Enhanced Multivariable TS Fuzzy Modeling with Multivariable Fuzzy Sets without Decomposition

Özer Ciftcioglu and I.S. Sariyildiz
Delft University of Technology, Faculty of Architecture
Berlageweg 1, 2628 CR, Delft, The Netherlands

ABSTRACT
Enhanced fuzzy modeling by multivariable fuzzy membership functions is described. From the interpretability issues viewpoint conventionally fuzzy modeling is carried out by means of decomposition of multivariable membership functions via projections on each variable component. However, due to decomposition there involves an error while reconstructing the model output from the contributions of each variable. To circumvent this error, in this work, the fuzzy modeling is accomplished by the multivariable fuzzy membership functions so that enhanced modeling performance is achieved. For the interpretability issues, still the conventional decomposition can be independently carried out as an informative support for better understanding the model properties while the accurate model functionality is maintained in the application involved.

Keywords: TS fuzzy modeling, Clustering, RBF networks, OLS algorithm.

1. INTRODUCTION
Fuzzy modeling techniques have been applied to nonlinear, uncertain and complex systems. A number of researches have been reported in the literature in a variety of disciplines spanning exact sciences and soft sciences. One of the aspects that is claimed for fuzzy modeling is that it distinguishes itself from other black-box approaches like neural networks. That is, fuzzy models are to a certain degree transparent to interpretation and analysis. However absence of a clear procedure for general fuzzy modeling already is a firm indication of the complexity of the task providing transparency. Nevertheless, for certain type of fuzzy modeling rather well-established methods are available for practical applications. One of such modeling techniques is known as the method of Takagi-Sugeno (TS) model [1] in fuzzy systems literature. It has been one of the major topics in theoretical studies and practical applications of fuzzy modeling and control. The basic idea of this method is to decompose the input space into fuzzy regions and to approximate the system in every region by a simple model. The TS model is usually identified in two steps. First, the fuzzy sets in the rule antecedents are determined, producing partitioning of the input space. Then, when the rule antecedents are fixed, the consequent parameters are determined. Since, in most cases the knowledge about the system being modeling is not enough to place the fuzzy sets and consequent parameters, data driven models are ubiquitously used. For this purpose least square estimation from the observation data is commonly considered. This can be employed in two ways [2]. One solves a local, weighted least-squares problems, one for each rule. The other solves a global least-squares problem. Local least squares gives more reliable local models, while global least squares gives a minimal prediction error estimate. In determining the antecedents fuzzy sets, in the multivariable input case, one clusters the data to obtain multivariable fuzzy sets and by means of projection on each input variable one obtains the respective fuzzy sets. However, during the projection process an irrecoverable information loss occurs so that, the fuzzy modeling from the projected fuzzy sets becomes an issue. As to one-dimensional input case, there is no need for projection and therefore the model is satisfactorily accurate. This explains the reason why the majority of fuzzy modeling applications concern merely this simple case. However, in two-dimensional case, the model becomes less precise and the quality of the model exponentially aggravated with the increase of the number of fuzzy sets employed in each dimension. Note that this is not only because of curse of dimensionality but also because of information loss due to decomposition of the multivariable fuzzy sets. The situation becomes already a marked issue even in low dimensional input case, which can be as low as two. This explains the fact that, the majority of the fuzzy logic applications use at most two-dimensional input space and not more although the vast number of theoretical studies reported may extends to any dimension without actual reference to the information loss which occurs in the projection process. Because of the curse of dimensionality of fuzzy sets, higher dimensional fuzzy modeling endeavors become extremely cumbersome and contradict the claimed effectiveness and practicality of fuzzy logic. This work aims to highlight above-mentioned accurate multivariable modeling issues of fuzzy modeling and introduce multivariable membership functions directly into modeling.

