

Variational Bayes for a Mixed Stochastic/Deterministic Fuzzy Filter

Mohit Kumar, Norbert Stoll, and Regina Stoll

Abstract—This study, under variational Bayes (VB) framework, infers the parameters of a Takagi-Sugeno fuzzy filter having deterministic antecedents and stochastic consequents. The motivation of the study is to take advantages of the VB framework in designing fuzzy filtering algorithms. These advantages include an automated regularization, incorporation of statistical noise models, and model comparison capability. The VB method can be easily applied to the linear-in-parameters models. This paper applies VB method to the nonlinear fuzzy filters without using Taylor expansion for a linear approximation of some nonlinear function. It is assumed that the nonlinear parameters (i.e. antecedents) of the fuzzy filter are deterministic while linear parameters are stochastic. The VB algorithm, by maximizing a strict lower bound on the data evidence, makes the approximate posterior of linear parameters as close to the true posterior as possible. The nonlinear deterministic parameters are tuned in a way to further increase the lower bound on data evidence. The VB paradigm can be used to design an algorithm that automatically selects the most suitable fuzzy filter out of the considered finite set of fuzzy filters. This is done by fitting the observed data as a stochastic combination of the different Takagi-Sugeno fuzzy filters such that the individual filters compete with one another to model the data.

Index Terms—Fuzzy filtering, variational Bayes, probability distribution, Takagi-Sugeno fuzzy model.

I. INTRODUCTION

FUZZY systems based on fuzzy set theory [1], [2] have been proposed in the literature to deal with the uncertainties. Our recent work on fuzzy methods for a proper handling of the uncertainties associated to a few real-world applications has been reported in [3]–[10]. The use of fuzzy systems in data driven modeling is a topic that is widely studied by the researchers [11]–[30] due to the successful applications of fuzzy techniques in data mining, prediction, control, classification, simulation, and pattern recognition.

The robustness of an identification method becomes a key issue in presence of the uncertainties in the data. Therefore, several robust methods of fuzzy identification have been developed [31]–[49]. The robustness against outliers was achieved in [31] via optimizing a robust objective function. Regularization is a general method for improving the robustness of identification algorithms [34]. Regularization was used in [32],

[37] to convert the fuzzy identification problem to a well-posed problem. [35] suggests a regularized orthogonal least squares algorithm combined with a D-optimality used for subspace based rule selection for a linear-in-parameters fuzzy model. A learning algorithm based on input-to-state stability approach was introduced in [33]. [39] considers the identification of fuzzy models with uncertain data using semidefinite programming and second-order cone programming. The min-max approach to fuzzy model parameters estimation, that tries to minimize the worst-case effect of disturbances on the estimation errors, was studied in [38], [40], [41], [43], [44]. The criterion of integral squared error with exponential forgetting was used in [42] for a robust estimation of fuzzy model parameters. A clustering algorithm (termed as robust fuzzy regression agglomeration) and a robust cost function were used in [45] for the fuzzy modeling with outliers. The fuzzy modeling problem was studied in a probabilistic Bayesian learning framework using the extended relevance vector machine [36]. Support vector regression techniques have been integrated with the fuzzy systems for a robust behavior [48], [49]. The fuzzy model parameters were learned by a combination of fuzzy clustering and linear support vector regression [47]. The ϵ -insensitive learning of fuzzy model parameters was suggested in [46]. The fuzzy model parameters estimation algorithms presented in [28], [30] have following features:

- 1) Several researchers estimate the membership functions related parameters (i.e. antecedents) based on some criterion (e.g. fuzzy clustering criterion) while the estimation of linear parameters (i.e. consequents) is based on a different criterion (e.g. support vector regression). However, a more elegant approach is to estimate both antecedents and consequents based on the filtering criterion (e.g. H^∞ – optimal filtering).
- 2) A mathematical framework should be available for designing the fuzzy filtering algorithms and for their analysis in terms of stability, robustness, and steady-state error.

Bayesian framework based on Bayes' theorem is a powerful technique for the statistical inference of model parameters. The VB framework has the advantage of being less computationally intensive than other Bayesian methods. The VB method approximates the posterior distributions over a model in an analytical manner [50]. The VB method minimizes the Kullback-Leibler (KL) divergence of the approximate posterior from the true posterior density [51]. The Expectation Maximization (EM) algorithm is a special case of VB [52]. The

This work was supported by Center for Life Science Automation, Rostock, Germany.

M. Kumar is with the Center for Life Science Automation, F.-Barnewitz-Str. 8, D-18119 Rostock, Germany.

N. Stoll is with the Institute of Automation, College of Computer Science and Electrical Engineering, University of Rostock, Richard-Wagner-Str. 31, D-18119 Rostock-Warnemünde, Germany.

R. Stoll is with the Institute of Preventive Medicine, Faculty of Medicine, University of Rostock, St.-Georg-Str. 108, D-18055 Rostock, Germany.

VB by virtue of being Bayesian method has its advantages: it can incorporate regularizing priors, complex noise models, and perform model comparisons. The VB method has been applied to the nonlinear models by the authors in [53]–[55] using a Taylor expansion. However, the convergence of the VB method with the use of Taylor expansion (as in [55]) is not guaranteed.

The Bayesian inference problem determines the parameters w of a model \mathbf{m} using available data y based on Bayes's theorem:

$$p(w|y, \mathbf{m}) = \frac{p(y|w, \mathbf{m})p(w|\mathbf{m})}{p(y|\mathbf{m})}.$$

Bayes's theorem provides the *posterior* probability of the parameters given the data and the model. The analytical evaluation of posterior probability distribution is not possible in every case. Thus, it is approximated by a variational distribution:

$$q(w) \approx p(w|y, \mathbf{m})$$

where $q(w)$ is restricted to belong to a family of distributions of simpler form. This form is selected by minimizing the difference (in term of Kullback-Leibler divergence) between q and true posterior. The Kullback-Leibler (KL) divergence of $p(w|y, \mathbf{m})$ from $q(w)$ is defined as

$$KL(q||p) = \int q(w) \log \frac{q(w)}{p(w|y, \mathbf{m})} dw.$$

The logarithmic *evidence* for the data is given as

$$\begin{aligned} \log p(y|\mathbf{m}) &= \log \int p(y, w|\mathbf{m}) dw \\ &= \log \int q(w) \frac{p(y, w|\mathbf{m})}{q(w)} dw \\ &\geq \int q(w) \log \frac{p(y, w|\mathbf{m})}{q(w)} dw \\ &\equiv \mathcal{F}(q(w), \mathbf{m}) \end{aligned}$$

where we have made use of the Jensen's inequality. Any probability distribution $q(w)$ gives rise to a lower bound $\mathcal{F}(q(w), \mathbf{m})$ on the logarithmic evidence. The lower bound $\mathcal{F}(q(w), \mathbf{m})$ is the negative of a quantity known as *free energy*. Since

$$\log p(y|\mathbf{m}) = \mathcal{F}(q(w), \mathbf{m}) + KL(q||p),$$

minimizing $KL(q||p)$ is equivalent to maximizing $\mathcal{F}(q(w), \mathbf{m})$ over $q(w)$. Therefore, posterior distribution $p(w|y, \mathbf{m})$ is inferred by estimating $q(w)$ correctly, i.e., by maximizing $\mathcal{F}(q(w), \mathbf{m})$ over $q(w)$.

A Takagi-Sugeno fuzzy filter, as explained in Appendix A, can be mathematically represented as

$$y_f = G^T(x, \theta)\alpha, \quad c\theta \geq h. \quad (1)$$

The filter, as seen from (1), is characterized by two types of parameters: antecedents (θ) and consequents (α). Expression (1) shows that the output of the fuzzy filter is linear in consequents (i.e. in the elements of vector α) while nonlinear in antecedents (i.e. in the elements of vector θ). We study a type of filter with

- the nonlinear parameters θ being considered as deterministic, and
- the linear parameters α being considered as random variables.

We focus on a process with n -inputs (represented by the vector $x \in R^n$) and a single output (represented by the scalar y). It is assumed that inputs-output data pairs $\{x(j), y(j)\}$ are related via

$$y(j) = G^T(x(j), \theta)\alpha + n_j, \quad (2)$$

where n_j is the additive Gaussian uncertainty with mean 0 and a variance of $1/\phi$. The fuzzy filtering algorithms should seek to estimate the vector θ and evaluate the posterior probability distribution of α . The considered fuzzy filtering problem in the VB framework is stated in Problem 1.

Problem 1: Given N pairs of inputs-output data $\{x(j), y(j)\}_{j=1}^N$ and a structure \mathbf{m} (i.e. membership type, number of membership functions and rules) of a Takagi-Sugeno filter of type (1) such that data satisfy (2), estimate θ and the variational distributions $(q(\alpha), q(\phi))$ by maximizing the lower bound on the quantity: $\log p(y(1), \dots, y(N)|x(1), \dots, x(N), \theta, \mathbf{m})$.

Introduce the notations:

$$Y = \begin{bmatrix} y(1) \\ \vdots \\ y(N) \end{bmatrix} \in R^N, \quad B(\theta) = \begin{bmatrix} G^T(x(1), \theta) \\ \vdots \\ G^T(x(N), \theta) \end{bmatrix} \in R^{N \times K},$$

$$v = [n_1 \dots n_N]^T \in R^N.$$

Now, we have

$$Y = B(\theta)\alpha + v, \quad (3)$$

where v is an additive Gaussian uncertainty with mean 0 and a variance of $1/\phi$:

$$p(v) \sim N(0, \phi^{-1}I).$$

Problem 1 can be rewritten as

Problem 2: Given N pairs of inputs-output data $\{x(j), y(j)\}_{j=1}^N$ and a structure \mathbf{m} (i.e. membership type, number of membership functions and rules) of a Takagi-Sugeno filter of type (1) such that data satisfy (3), estimate θ and the variational distributions $(q(\alpha), q(\phi))$ by maximizing the lower bound on the quantity: $\log p(Y|B(\theta), \mathbf{m})$.

Our next concern is to fit the observed data as a stochastic combination of the different Takagi-Sugeno fuzzy filters such that the individual filters compete with one another to model the data. Assume that there are S number of fuzzy filters (with their structures as $\{\mathbf{m}^i\}_{i=1}^S$) such that the output of the i -th filter (i.e. $B(\theta^i)\alpha^i$) should match to the observed output vector Y in some optimal manner. Let s_i (where $s_i = 1, 2, \dots, S$) be a discrete indicator random variable whose value represents the chosen filter for data modeling. That is,

$$\begin{aligned} \text{If } s_i = 1, & \quad Y = B(\theta^1)\alpha^1 + v \\ & \quad \vdots \\ \text{If } s_i = S, & \quad Y = B(\theta^S)\alpha^S + v \end{aligned} \quad (4)$$

Let $\pi = [\pi_1 \dots \pi_S]^T \in R^S$, with $0 \leq \pi_{s_i} \leq 1$ and $\sum_{s_i=1}^S \pi_{s_i} = 1$, be a vector of mixing proportions (i.e. the

proportions by which individual fuzzy filters' outputs are mixed to match the observed output vector). The discrete distribution of the indicator variable s_i is given as

$$p(s_i = 1|\pi) = \pi_1, \dots, p(s_i = S|\pi) = \pi_S.$$

We model the probability density function of the observed output data as a weighted average of the individual fuzzy filters' output density functions:

$$\begin{aligned} p(Y|\pi, \{B(\theta^{s_i})\}_{s_i=1}^S, \{\alpha^{s_i}\}_{s_i=1}^S, \phi, \{\mathbf{m}^{s_i}\}_{s_i=1}^S) \quad (5) \\ = \sum_{s_i=1}^S p(s_i|\pi) p(Y|s_i, B(\theta^{s_i}), \alpha^{s_i}, \phi, \mathbf{m}^{s_i}). \end{aligned}$$

Problem 3: Given N pairs of inputs-output data $\{x(j), y(j)\}_{j=1}^N$ and S different structures $\{\mathbf{m}^{s_i}\}_{s_i=1}^S$ of the Takagi-Sugeno filters of type (1) such that data satisfy (4), estimate $\{\theta^{s_i}\}_{s_i=1}^S$ and the variational distributions $(\{q(\alpha^{s_i}), q(s_i)\}_{s_i=1}^S, q(\pi), q(\phi))$ by maximizing the lower bound on the quantity: $\log p(Y|\{B(\theta^{s_i})\}_{s_i=1}^S, \{\mathbf{m}^{s_i}\}_{s_i=1}^S)$.

Remark 1: Our approach looks similar to the well known statistical modeling technique of "finite mixture models". The finite mixture models are widely used for clustering where it is assumed that there exists a finite set of random sources whose convex combination might have generated the observed data. Our approach, however, is different in the sense that a component model (i.e. individual filter) tries to fit all the N pairs of data rather than a subset of the complete data set. That is, our approach, unlike finite mixture modeling, doesn't partition the total data set into S different clusters such that the data belonging to a cluster is modeled as having been generated by one of the component models in the set. Therefore, Problem 3 allows the different filters to compete with one another to model the data.

