\,1,\frac{1}{\beta})^{\frac{1}{K}}, \tag{8}$$

where the first term $\prod_{k=1}^{K}(u_{nk}^{m}/K)^{-d/2}$ is a counter-balance to cancel the $\prod_{k=1}^{K}(u_{nk}^{m}/K)^{d/2}$ term in $p(\mathbf{X}\,|\,K,\mathbf{U},\mathbf{C},\mathbf{Y},\mathbf{V})$. The second term $Dirichlet(\mathbf{u}_n\,|\,\boldsymbol{\alpha})^{1/K}$ is a Dirichlet likelihood parameterized by vector $\boldsymbol{\alpha}=[\alpha_1,\alpha_2,...,\alpha_K]^T$, i.e.,

$$Dirichlet(\mathbf{u}_n\,|\,\boldsymbol{\alpha}) = \frac{1}{\mathbf{B}(\boldsymbol{\alpha})}\prod_{k=1}^{K}u_{nk}^{\alpha_k-1}\ , \tag{9}$$

where $\mathbf{B}(\boldsymbol{\alpha})$ is a normalizing constant. The third term $\prod_{k=1}^{K} Laplace(u_{nk} \mid 1, \frac{1}{\beta})^{\frac{1}{K}}$ on the fuzzy cluster memberships is a K-dimensional Laplace distribution, and results in

$$\prod_{k=1}^{K} Laplace(u_{nk} \mid 1, \frac{1}{\beta})^{\frac{1}{K}} = \prod_{k=1}^{K} (\beta / 2exp(-\beta \left| u_{nk} - 1 \right|))^{\frac{1}{K}} = \frac{\beta}{2} exp(-\beta) exp(\frac{\beta}{K}). \tag{10}$$

In summary, we can define the following conditional likelihood

$$p(\mathbf{U} \mid K, \mathbf{C}, \mathbf{Y}, \mathbf{V}) = \prod_{n=1}^{N} p(\mathbf{u}_n \mid K, \mathbf{C})$$
$$\propto \prod_{n=1}^{N} [(\prod_{k=1}^{K} (\frac{u_{nk}^{m}}{K})^{-d/2}) Dirichlet(\mathbf{u}_n \mid \boldsymbol{\alpha})^{1/K}] \times (K^{-Kd/2}) exp(\frac{\beta N}{K}) . \tag{11}$$

Thirdly, let us consider the conditional likelihood $p(\mathbf{C}, \mathbf{Y}, \mathbf{V} \mid K)$. Given the number of fuzzy rules, we can simply define this likelihood as $p(\mathbf{C}, \mathbf{Y}, \mathbf{V} \mid K) = \prod_{n=1}^{N} p(\mathbf{C}, \mathbf{y}_n, \mathbf{V} \mid K)$. With the local weighted least squares, referring to the discussions in [28], we can take $p(\mathbf{C}, \mathbf{y}_n, \mathbf{V} \mid K) \propto exp(-\frac{1}{2K} \sum_{k=1}^{K} \tilde{w}_{k,n} (y_n - \mathbf{v}_k^T \tilde{\mathbf{x}}_n)^2)$, and we can define the following conditional likelihood

$$p(\mathbf{C}, \mathbf{Y}, \mathbf{V} \mid K) \propto exp(-\frac{1}{2K} \sum_{n=1}^{N} \sum_{k=1}^{K} \tilde{w}_{k,n} (y_n - \mathbf{v}_k^T \tilde{\mathbf{x}}_n)^2) \ , \tag{12}$$

where $\tilde{w}_{k,n}$ is the weight to the kth rule for the sample $\mathbf{x}_n$, and $\tilde{w}_{k,n}$ is equal to the normalized fuzzy membership $\tilde{\mu}_k(\mathbf{x}_n)$. The consequent parameter $\mathbf{v}_k$ can be computed as

$$\mathbf{v}_k = (\mathbf{X}_e^T \mathbf{D}_k \mathbf{X}_e)^{-1} \mathbf{X}_e^T \mathbf{D}_k \mathbf{Y}, \tag{13}$$

where $\mathbf{X}_e = [\tilde{\mathbf{x}}_1, \tilde{\mathbf{x}}_2, ..., \tilde{\mathbf{x}}_N]^T$, $\mathbf{D}_k = diag(\tilde{\mathbf{w}}_k)$, $\tilde{\mathbf{w}}_k = [\tilde{\mu}_k(\mathbf{x}_1), \tilde{\mu}_k(\mathbf{x}_2), ..., \tilde{\mu}_k(\mathbf{x}_N)]$. In order to compute $\tilde{\mu}_k(\mathbf{x}_n)$, the width parameter δ_{ki} in Gaussian membership function (2) can be given as follows [16]

$$\delta_{ki}^2 = \sum_{\mathbf{x} \in \mathbf{c}_k} (x_{ji} - c_{ki})^2 / N_k , i = 1, ..., d, \ k = 1, ..., K, \tag{14}$$

where N_k is the size of the kth cluster, and c_{ki} is the mean of the ith component of the data samples in the kth cluster.

Finally, let us define the fourth factor in (5). Since the number of fuzzy rules is a positive integer, thus its distribution should be a discrete distribution. Here we take the prior distribution $p(K)$ on the number K of fuzzy rules as the following Poisson distribution with the shape parameter λ .

$$p(K) = Poisson(K, \lambda) = \lambda^K exp(-\lambda) / K! . \tag{15}$$

Following [29], λ is set to be $\lambda = logN$. In addition, the number of fuzzy rules can be sampled from different distributions according to practical requirements. For example, we may assume it follows the discrete uniform distribution $p(K) = discrete(K) = 1 / l$, where $K_{min} \le K \le K_{max}$, l is the number of optional rules between K_{min} and K_{max} . If the discrete uniform distribution is adopted, the number of fuzzy rules can be selected from a given search grid.