This paper presents a new approach to enhance fuzzy modeling using multidimensional fuzzy sets directly in the fuzzy model in place of projected fuzzy sets which results in decomposed multivariable fuzzy sets. The local multivariable linear models are established as follows. The antecedent fuzzy sets, that is, fuzzy membership functions having been obtained in the form of multivariable functions, these functions are represented in continuous form with respect to input variables. This is accomplished by using a multi input and multi output neural net, which is in particular radial basis functions network (RBF) providing the required function approximation properties in a fuzzy model. Note that in this network the regression matrix is not normalized as to another case where such normalization is required for the equivalence of an RBF network to a fuzzy system. The inputs of the network are the fuzzy model input parameters, outputs are the membership function values as to the fuzzy sets involved in the model input. Thus, each output is restricted to one multivariable fuzzy set. The neural net is trained according to the multidimensional fuzzy sets, which are identified by means of initial fuzzy
clustering of the data. Based on this, the linear model parameters are determined by the method of least squares in a straightforward manner. The accurate fuzzy modeling having been thus guaranteed, the transparency issues can be handled by means of projection process to have information about the shape and locations of the membership functions pertinent to each variable at the input. This information is enough to establish the fuzzy rules with their respective validity regions. The redundency of the number of multivariable fuzzy sets can be circumvented by classical cluster merging algorithms available.

The organization of the paper is as follows. It describes briefly the foundations of fuzzy modeling and explains the bottleneck of multivariable fuzzy modeling in applications. It explains theoretically the removal of this bottleneck by means of incorporating a neural network into fuzzy modeling. By doing so, curse of dimensionality is greatly alleviated. Next to it, the model performance is greatly improved thereby the applicability of fuzzy logic. These are alleviated. Next to it, the model performance is greatly improved thereby the applicability of fuzzy logic.

2. FUZZY MODELING

A. Takagi-Sugeno Fuzzy Modeling

Takagi-Sugeno (TS) type fuzzy modelling [1] consists of set of fuzzy rules a local input-output relation in a linear form as

\[ R_i: \text{If } x_1 \text{ is } A_{i1} \text{ and } \ldots x_n \text{ is } A_{in} \text{ then } \]

\[ \hat{y}_i = a_ix+b_i, \quad i=1,2,\ldots,K \]

where \( R_i \) is the ith rule, \( x=[x_1,x_2,\ldots,x_n]^T \in \Xi \) is the vector of input variables; \( A_{i1}, A_{i2},\ldots,A_{in} \) are fuzzy sets and \( y_i \) is the rule output; \( K \) is the number of rules. The output of the model is calculated through the weighted average of the rule consequents of the form

\[ \hat{y} = \frac{\sum \beta_i(x)x_i}{\sum \beta_i(x)} \]

(2)

In (2), \( \beta_i(x) \) is the degree of activation of the ith rule

\[ \beta_i(x) = \prod_{j=1}^{n} \mu_{A_{ij}}(x_j), \quad i=1,2,\ldots,K \]

(3)

where \( \mu_{A_{ij}}(x_j) \) is the membership function of the fuzzy set \( A_{ij} \) at the input (antecedent) of \( R_i \). To form the fuzzy system model from the data set with \( N \) data samples, given by

\[ X=[x_1, x_2, \ldots, x_n]^T, \quad Y=[y_1, y_2, \ldots, y_N]^T \]

(4)

where each data sample has a dimension of \( n \) (\( N>n \)). First the structure is determined and afterwards the parameters of the structure are identified. The number of rules characterizes the structure of a fuzzy system. The number of rules is determined by clustering methods. Fuzzy clustering in the Cartesian product-space \( \Xi \times \Psi \) is applied to partition the training data. The partitions correspond to the characteristic regions where the system’s behaviour is approximated by local linear models in the multidimensional space.

Given the training data \( \Gamma \) and the number of clusters \( K \), a suitable clustering algorithm [3] is applied. One of such clustering algorithms is known as Gustafson-Kessel (GK) [4]. As result of the clustering process a fuzzy partition matrix \( U \) is obtained. The fuzzy sets in the antecedent of the rules are identified by means of the partition matrix \( U \) which has dimensions \( [N \times K] \); \( N \) is the size of the data set. The \( ik \)-th element of \( \mu_{ik} \in [0,1] \) is the membership degree of the \( ih \) data item in cluster \( k \); that is, the \( ih \) row of \( U \) contains the point wise description of a multidimensional fuzzy set. One-dimensional fuzzy sets \( A_{ij} \) are obtained from the multidimensional fuzzy sets by projections onto the space of the input variables \( x_i \). This is expressed by the point-wise projection operator of the form

\[ \mu_{A_{ij}}(x_i) = proj_i(\mu_{ik}) \]

(5)

The point-wise defined fuzzy sets \( A_{ij} \) are then approximated by appropriate parametric functions. The consequent parameters for each rule are obtained by means of linear least square estimation. For this, consider the matrices

\[ X=[x_1, x_2, \ldots, x_n]^T, \quad X_e=[X, I] \]