The aim of this study is to develop fuzzy filtering algorithms based on the solutions of Problems 2 and 3. The motivation of the work is to take simultaneously the following advantages of the VB framework in designing fuzzy filtering algorithms:

- 1) an automated regularization through priors,
- 2) model comparison capability (i.e. an automatic selection of the most suitable model),
- 3) handling uncertainty via incorporating statistical noise models.

The VB method is not a new technique and has been widely studied by the researchers. The VB method can be easily applied to the linear-in-parameters models. A contribution of this study is to extend VB method to the nonlinear fuzzy filters. The nonlinear deterministic parameters are estimated in such a way that the lower bound on data evidence is increased. The text also provides an algorithm that automatically selects the most suitable fuzzy filter out of the considered finite set of fuzzy filters and infers its parameters. To the best knowledge of the authors, this is the first study that applies VB method to the introduced mixed stochastic/deterministic fuzzy filter for solving Problems 2 and 3.

II. VARIATIONAL BAYESIAN INFERENCE OF A STOCHASTIC COMBINATION OF FUZZY FILTERS

This section presents our approach to solve the Problem 3. Following distributions are chosen for the parameters priors:

$$\begin{aligned} p(\alpha^{s_i}|m_0^{s_i}, \Lambda_0^{s_i}) &= N(\alpha^{s_i}|m_0^{s_i}, (\Lambda_0^{s_i})^{-1}) \\ p(\phi|a_0, b_0) &= Ga(\phi|a_0, b_0) \\ p(\pi|c_0 d_0) &= Dir(\pi|c_0 d_0), \quad d_0 = [\frac{1}{S} \dots \frac{1}{S}]^T \in R^S \end{aligned}$$

where Gamma and Dirichlet distributions are defined as follows

$$Ga(\phi|a_0, b_0) = \frac{1}{\Gamma(b_0)} \frac{\phi^{b_0-1}}{a_0^{b_0}} e^{-\frac{\phi}{a_0}}, \quad \text{for } \phi > 0 \text{ and } a_0, b_0 > 0.$$

$$Dir(\pi|c_0 d_0) = \frac{\Gamma(c_0)}{(\Gamma(\frac{c_0}{S}))^S} \pi_1^{\frac{c_0}{S}-1} \dots \pi_S^{\frac{c_0}{S}-1}$$

where $\pi_1, \dots, \pi_S \geq 0$, $\sum_{j=1}^S \pi_j = 1$, $c_0 > 0$. The logarithmic evidence for the data is given as

$$\begin{aligned} \log p(Y|\{B(\theta^{s_i})\}_{s_i=1}^S, \{\mathbf{m}^{s_i}\}_{s_i=1}^S) \\ = \log \int d\pi d\alpha^1 \dots d\alpha^S d\phi p(\pi|c_0 d_0) p(\alpha^1|m_0^1, \Lambda_0^1) \dots \\ \dots p(\alpha^S|m_0^S, \Lambda_0^S) p(\phi|a_0, b_0) \\ p(Y|\pi, \{B(\theta^{s_i})\}_{s_i=1}^S, \{\alpha^{s_i}\}_{s_i=1}^S, \phi, \{\mathbf{m}^{s_i}\}_{s_i=1}^S). \end{aligned}$$

Using (5),

$$\begin{aligned} \log p(Y|\{B(\theta^{s_i})\}_{s_i=1}^S, \{\mathbf{m}^{s_i}\}_{s_i=1}^S) \\ = \log \int d\pi d\alpha^1 \dots d\alpha^S d\phi p(\pi|c_0 d_0) p(\alpha^1|m_0^1, \Lambda_0^1) \dots \\ \dots p(\alpha^S|m_0^S, \Lambda_0^S) p(\phi|a_0, b_0) \\ \sum_{s_i=1}^S p(s_i|\pi) p(Y|s_i, B(\theta^{s_i}), \alpha^{s_i}, \phi, \mathbf{m}^{s_i}). \end{aligned}$$

For simplicity, we define $\alpha = \{\alpha^1, \dots, \alpha^S\}$, $m_0 = \{m_0^1, \dots, m_0^S\}$, $\Lambda_0 = \{\Lambda_0^1, \dots, \Lambda_0^S\}$ with $p(\alpha|m_0, \Lambda_0) = \prod_{s_i=1}^S p(\alpha^{s_i}|m_0^{s_i}, \Lambda_0^{s_i})$. The above integral can be written as

$$\begin{aligned} \log p(Y|\{B(\theta^{s_i})\}_{s_i=1}^S, \{\mathbf{m}^{s_i}\}_{s_i=1}^S) \\ = \log \int d\pi d\alpha d\phi p(\pi|c_0 d_0) p(\alpha|m_0, \Lambda_0) p(\phi|a_0, b_0) \\ \sum_{s_i=1}^S p(s_i|\pi) p(Y|s_i, B(\theta^{s_i}), \alpha^{s_i}, \phi, \mathbf{m}^{s_i}). \end{aligned}$$

Introducing an arbitrary distribution $q(\pi, \alpha, \phi)$ to lower bound the data evidence:

$$\begin{aligned} \log p(Y|\{B(\theta^{s_i})\}_{s_i=1}^S, \{\mathbf{m}^{s_i}\}_{s_i=1}^S) \geq \int d\pi d\alpha d\phi q(\pi, \alpha, \phi) \\ p(\pi|c_0 d_0) p(\alpha|m_0, \Lambda_0) p(\phi|a_0, b_0) \sum_{s_i=1}^S p(s_i|\pi) p(Y|s_i, \dots) \\ \log \frac{\quad}{q(\pi, \alpha, \phi)} \end{aligned}$$

where $p(Y|s_i, \dots)$ should be understood as $p(Y|s_i, B(\theta^{s_i}), \alpha^{s_i}, \phi, \mathbf{m}^{s_i})$. We restrict our method to use the approximation:

$$\begin{aligned} q(\pi, \alpha, \phi) &\approx q(\pi)q(\alpha)q(\phi) \\ &= q(\pi) \prod_{s_i=1}^S q(\alpha^{s_i})q(\phi) \end{aligned}$$

This results in

$$\begin{aligned} &\log p(Y|\{B(\theta^{s_i})\}_{s_i=1}^S, \{\mathbf{m}^{s_i}\}_{s_i=1}^S) \\ &\geq \int d\pi q(\pi) \log \frac{p(\pi|c_0 d_0)}{q(\pi)} + \int d\alpha q(\alpha) \log \frac{p(\alpha|m_0, \Lambda_0)}{q(\alpha)} \\ &\quad + \int d\phi q(\phi) \log \frac{p(\phi|a_0, b_0)}{q(\phi)} \\ &\quad + \int d\pi d\alpha d\phi q(\pi)q(\alpha)q(\phi) \log \left(\sum_{s_i=1}^S p(s_i|\pi) p(Y|s_i, \dots) \right). \end{aligned}$$

Again, a discrete distribution $q(s_i)$ with $\sum_{s_i=1}^S q(s_i) = 1$ is introduced to further lower bound the data evidence:

$$\begin{aligned} &\log p(Y|\{B(\theta^{s_i})\}_{s_i=1}^S, \{\mathbf{m}^{s_i}\}_{s_i=1}^S) \\ &\geq \int d\pi q(\pi) \log \frac{p(\pi|c_0 d_0)}{q(\pi)} + \int d\alpha q(\alpha) \log \frac{p(\alpha|m_0, \Lambda_0)}{q(\alpha)} \\ &\quad + \int d\phi q(\phi) \log \frac{p(\phi|a_0, b_0)}{q(\phi)} + \\ &\quad \int d\pi d\alpha d\phi q(\pi)q(\alpha)q(\phi) \sum_{s_i=1}^S q(s_i) \log \frac{p(s_i|\pi) p(Y|s_i, \dots)}{q(s_i)}. \end{aligned}$$

That is,

$$\begin{aligned} &\log p(Y|\{B(\theta^{s_i})\}_{s_i=1}^S, \{\mathbf{m}^{s_i}\}_{s_i=1}^S) \\ &\geq \int d\pi q(\pi) \log \frac{p(\pi|c_0 d_0)}{q(\pi)} + \int d\alpha q(\alpha) \log \frac{p(\alpha|m_0, \Lambda_0)}{q(\alpha)} \\ &\quad + \int d\phi q(\phi) \log \frac{p(\phi|a_0, b_0)}{q(\phi)} \\ &\quad + \sum_{s_i=1}^S q(s_i) \int d\alpha d\phi q(\alpha)q(\phi) \log p(Y|s_i, \dots) \\ &\quad + \sum_{s_i=1}^S q(s_i) \int d\pi q(\pi) \log \frac{p(s_i|\pi)}{q(s_i)}. \end{aligned}$$

The lower bound is defined as a functional of the variational posterior distributions as follows:

$$\begin{aligned} &\mathcal{F}(q(\pi), q(\alpha), q(\phi), \{q(s_i)\}_{s_i=1}^S, \\ &\quad \{B(\theta^{s_i})\}_{s_i=1}^S, c_0, m_0, \Lambda_0, a_0, b_0, \{\mathbf{m}^{s_i}\}_{s_i=1}^S) \\ &= \int d\pi q(\pi) \log \frac{p(\pi|c_0 d_0)}{q(\pi)} + \int d\alpha q(\alpha) \log \frac{p(\alpha|m_0, \Lambda_0)}{q(\alpha)} \\ &\quad + \int d\phi q(\phi) \log \frac{p(\phi|a_0, b_0)}{q(\phi)} \\ &\quad + \sum_{s_i=1}^S q(s_i) \int d\alpha d\phi q(\alpha)q(\phi) \log p(Y|s_i, \dots) \\ &\quad + \sum_{s_i=1}^S q(s_i) \int d\pi q(\pi) \log \frac{p(s_i|\pi)}{q(s_i)}. \end{aligned}$$

Now, we are in a position to present our method for inferring the parameters of the fuzzy filters combination. Algorithm 1 lists the various steps involved in our method.

Algorithm 1 VB inference of the fuzzy filters combination

Require: Data pairs $\{x(j), y(j)\}_{j=1, \dots, N}$.

- 1: Choose a total of S fuzzy filters' structures $\{\mathbf{m}^{s_i}\}_{s_i=1}^S$; hyper-parameters $c_0, m_0, \Lambda_0, a_0, b_0$ which define the regularizing priors; parameters $\{c^{s_i}, h^{s_i}\}_{s_i=1}^S$ such that the interpretability constraints on the membership functions of the s_i -th filter can be formulated as $c^{s_i} \theta^{s_i} \geq h^{s_i}$.

- 2: Define

$$\begin{aligned} &\mathcal{F}(q(\pi), q(\alpha), q(\phi), \{q(s_i)\}_{s_i=1}^S, \{B(\theta^{s_i})\}_{s_i=1}^S, c_0, m_0, \Lambda_0, a_0, \\ &\quad b_0, \{\mathbf{m}^{s_i}\}_{s_i=1}^S) = \\ &\quad \int d\pi q(\pi) \log \frac{p(\pi|c_0 d_0)}{q(\pi)} + \int d\alpha q(\alpha) \log \frac{p(\alpha|m_0, \Lambda_0)}{q(\alpha)} \\ &\quad + \int d\phi q(\phi) \log \frac{p(\phi|a_0, b_0)}{q(\phi)} \\ &\quad + \sum_{s_i=1}^S q(s_i) \int d\alpha d\phi q(\alpha)q(\phi) \log p(Y|s_i, B(\theta^{s_i}), \alpha^{s_i}, \phi, \mathbf{m}^{s_i}) \\ &\quad + \sum_{s_i=1}^S q(s_i) \int d\pi q(\pi) \log \frac{p(s_i|\pi)}{q(s_i)}. \end{aligned}$$

- 3: Derive analytically the correct expressions for distributions $(q(\pi), q(\alpha), q(\phi), q(s_i))$ by maximizing \mathcal{F} over $(q(\pi), q(\alpha), q(\phi), q(s_i))$ (i.e. by setting the functional derivatives of \mathcal{F} with respect to each free distribution equal to zero). Let $(q^*(\pi), q^*(\alpha), q^*(\phi), q^*(s_i))$ denote the obtained expressions for the variational distributions.