After the above discussions, by multiplying (7), (11), (12) and (15) together, we can represent the joint likelihood of the data and all relevant parameters in the proposed Bayesian TSK fuzzy model as

$$p(\mathbf{X}, K, \mathbf{U}, \mathbf{C}, \mathbf{Y}, \mathbf{V}) = p(\mathbf{X} \mid K, \mathbf{U}, \mathbf{C}, \mathbf{Y}, \mathbf{V}) p(\mathbf{U} \mid K, \mathbf{C}, \mathbf{Y}, \mathbf{V}) p(\mathbf{C}, \mathbf{V}, \mathbf{Y} \mid K) p(K)$$
$$\propto exp(-\frac{1}{2K} \sum_{n=1}^{N} \sum_{k=1}^{K} u_{nk}^{m} \| \mathbf{x}_n - \mathbf{c}_k \|^2) \times \left(\prod_{n=1}^{N} \prod_{k=1}^{K} (u_{nk}^{\alpha_k - 1})^{1/K} \right) \times exp(\frac{\beta N}{K})$$
$$\times exp(-\frac{1}{2K} \sum_{n=1}^{N} \sum_{k=1}^{K} \tilde{w}_{k,n} (y_n - \mathbf{v}_k^T \tilde{\mathbf{x}}_n)^2) \times \frac{\lambda^K}{K!} exp(-\lambda). \tag{16}$$

Therefore, the objective function of the proposed Bayesian TSK fuzzy model can be obtained as the logarithm of (16) as

$$J = -\frac{1}{2K} \sum_{n=1}^{N} \sum_{k=1}^{K} u_{nk}^{m} \| \mathbf{x}_n - \mathbf{c}_k \|^2 + \frac{1}{K} \left(\sum_{n=1}^{N} \sum_{k=1}^{K} (\alpha_k - 1) log u_{nk} \right) + \frac{\beta N}{K}$$
$$- \frac{1}{2K} \sum_{n=1}^{N} \sum_{k=1}^{K} \tilde{w}_{k,n} (y_n - \mathbf{v}_k^T \tilde{\mathbf{x}}_n)^2 + K log \lambda - \sum_{k=1}^{K} log k. \tag{17}$$

Please note, since λ is a given constant, i.e., $\lambda = logN$, we omit $log(exp(-\lambda))$ in (17). For the choice of β , our experiments suggest that it may be appropriately selected from $\{1, 2, ..., 8\}$.

Obviously, the negative value of the first term in (17) is the objective function of the classical fuzzy C-means clustering algorithm FCM [30], which keeps the same as the fact that most of the existing fuzzy systems use FCM or its variants to determine the antecedent parts of fuzzy rules. The negative value of the fourth term in (17) is essentially the local weighted least squares, which is similar to the commonly global least squares $\sum_{n=1}^{N} (y_n - \mathbf{v}_k^T \tilde{\mathbf{x}}_n)^2$ but much more efficient in reducing overfitting [31]. Therefore, (17) indeed provides a new generalized objective function of training a TSK fuzzy system, and its natural benefit exists in the fact that the number of fuzzy rules, both antecedent and consequent parts of fuzzy rules are jointly considered. In contrast to most of existing TSK fuzzy systems in which the consequent parameters are learned just after determination of the structure, we can easily observe from (17) that both the structure identification and parameter estimation are mutually dependent and closely related in the proposed Bayesian TSK fuzzy model. In such a new fuzzy model, the natural link between the input and output space can be well characterized. Moreover, all unknown parameters {K, $\mathbf{U}$, $\mathbf{C}$, $\mathbf{V}$} in the proposed Bayesian fuzzy model can be optimized simultaneously when the objective function value of (17) tends toward the maximization.

B. Joint Learning of Structure Identification and Parameter Estimation

For implementing the objective function in (17), we use the particle filter method [32] to find the MAP parameters within a maximum likelihood context. As a type of Sequential Monte Carlo, a particle filter can be used to deal with non-Gaussian densities representing models having uncertain parameters represented by arbitrary probability density functions. Besides, a particle filter is very noise tolerant for the underlying state

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TII.2018.2813977, IEEE Transactions on Industrial Informatics

estimation process and it is not very sensitive to the initial values of the states [33]. Based on the MAP principle, the particle filter for BTSK-JL approximates the optimal solution of parameters $\{K, \mathbf{U}, \mathbf{C}, \mathbf{V}\}$ by generating a set of weighted discrete particles $WDP[r] = \{\{K, \mathbf{U}, \mathbf{C}, \mathbf{V}\}, ll\}^T$ (r=1, 2,…, P), where P is the number of particles, and ll is the objective function value of the particle obtained by (17). The particle filter for BTSK-JL operates by executing the iterations of the following steps:

(1) Parameter sampling step. In this step, a random new rule number K^* is drawn from a Poisson distribution with a mean value of the particle's current rule number by (15). If the value of K^* is smaller than the current number K, then K^* fuzzy cluster centers are randomly selected from the cluster center matrix **C**. On the other hand, if the value of K^* is greater than K, then the new $(K^* - K)$ cluster centers are sampled from a d-dimensional Laplace distribution [34] given as

$$\mathbf{c}_k \sim \prod_{i=1}^{d} Laplace(e_i, \gamma) = \prod_{i=1}^{d} \frac{1}{2\gamma} exp(-\frac{|e_i|}{\gamma}), \quad (18)$$

where the location parameter $\mathbf{e} = [e_1, e_2, ..., e_d]^T$ is the mean of the data **X**, and the scale parameter γ determines the width of the distribution. Based on our extensive experiments, the value of γ is set to be $\gamma = 5$ in this study.

(2) Parameter estimation step. The parameter estimation procedure consists of three sub-steps: estimation of the fuzzy partition matrix **U**, estimation of the cluster center matrix **C** and estimation of the consequent parameter matrix **V**.

a) Estimation of **U**. With the change of the number of fuzzy rules, the values of fuzzy memberships also change accordingly. Because there is no prior knowledge on the membership values, here the flat Dirichlet distribution [35] should be used. In this case, the prior on the membership values is uninformative, and each element in $\boldsymbol{\alpha}$ is set to be one. Considering Dirichlet's characteristic of the positivity and sum-to-one constraints on u_{nk}, the optimal value of u_{nk} can be computed by setting the Lagrangian of (17) with the equality constraint $\sum_{k=1}^{K} u_{nk} = 1$, i.e., $-\frac{1}{2K}\sum_{n=1}^{N}\sum_{k=1}^{K} u_{nk}^m \| \mathbf{x}_n - \mathbf{c}_k \|^2 + \frac{\beta N}{K} - \frac{1}{2K}\sum_{n=1}^{N}\sum_{k=1}^{K} \tilde{w}_{k,n}(y_n - \mathbf{v}_k^T \tilde{\mathbf{x}}_n)^2 + Klog\lambda - \sum_{k=1}^{K} logk$ $+\sum_{n=1}^{N} \eta_n (\sum_{k=1}^{K} u_{nk} - 1)$, where η_n is the Lagrange multiplier. By setting the Lagrangian's gradient to zero, u_{nk} $(1 \le n \le N, 1 \le k \le K)$ can be optimized with the following update equation

$$u_{nk} = \left\| \mathbf{x}_n - \mathbf{c}_k \right\|^{\frac{2}{1-m}} / \sum_{k=1}^{K} \left\| \mathbf{x}_n - \mathbf{c}_k \right\|^{\frac{2}{1-m}} \ . \quad (19)$$