(6)

where \( X \) is the training data matrix and \( X_e \) is the extended matrix of \( [N \times (n+1)] \) and \( X_e=[(A_1X_1); (A_2X_2); \ldots; (A_KX_K)] \) ([\( N \times K(n+1) \)]

(6)

where the diagonal matrix \( \Lambda_i \) consists of normalized membership degree as its k-th diagonal element

\[ \beta_i(x_i) = \frac{\beta_i(x_i)}{\sum_{j=1}^{K} \beta_j(x_i)} \]

(7)

The parameter vector \( \beta \) dimension of \([K \times (n+1)]\) is given by

\[ \beta = [g_1^T, g_2^T, \ldots, g_K^T]^T \]

(8)

where \( g_i^T=[a_i^T, b_i] \), \( 1 \leq i \leq K \).

Now, if we denote the input and output data sets as \( X_e \) and \( Y \) respectively then, the fuzzy system can be represented as a regression model of the matrix form

\[ Y=X_e \hat{g}+e \]

(9)

Due to projection process in (5), there is an information loss of multidimensional fuzzy sets which is reflected as degradation in the fuzzy modeling. To circumvent this, the multidimensional fuzzy sets are directly used for modeling without decomposition through projections. For a model with single output (9) becomes

\[ y=X \hat{g}+e \]

(10)

where

\[ \hat{g}=a^T b \]

(11)

B. GK Clustering Algorithm

For clustering, there are several effective fuzzy clustering algorithms available. Gustafson-Kessel algorithm is the one which is commonly used due to some desirable features. In order to detail the modeling enhancement achieved in this work, GK algorithm is briefly explained below.
GK Algorithm: Given the data Z with the number of data samples N, number of clusters M, fuzziness parameter m>1, and the termination tolerance ε>0, initialize the fuzzy partition matrix $U=[u_{ki}]$ for $i=1,2,\ldots,M$, randomly. Then

**Repeat for** $l=1,2,\ldots$.

**Step 1:** Compute cluster prototypes:

$$v_{\ell}^{(l)} = \sum_{i=1}^{N} \left( u_{ki}^{(l-1)} ight)^m z_i, \quad 1 \leq i \leq M$$

(12)

**Step 2:** Compute covariance matrices:

$$F_{\ell} = \sum_{i=1}^{N} \left( u_{ki}^{(l-1)} ight)^m (z_i - v_{\ell}^{(l)}) (z_i - v_{\ell}^{(l)})^T \sum_{i=1}^{N} \left( u_{ki}^{(l-1)} ight)^m, 1 \leq i \leq M$$

(13)

**Step 3:** Compute distances to cluster prototypes:

$$d_{kl} = (z_l - v_{\ell}^{(l)})^T D_{l} (z_l - v_{\ell}^{(l)}) / d_{kl}, 1 \leq i \leq M, 1 \leq k \leq N$$

where

$$D_{l} = \left[ \det(F_{l}) \right]^{1/(n+1)} F_{l}^{-1}$$

(15)

**Step 4:** Update the partition matrix:

$$u_{ki}^{(l)} = \frac{1}{\sum_{j=1}^{M} \left( d_{kj} / d_{kl} \right)^{2(m-1)}}$$

if $d_{kl} > 0$

$u_{ki}^{(l)} = 1$ if $d_{kl} = 0$

$$u_{ki}^{(l)} \in [0,1] \text{ with } \sum_{i=1}^{M} u_{ki}^{(l)} = 1$$

(16)

(17)

until $\| U^{(l)} - U^{(l-1)} \| < \varepsilon$

Note that the product $d_{kl}^2 = (z_l - v_{\ell}^{(l)})^T D_{l} (z_l - v_{\ell}^{(l)})$ measures the distance of the antecedent vector $z$ from the projection of the cluster centre $v_{\ell}^{(l)}$. In step four, an inversion is applied to obtain the membership degree where the expression computes the degree of fulfilment of one rule relative to the other rules and the sum of the membership degrees of all the rules equal to one. This is, in fact, a probabilistic constraint. A drawback of the GK algorithm is that it is effective to find clusters of approximately equal volumes. However, the eigenstructure of the cluster covariance matrix provides information about the shape and orientation of the cluster. This information can be used to compute optimal local linear models from the covariance matrix, as this is exploited in this work for comparison purpose with the RBF network based approach. A desirable feature of the GK algorithm over other fuzzy clustering methods is that GK can detect clusters of different shape and orientation in one data set, yielding better positioning of the fuzzy sets projected on the individual antecedent variables in a fuzzy modeling.