- 4: Estimate the optimal values of antecedents of fuzzy filters, i.e. $(\theta^{1,*}, \dots, \theta^{S,*})$, via maximizing \mathcal{F} further over $(\theta^1, \dots, \theta^S)$. That is, solve the following constrained nonlinear optimization problem:

$$(\theta^{1,*}, \dots, \theta^{S,*}) = \arg \max_{(\theta^1, \dots, \theta^S)} [\mathcal{F}_\theta(\cdot); c^{s_i} \theta^{s_i} \geq h^{s_i}]$$

where

$$\begin{aligned} &\mathcal{F}_\theta(\cdot) = \mathcal{F}(q^*(\pi), q^*(\alpha), q^*(\phi), \{q^*(s_i)\}_{s_i=1}^S, \{B(\theta^{s_i})\}_{s_i=1}^S, \\ &\quad c_0, m_0, \Lambda_0, a_0, b_0, \{\mathbf{m}^{s_i}\}_{s_i=1}^S). \end{aligned}$$

- 5: **return** $(\theta^{1,*}, \dots, \theta^{S,*})$ and $(q^*(\pi), q^*(\alpha), q^*(\phi), \{q^*(s_i)\}_{s_i=1}^S)$.
-

III. OPTIMIZATION OF A LOWER BOUND ON THE DATA EVIDENCE

A. Optimization w.r.t. $q(\pi)$

\mathcal{F} will be stationary w.r.t. distribution $q(\pi)$, if

$$\begin{aligned} &\log(p(\pi|c_0 d_0)) - \log(q(\pi)) \\ &\quad + \sum_{s_i=1}^S q(s_i) \log(p(s_i|\pi)) + \text{cons}\{q(\pi)\} = 0, \text{ i.e.,} \end{aligned}$$

$$\log\left(\prod_{s_i=1}^S \frac{c_0}{\pi_{s_i}^S} - 1\right) - \log(q(\pi))$$

$$+ \sum_{s_i=1}^S \log(\pi_{s_i}^{q(s_i)}) + \text{cons}\{q(\pi)\} = 0, \text{ i.e.,}$$

$$\log\left(\prod_{s_i=1}^S \pi_{s_i}^{\frac{c_0}{S}} + q(s_i) - 1\right) - \log(q(\pi)) + \text{cons}\{q(\pi)\} = 0.$$

This implies that

$$q^*(\pi) = \text{Dir}(\pi|cd), \quad d = [d_1 \cdots d_S]^T \in R^S, \quad \sum_{s_i=1}^S d_{s_i} = 1,$$

such that

$$cd_{s_i} = \frac{c_0}{S} + q(s_i).$$

Using $\sum_{s_i=1}^S d_{s_i} = 1$ and $\sum_{s_i=1}^S q(s_i) = 1$, we have

$$c = c_0 + 1.$$

Thus,

$$d_{s_i} = \frac{1}{c_0 + 1} \left(\frac{c_0}{S} + q(s_i) \right).$$

B. Optimization w.r.t. $q(\alpha^1), \dots, q(\alpha^S)$

\mathcal{F} will be stationary w.r.t. distribution $q(\alpha^{s_i})$ where $s_i = 1, \dots, S$, if

$$\begin{aligned} & \log(p(\alpha^{s_i} | m_0^{s_i}, \Lambda_0^{s_i})) - \log(q(\alpha^{s_i})) \\ & + q(s_i) \int d\phi q(\phi) \log(p(Y|s_i, B(\theta^{s_i}), \alpha^{s_i}, \phi, \mathbf{m}^{s_i})) \\ & + \text{cons}\{q(\alpha^{s_i})\} = 0. \end{aligned}$$

The substitutions

$$p(\alpha^{s_i} | m_0^{s_i}, \Lambda_0^{s_i}) \propto \exp\left(-\frac{1}{2}(\alpha^{s_i} - m_0^{s_i})^T \Lambda_0^{s_i} (\alpha^{s_i} - m_0^{s_i})\right),$$

$$\begin{aligned} & p(Y|s_i, B(\theta^{s_i}), \alpha^{s_i}, \phi, \mathbf{m}^{s_i}) \\ & \propto \exp\left(-\frac{\phi}{2}(Y - B(\theta^{s_i})\alpha^{s_i})^T (Y - B(\theta^{s_i})\alpha^{s_i})\right), \end{aligned}$$

result in

$$\begin{aligned} & -\frac{1}{2}(\alpha^{s_i} - m_0^{s_i})^T \Lambda_0^{s_i} (\alpha^{s_i} - m_0^{s_i}) - \log(q(\alpha^{s_i})) \\ & - \frac{q(s_i)}{2} \left(\int d\phi q(\phi) \phi \right) (Y - B(\theta^{s_i})\alpha^{s_i})^T (Y - B(\theta^{s_i})\alpha^{s_i}) \\ & + \text{cons}\{q(\alpha^{s_i})\} = 0. \end{aligned}$$

This implies that

$$q^*(\alpha^{s_i}) = N(\alpha^{s_i} | m^{s_i}, (\Lambda^{s_i})^{-1}), \quad \text{such that}$$

$$\Lambda^{s_i} = \Lambda_0^{s_i} + q(s_i) \left(\int d\phi q(\phi) \phi \right) (B(\theta^{s_i}))^T B(\theta^{s_i}),$$

$$\Lambda^{s_i} m^{s_i} = \Lambda_0^{s_i} m_0^{s_i} + q(s_i) \left(\int d\phi q(\phi) \phi \right) (B(\theta^{s_i}))^T Y.$$

Thus,

$$m^{s_i} = (\Lambda^{s_i})^{-1} \left[\Lambda_0^{s_i} m_0^{s_i} + q(s_i) \left(\int d\phi q(\phi) \phi \right) (B(\theta^{s_i}))^T Y \right].$$

The term $\int d\phi q(\phi) \phi$, appearing in the expressions for Λ^{s_i} and m^{s_i} , will be evaluated after obtaining the correct expression for $q(\phi)$ in the coming part of the text.

C. Optimization w.r.t. $q(\phi)$

Before equating the derivative of \mathcal{F} w.r.t. $q(\phi)$ equal to zero, note that

$$\begin{aligned} & \log(p(Y|s_i, B(\theta^{s_i}), \alpha^{s_i}, \phi, \mathbf{m}^{s_i})) \\ & = \frac{N}{2} \log(\phi) - \frac{\phi}{2} (Y - B(\theta^{s_i})\alpha^{s_i})^T (Y - B(\theta^{s_i})\alpha^{s_i}) \\ & \quad - \frac{N}{2} \log(2\pi). \end{aligned}$$

\mathcal{F} will be stationary w.r.t. distribution $q(\phi)$, if

$$\begin{aligned} & \log(p(\phi|a_0, b_0)) - \log(q(\phi)) + \text{cons}\{\phi\} \\ & + \sum_{s_i=1}^S q(s_i) \int d\alpha^{s_i} q(\alpha^{s_i}) \left[\frac{N}{2} \log(\phi) \right. \\ & \quad \left. - \frac{\phi}{2} (Y - B(\theta^{s_i})\alpha^{s_i})^T (Y - B(\theta^{s_i})\alpha^{s_i}) \right] = 0, \quad \text{i.e.,} \end{aligned}$$

$$\log(p(\phi|a_0, b_0)) - \log(q(\phi)) + \text{cons}\{\phi\}$$

$$\begin{aligned} & - \frac{\phi}{2} \sum_{s_i=1}^S q(s_i) \int d\alpha^{s_i} q(\alpha^{s_i}) (Y - B(\theta^{s_i})\alpha^{s_i})^T (Y - B(\theta^{s_i})\alpha^{s_i}) \\ & + \frac{N}{2} \log(\phi) = 0. \end{aligned}$$

Using the facts

$$\log(p(\phi|a_0, b_0)) = (b_0 - 1) \log(\phi) - \frac{\phi}{a_0} + \text{cons}\{\phi\},$$

$$\begin{aligned} & \int d\alpha^{s_i} q(\alpha^{s_i}) (Y - B(\theta^{s_i})\alpha^{s_i})^T (Y - B(\theta^{s_i})\alpha^{s_i}) \\ & = (Y - B(\theta^{s_i})m^{s_i})^T (Y - B(\theta^{s_i})m^{s_i}) \\ & \quad + \text{Tr}((\Lambda^{s_i})^{-1} (B(\theta^{s_i}))^T B(\theta^{s_i})), \end{aligned}$$

we have

$$\begin{aligned} & (b_0 - 1) \log(\phi) - \frac{\phi}{a_0} - \log(q(\phi)) + \text{cons}\{\phi\} + \frac{N}{2} \log(\phi) \\ & - \frac{\phi}{2} \sum_{s_i=1}^S q(s_i) [(Y - B(\theta^{s_i})m^{s_i})^T (Y - B(\theta^{s_i})m^{s_i}) \\ & \quad + \text{Tr}((\Lambda^{s_i})^{-1} (B(\theta^{s_i}))^T B(\theta^{s_i}))] = 0. \end{aligned}$$

This implies that

$$q^*(\phi) = \text{Ga}(\phi|a, b), \quad \text{such that}$$

$$\begin{aligned} \frac{1}{a} &= \frac{1}{a_0} + \frac{1}{2} \sum_{s_i=1}^S q(s_i) [(Y - B(\theta^{s_i})m^{s_i})^T (Y - B(\theta^{s_i})m^{s_i}) \\ & \quad + \text{Tr}((\Lambda^{s_i})^{-1} (B(\theta^{s_i}))^T B(\theta^{s_i}))], \\ b &= b_0 + \frac{N}{2}. \end{aligned}$$

D. Optimization w.r.t. $q(s_i)$

\mathcal{F} will be stationary w.r.t. distribution $q(s_i)$, if

$$\int d\pi q(\pi) \log(p(s_i|\pi)) - \log(q(s_i)) \\ + \int d\alpha^{s_i} d\phi q(\alpha^{s_i}) q(\phi) \log p(Y|s_i, B(\theta^{s_i}), \alpha^{s_i}, \phi, \mathbf{m}^{s_i}) = 0$$

As per a result of Dirichlet distributions,

$$\int d\pi q(\pi) \log(p(s_i|\pi)) = \Psi(cd_{s_i}) - \Psi(c),$$

where $\Psi(\cdot)$ is the digamma function. Now, consider the term

$$\int d\alpha^{s_i} d\phi q(\alpha^{s_i}) q(\phi) \log p(Y|s_i, B(\theta^{s_i}), \alpha^{s_i}, \phi, \mathbf{m}^{s_i}) \\ = \int d\phi q(\phi) \left(\int d\alpha^{s_i} q(\alpha^{s_i}) \log p(Y|s_i, B(\theta^{s_i}), \alpha^{s_i}, \phi, \mathbf{m}^{s_i}) \right) \\ = \frac{N}{2} \int d\phi q(\phi) \log(\phi) - \frac{N}{2} \log(2\pi) \\ - \frac{\int d\phi q(\phi) \phi}{2} [(Y - B(\theta^{s_i}) \mathbf{m}^{s_i})^T (Y - B(\theta^{s_i}) \mathbf{m}^{s_i}) \\ + Tr((\Lambda^{s_i})^{-1} (B(\theta^{s_i}))^T B(\theta^{s_i}))] \\ = \frac{N}{2} (\Psi(b) + \log(a)) - \frac{N}{2} \log(2\pi) \\ - \frac{ab}{2} [(Y - B(\theta^{s_i}) \mathbf{m}^{s_i})^T (Y - B(\theta^{s_i}) \mathbf{m}^{s_i}) \\ + Tr((\Lambda^{s_i})^{-1} (B(\theta^{s_i}))^T B(\theta^{s_i}))].$$

Here, we have used some results of Gamma distribution. Finally, the equilibrium equation becomes

$$\Psi(cd_{s_i}) - \Psi(c) - \log(q(s_i)) + \frac{N}{2} (\Psi(b) + \log(a)) \\ - \frac{N}{2} \log(2\pi) - \frac{ab}{2} [(Y - B(\theta^{s_i}) \mathbf{m}^{s_i})^T (Y - B(\theta^{s_i}) \mathbf{m}^{s_i}) \\ + Tr((\Lambda^{s_i})^{-1} (B(\theta^{s_i}))^T B(\theta^{s_i}))] = 0.$$

This implies that

$$q^*(s_i) = \frac{1}{\mathcal{Z}} \exp \left(\begin{array}{c} \Psi(cd_{s_i}) - \Psi(c) + \frac{N}{2} (\Psi(b) + \log(a)) \\ - \frac{N}{2} \log(2\pi) - \frac{ab}{2} r(m^{s_i}, \Lambda^{s_i}, \theta^{s_i}) \end{array} \right),$$

where \mathcal{Z} is the normalization constant such that $\sum_{s_i=1}^S q(s_i) = 1$ and

$$r(m^{s_i}, \Lambda^{s_i}, \theta^{s_i}) = (Y - B(\theta^{s_i}) \mathbf{m}^{s_i})^T (Y - B(\theta^{s_i}) \mathbf{m}^{s_i}) \\ + Tr((\Lambda^{s_i})^{-1} (B(\theta^{s_i}))^T B(\theta^{s_i})).$$