Then the current width matrix $\boldsymbol{\delta}$ in the membership function and the current consequent parameter matrix **V** can be computed by (14) and (13), respectively.

b) Estimation of **C**. When the number of clusters K and the fuzzy partition matrix **U** are fixed, while using the current consequent parameter matrix **V**, the optimal value of c_{ki} in the center $\mathbf{c}_k$ can be computed by taking the derivative of (17) with respect to c_{ki}, then c_{ki} can be optimized,

$$c_{ki} = \frac{\dfrac{\sum_{n=1}^{N} u_{nk}^m x_{ni}}{K} + 2\sum_{n=1}^{N}\sum_{k=1}^{K}(y_n - \mathbf{v}_k^T \tilde{\mathbf{x}}_n)^2 \times \dfrac{\sum_{k^{'}=1}^{K} \mu_{k^{'}}(\mathbf{x}_n) - \mu_k(\mathbf{x}_n)}{(\sum_{k^{'}=1}^{K} \mu_{k^{'}}(\mathbf{x}_n))^2} \times \mu_k(\mathbf{x}_n) \times \dfrac{x_{ni}}{\delta_{ki}}}{\dfrac{\sum_{n=1}^{N} u_{nk}^m}{K} + 2\sum_{n=1}^{N}\sum_{k=1}^{K}(y_n - \mathbf{v}_k^T \tilde{\mathbf{x}}_n)^2 \times \dfrac{\sum_{k^{'}=1}^{K} \mu_{k^{'}}(\mathbf{x}_n) - \mu_k(\mathbf{x}_n)}{(\sum_{k^{'}=1}^{K} \mu_{k^{'}}(\mathbf{x}_n))^2} \times \mu_k(\mathbf{x}_n) \times \dfrac{1}{\delta_{ki}}}$$

(20)

c) Estimation of **V**. After tuning the fuzzy partition matrix **U** and the cluster center matrix **C**, the optimal value of consequent parameter $\mathbf{v}_k$ can be estimated by taking the derivative of (17) with respect to $\mathbf{v}_k$. Then $\mathbf{v}_k$ can be optimized with the same equation as in (13).

d) Computation of ll. According to (18)-(20), we can obtain the optimal parameters {K, **C**, **U**, **V**} for each particle. The objective function value ll of each particle can be computed by (17).

(3) Particle updating step. If $WDP[r]$ (r=1, 2,…, P) improves the MAP value for current K fuzzy rules through (17), it can be considered as a candidate for the MAP solution, i.e.,

$$CAND[K] = \begin{cases} WDP[r], \ if \ \ WDP[r].ll > CAND[K].ll, \\ CAND[K], otherwise, \end{cases} \quad (21)$$

where $CAND$ is the set of candidate particles for the MAP solution, and its element $CAND[K]$ is the candidate particle for the K fuzzy rules. Then a set of particles called PS, whose elements contain the potential MAP solutions, are constructed

PS={WDP, $CAND$}. (22)

(4) Weighting and resampling step. Weighting and resampling step can reduce the effects of degeneracy of particles. The resampling process is done with replacement, and particles with large weights are likely to be duplicated multiple copies and particles with very small weights may be always discarded. The normalized importance weight w_i of each particle in PS can be computed as follows

$$w_i = exp(PS[i].ll) / \sum_{i=1}^{|PS|} exp(PS[i].ll) \ . \quad (23)$$

where $|PS|$ is the number of particles in PS. In this step, a set of P particles with big weights is drawn from the set PS in which the probability of resampling of each particle is proportional to the value of w_i.

After t_{max} iterations, the parameters {K, **U**, **C**, **V**} corresponding to the largest value ll in $CAND$ will be taken to form the fuzzy rules and thus the output function of the proposed fuzzy model can be accordingly determined by (3). The joint learning method BTSK-JL for the proposed Bayesian TSK fuzzy model is summarized below.

Here, let us give a remark about algorithm 1 (i.e., BTSK-JL). Different from the standard weighting and resampling step in the particle filter [32] where only the particle set WDP with the fixed number of particles is involved, algorithm 1 uses the particle set *PS* in its weighting and resampling step. *PS* consists of two parts. One is the particle set *WDP* which contains the sampling particles generated in each iteration, and the other is the particle set *CAND* which consists of the particles corresponding to the optimal objective function values, denoted as *ll* in algorithm 1, for different numbers of rules. The role of *CAND* is to speed up the resampling convergence. Our extensive experiments indicate that this strategy can be helpful in reducing the number of iterations greatly in algorithm 1.

Algorithm 1: Joint learning method BTSK-JL for the proposed Bayesian TSK fuzzy model.

```
//Initialization
1) Create particle set WDP, set particle WDP[1] in which $WDP[1].K=1$, $WDP[1].\mathbf{c}_1$ by (18), $WDP[1].u_{i1}$ = 1 ($i$ =1,2, …, $N$), and compute WDP[1].ll by (17); Set each particle WDP[r] with sample WDP[1] ($r$=2, 3,…, $P$).
2) Create particle set CAND, set particle CAND[1]= WDP[1];
3) Set the iterative indexes $t$=1, $r$=1;
//Structure identification and parameter estimation
Repeat $t=t+1$;
{ { Repeat $r=r+1$;
  1) Sample new number of clusters $K^*$ by (15):
  If $K^* \le WDP[r].K$, randomly select $K^*$ centers from $WDP[r].\mathbf{C}$, then update $WDP[r].\mathbf{C}$,
  If $K^* > WDP[r].K$, sample new $\mathbf{c}_k (k=WDP[r].K+1, WDP[r].K+2,..., K^*)$ centers by (15), then update $WDP[r].\mathbf{C}$ ;
  2) Update $WDP[r].K= K^*$ ;
  3) Estimate $WDP[r].\mathbf{U}$ by (19);
  4) Estimate $WDP[r].\mathbf{C}$ by (20);
  5) Estimate $WDP[r].\mathbf{V}$ by (13);
  6) Compute $WDP[r].ll$ by (17);
  7) Obtain the candidate particle set CAND by (21);
  8) Construct the particle set PS={WDP, CAND};
  Until $r > P$}
  9) Compute each particle weight $w_i$ in PS by (23);
10) Sample P particles in PS according to the sampling probability being equal
    to $w_i$, then update WDP with these P particles;
Until $t \ge t_{max}$ or
  $count(|max\{PS[r_1].ll\}_t - max\{PS[r_2].ll\}_{t-1}| < \varepsilon) > miter \ (r_1, r_2 = 1, 2, ..., |PS|)$ };
11) Select the particle with the largest value ll in CAND, pick its $K$, $\mathbf{U}$, $\mathbf{C}$ and $\mathbf{V}$;
//Construction the fuzzy rules
Use parameters {$K$, $\mathbf{U}$, $\mathbf{C}$, $\mathbf{V}$} to construct the fuzzy rules by (1), and obtain the output function by (3).
```