As to fuzzy modeling with multivariate membership functions, the degree of fire (DOF) values for each variable can be obtained via norm inducing matrix and cluster centres information via (equation). However in this case, the membership function values are normalized after these computations, the normalization factor for each cluster is different than the normalization factor used for obtaining the fuzzy model. In other words, new conditions are imposed on the formation of the DOF determination rather than keeping the modeling conditions fixed. This situation introduces another source of error similar to the reconstruction error which occurs in the case of decomposed multivariable fuzzy membership functions via projection. To avoid this, the multivariable membership functions are fixed by means of a network which is radial basis function network, in this case. By doing so, the modeling conditions are kept constant and DOF is obtained simply as output of this network. This is schematically shown in figure 1 where the number of the input variables are three forming three dimensional antecedent space, and the number of clusters of fuzzy model is two, forming two dimensional output space.

![Figure 1. RBF network modeling the multidimensional fuzzy fuzzy membership functions obtained from the clustering.](image)

The RBF network is briefly described below.

**B. RBF Network and OLS Algorithm**

Radial basis functions (RBF) network is a special type of feed-forward neural network. It has been described in detail in literature [6]. RBF networks have their origin in the solution of the multivariate interpolation problem [7-9]. It has been shown that, in compact domains the function space realized by RBF network is dense in the space of all continuous functions [10]. that is, it is universal approximator of continuous functions. An RBF network, consists of three layers, namely inputs, nodes and outputs layers. The connection weight vectors between the inputs and nodes-layer are denoted as $x$ and the vectors between the nodes-layer and output are denoted as $w$. The vectors representing the basis function centres is denoted by $c$. The outputs are formed a weighted linear combination of the basis functions outputs in the nodes-layer. The basis
functions are commonly Gaussian functions. The output \( \phi \) of the basis function \( j \) is given by

\[
\phi_j = \exp \left\{ -\frac{\|x - c_j\|^2}{\sigma_j^2} \right\}
\]

where

\[
\|x - c_j\| = \left[ (x - c_j)^T (x - c_j) \right]^{1/2}
\]

is the norm. \( \sigma_j^2 \) is the variance determining the width of the gaussian. The symmetric matrix \( \Phi \) formed by the elements

\[
\phi_{ij} = \exp \left\{ -\frac{\|x_i - c_j\|^2}{\sigma_j^2} \right\}
\]

is referred to as RBF matrix which has all one along the main diagonal. For a single output model which models the function \( y = f(x) \), the function approximation of the model in matrix form can be expressed as

\[
y = \Phi w + e
\]

where \( e \) is the vector modelling the approximation errors. Generally, the width parameters are taken to be equal for simplicity. The form given by (21) is exactly a conventional perceptron neural network formalism. Here, in place of sigmoidal functions, radial basis functions are used. Therefore, for the determination of weights, centres and width parameters of the RBF network well-known back-propagation algorithm can be used. However, for the present purpose the more suitable training algorithm is known as orthogonal least squares (OLS) algorithm for determining the basis function centres \( c \) and the weight vectors \( w \) due to continuous multivariable functions approximations of the RBF network. The width parameter is taken as constant. The method of OLS is well-described in the literature [11,12]. Basically it performs a supervised clustering of the data samples where radial basis function centres are selected among the data samples. The gradation of the centres is based on their energy contribution to the output; that is, when the energy contribution of an RBF centre is higher relative to others, it is considered to be more influential among others. Therefore, in the OLS an iteration procedure is executed which is equal to the data samples. At each iteration process the most influential centre among the available ones is determined and at the end of the process, all centres are hierarchically graded according to their contribution to the output of the RBF net.

### 3. EXPERIMENTAL STUDY OF FUZZY MODELING

For the experimental study, modeling of a two input nonlinear function [13,14]

\[
f(x_1, x_2) = \sin \left( \frac{\pi x_1}{10} \right) \sin \left( \frac{\pi x_2}{10} \right)
\]

is considered where we can assume that a system is given by

\[
y(t) = F(x(t), x(t))
\]

and it will be modeled with the a priori unknown nonlinear mapping

\[
y = \sin \left( \frac{\pi x_1}{10} \right) \sin \left( \frac{\pi x_2}{10} \right)
\]

This is shown in figure 1 together with the clustering results for four clusters. The multivariable membership functions corresponding the clusters in figure 1 are identified as shown in figure 2.