The constant terms in the expression of $q^*(s_i)$, which don't vary as s_i varies from 1 to S , can be included in the normalization constant. This simplifies the expression for $q^*(s_i)$ as

$$q^*(s_i) = \frac{1}{\mathcal{Z}} \exp \left(\Psi(cd_{s_i}) - \frac{ab}{2} r(m^{s_i}, \Lambda^{s_i}, \theta^{s_i}) \right).$$

E. Optimization w.r.t. $(\theta^1, \dots, \theta^S)$

The optimal values of antecedents of fuzzy filters are obtained via maximizing

$$\mathcal{F}(q^*(\pi), q^*(\alpha), q^*(\phi), \{q^*(s_i)\}_{s_i=1}^S, \{B(\theta^{s_i})\}_{s_i=1}^S, c_0, m_0, \\ \Lambda_0, a_0, b_0, \{\mathbf{m}^{s_i}\}_{s_i=1}^S)$$

over $(\theta^1, \dots, \theta^S)$. The lower bound on the logarithmic evidence for the data can be expressed as

$$\mathcal{F}(q(\pi), q(\alpha), q(\phi), \{q(s_i)\}_{s_i=1}^S, \{B(\theta^{s_i})\}_{s_i=1}^S, c_0, m_0, \\ \Lambda_0, a_0, b_0, \{\mathbf{m}^{s_i}\}_{s_i=1}^S) \\ = \sum_{s_i=1}^S q(s_i) \int d\alpha d\phi q(\alpha) q(\phi) \log p(Y|s_i, B(\theta^{s_i}), \alpha^{s_i}, \phi, \mathbf{m}^{s_i}) \\ + \sum_{s_i=1}^S q(s_i) \int d\pi q(\pi) \log \frac{p(s_i|\pi)}{q(s_i)} \\ - \int d\pi q(\pi) \log \frac{q(\pi)}{p(\pi|c_0 d_0)} - \int d\alpha q(\alpha) \log \frac{q(\alpha)}{p(\alpha|m_0, \Lambda_0)} \\ - \int d\phi q(\phi) \log \frac{q(\phi)}{p(\phi|a_0, b_0)}.$$

The value of \mathcal{F} at obtained optimal distributions (i.e. at $q(\pi) = q^*(\pi)$, $q(\alpha) = q^*(\alpha)$, $q(\phi) = q^*(\phi)$, $q(s_i) = q^*(s_i)$) is given as

$$\mathcal{F}(q^*(\pi), q^*(\alpha), q^*(\phi), \{q^*(s_i)\}_{s_i=1}^S, \{B(\theta^{s_i})\}_{s_i=1}^S, c_0, m_0, \\ \Lambda_0, a_0, b_0, \{\mathbf{m}^{s_i}\}_{s_i=1}^S) \\ = \frac{N}{2} (\Psi(b) + \log(a) - \log(2\pi)) - \frac{ab}{2} \sum_{s_i=1}^S q^*(s_i) r(m^{s_i}, \Lambda^{s_i}, \theta^{s_i}) \\ + \sum_{s_i=1}^S q^*(s_i) [\Psi(cd_{s_i}) - \Psi(c) - \log(q^*(s_i))] - \log \left(\frac{\Gamma(c)}{\Gamma(c_0)} \right) \\ - \sum_{s_i=1}^S \log \left(\frac{\Gamma(c_0/S)}{\Gamma(cd_{s_i})} \right) - \sum_{s_i=1}^S [cd_{s_i} - \frac{c_0}{S}] [\Psi(cd_{s_i}) - \Psi(c)] \\ - \sum_{s_i=1}^S \left[\frac{1}{2} \log \left(\frac{|\Lambda_0^{s_i} - 1|}{|(\Lambda^{s_i})^{-1}|} \right) + \frac{1}{2} Tr(\Lambda_0^{s_i} (\Lambda^{s_i})^{-1}) \right] \\ + \frac{1}{2} (m^{s_i} - m_0^{s_i})^T \Lambda_0^{s_i} (m^{s_i} - m_0^{s_i}) - \frac{K}{2} \Big] - \log(\Gamma(b_0)) \\ - b_0 \log(a_0) + \log(\Gamma(b)) + b_0 \log(a) - b\Psi(b) + b_0\Psi(b) \\ - b \frac{a}{a_0} + b.$$

At this point, we observe that for the given values of probability mass functions $\{q^*(s_i)\}_{s_i=1}^S$ and parameters $(c, \{d_{s_i}\}_{s_i=1}^S, \{\Lambda^{s_i}\}_{s_i=1}^S, \{\mathbf{m}^{s_i}\}_{s_i=1}^S, a, b)$, an increase in the value of \mathcal{F} will occur if parameters $\{\theta^{s_i}\}_{s_i=1}^S$ are optimized based on the following optimization problem:

$$\theta^{s_i,*} = \arg \min_{\theta^{s_i}} [r(m^{s_i}, \Lambda^{s_i}, \theta^{s_i}); c^{s_i} \theta^{s_i} \geq h^{s_i}].$$

IV. THE ALGORITHMS

The rules for updating the parameters of the distributions $(q(\pi), q(\alpha), q(\phi), q(s_i))$ and antecedents of fuzzy filters are summarized in the followings.

$$\theta^{s_i,*} = \arg \min_{\theta^{s_i}} [r(m^{s_i}, \Lambda^{s_i}, \theta^{s_i}); c^{s_i} \theta^{s_i} \geq h^{s_i}]$$

$$\Lambda^{s_i} = \Lambda_0^{s_i} + q^*(s_i)ab(B(\theta^{s_i,*}))^T B(\theta^{s_i,*})$$

$$m^{s_i} = [\Lambda_0^{s_i} + q^*(s_i)ab(B(\theta^{s_i,*}))^T B(\theta^{s_i,*})]^{-1} [\Lambda_0^{s_i} m_0^{s_i} + q^*(s_i)ab(B(\theta^{s_i,*}))^T Y]$$

$$q^*(s_i) = \frac{1}{Z} \exp\left(\Psi(cd_{s_i}) - \frac{ab}{2}r(m^{s_i}, \Lambda^{s_i}, \theta^{s_i,*})\right)$$

where Z is s.t. $\sum_{s_i=1}^S q^*(s_i) = 1$.

$$b = b_0 + \frac{N}{2}$$

$$\frac{1}{a} = \frac{1}{a_0} + \frac{1}{2} \sum_{s_i=1}^S q^*(s_i)r(m^{s_i}, \Lambda^{s_i}, \theta^{s_i,*})$$

$$c = c_0 + 1$$

$$d_{s_i} = \frac{1}{c_0 + 1} \left(\frac{c_0}{S} + q^*(s_i) \right)$$

Here, in the expressions for Λ^{s_i} and m^{s_i} , the term $\int d\phi q(\phi)\phi$ has been substituted as ab .

Each of these update rules is guaranteed to monotonically increase the objective function \mathcal{F} . Therefore, several iterations of update rules can be performed to increase \mathcal{F} until a consistent solution is reached. The iterative optimization process can be terminated if the increase in \mathcal{F} from an iteration to the next is less than a tolerance limit. Algorithm 2 summarizes our method for inferring the parameters of the fuzzy filters combination via maximizing \mathcal{F} .

Remark 2: The Problem 2 is simpler and a particular case of the Problem 3 where there is only one fuzzy filter whose parameters need to be inferred. The update rules in this case are summarized in algorithm 3.

Remark 3: Algorithm 3 can be started with an initial guess of $m|_0 = m_0$, $\Lambda|_0 = \Lambda_0$, $a|_0 = a_0$, $b|_0 = b_0$. And $\theta^*|_0$, in the case of grid partitioning, can be set for the uniformly distributed membership functions in the corresponding inputs' ranges. Regarding the starting point of algorithm 2, one can do the following:

- 1) Infer independently the parameters of each of the S filters using algorithm 3.
- 2) Let $(\theta_{single}^{s_i,*}, \Lambda_{single}^{s_i}, m_{single}^{s_i}, a_{single}^{s_i}, b_{single}^{s_i})$ denote the parameters of the s_i -th filter returned by algorithm 3 and $\mathcal{F}_{single}^{s_i}$ be the corresponding value of the optimized objective function. Now, the initial guess can be chosen as

$$q^*(s_i)|_0 \propto \exp\left(\mathcal{F}_{single}^{s_i}\right), \text{ where } \sum_{s_i=1}^S q^*(s_i)|_0 = 1;$$

$$\theta^{s_i,*}|_0 = \theta_{single}^{s_i,*}; m^{s_i}|_0 = m_{single}^{s_i}, \Lambda^{s_i}|_0 = \Lambda_{single}^{s_i};$$

$$\frac{1}{a|_0} = \frac{1}{a_0} + \frac{1}{2} \sum_{s_i=1}^S q^*(s_i)|_0 r(m_{single}^{s_i}, \Lambda_{single}^{s_i}, \theta_{single}^{s_i,*});$$

$$b|_0 = b_0 + \frac{N}{2}; c|_0 = c_0 + 1;$$

Algorithm 2 An algorithm for VB inference of the fuzzy filters combination

Require: Data pairs $\{x(j), y(j)\}_{j=1, \dots, N}$.

- 1: Choose a total of S fuzzy filters' structures $\{m^{s_i}\}_{s_i=1}^S$; hyperparameters $c_0, m_0, \Lambda_0, a_0, b_0$ which define the regularizing priors; parameters $\{c^{s_i}, h^{s_i}\}_{s_i=1}^S$ such that the interpretability constraints on the membership functions of the s_i -th filter can be formulated as $c^{s_i}\theta^{s_i} \geq h^{s_i}$.
- 2: Set iteration count $t = 0$ and choose a tolerance limit (say, equal to 0.01%).
- 3: **if** $\left(\max_{s_i} abs(q^*(s_i)|_{t+1} - q^*(s_i)|_t) < 0.0001 \right)$ & $(t > 0)$ **then**
- 4: **return** $(\theta^{1,*}|_{t+1}, \dots, \theta^{S,*}|_{t+1})$ and $(c|_{t+1}, \{d_{s_i}|_{t+1}\}_{s_i=1}^S, \{\Lambda^{s_i}|_{t+1}\}_{s_i=1}^S, \{m^{s_i}|_{t+1}\}_{s_i=1}^S, \{q^*(s_i)|_{t+1}\}_{s_i=1}^S, a|_{t+1}, b|_{t+1})$.
- 5: **else**
- 6: Update the parameters as follows

$$\theta^{s_i,*}|_{t+1} = \arg \min_{\theta^{s_i}} [r(m^{s_i}|_t, \Lambda^{s_i}|_t, \theta^{s_i}); c^{s_i}\theta^{s_i} \geq h^{s_i}]$$

$$\Lambda^{s_i}|_{t+1} = \Lambda_0^{s_i} + q^*(s_i)|_t a|_t b|_t (B(\theta^{s_i,*}|_{t+1}))^T B(\theta^{s_i,*}|_{t+1})$$

$$m^{s_i}|_{t+1} = [\Lambda_0^{s_i} + q^*(s_i)|_t a|_t b|_t (B(\theta^{s_i,*}|_{t+1}))^T B(\theta^{s_i,*}|_{t+1})]^{-1} [\Lambda_0^{s_i} m_0^{s_i} + q^*(s_i)|_t a|_t b|_t (B(\theta^{s_i,*}|_{t+1}))^T Y]$$

$$q^*(s_i)|_{t+1} = \frac{1}{Z} \exp\left(\Psi(c|_t d_{s_i}|_t) - \frac{a|_t b|_t}{2} r(m^{s_i}|_{t+1}, \Lambda^{s_i}|_{t+1}, \theta^{s_i,*}|_{t+1})\right)$$

where Z is s.t. $\sum_{s_i=1}^S q^*(s_i)|_{t+1} = 1$.

$$b|_{t+1} = b_0 + \frac{N}{2}$$

$$\frac{1}{a|_{t+1}} = \frac{1}{a_0} + \frac{1}{2} \sum_{s_i=1}^S q^*(s_i)|_{t+1} r(m^{s_i}|_{t+1}, \Lambda^{s_i}|_{t+1}, \theta^{s_i,*}|_{t+1})$$

$$c|_{t+1} = c_0 + 1$$

$$d_{s_i}|_{t+1} = \frac{1}{c_0 + 1} \left(\frac{c_0}{S} + q^*(s_i)|_{t+1} \right)$$

Here, $(\{m^{s_i}|_0\}_{s_i=1}^S, \{\Lambda^{s_i}|_0\}_{s_i=1}^S, a|_0, b|_0, \{\theta^{s_i,*}|_0\}_{s_i=1}^S, \{q^*(s_i)|_0\}_{s_i=1}^S, c|_0, \{d_{s_i}|_0\}_{s_i=1}^S)$ denote the initial guess.