C. Time Complexity and Convergence

As for the time complexity of the joint learning method BTSK-JL, it mainly lies in parameter estimation step, i.e., estimating the parameters {**U**, **C**, **V**}. The complexity of computing each element of **U** by using (19) is $O(NK)$. The complexity of computing each element of $\mathbf{c}_k$ by using (20) is $O(NK^2(d+1))$. The time complexity is $O(N(d+1)^2)$ to solve the weighted least squares problem for each rule [28]. Thus estimating the consequent parameters of a fuzzy system composed of *K* fuzzy rules will require $O(KN(d+1)^2)$ operations. Taking into account these three parameters in a single outer iteration, the time complexity of the joint learning is $O(P(NK+KN(d+1)(K+d+1)))$, where *N*, *K*, *d* and *P* are the number of training samples, the number of fuzzy rules, and the dimension of samples and the number of particles, respectively. Obviously, since the time complexity is dependent on both N^2 and d^2, BTSK-JL will become very time-consuming and even be impractical for high-dimensional and/or large-scale datasets. In other words, how to speed up BTSK-JL for high-dimensional and/or large-scale datasets is worthy to be studied in the future.

Here we briefly discuss the convergence of BTSK-JL. According to [36], a particle filter method will converge to the correct distribution when the number of particles tends to infinity. Therefore, with a limited number of particles in practice, we should make a tradeoff between the convergence speed and performance of a particle filter method. In this study, we set a given threshold ε in the termination conditions. That is to say, algorithm 1 records the occurrence frequency of the condition in which the difference between the optimal objective function values in *PS* at current iteration and the last iteration is less than ε. When the occurrence frequency is more than the other threshold $miter$, BTSK-JL will terminate. Otherwise, BTSK-JL will run until the maximal number of iterations is achieved. With the execution of algorithm 1, the number of rules and the obtained cluster centers will not change a lot, hence the output errors computed by the local weighted least squares strategy on all training samples become almost unchanged. In this case, the performance of the proposed Bayesian fuzzy model can not be improved. In other words, as a random search of the particle filter, BTSK-JL takes the particle with the biggest value of the objective function as the final output, it can guarantee to a local optimal solution [37].

III. Experimental Results

We conduct experiments using the proposed joint learning method BTSK-JL on four time series datasets and a real dataset about glutamic acid fermentation process in comparison with four state-of-the-art algorithms. In subsection III-A, the experimental settings are described. In subsection III-B, the experimental results on the time series datasets are reported. In subsection III-C, a case study about glutamic acid fermentation process is given.

A. Experimental Settings

For comparison purpose, four state-of-the-art TSK related regression algorithms are applied to compare against our proposed joint learning method BTSK-JL. They are TSK-IRL-R [14], MOGUL-TSK-R [31], L2-TSK-FS [15] and B-ZTSK-FS [21]. The descriptions of these algorithms are shown in Table I, and their same parameters settings and the default search grids are taken from their respective references, as listed in Table II.

In our experiments, we consider a fivefold cross-validation strategy, i.e., we randomly split each dataset into five equal partitions (each one with 20% samples) and then take all possible combinations of four partitions as the training set and the remaining one as the testing set. In order to evaluate the performance of the algorithms, both the mean squared error (*MSE*) and the average number of fuzzy rules for five runs are

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TII.2018.2813977, IEEE Transactions on Industrial Informatics

recorded. The *MSE* is computed as $MSE=\sum_{i=1}^{M}(y_i-f(\mathbf{x}_i))^2/M$, where M is the number of test samples, y_i and $f(\mathbf{x}_i)$ is the original output and actual output obtained by the corresponding regression algorithms, respectively. In our experiments, L2-TSK-FS, B-ZTSK-FS, and BTSK-JL are implemented by MATLAB; while TSK-IRL-R and MOGUL-R implemented by JAVA are taken from KEEL software toolbox [38]. Moreover, the environment in the experiments is a computer with Intel Core i5-3317U 1.70GHz CPU and 8 GB RAM.

TABLE I
ALGORITHMS ADOPTED FOR PERFORMANCE COMPARISON

Algorithms	Descriptions
TSK-IRL-R [14]	The TSK fuzzy system by using a two-stage evolutionary process and the iterative rule learning methodology.
MOGUL-R [31]	The TSK fuzzy system by using the local evolutionary learning and MOGUL method.
L2-TSK-FS [15]	The TSK fuzzy system by using FCM for antecedent parameters, and using L2 norm penalty and ε -insensitive criterion for consequent parameters.
B-ZTSK-FS [21]	The zero-order TSK fuzzy system by using a Metropolis-Hastings (MH) sampling method to simultaneously learn antecedent/consequent parameters.
BTSK-JL	The proposed Bayesian TSK fuzzy system in this study.

TABLE II
PARAMETER SETTINGS AND SEARCH GRIDS FOR CROSS-VALIDATION

Algorithms	Descriptions
MOGUL-R	Evolutionary strategy application=1, type of niches=1, evolutionary strategy=100, matching degree of the positive samples parameter =0.05, the percentage of allowed negative samples=1.5, the minimum matching degree in the fitness function=0.1, the number of labels =5, the number of parents for the evolutionary strategy mogul=15, the size of the standard deviation string mogul=6.
TSK-IRL-R	Evolutionary strategy iterations=500, the number of parents for the evolutionary strategy =15, the number of parents for the evolutionary strategy=100, recombination operation for the solution string=3, recombination operation for the deviation string=2, the number of parents to recombine the solution string and deviation string =15, the number of parents to recombine the angle string=1, the matching degree of the positive samples =0.05, the percentage of allowed negative samples=0.1, the number of labels =5, the population size tuning=61, the tuning parameter *a* =0.35, the tuning parameter *b* =5, the cross probability per individual tuning=0.1.
L2-TSK-FS	The number of rules $K\in\{2^2,3^2,\dots,11^2\}$, the scale parameter $h\in\{0.2^2,0.4^2,\dots,2^2\}$, the fuzzy index *m*=2, the regularization parameter $C\in\{2^{-4},2^0,\dots,2^7\}$.
B-ZTSK-FS	The number of rules $K\in\{2^2,3^2,\dots,11^2\}$, the scale parameter $h\in\{0.2^2,0.4^2,\dots,2^2\}$, the fuzzy index *m*=2, the maximum number of iterations = 1000 in MH sampling, the Dirichlet index=$\mathbf{1_K}$.
BTSK-JL	The fuzzy index *m*=2, the threshold ε =10^{-3}, the convergence threshold *miter* =50, the maximum number of iterations = 500, the model sparsity parameter $\beta\in\{1,2,\dots,8\}$, the number of particles *P*=10.