Based on the multivariable membership functions, the projected membership functions of the variables \( x_1 \) and \( x_2 \) are shown in figure 3 where also the identified surface together with the data points is shown.
In figure 3 it is clear that some data points are not matching the identified surface. To illustrate this more explicitly, total 121 data points are shown in figure 4 together with their counterparts forming the surface subject to modeling. In this figure the points forming the surface and the data points obtained as the output of the model are shown together for 121 points (upper plot), 144 points (middle plot) and 169 point (lower plot), respectively. Therefore, the uppermost plot in figure 4 corresponds to the lowermost plot in figure 3. The fuzzy modeling errors, even in such 2-dimensional simple exercise are clearly observed. The main source of the errors is attributed to the reconstruction error, which is caused by the projection of the multivariable membership functions.

The same identification task is carried out by means of representing the multivariable fuzzy membership functions by an RBF network. In this case, the identification results are shown in figure 5.

In figure 5, the lowermost plot is the counterpart of the uppermost plot in figure 4. By the comparison of these plots, the improvement achieved by using an RBF network is clearly seen. The further comparisons for 144 and 169 points are shown in figure 6, which includes the plot with 121 points, as shown already in figure 5. The figure 6 is to compare with figure 4, and the marked improvement by RBF network support is to observe.

In the case of forming multivariable fuzzy membership functions using the norm inducing matrix $D_i = [\det(F_i)]^{1/(m-1)}F_i^{-1}$ and the computing the membership function by $u_{kj}(x_k) = \frac{1}{\sum_{j=1}^{M}(d_{kj}/d_{kj})^{1/(m-1)}}$, one obtains the results as presented in figure 7.
The results presented in figure 7 are rather informative from the fuzzy modeling viewpoint. In the first place, comparison of figure 4 and figure 7 reveals that, the employment of multivariable fuzzy sets provides more accurate modelling as this is intended to demonstrate in this work. Secondly, the accuracy can still be improved by integrating an RBF net, as this to conclude by the comparison of figure 6 and figure 7. In figure 6, a marked improvement as to figure 7, is clearly observed. It is noteworthy to mention two observations. Firstly, in figure 6, the deviation between the real and approximated system outputs is reasonably small and therefore the approximation is satisfactory. The deviations are relatively large at the boundaries where the output is zero, at the same time (starting and ending regions of the plots). This is due to the unfavourable interpolation conditions at these regions. The rest is satisfactory. Secondly, the model performance is satisfactorily stable for the data sets with varying number of data samples, implying satisfactory precision of the model. Note that, the data samples in the data sets are entirely different, that is, the sets with lower number of data samples are not subsets of the sets with higher number of data samples. It should be pointed out that, next to RBF support in fuzzy modeling, additionally, projection of the multivariable membership functions can be implemented in a conventional way as well for informative reasons to improve the transparency of the model by additional information provided by this additional process.

4. CONCLUSIONS

The precision and consequently accuracy of fuzzy modelling rapidly diminishes in multivariable modelling environment due to curse of dimensionality. In such cases, to improve the model performance several propositions are made in literature such as similarity measures [15], evolutionary algorithm [16] and so on. Due to the model performance degradation, the majority of fuzzy modeling applications reported are in one-dimensional space and there is only few investigations reported on the degradation of fuzzy modelling in multidimensional case due to curse of dimensionality and reconstruction error occurring as result of multivariable membership function projection. In contrast with this scant attention, there are abundance of theoretical reported elaborations without thorough verifications in the real-life applications. Therefore, this work intended to point out the implications of multidimensional fuzzy modelling as to its performance. Having noted the rapid degradation of modelling performance in multidimensional antecedent space, support of radial basis function network is presented operating with multidimensional fuzzy sets, as a means of progressive improvement of this issue. The marked improvement of this approach is shown. Fuzzy modelling is an essential machinery of soft computing which deals with machine intelligence via computational methods. Soft computing is especially effective in the applications in soft sciences. In particular, in the consideration of design related soft aspects which generally pose ill-defined issues in highly multidimensional space, model precision is tough to maintain and the multidimensionality is much beyond the concern of engineering applications. As to the present approach, one of such soft areas is the architectural design [17] and the approach implies essential contribution for the improvement of the design performance.

5. REFERENCES