7: **end if**

$$d_{s_i}|_0 = \frac{1}{c_0 + 1} \left(\frac{c_0}{S} + q^*(s_i)|_0 \right).$$

Here, the posterior distribution of the indicator variable is initiated proportional to the exponential of the lower bound on logarithmic evidence. The reason being that algorithm 2, like expectation-maximization (EM) algorithm, has the chances of being trapped in a local maxima. Therefore, a good starting point (obtained by the inference of component models using algorithm 3) can be taken to reduce the chances of algorithm convergence to a local maxima.

Remark 4: After optimizing \mathcal{F} via algorithm 2, one finds for a model m^{i^*} ,

$$q^*(s_i = i^*) \approx 1 \text{ and}$$

$$q^*(s_i) \approx 0 \text{ for } s_i = 1, \dots, i^* - 1, i^* + 1, \dots, S.$$

Algorithm 3 An algorithm for VB inference of fuzzy filter parameters

Require: Data pairs $\{x(j), y(j)\}_{j=1, \dots, N}$.

- 1: Choose a fuzzy filter structure m ; hyper-parameters m_0, Λ_0, a_0, b_0 which define the regularizing priors; parameters c, h such that the interpretability constraints on the membership functions can be formulated as $c\theta \geq h$.
- 2: Set iteration count $t = 0$ and choose a tolerance limit (say, equal to 0.01%).
- 3: **if** $(\mathcal{F}|_{t+1} - \mathcal{F}|_t < 0.0001\mathcal{F}|_t)$ & $(t > 0)$ **then**
- 4: **return** $(\theta^*|_{t+1}, \Lambda|_{t+1})$ and $(m|_{t+1}, a|_{t+1}, b|_{t+1})$.
- 5: **else**
- 6: Update the parameters as follows

$$\theta^*|_{t+1} = \arg \min_{\theta} [r(m|_t, \Lambda|_t, \theta); c\theta \geq h].$$

$$\Lambda|_{t+1} = \Lambda_0 + a|_t b|_t (B(\theta^*|_{t+1}))^T B(\theta^*|_{t+1})$$

$$m|_{t+1} = [\Lambda_0 + a|_t b|_t (B(\theta^*|_{t+1}))^T B(\theta^*|_{t+1})]^{-1} [\Lambda_0 m_0 + a|_t b|_t (B(\theta^*|_{t+1}))^T Y]$$

$$b|_{t+1} = b_0 + \frac{N}{2}$$

$$\frac{1}{a|_{t+1}} = \frac{1}{a_0} + \frac{1}{2} r(m|_{t+1}, \Lambda|_{t+1}, \theta^*|_{t+1})$$

Here, $(m|_0, \Lambda|_0, a|_0, b|_0, \theta^*|_0)$ denote the initial guess and $\mathcal{F}|_t$ is computed as

$$\begin{aligned} \mathcal{F}|_t = & \frac{N}{2} (\Psi(b|_t) + \log(a|_t)) - \frac{N}{2} \log(2\pi) \\ & - \frac{a|_t b|_t}{2} r(m|_t, \Lambda|_t, \theta^*|_t) - \frac{1}{2} \log \left(\frac{|\Lambda_0|^{-1}}{|\Lambda|_t|^{-1}} \right) \\ & - \frac{1}{2} Tr(\Lambda_0 (\Lambda|_t)^{-1}) - \frac{1}{2} (m|_t - m_0)^T \Lambda_0 (m|_t - m_0) \\ & + \frac{K}{2} - \log(\Gamma(b_0)) - b_0 \log(a_0) + \log(\Gamma(b|_t)) \\ & + b_0 \log(a|_t) - b|_t \Psi(b|_t) + b_0 \Psi(b|_t) - b|_t \frac{a|_t}{a_0} + b|_t. \end{aligned}$$

7: **end if**

That is, i^* -th fuzzy filter is the winner model that takes the most responsibility of the data. Therefore, algorithm 2 is capable of automatically selecting the most suitable fuzzy filter out of the set and inferring its parameters.

Remark 5: The algorithms 2 and 3 were implemented in MATLAB 6.5 [56]. The first update of the step 6 in both algorithms involves a nonlinear constrained optimization problem. The parameters estimation, based on the nonlinear optimization problem, was performed by running a single iteration of the algorithm “*fmincon*” available in MATLAB Optimization Toolbox [57].

V. SIMULATION STUDIES

A. Study 1

The first example, taken from [45], deals with the identification of a noisy time series:

$$\begin{aligned} x_j &= 1.5x_{j-1} \exp\left(-\frac{x_{j-1}^2}{4}\right) + \epsilon_j, \quad \epsilon_j \sim N(0, 1) \\ y_j &= x_j + v_j. \end{aligned}$$

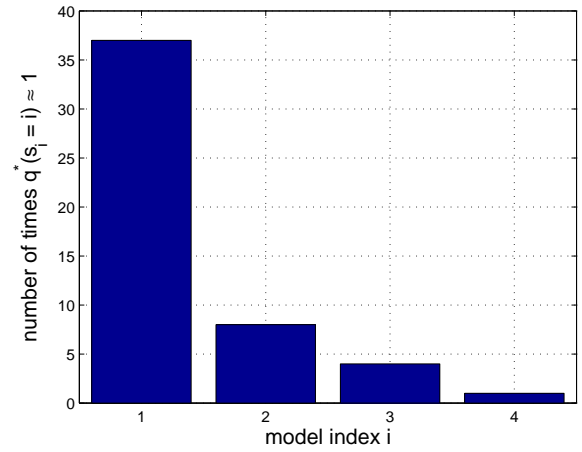


Fig. 1. The histogram of winner model index for study 1

Here, v_j is generated by a gross error model defined as

$$F = (1 - \epsilon)G + \epsilon H$$

where F is the noise distribution and G, H are the probability distributions that occur with probabilities $1 - \epsilon$ and ϵ , respectively. As in [45], the gross error model with $\epsilon = 0.05$, $G \sim N(0, 0.05)$, and $H \sim N(0, 3)$ was taken for the simulation study. It is required to train a fuzzy model with noisy input-output pairs $\{y_{j-1}, y_j\}$ to approximate the true function (i.e. $f(x) = 1.5x \exp(-\frac{x^2}{4})$). The training data set consists of 200 input-output pairs and the testing time series has 400 input-output pairs generated from the true function.

A set of 4 different fuzzy filters (say, m^1, m^2, m^3, m^4), of the type discussed in Appendix A-A which define one-dimensional clustering based membership functions on the input variable, was considered. The m^1, m^2, m^3, m^4 define 3, 4, 5, 6 number of membership functions respectively. Algorithm 2, initialized as per the suggestions of remark 3, was used to infer the parameters of fuzzy filters combination. The priors were taken as follows:

- 1) $m_0^{s_i}$ equal to the zero vector,
- 2) $\Lambda_0^{s_i}$ equal to the unity matrix,
- 3) $a_0 = 10^6, b_0 = 10^{-6}$ (i.e. relatively noninformative priors for the uncertainty).
- 4) c_0 equal to S .

The matrix c^{s_i} and vector h^{s_i} were designed to incorporate the following constrains on membership functions:

- 1) Any two consecutive knots must be separated at least by a distance of 0.1.
- 2) The extreme left knot must be greater than the minimum value of input variable in training set and the extreme right one less than the maximum value of input variable in the training set.

The experiment was repeated 50 times on the independently generated data sets. As stated in remark 4, each time the algorithm converged to choose the winner model. The minimum and maximum iterations of the algorithm during the experiments were 2 and 6 respectively. Figure 1 plots the counts that a model was chosen as the winner one. The

TABLE I
SIMULATIONS RESULTS OF STUDY 1

method	number of fuzzy rules	number of model parameters	RMSE
algorithm 2	3.38 ± 0.7253	10.14 ± 2.176	0.3433 ± 0.1221
Best known result from [45]	5	20	0.4200

performance of the winner fuzzy model was evaluated by calculating the root mean square error (RMSE) on the testing data set. The first row of table I reports the mean and standard deviation of the results during 50 runs of the experiment.

B. Study 2

The second example has been taken from [46] where the aim is to approximate the following nonlinear function:

$$f(x) = \left[a_1 + a_2 \frac{x}{b} + a_3 \left(\frac{2x^2}{b^2} - 1 \right) \right] \exp \left(-\frac{x^2}{2b^2} \right)$$

where $a_1 = 1$, $a_2 = 2$, $a_3 = -2$, and $b = 1.8$. The training data set consists of 100 data pairs $\{x(j), y(j)\}_{j=1}^{100}$. Here, $x(j)$ is a random number generated from a uniform distribution on $[-10, 10]$ and $y(j) = f(x(j)) + n_j$, where n_j is a uniform random number on $[-2.5, 2.5]$. The approximation quality was measured in term of root mean squared error (RMSE) on uniform grid of 201 points on $[-10, 10]$.

A set of 4 different fuzzy filters (say, m^1, m^2, m^3, m^4), of the type discussed in Appendix A-B, was considered. Following the approach of [46], fuzzy c-mean clustering was used to initialize the membership functions. The m^1, m^2, m^3, m^4 define 4, 5, 6, 7 membership functions respectively. Algorithm 2 was used to infer the parameters of fuzzy filters combination. The priors were same as in study 1 except that $m_0^{s_i} = m_{single}^{s_i}$ and $\Lambda_0^{s_i} = \Lambda_{single}^{s_i}$, where the parameters $(\Lambda_{single}^{s_i}, m_{single}^{s_i})$ were returned by algorithm 3 taking $m_0^{s_i}$ equal to the zero vector and $\Lambda_0^{s_i}$ equal to the unity matrix. The tuning of membership functions was constrained as follows:

- 1) The variances of Gaussian membership functions should take their values within 50 – 150% of their initial values obtained by clustering.
- 2) The centers of Gaussian membership functions should take their values within $\pm\sqrt{\text{variance}}$ of their initial values obtained by clustering.

As in [46], the experiment was repeated independently 50 times. The number of iterations of algorithm 2 was observed to be varying from 3 to 164 with an average equal to 13.66. Figure 2 and table II show the obtained results.

C. Study 3

A real-world example of modeling an ECG signal was taken from [46]. The ECG signal data belong to the MIT-BIH database. The aim is to identify a model that predicts the current signal value $s(j)$ using the previous four values $s(j-1)$, $s(j-2)$, $s(j-3)$, and $s(j-4)$. The model inputs and output are defined as

$$x(j) = [s(j-1)s(j-2)s(j-3)s(j-4)]^T \in R^4,$$

$$y(j) = s(j).$$

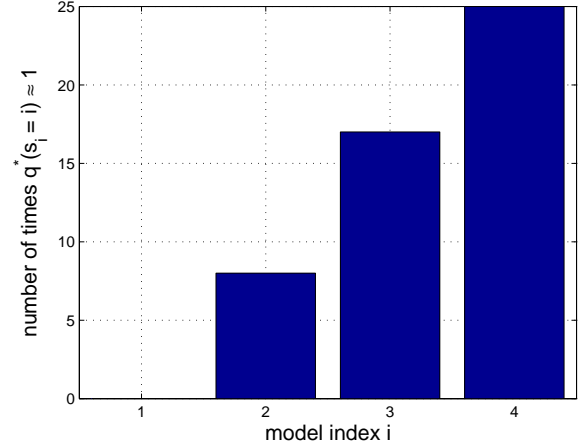


Fig. 2. The histogram of winner model index for study 2

As in [46], the fuzzy model was trained with 450 data pairs and tested on another 5000 pairs.

Algorithm 3 was used to infer the parameters of a fuzzy filter (of the type of Appendix A-B) that defines 3 different clusters on the input data. Following the method of [46], the initial parameters of the membership functions were obtained from fuzzy clustering of the input training data. The tuning of membership functions was constrained in the same way as in study 2. The priors were taken as follows:

- 1) m_0 equal to the zero vector,
- 2) Λ_0 equal to the unity matrix,
- 3) $a_0 = 10^6, b_0 = 10^{-6}$.

The algorithm converged after 25 iterations. The results of the experiment are shown in table III.

D. Study 4

Consider a process model

$$y = f(x_1, x_2) + n,$$

$$f(x_1, x_2) = (1 - x_1 x_2) e^{-(x_1 + x_2)^2} - \cos(4x_1 x_2) + \log(1 + x_1 x_2),$$

where x_1, x_2 are chosen from a uniform distribution over $[-0.9, 0.9]$, and n is a random variable normally distributed with zero mean and some fixed variance. The aim is to filter the uncertainty n from y using a fuzzy model. A total 500 pairs of inputs-output data $(\{x(j) = [x_1(j) \ x_2(j)]^T, y(j)\}_{j=1}^{500})$ were extracted for inferring the parameters of a fuzzy filter. Table IV lists the details of the considered fuzzy models for the purpose of filtering. The performance of a filter was measured

TABLE II
SIMULATIONS RESULTS OF STUDY 2

method	number of fuzzy rules	number of model parameters	RMSE
algorithm 2	6.34 ± 0.7453	25.36 ± 2.9813	0.4737 ± 0.1022
Best known result from [46]	8	32	0.4836 ± 0.0820

TABLE III
SIMULATIONS RESULTS OF STUDY 3

method	number of fuzzy rules	number of model parameters	RMSE
algorithm 2	3	39	0.0150
Best known result from [46]	8	104	0.0214

TABLE IV
THE DIFFERENT FUZZY FILTER STRUCTURES IN STUDY 4

model	number of membership functions for each input	membership functions type (grid partitioning)	model order
m ¹	5	triangular	zero
m ²	4	gaussian	zero
m ³	3	clustering	zero
m ⁴	2	triangular	zero
m ⁵	2	gaussian	zero
m ⁶	2	clustering	zero

by calculating the energy of filtering errors defined as

$$FEE = \sum_{j=1}^{500} |f(x_1(j), x_2(j)) - G^T(x(j), \theta^*)m|^2$$

where (m, θ^*) are returned by algorithm 3. Algorithm 3 was used to infer the parameters of each of the 6 filters with the same priors and constraints as in study 1. The nonlinear optimization problem was solved using the MATLAB algorithm “*fmincon*” with its default settings regarding the maximum number of iterations and other parameters.