B. Time Series Problems

In this subsection, the performance of the proposed joint learning method BTSK-JL is evaluated on the following four time series tasks:

(1) Chaotic time series dataset (called *chaotic* for simplicity) [20]: This dataset consists of 1000 samples, which is from the Mackey-Glass chaotic time series from the lowing delay differential equation: $\frac{dx(t)}{dt}=\frac{0.2x(t-\tau)}{1+x^{10}(t-\tau)}-0.1x(t)$, where $\tau>17$. Following [20], τ is set to be $\tau=30$, and given nine past values $x(k-8),x(k-7),\ ...,x(k)$, predict the value of $x(k+1)$ at the next time.

(2) Laser time series (called *laser* for simplicity) [31]: This time series is generated by a far-infrared laser in a chaotic state. In accordance with [31], five past values $x(k-4),\ ...,x(k)$ is used to predict the value of $x(k+1)$ at the next time. Fig. 2 shows the 995 points of the laser time series (black line) that are taken as the dataset used in the experiment.

(3) Neural network forecasting competition 5 (called *nn5* for simplicity) [39]: This dataset consists of 782 samples with five features, and it includes daily time series of different ATMs measured over two years.

(4) Earth telescope observations data (called *edat* for simplicity) [20]: This dataset consists of 1665 samples with five features, and it includes a time series of light curve of the white dwarf star PG1159-035 during March 1989.

TABLE III
PERFORMANCE OF FIVE ALGORITHMS ON FOUR TIME SERIES

chaotic	TSK-IRL-R			MOGUL-R		
	MSE (std)	Rules	Running time (std)	*MSE* (std)	Rules	Running time (std)
	2.7412E-04 (2.168E-06)	716.8	---	2.5632E-04 (2.076E-06)	39.0	---
	L2-TSK-FS			B-ZTSK-FS		
	MSE (std)	Rules	Running time (std)	*MSE* (std)	Rules	Running time (std)
	2.2471E-04 (3.006E-06)	9.0	10.43 (0.43)	2.2451E-04 (2.715E-06)	4.0	18.74 (1.04)
	BTSK-JL					
	MSE (std)	Rules	Running time (std)			
	2.0529E-04 (2.548E-06)	**2.4**	13.06 (0.69)			
laser	TSK-IRL-R			MOGUL-R		
	MSE (std)	Rules	Running time (std)	*MSE* (std)	Rules	Running time (std)
	24.6532 (2.033)	221.6	---	27.4323 (1.158)	90.4	---
	L2-TSK-FS			B-ZTSK-FS		
	MSE (std)	Rules	Running time (std)	*MSE* (std)	Rules	Running time (std)
	20.6643 (1.654)	9.0	39.16 (0.78)	18.8678 (1.729)	9.0	52.37 (1.86)
	BTSK-JL					
	MSE (std)	Rules	Running time (std)			
	16.6325 (1.028)	**5.0**	47.84 (1.55)			
nn5	TSK-IRL-R			MOGUL-R		
	MSE (std)	Rules	Running time (std)	*MSE* (std)	Rules	Running time (std)
	68.8432 (4.912)	410.6	---	69.2772 (3.892)	111.8	---
	L2-TSK-FS			B-ZTSK-FS		
	MSE (std)	Rules	Running time (std)	*MSE* (std)	Rules	Running time (std)
	70.7643 (5.462)	36.0	50.39 (0.94)	69.6995 (5.007)	36.0	75.38 (2.59)
	BTSK-JL					
	MSE (std)	Rules	Running time (std)			
	69.4523 (4.745)	**23.0**	57.58 (1.24)			
edat	TSK-IRL-R			MOGUL-R		
	MSE (std)	Rules	Running time (std)	*MSE* (std)	Rules	Running time (std)
	3.5024E-03 (3.130E-04)	285.2	---	3.1492E-03 (3.595E-04)	53.6	---
	L2-TSK-FS			B-ZTSK-FS		
	MSE (std)	Rules	Running time (std)	*MSE* (std)	Rules	Running time (std)
	3.3675E-03 (3.132E-04)	49.0	104.60 (3.09)	3.3571E-03 (3.213E-04)	36.0	169.32 (4.82)
	BTSK-JL					
	MSE (std)	Rules	Running time (std)			
	3.2593E-03 (3.154E-04)	**23.6**	128.89 (3.57)			

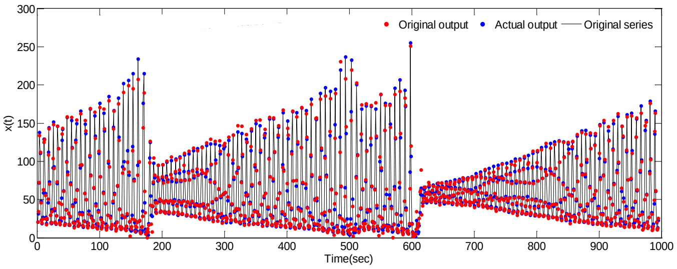


Fig.2. Outputs of BTSK-JL and the actual outputs for the series *laser*

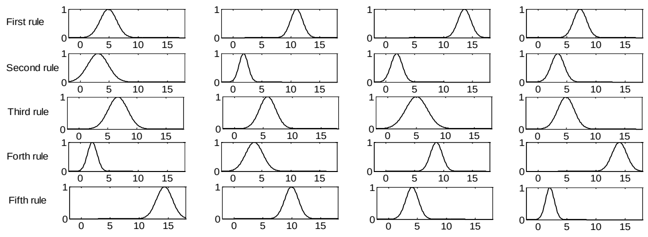


Fig.3 Fuzzy membership functions in the antecedent parts of five fuzzy rules obtained by BTSK-JL on *laser*

Table III shows the average numbers of rules, the average *MSE* results with the standard deviations, and the average running time (containing training time and testing time in seconds) with the standard deviations obtained by different algorithms on four time-series datasets. As pointed out in the above, both TSK-IRL-R and MOGUL-R are implemented by JAVA rather than MATLAB, so we do not report their running time (denoted as "---") in Table III.

These results demonstrate that:

a) In the sense of the *MSE* index, BTSK-JL performs better than the other algorithms on three time-series datasets. Only for time-series *nn5*, the performance of BTSK-JL is a little worse. Comparing with TSK-IRL-R, L2-TSK-FS and B-ZTSK-F and MOGUL-R, BTSK-JL is always more effective in terms of the *MSE* index. We can infer that revealing the inherent relation between structure identification and parameter estimation has a favorable effect on the predictive performance of a TSK fuzzy model. Meanwhile, as for L2-TSK-FS and B-ZTSK-FS, it is not an easy work to select a reasonable search grid for fuzzy rules. In order to better observe how BTSK-JL behaves, as an example, Fig.2 depicts the actual outputs of BTSK-JL (blue dots) and original outputs (red dots) on series *laser*. Obviously, the actual outputs generated by BTSK-JL are quite close to the original outputs.

b) In terms of the average number of fuzzy rules, BTSK-JL achieves less number of fuzzy rules compared to other four rule based algorithms on four time-series datasets. Let us keep in mind that the number of rules can heavily affect the interpretability of a TSK fuzzy system. We can see that the joint learning mechanism in BTSK-JL is of a great help to find a compact set of fuzzy rules with satisfactory performance. Fig.3 graphically presents its corresponding fuzzy membership functions in the antecedent parts of the fuzzy rules in a certain run. Since BTSK-JL obtains a compact set of fuzzy rules with high interpretability, it is very applicable for time-series prediction.

c) In terms of the average running time, BTSK-JL is slower than L2-TSK-FS on four time series datasets, while BTSK-JL is faster than B-ZTSK-FS. It is because the training problem of L2-TSK-FS does not need an iterative sampling procedure and it can be trained by taking an efficient quadratic programming (QP) solver [15]. B-ZTSK-FS simultaneously learns the antecedent and consequent parameters by taking the MH sampling technique. However in the iteration process of MH sampling, each fuzzy cluster membership for *N* data samples and each cluster center are selected from the given distributions respectively, this learning strategy for B-ZTSK-FS tends to accompany with slow convergence.

Table IV is arranged to report how only a free parameter, i.e., model sparsity parameter β, affects the average results of BTSK-JL about the number of fuzzy rules, *MSE* and the number of iterations (i.e., convergence speed) on four time series datasets. According to Table IV, a smaller β tends to obtain more fuzzy rules through more thorough exploration of the search space for rules with more iteration. On the contrary, a larger β tends to obtain a smaller number of rules by exploring a more compact search space for rules with less iteration. What is more, after considering the trade-off between performance and interpretability, we can see from Table IV that $\beta \in \{1, 2, ...,8\}$ may be an appropriate choice, which is in accordance with the suggestion below (17).

TABLE IV

AVERAGE RESULTS ON THE NUMBER OF FUZZY RULES, MSE, AND THE NUMBER OF ITERATIONS WITH DIFFERENT β VALUES ON FOUR TIME SERIES

Datasets		$\beta=1$	$\beta=2$	$\beta=3$	$\beta=4$	$\beta=5$	$\beta=6$	$\beta=7$	$\beta=8$
chaotic	*MSE*	3.5643E-04	3.6753E-04	3.0124E-04	3.7002E-04	2.8235E-04	2.0529E-04	2.0759E-04	2.532E-04
	Rules	12.2	10.0	7.8	6.0	4.6	2.4	2.4	2.2
	Iterations	98.8	80.4	74.8	68.6	64.4	63.4	62.2	60.8
laser	*MSE*	18.0467	17.7285	16.8278	16.6325	17.1864	17.1936	17.6743	17.6711
	Rules	14.2	10.8	7.4	5.0	4.8	4.8	4.6	4.5
	Iterations	196.4	165.8	146.0	120.8	112.8	110.8	108.6	106.8
nn5	*MSE*	73.4876	73.5512	69.4523	70.6802	70.9823	83.8963	83.9027	83.9216
	Rules	39.8	37.6	23.0	22.2	20.6	16.8	16.8	16.2
	Iterations	212.8	206.4	180.4	178.0	169.6	165.8	166.4	165.2
edat	*MSE*	3.7657E-03	3.7606E-03	3.6435E-03	3.5245E-03	3.2593E-03	3.7578E-03	3.9244E-03	4.0864E-03
	Rules	58.2	44.0	36.0	28.6	23.6	22.6	21.8	21.2
	Iterations	198.8	186.8	180.6	172.4	170.2	165.6	164.0	165.6

C. Glutamic Acid Fermentation Process

Glutamic acid fermentation process is a typical intermittence industry process which is a highly nonlinear of complex dynamic batch process. The dataset is established based on the glutamic acid fermentation process [15]. The input variables of the dataset include 6 features: fermentation time *k*, glucose concentration *S*(*k*), thalli concentration *X*(*k*), glutamic acid concentration *P*(*k*), stirring speed *R*(*k*), and ventilation *Q*(*k*) at time *k*. The output variables are glucose concentration *S*(*k* +2), thalli concentration *X*(*k*+2), and glutamic acid concentration *P*(*k*+2) at a future time *k*+2. The data in this experiment are collected from a 160 batch-fermentation process and each batch containing 14 effective samples, i.e., 2240 samples in the dataset. In this study, three biochemical process prediction models containing 6 features are constructed respectively for three outputs: *S*(*k*+2), *X*(*k*+2) and *P*(*k*+2). With the same experimental settings as in the last subsection, the average number of fuzzy rules, the average *MSE* results with the standard deviations and the average running time (in seconds) with the standard deviations for glutamic acid fermentation process are recorded in Table V.

It is noticeable that BTSK-JL achieves the best *MSE* among all the five algorithms. While BTSK-JL uses less number of fuzzy rules to achieve satisfactory predictive performance, resulting in high interpretability from the perspective of the number of fuzzy rules. Both TSK-IRL-R and L2-TSK-FS need more fuzzy rules to obtain good results in terms of the *MSE* index. Although B-ZTSK-FS simultaneously learns antecedent and consequent parameters, it does not establish a relationship between structure identification and parameter learning for a TSK fuzzy system. That is why B-ZTSK-FS requires more fuzzy rules but obtains worse performance than BTSK-JL. MOGUL-R as a joint learning model performs a local identification of prototypes to obtain a set of initial rules and then a genetic tuning process to refine the fuzzy model. However, MOGUL-R obtains much more fuzzy rules for this glutamic acid fermentation process.

As for the running time, the same observation as in the last subsection can be seen from Table V. Unlike L2-TSK-FS involves three free parameters to be determined using the grid search strategy, BTSK-JL only involves a free parameter and hence it becomes very applicable to industry process.

IV. Conclusions

In this study, a Bayesian probabilistic model is constructed for the TSK fuzzy model, in which fuzziness and probability can work well for fuzzy regression tasks in a collaborative manner. With the help of the particle filter method together with the MAP principle, a joint leaning method is designed for simultaneous learning of the number of rules and the antecedent/consequent parameters of the fuzzy rules. Since the proposed joint learning method BTSK-JL can capture the subtle interaction between input and output spaces, our experimental results indicate that it has promising predictive performance with a compact set of interpretable fuzzy rules, in contrast to other four regression algorithms.