For a fixed variance of n , algorithm 3 was run several times on the different independently generated 500 pairs of inputs-output data. The results of 40 independent runs of algorithm 3 are visualized in figures 3 and 4. Figure 3 shows for each model the scatter plots between filtering errors energy and negative free energy. Following inferences can be made from figures 3 and 4:

- The higher values of filtering errors energy correspond to the lower values of negative free energy and vice versa. This demonstrates the negative free energy based models comparison capability of the approach.
- m¹, as seen from figure 4(a), is the most suitable model in the case of lower magnitude uncertainties while m³, due to its slightly better performance in terms of filtering errors energy, should be preferred in the case of higher magnitude uncertainties.

Algorithm 2 was run on each of the 40 independently generated data sets. Figure 5 plots the counts that a model was chosen as the winner one.

Following remarks can be made on the aforementioned simulation studies:

- 1) A comparison between first and second row of tables I, II, and III demonstrates the effectiveness of our VB based approach in terms of performance and model complexity.
- 2) Figure 5 shows the effectiveness of the algorithm 2 in choosing the right structure of the fuzzy model, i.e., m¹ in the case of $n \sim N(0, 0.05)$ and m³ in the case of $n \sim N(0, 0.5)$.

VI. CONCLUDING REMARKS

This study has introduced a mixed Takagi-Sugeno fuzzy filter whose antecedents are deterministic while the consequents are random variables. The parameters of the fuzzy filter are inferred under VB framework. The desired features (i.e. “automated regularization”, “model comparisons”, and “handling uncertainty via incorporating statistical noise models”) of the VB framework have been exploited to suggest an algorithm that selects the most suitable fuzzy filter out of the set and infers its parameters. The simulation studies verify the feasibility of the method.

The limitations of our method are following.

- 1) The algorithms, like expectation-maximization (EM) algorithm, might have the chances of being trapped in a local maxima.
- 2) A closed-form expression for the updating of nonlinear antecedent, unlike other parameters updates, is not avail-

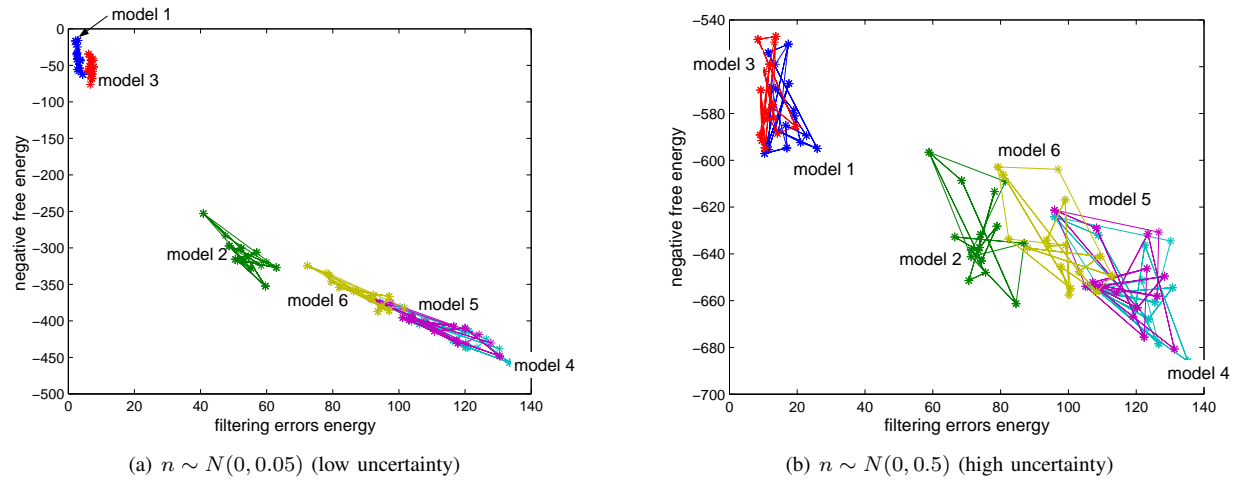


Fig. 3. The scatter plots between the values of filtering errors energy and negative free energy

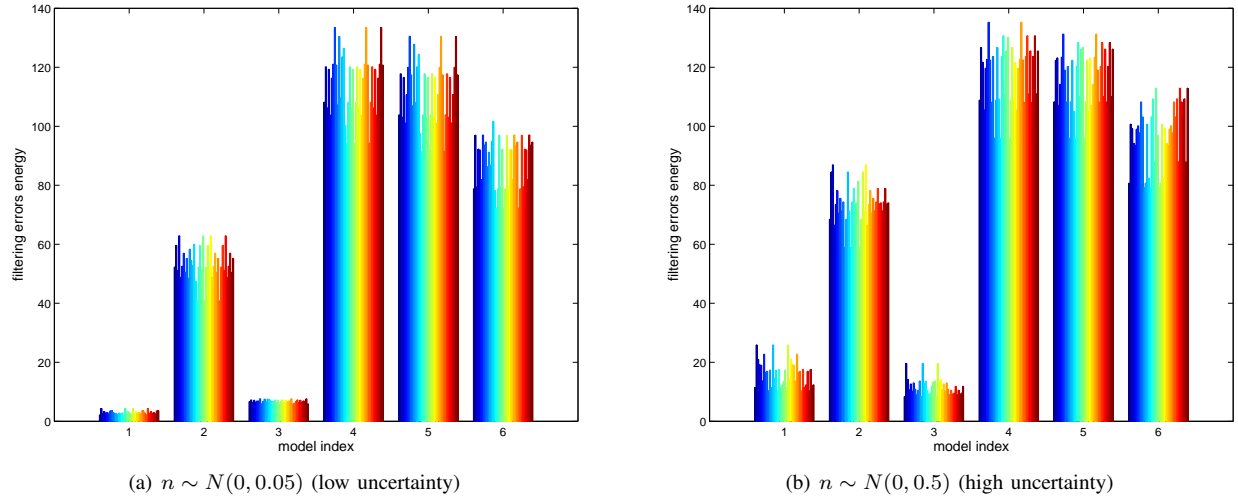


Fig. 4. The bar plots of filtering errors energy values obtained during 40 independent experiments

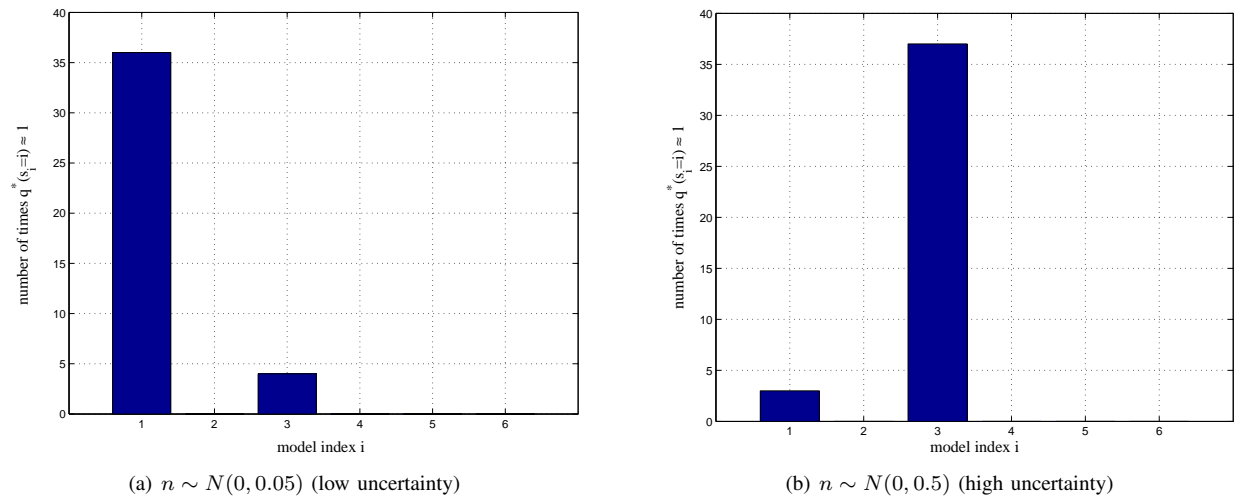


Fig. 5. The histogram of winner model index for study 4

able. This is computationally the most expensive step of the algorithms.

These two issues will be addressed in our future study. The scope for future research includes

- 1) a deterministic robustness analysis of the algorithm 3,
- 2) the mean-squared-error analysis of the algorithm 3,
- 3) the generalization of the proposed algorithms to non Gaussian noise models,
- 4) an interpretation of the distribution $q(\alpha)$ as a multivariate membership function for consequents and thus constructing a functionally equivalent deterministic fuzzy filter.

The main motivation behind our study is derived from the idea of combining the uncertainties handling capabilities of fuzzy membership functions with that of statistical models. The authors are optimistic that this integrated framework (fuzzy modeling + statistical modeling) has much to offer in the field of computational intelligence and machine learning.

ACKNOWLEDGMENT

The authors would like to thank Sukhvir Singh, Department of Computer Science, Himachal Pradesh University for his support in the development of a software of the algorithms.

APPENDIX A A TAKAGI-SUGENO FUZZY FILTER

Consider a Takagi-Sugeno fuzzy model ($F_s : X \rightarrow Y$) that maps n -dimensional real input space ($X = X_1 \times X_2 \times \dots \times X_n$) to one dimensional real line.

A. Grid Partitioning of Input Space

A rule of the model is represented as

If x_1 is A_1 and \dots and x_n is A_n then $y_f = s_0 + \sum_{j=1}^n s_j x_j$.

Here (x_1, \dots, x_n) are the model input variables, y_f is the filtered output variable, (A_1, \dots, A_n) are the linguistic terms which are represented by fuzzy sets, and (s_0, s_1, \dots, s_n) are real scalars. Given a universe of discourse X_j , a fuzzy subset A_j of X_j is characterized by a mapping:

$$\mu_{A_j} : X_j \rightarrow [0, 1],$$

where for $x_j \in X_j$, $\mu_{A_j}(x_j)$ can be interpreted as the degree or grade to which x_j belongs to A_j . This mapping is called as membership function of the fuzzy set. Let us define, for j^{th} input, P_j non-empty fuzzy subsets of X_j (represented by $A_{1j}, A_{2j}, \dots, A_{P_jj}$). Let the i^{th} rule of the rule-base is represented as

$$R_i : \quad \text{If } x_1 \text{ is } A_{i1} \text{ and } \dots \text{ and } x_n \text{ is } A_{in} \text{ then} \\ y_f = s_{i0} + s_{i1}x_1 + \dots + s_{in}x_n,$$

where $A_{i1} \in \{A_{11}, \dots, A_{P_11}\}$, $A_{i2} \in \{A_{12}, \dots, A_{P_22}\}$ and so on. Now, the different choices of $A_{i1}, A_{i2}, \dots, A_{in}$ leads to the $K = \prod_{j=1}^n P_j$ number of fuzzy rules. For a given input vector $x = [x_1 \dots x_n]^T \in R^n$, the *degree of fulfillment* of the

i^{th} rule, by modeling the logic operator ‘and’ using product, is given by

$$g_i(x) = \prod_{j=1}^n \mu_{A_{ij}}(x_j).$$

The output of the fuzzy model to input vector x is computed by taking the weighted average of the output provided by each rule:

$$y_f = \frac{\sum_{i=1}^K (s_{i0} + s_{i1}x_1 + \dots + s_{in}x_n) g_i(x)}{\sum_{i=1}^K g_i(x)} \\ = \frac{\sum_{i=1}^K (s_{i0} + s_{i1}x_1 + \dots + s_{in}x_n) \prod_{j=1}^n \mu_{A_{ij}}(x_j)}{\sum_{i=1}^K \prod_{j=1}^n \mu_{A_{ij}}(x_j)}. \quad (6)$$

Let us define a real vector θ such that the membership functions of any type (e.g. trapezoidal, triangular, etc) can be constructed from the elements of vector θ . To illustrate the construction of membership functions based on knot vector (θ) , consider the following examples:

a) *Triangular membership functions:* Let

$$\theta = (t_1^0, t_1^1, \dots, t_1^{P_1-2}, t_1^{P_1-1}, \dots, t_n^0, t_n^1, \dots, t_n^{P_n-2}, t_n^{P_n-1})$$

such that for i^{th} input, $t_i^0 < t_i^1 < \dots < t_i^{P_i-2} < t_i^{P_i-1}$ holds for all $i = 1, \dots, n$. Now, P_i triangular membership functions for i^{th} input ($\mu_{A_{1i}}, \mu_{A_{2i}}, \dots, \mu_{A_{P_ii}}$) can be defined as:

$$\mu_{A_{1i}}(x_i, \theta) = \max \left(0, \min \left(1, \frac{t_i^1 - x_i}{t_i^1 - t_i^0} \right) \right)$$

$$\mu_{A_{ji}}(x_i, \theta) = \max \left(0, \min \left(\frac{x_i - t_i^{j-2}}{t_i^{j-1} - t_i^{j-2}}, \frac{t_i^j - x_i}{t_i^j - t_i^{j-1}} \right) \right),$$

for all $j = 2, \dots, P_i - 1$.

$$\mu_{A_{P_i i}}(x_i, \theta) = \max \left(0, \min \left(\frac{x_i - t_i^{P_i-2}}{t_i^{P_i-1} - t_i^{P_i-2}}, 1 \right) \right)$$

b) *One-dimensional clustering criterion based membership functions:* Let

$$\theta = (t_1^0, t_1^1, \dots, t_1^{P_1-2}, t_1^{P_1-1}, \dots, t_n^0, t_n^1, \dots, t_n^{P_n-2}, t_n^{P_n-1})$$

such that for i^{th} input, $t_i^0 < t_i^1 < \dots < t_i^{P_i-2} < t_i^{P_i-1}$ holds for all $i = 1, \dots, n$. Consider the problem of assigning two different memberships (say $\mu_{A_{1i}}$ and $\mu_{A_{2i}}$) to a point x_i such that $t_i^0 < x_i < t_i^1$, based on following clustering criterion:

$$[\mu_{A_{1i}}(x_i), \mu_{A_{2i}}(x_i)] = \arg \min_{[u_1, u_2]} [u_1^2(x_i - t_i^0)^2 \\ + u_2^2(x_i - t_i^1)^2, u_1 + u_2 = 1].$$

This results in

$$\mu_{A_{1i}}(x_i) = \frac{(x_i - t_i^1)^2}{(x_i - t_i^0)^2 + (x_i - t_i^1)^2}, \text{ and}$$

$$\mu_{A_{2i}}(x_i) = \frac{(x_i - t_i^0)^2}{(x_i - t_i^0)^2 + (x_i - t_i^1)^2}.$$

Thus, for i^{th} input, P_i membership functions can be defined as:

$$\mu_{A_{1i}} = \begin{cases} 1 & x_i \leq t_i^0 \\ \frac{(x_i - t_i^1)^2}{(x_i - t_i^0)^2 + (x_i - t_i^1)^2} & t_i^0 \leq x_i \leq t_i^1 \\ 0 & \text{otherwise} \end{cases}$$

$$\mu_{A_{ji}} = \begin{cases} \frac{(x_i - t_i^{j-2})^2}{(x_i - t_i^{j-2})^2 + (x_i - t_i^{j-1})^2} & t_i^{j-2} \leq x_i \leq t_i^{j-1} \\ \frac{(x_j - t_i^j)^2}{(x_i - t_i^{j-1})^2 + (x_i - t_i^j)^2} & t_i^{j-1} \leq x_i \leq t_i^j \\ 0 & \text{otherwise} \end{cases}$$

for $j = 2, \dots, P_i - 1$.

$$\mu_{A_{P_i i}} = \begin{cases} 1 & x_i \geq t_i^{P_i-1} \\ \frac{(x_i - t_i^{P_i-2})^2}{(x_i - t_i^{P_i-2})^2 + (x_i - t_i^{P_i-1})^2} & t_i^{P_i-2} \leq x_i \leq t_i^{P_i-1} \\ 0 & \text{otherwise} \end{cases}$$

c) *Gaussian membership functions:* Let

$$\theta = (t_1^0, t_1^1, \dots, t_1^{P_1-2}, t_1^{P_1-1}, \dots, t_n^0, t_n^1, \dots, t_n^{P_n-2}, t_n^{P_n-1})$$

such that for i^{th} input, $t_i^0 < t_i^1 < \dots < t_i^{P_i-2} < t_i^{P_i-1}$ holds for all $i = 1, \dots, n$. Now, P_i Gaussian membership functions for i^{th} input can be defined as:

$$\mu_{A_{ji}}(x_i, \theta) = e^{-(x_i - t_i^{j-1})^2}, \quad j = 1, \dots, P_i$$

For any choice of membership functions (which can be constructed from a vector θ), (6) can be rewritten as function of θ :

$$y_f = \sum_{i=1}^K (s_{i0} + s_{i1}x_1 + \dots + s_{in}x_n) \tilde{G}_i(x, \theta),$$

$$\tilde{G}_i(x, \theta) = \frac{\prod_{j=1}^n \mu_{A_{ij}}(x_j, \theta)}{\sum_{i=1}^K \prod_{j=1}^n \mu_{A_{ij}}(x_j, \theta)}.$$

Let us introduce the following notation:

$$\alpha = \begin{bmatrix} s_{10} \\ s_{11} \\ \vdots \\ s_{1n} \\ \vdots \\ s_{K0} \\ s_{K1} \\ \vdots \\ s_{Kn} \end{bmatrix}, \quad G(x, \theta) = \begin{bmatrix} \tilde{G}_1(x, \theta) \\ x_1 \tilde{G}_1(x, \theta) \\ \vdots \\ x_n \tilde{G}_1(x, \theta) \\ \vdots \\ \tilde{G}_K(x, \theta) \\ x_1 \tilde{G}_K(x, \theta) \\ \vdots \\ x_n \tilde{G}_K(x, \theta) \end{bmatrix}$$

Now, we have

$$y_f = G^T(x, \theta)\alpha.$$

In this expression, θ is not allowed to be any arbitrary vector, since the elements of θ must $\forall i = 1, \dots, n$, ensure

$$a_i \leq t_i^0 < t_i^1 < \dots < t_i^{P_i-2} < t_i^{P_i-1} \leq b_i, \quad \text{where } x_i \in [a_i, b_i].$$

These inequalities and any other membership functions related constraints (designed for incorporating a priori knowledge) can be written in the form of a matrix inequality $c\theta \geq h$. Hence, a Takagi-Sugeno type fuzzy filter can be represented as

$$y_f = G^T(x, \theta)\alpha, \quad c\theta \geq h.$$

B. Fuzzy Clustering Based Partitioning of Input Space

Several studies have used fuzzy c-mean (or its robust alternatives) to find clusters in the input space and thus obtaining the parameters of the membership functions. Such methods define multivariate membership functions and corresponding to each cluster, there exists a fuzzy rule of the Takagi-Sugeno form:

$$R_i : \text{ If } x \text{ is } A_i \text{ then } y_f = s_{i0} + s_{i1}x_1 + \dots + s_{in}x_n,$$

$i = 1, 2, \dots, K$. The fuzzy set A_i (with a membership function $A_i(x) : R^n \rightarrow [0, 1]$) is typically defined with the Gaussian function:

$$\mu_{A_i}(x) = \prod_{j=1}^n \exp\left(-\frac{|x_j - t_j^{i,0}|^2}{2t_j^{i,1}}\right)$$

where $t_j^{i,0}$ is the center and $t_j^{i,1}$ is the dispersion of the membership function on x_j defined by the i -th cluster. The parameters of the membership functions (i.e. $t_j^{i,0}, t_j^{i,1}; j = 1, \dots, n; i = 1, \dots, K$) can be obtained from fuzzy clustering of the input data, see e.g. [46]. Let the parameters of the membership functions are collected in a vector θ , defined as

$$\theta = (t_1^{1,0}, t_1^{1,1}, \dots, t_n^{1,0}, t_n^{1,1}, \dots, t_1^{K,0}, t_1^{K,1}, \dots, t_n^{K,0}, t_n^{K,1}).$$

It is easy now to see that the fuzzy filter, like the grid partitioning case, can be still functionally represented as

$$y_f = G^T(x, \theta)\alpha, \quad \text{where}$$

$$\alpha = \begin{bmatrix} s_{10} \\ s_{11} \\ \vdots \\ s_{1n} \\ \vdots \\ s_{K0} \\ s_{K1} \\ \vdots \\ s_{Kn} \end{bmatrix}, \quad G(x, \theta) = \begin{bmatrix} \tilde{G}_1(x, \theta) \\ x_1 \tilde{G}_1(x, \theta) \\ \vdots \\ x_n \tilde{G}_1(x, \theta) \\ \vdots \\ \tilde{G}_K(x, \theta) \\ x_1 \tilde{G}_K(x, \theta) \\ \vdots \\ x_n \tilde{G}_K(x, \theta) \end{bmatrix}$$

$$\tilde{G}_i(x, \theta) = \frac{\mu_{A_i}(x, \theta)}{\sum_{i=1}^K \mu_{A_i}(x, \theta)}.$$

Although the parameters of membership functions (i.e. elements of vector θ) are obtained by fuzzy clustering, one may prefer a fine tuning of the elements of vector θ under a filtering

performance criterion. In this case, the tuning process can be constrained. A necessary constraint is that the variances of Gaussian membership functions must be greater than zero. Any type of constraints on the parameters of membership functions can be formulated as a matrix inequality $c\theta \geq h$.