The future works can be focused from two aspects. One is how to reduce the time complexity of BTSK-JL such that it is suitable for high dimensional and/or large scale datasets. The other is how to extend the idea of BTSK-JL for classification tasks.

TABLE V
PERFORMANCE OF FIVE ALGORITHMS FOR GLUTAMIC ACID FERMENTATION PROCESS

S(*k* +2)	TSK-IRL-R			MOGUL-R		
	MSE (std)	Rules	Running time (std)	*MSE* (std)	Rules	Running time (std)
	0.2654 (0.011)	228.8	---	0.7312 (0.099)	40.6	---
	L2-TSK-FS			B-ZTSK-FS		
	MSE (std)	Rules	Running time (std)	*MSE*(std)	Rules	Running time (std)
	0.3479 (0.057)	16.0	7.45 (0.12)	0.2671 (0.017)	9.0	14.65 (1.99)
	BTSK-JL					
	MSE (std)	Rules	Running time (std)			
	0.2311 (0.009)	**6.4**	10.92 (0.73)			
X(*k*+2)	TSK-IRL-R			MOGUL-R		
	MSE (std)	Rules	Running time (std)	*MSE* (std)	Rules	Running time (std)
	3.6597E-03 (3.019E-05)	224.6	---	5.0922E-03 (3.852E-05)	44.0	---
	L2-TSK-FS			B-ZTSK-FS		
	MSE (std)	Rules	Running time (std)	*MSE* (std)	Rules	Running time (std)
	4.0455E-03 (3.363E-05)	16.0	7.99 (0.08)	3.5537E-03 (3.227E-05)	9.0	13.96 (0.94)
	BTSK-JL					
	MSE (std)	Rules	Running time (std)			
	2.2568E-03 (2.026E-04)	**6.2**	11.58 (0.72)			
P(*k*+2)	TSK-IRL-R			MOGUL-R		
	MSE (std)	Rules	Running time (std)	*MSE* (std)	Rules	Running time (std)
	0.0843 (3.082E-03)	221.8	---	0.0917 (3.230E-03)	40.4	---
	L2-TSK-FS			B-ZTSK-FS		
	MSE (std)	Rules	Running time (std)	*MSE* (std)	Rules	Running time (std)
	0.0896 (3.117E-03)	16.0	8.73 (0.10)	0.0789 (3.054E-03)	9.0	13.84 (0.79)
	BTSK-JL					
	MSE (std)	Rules	Running time (std)			
	0.0719 (3.002E-03)	**6.4**	10.03 (0.56)			

References

[1] T. Efendigil, S. Önüt, C. Kahraman, A decision support system for demand forecasting with artificial neural networks and neuro-fuzzy models: a comparative analysis, *Expert Systems with Applications*, vol.36, no.3, pp. 6697-6707, 2009.

[2] J. L. Zhu, Z. Q. Ge, Z. H. Song, Variational Bayesian Gaussian mixture regression for soft sensing key variables in Non-Gaussian industrial processes, *IEEE Trans. Control Systems Technology*, vol. 25, no.3, pp. 1092-1099, 2017.

[3] G. Zhang, H. X. Li, M. Gan, Design a wind speed prediction model using probabilistic fuzzy System, *IEEE Trans. Industrial Informatics*, vol. 8, no. 4, pp. 819-827, 2012.

[4] P. Singh, N. R. Pal, S. Verma, O. P. Vyas, Fuzzy rule-based approach for software fault prediction, *IEEE Trans. Systems, Man, and Cybernetics: Systems*, vol. 47, no.5, pp .826-837, 2017.

[5] H. C. Huang, Fusion of Modified Bat Algorithm Soft Computing and Dynamic Model Hard Computing to Online Self-Adaptive Fuzzy Control of Autonomous Mobile Robots, *IEEE Trans. Industrial Informatics*, vol. 12, no. 3, pp. 972-979, 2016

[6] J. Richardson, J. Korniak, P. D. Reiner, B. M. Wilamowski, Nearest-Neighbor Spline Approximation (NNSA) Improvement to TSK Fuzzy Systems, *IEEE Trans. Industrial Informatics*, vol. 12, no. 1, pp. 169-178, 2016

[7] F. J. Luo, Z. Y. Dong, G. Chen, Y. Xu, K. Meng, Y. Y. Chen, K. P. Wong, Advanced pattern discovery-based fuzzy classification method for power system dynamic security assessment, *IEEE Trans. Industrial Informatics*, vol.11, no.2, pp. 416-426, 2015,

[8] T. Takagi, M. Sugeno, Fuzzy identification of systems and its application to modeling and control, *IEEE Trans. Systems, Man, and Cybernetics*,

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TII.2018.2813977, IEEE Transactions on Industrial Informatics

vol.15, no.1, pp.116-132, 1985.

[9] M. Luo, F. C. Sun, H. P. Liu, Hierarchical Structured Sparse Representation, *IEEE Trans. Fuzzy System*, vol.21, no.6, pp.1032-1043, 2013.

[10] H. Pomares, I. Rojas, J. González, A. Prieto, Structure identification in complete rule-based fuzzy systems, *IEEE Trans. Fuzzy Systems*, vol.10, no.3, pp. 349-359, 2002.

[11] M. N. Luo, F. C. Sun, H. P. Liu, Joint block structure sparse representation for multi-input–multi-output (MIMO) T-S fuzzy system identification, *IEEE Trans. Fuzzy Systems*, vol.22, no.6, pp. 1387-1400, 2014.

[12] L. X. Ren, G. W. Irwin, Robust fuzzy Gustafson-Kessel clustering for nonlinear system identification, *International Journal of Systems Science*, vol.34, no.14, pp. 787-803, 2003.

[13] R. Alcalá, J. Alcala-Fdez, F. Herrera, A Proposal for the genetic lateral tuning of linguistic fuzzy systems and its interaction with rule selection, *IEEE Trans. Fuzzy Systems*, vol.15, no.4, pp.616-635, 2007.

[14] O. Cordón, F. Herrera, A two-stage evolutionary process for designing TSK fuzzy rule-based systems, *IEEE Trans. Systems, Man and Cybernetics, Part B: Cybernetics*, vol.29, no.6, pp.703-715, 1999.

[15] Z. H. Deng, K. S. Choi, S. T. Wang. Scalable TSK fuzzy modeling for very large datasets using minimal-enclosing-ball approximation, *IEEE Trans. Fuzzy Systems*, vol.19, no.4, pp.210-226, 2011.

[16] N. R. Pal, S. Saha, Simultaneous structure identification and fuzzy rule generation for Takagi-Sugeno models, *IEEE Trans. Systems, Man, and Cybernetics, Part B: Cybernetics*, vol.38, no.6, pp.1626-1638, 2008.