REFERENCES

- [1] L. A. Zadeh, "Outline of a new approach to the analysis of complex systems and decision processes," *IEEE Trans. on Systems, Man, and Cybernetics*, vol. 3, pp. 28–44, Jan. 1973.
- [2] L. A. Zadeh, "The role of fuzzy logic in the management of uncertainty in expert systems," *Fuzzy Sets Systems*, vol. 11, pp. 199–227, 1983.
- [3] M. Kumar, M. Weippert, R. Vilbrandt, S. Kreuzfeld, and R. Stoll, "Fuzzy evaluation of heart rate signals for mental stress assessment," *IEEE Transactions on Fuzzy Systems*, vol. 15, no. 5, pp. 791–808, 2007.
- [4] M. Kumar, D. Arndt, S. Kreuzfeld, K. Thurow, N. Stoll, and R. Stoll, "Fuzzy techniques for subjective workload score modelling under uncertainties," *IEEE Transactions on Systems, Man, and Cybernetics-Part B: Cybernetics*, vol. 38, no. 6, pp. 1449–1464, 2008.
- [5] S. Kumar, M. Kumar, R. Stoll, and U. Kragl, "Handling uncertainties in toxicity modelling using a fuzzy filter," *SAR and QSAR in Environmental Research*, vol. 18, no. 7-8, pp. 645–662, 2007.
- [6] S. Kumar, M. Kumar, K. Thurow, R. Stoll, and U. Kragl, "Fuzzy filtering for robust bioconcentration factor modelling," *Environmental Modelling & Software*, vol. 24, no. 1, pp. 44–53, 2009.
- [7] M. Kumar, K. Thurow, N. Stoll, and R. Stoll, "Fuzzy handling of uncertainties in modeling the inhibition of glycogen synthase kinase-3 by paullones," in *Proc. IEEE International Conference on Automation Science and Engineering (CASE 2007)*, Scottsdale, Arizona USA, Sep. 2007, pp. 237–242.
- [8] M. Kumar, N. Stoll, D. Kaber, K. Thurow, and R. Stoll, "Fuzzy filtering for an intelligent interpretation of medical data," in *Proc. IEEE International Conference on Automation Science and Engineering (CASE 2007)*, Scottsdale, Arizona USA, Sep. 2007, pp. 225–230.
- [9] M. Kumar, K. Thurow, N. Stoll, and R. Stoll, "A fuzzy system for modeling the structure-activity relationships in presence of uncertainties," in *Proc. IEEE International Conference on Automation Science and Engineering (CASE 2008)*, Washington DC, USA, Aug. 2008, pp. 1025–1030.
- [10] M. Kumar, K. Thurow, N. Stoll, and R. Stoll, "Robust fuzzy mappings for QSAR studies," *European Journal of Medicinal Chemistry*, vol. 42, pp. 675–685, 2007.
- [11] L. X. Wang and J. M. Mendel, "Generating fuzzy rules by learning from examples," *IEEE Trans. on Systems, Man, and Cybernetics*, vol. 22, no. 6, pp. 1414–1427, 1992.
- [12] K. Nozaki, H. Ishibuchi, and H. Tanaka, "A simple but powerful heuristic method for generating fuzzy rules from numerical data," *Fuzzy Sets and Systems*, vol. 86, pp. 251–270, 1997.
- [13] J. J. Shan and H. C. Fu, "A fuzzy neural network for rule acquiring on fuzzy control systems," *Fuzzy Sets and Systems*, vol. 71, pp. 345–357, 1995.
- [14] D. Nauck and R. Kruse, "A neuro-fuzzy approach to obtain interpretable fuzzy systems for function approximation," in *Proc. IEEE International Conference on Fuzzy Systems 1998 (FUZZ-IEEE'98)*, Anchorage, AK, May 1998, pp. 1106–1111.
- [15] J.-S. R. Jang, "ANFIS: Adaptive-network-based fuzzy inference systems," *IEEE Transactions on Systems, Man, and Cybernetics*, vol. 23, pp. 665–685, May 1993.
- [16] P. Thrift, "Fuzzy logic synthesis with genetic algorithms," in *Proc. of the 4th Int. Conf. on Genetic Algorithms*, 1991, pp. 509–513.
- [17] J. Liska and S. S. Melsheimer, "Complete design of fuzzy logic systems using genetic algorithms," in *Proc. of the 3rd IEEE Int. Conf. on Fuzzy Systems*, 1994, pp. 1377–1382.
- [18] F. Herrera, M. Lozano, and J. Verdegay, "Generating fuzzy rules from examples using genetic algorithms," in *Proc. 5th Int. Conf. on Information Processing and Management of Uncertainty in Knowledge-Based Systems (IPMU'94)*, Paris, France, Jul. 1994, pp. 675–680.
- [19] A. González and R. Pérez, "Completeness and consistency conditions for learning fuzzy rules," *Fuzzy Sets and Systems*, vol. 96, pp. 37–51, 1998.
- [20] R. Babuška and H. Verbruggen, "Constructing fuzzy models by product space clustering," in *Fuzzy Model Identification: Selected Approaches*, H. Hellendoorn and D. Driankov, Eds. Berlin, Germany: Springer, 1997, pp. 53–90.
- [21] R. Babuška, *Fuzzy Modeling for Control*. Boston: Kluwer Academic Publishers, 1998.
- [22] J. Abonyi, R. Babuška, and F. Szeifert, "Modified Gath-Geva Fuzzy Clustering for Identification of Takagi-Sugeno Fuzzy Models," *IEEE Trans. on System, Man and Cybernetics, Part B*, pp. 612–621, Oct. 2002.
- [23] D. Simon, "Design and rule base reduction of a fuzzy filter for the estimation of motor currents," *International Journal of Approximate Reasoning*, vol. 25, pp. 145–167, Oct. 2000.
- [24] D. Simon, "Training fuzzy systems with the extended kalman filter," *Fuzzy Sets and Systems*, vol. 132, pp. 189–199, Dec. 2002.
- [25] J. S. R. Jang, C. T. Sun, and E. Mizutani, *Neuro-Fuzzy and Soft Computing: a Computational Approach to Learning and Machine Intelligence*. Upper Saddle River: Prentice-Hall, 1997.
- [26] W. Wang and J. Vrbanek, "An evolving fuzzy predictor for industrial applications," *IEEE Trans. on Fuzzy Systems*, vol. 16, no. 6, pp. 1439–1449, Dec. 2008.
- [27] E. Lughofer, "FLEXFIS: A Robust Incremental Learning Approach for Evolving TS Fuzzy Models," *IEEE Trans. on Fuzzy Systems*, vol. 16, no. 6, pp. 1393–1410, Dec. 2008.
- [28] M. Kumar, N. Stoll, and R. Stoll, "On the estimation of parameters of takagi-sugeno fuzzy filters," *IEEE Transactions on Fuzzy Systems*, vol. 17, no. 1, pp. 150–166, Feb. 2009.
- [29] C.-J. Lin, C.-H. Chen, and C.-T. Lin, "Efficient self-evolving evolutionary learning for neuro-fuzzy inference systems," *IEEE Trans. on Fuzzy Systems*, vol. 16, no. 6, pp. 1476–1490, Dec. 2008.
- [30] M. Kumar, N. Stoll, and R. Stoll, "Adaptive fuzzy filtering in a deterministic setting," *IEEE Transactions on Fuzzy Systems*, vol. 17, no. 4, pp. 763–776, Aug. 2009.
- [31] W. Y. Wang, T. T. Lee, C. L. Liu, and C. H. Wang, "Function approximation using fuzzy neural networks with robust learning algorithm," *IEEE Trans. Syst., Man., Cybern. B*, vol. 27, pp. 740–747, Sep. 1997.
- [32] M. Burger, H. Engl, J. Haslinger, and U. Bodenhofer, "Regularized data-driven construction of fuzzy controllers," *J. Inverse and Ill-posed Problems*, vol. 10, pp. 319–344, 2002.
- [33] W. Yu and X. Li, "Fuzzy identification using fuzzy neural networks with stable learning algorithms," *IEEE Trans. on Fuzzy Systems*, vol. 12, no. 3, pp. 411–420, Jun. 2004.
- [34] T. Johansen, "Robust identification of takagi-sugeno-kang fuzzy models using regularization," in *Proc. IEEE conf. Fuzzy Systems*, New Orleans, USA, 1996, pp. 180–186.
- [35] X. Hong, C. J. Harris, and S. Chen, "Robust neurofuzzy rule base knowledge extraction and estimation using subspace decomposition combined with regularization and D-optimality," *IEEE Trans. Syst., Man., Cybern. B*, vol. 34, no. 1, pp. 598–608, 2004.
- [36] J. Kim, Y. Suga, and S. Won, "A new approach to fuzzy modeling of nonlinear dynamic systems with noise: Relevance vector learning mechanism," *IEEE Trans. on Fuzzy Systems*, vol. 14, no. 2, pp. 222–231, Apr. 2006.
- [37] M. Kumar, R. Stoll, and N. Stoll, "Robust solution to fuzzy identification problem with uncertain data by regularization. Fuzzy approximation to physical fitness with real world medical data: An application," *Fuzzy Optimization and Decision Making*, vol. 3, no. 1, pp. 63–82, Mar. 2004.
- [38] M. Kumar, R. Stoll, and N. Stoll, "Robust adaptive fuzzy identification of time-varying processes with uncertain data. Handling uncertainties in the physical fitness fuzzy approximation with real world medical data: An application," *Fuzzy Optimization and Decision Making*, vol. 2, no. 3, pp. 243–259, Sep. 2003.
- [39] M. Kumar, R. Stoll, and N. Stoll, "SDP and SOCP for outer and robust fuzzy approximation," in *Proc. 7th IASTED International Conference on Artificial Intelligence and Soft Computing*, Banff, Canada, Jul. 2003.
- [40] M. Kumar, R. Stoll, and N. Stoll, "A robust design criterion for interpretable fuzzy models with uncertain data," *IEEE Trans. on Fuzzy Systems*, vol. 14, no. 2, pp. 314–328, Apr. 2006.
- [41] M. Kumar, R. Stoll, and N. Stoll, "A min-max approach to fuzzy clustering, estimation, and identification," *IEEE Trans. on Fuzzy Systems*, vol. 14, no. 2, pp. 248–262, Apr. 2006.
- [42] M. Kumar, R. Stoll, and N. Stoll, "Robust adaptive identification of fuzzy systems with uncertain data," *Fuzzy Optimization and Decision Making*, vol. 3, no. 3, pp. 195–216, Sep. 2004.
- [43] M. Kumar, N. Stoll, and R. Stoll, "An energy-gain bounding approach to robust fuzzy identification," *Automatica*, vol. 42, no. 5, pp. 711–721, May 2006.
- [44] M. Kumar, R. Stoll, and N. Stoll, "Deterministic approach to robust adaptive learning of fuzzy models," *IEEE Trans. Syst., Man., Cybern. B*, vol. 36, no. 4, pp. 767–780, Aug. 2006.

- [45] C. C. Chuang, S. F. Su, and S. S. Chen, "Robust TSK Fuzzy Modeling for Function Approximation With Outliers," *IEEE Transactions on Fuzzy Systems*, vol. 9, no. 6, pp. 810–821, Dec. 2001.
- [46] J. M. Leski, "TSK-Fuzzy Modeling Based on ϵ -Insensitive Learning," *IEEE Transactions on Fuzzy Systems*, vol. 13, no. 2, pp. 181–193, Apr. 2005.
- [47] C. F. Juang and C. D. Hsieh, "TS-fuzzy system-based support vector regression," *Fuzzy Sets and Systems*, vol. 160, no. 17, pp. 2486–2504, Sep. 2009.
- [48] L. Wang, Z. Mu, and H. Guo, "Fuzzy rule-based support vector regression system," *Journal of Control Theory and Applications*, vol. 3, no. 3, pp. 230–234, Aug. 2005.
- [49] C. T. Lin, S. F. Liang, C. M. Yeh, and K. W. Fan, "Fuzzy neural network design using support vector regression for function approximation with outliers," in *Proc. IEEE International Conference on System, Man, and Cybernetics*, vol. 3, Oct. 2005, pp. 2763–2768.
- [50] H. Attias, "A variational bayesian framework for graphical models," in *In Advances in Neural Information Processing Systems 12*. MIT Press, 2000, pp. 209–215.
- [51] H. Lappalainen and J. W. Miskin, "Ensemble learning," in *Advances in Independent Component Analysis*, M. Girolami, Ed. Springer-Verlag, 2000.
- [52] M. J. Beal, "Variational algorithms for approximate bayesian inference," Ph.D. dissertation, The Gatsby Computational Neuroscience Unit, University College London, London, UK, 2003.
- [53] M. W. Woolrich and T. E. Behrens, "Variational bayes inference of spatial mixture models for segmentation," *IEEE Transactions on Medical Imaging*, vol. 25, no. 10, pp. 1380–1391, Oct. 2006.
- [54] K. Friston, J. Mattout, N. Trujillo-Barreto, J. Ashburner, and W. Penny, "Variational free energy and the laplace approximation," *NeuroImage*, vol. 34, no. 1, pp. 220–234, 2007.
- [55] M. A. Chappell, A. R. Groves, B. Whitcher, and M. W. Woolrich, "Variational bayesian inference for a nonlinear forward model," *IEEE Transactions on Signal Processing*, vol. 57, no. 1, pp. 223–236, Jan. 2009.
- [56] "MATLAB - The Language of Technical Computing." [Online]. Available: <http://www.mathworks.com/products/matlab/>
- [57] "Optimization Toolbox - MATLAB." [Online]. Available: <http://www.mathworks.com/products/optimization/>



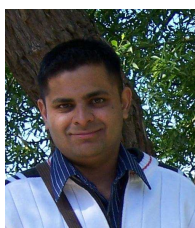
Norbert Stoll received the diploma (Dip.-Ing.) in automation engineering in 1979 and the Ph.D. degree in measurement technology in 1985 from Rostock University, Germany.

He served as head of section analytical chemistry at the Academy of Sciences of GDR, Central Institute for Organic Chemistry till 1991. From 1992 to 1994, he was the associate director of Institute of Organic Catalysis, Rostock, Germany. Since 1994, he is a professor of measurement technology in the engineering faculty of Rostock University. From 1994 to 2000, he directed the Institution of Automation in Rostock University. He is also holding, since 2003, the position of vice president in Center for Life Science Automation, Rostock. His fields of interests include medical process measurement, lab automation, and smart systems and devices.



Regina Stoll received the diploma in medicine (Dip.-Med.), the degree of "Dr.med." in occupational medicine, and the degree of "Dr.med.habil" in occupational and sports medicine from Rostock University, Germany in 1980, 1984, and 2002, respectively.

She is head of the Institute of Preventive Medicine, Rostock, Germany. She is a faculty member in the medicine faculty and faculty associate in the College of Computer Science and Electrical Engineering of Rostock University. She also holds the adjunct faculty member position in the industrial engineering department of North Carolina State University. Her research interests include occupational physiology, preventive medicine, and cardiopulmonary diagnostics.



Mohit Kumar received the B. Tech. degree in electrical engineering from National Institute of Technology, Hamirpur, India in 1999, the M. Tech. degree in control engineering from Indian Institute of Technology, Delhi, India in 2001, the Ph.D. degree (*summa cum laude*) in electrical engineering from Rostock University, Germany in 2004, and the degree of "Dr.-Ing. habil." with a *venia legendi* for automation engineering from Rostock University, Germany in 2009.

He served as a research scientist in the Institute of Occupational and Social Medicine, Rostock from 2001 to 2004. Currently, he is the head of the research group "Life Science Automation - Technologies" at the Center for Life Science Automation, Rostock. His research interests include modelling of the complex and uncertain processes with applications to the life science. He took an initiative in intelligent fuzzy computing to build a mathematical bridge between artificial intelligence and real-world applications (www.fuzzymodeling.com).