[17] C. F. Juang, C. D. Hsieh, A locally recurrent fuzzy neural network with support vector regression for dynamic-system modeling, *IEEE Trans. Fuzzy Systems*, vol.18, no.2, pp.261-273, 2010.

[18] N. R. Pal, R. Mudi, K. Pal, D. Patranabish, Rule extraction through exploratory data analysis for self-tuning fuzzy controllers, *International Journal of Fuzzy Systems*, vol.6, no.2, pp.71-80, 2004.

[19] Y. Y. Lin, J. Y. Chang, C. T. Lin, Identification and prediction of dynamic systems using an interactively recurrent self-evolving fuzzy neural network, *IEEE Trans. Neural Networks and Learning Systems*, vol.24, no. 2, pp. 310-321, 2013.

[20] C. F. Juang, C. D. Hsieh, A fuzzy system constructed by rule generation and iterative linear SVR for antecedent and consequent parameter optimization, *IEEE Trans. Fuzzy System*s, vol.20, no.2, pp.372-384, 2012.

[21] J. F. Liu, F. L. Chung, S. T. Wang, Bayesian zero-order TSK fuzzy system modeling, *Applied Soft Computing*, vol.55, no.6, pp.253-264, 2017.

[22] C. F. Juang, C. D. Hsieh, A fuzzy system constructed by rule generation and iterative linear SVR for antecedent and consequent parameter optimization, *IEEE Trans. Fuzzy System*s, vol.20, no.2, pp.372-384, 2012.

[23] L. Zadeh, Discussion: Probability Theory and Fuzzy Logic Are Complementary Rather Than Competitive, *Technometrics*, vol. 37, no. 3, pp. 271-276, 1995.

[24] J. M. Garibaldi, M. Jaroszewski, and S. Musikasuwan, Nonstationary fuzzy sets, *IEEE Trans. Fuzzy Systems*, vol. 16, no. 4, pp. 1072-1086, 2008.

[25] M. Gan, M. Hanmandlu, A. Tan, From a Gaussian mixture model to additive fuzzy systems, *IEEE Trans. Fuzzy System*s, vol. 16, no. 4, pp. 1072-1086, 2008.

[26] C. T. Lin, N. R. Pal, S. L. Wu, Y. T. Liu, Y. Y. Lin, An interval type-2 neural fuzzy system for online system identification and feature elimination, *IEEE Trans. Neural Networks and Learning Systems*, vol. 26, no. 7, pp.1442-1455, 2015.

[27] R. Granell, C. J. Axon, D. C.H. Wallom, Clustering disaggregated load profiles using a dirichlet process mixture model, *Energy Conversion and Management*, vol. 92, no. 3, pp.507-516, 2015.

[28] M. Cococcioni, B. Lazzerini, F. Marcelloni, Towards Efficient Multi-objective Genetic Takagi-Sugeno Fuzzy Systems for High Dimensional Problems, In: Y. Tenne, C. K. Goh (eds.) Computational Intelligence in Expensive Optimization Problems. pp. 397-422, 2009.

[29] X. Q. Gu, F. L. Chung, H. Ishibuchi, S. T. Wang: Imbalanced TSK Fuzzy Classifier by Cross-Class Bayesian Fuzzy Clustering and Imbalance Learning. *IEEE Trans. Systems, Man, and Cybernetics: Systems*, vol.47, no.8, pp. 2005-2020, 2017.

[30] J. C. Bezdek, Pattern recognition with fuzzy objective function algorithms. Kluwer Academic Publishers, 1981

[31] R. Alcalá, J. Alcala-Fdez, J. Casillas, O. Cordón, F. Herrera, Local identification of prototypes for genetic learning of accurate TSK fuzzy rule-based systems, *International Journal of Intelligent Systems*, vol.22 no. 9, pp.909-941, 2007.

[32] N. Chopin, A sequential particle filter method for static models, *Biometrika*, vol.89, no. 3, pp. 539-551, 2002.

[33] A. G. Daronkolaei, S. Shiry, M. B. Menhaj, Multiple target tracking for mobile robots using the JPDAF algorithm, in *Proc: 19th IEEE International Conference on Tools with Artificial Intelligence*. pp. 137-145, 2007.

[34] A. Cord, C. Ambroise, J. Cocquerez, Feature selection in robust clustering based on Laplace mixture, *Pattern Recognition Letters*, vol.27, no. 6, pp. 627-635, 2006.

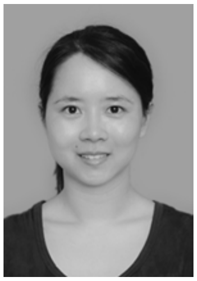
[35] B. A. Frigyik, A. Kapila, M. R. Gupta, Introduction to the Dirichlet distribution and related processes, Technical Report UWEETR-2010-006, University of Washington Department of Electrical Engineering, 2012.

[36] V. Elvira, J. Míguez, P. M. Djurić, Adapting the Number of Particles in Sequential Monte Carlo Methods through an Online Scheme for Convergence Assessment, *IEEE Transactions on Signal Processing*, vol. 65, no. 7, pp. 1781-1794, 2017.

[37] W. C. Cheng, PSO algorithm particle filters for improving the performance of lane detection and tracking systems in difficult roads, *Sensors*, vol. 12, no. 12, pp.17168-17185, 2012.

[38] KEEL Software and KEEL Datasets. http://sci2s.ugr.es/keel.

[39] J. D. Wichard, Forecasting the NN5 time series with hybrid models, *International Journal of Forecasting*, vol. 27, no. 3, pp. 700-707, 2011.



Xiaoqing Gu received the Ph.D. degree in light industry information technology and engineering from Jiangnan University, Wuxi, China, in 2017.

She is a Lecturer in the School of Information Science and Engineering, Changzhou University, Changzhou, China. She has published more than ten papers in international/national journals, including the IEEE TRANSACTIONS ON FUZZY SYSTEMS and IEEE TRANSACTIONS ON SYSTEMS, MAN AND CYBERNETICS: SYSTEMS. Her current research interests include pattern recognition and machine learning.

ShitongWang received the M.S. degree in computer science from Nanjing University of Aeronautics and Astronautics, Nanjing, China, in 1987. He has visited London University and Bristol University in U.K., Hiroshima International University and Osaka Prefecture University in Japan, HongKong University of Science and Technology, Hong Kong Polytechnic University, as a Research Scientist, for more than seven years.

He is currently a Full Professor of the School of Digital Media, Jiangnan University, China. His research interests include artificial intelligence, neuro-fuzzy systems, pattern recognition, and image processing. He has published about 100 papers in international/national journals and has authored seven books.