Brief Papers

Constructing Fuzzy Models with Linguistic Integrity from Numerical Data-AFRELI Algorithm

Jairo Espinosa and Joos Vandewalle

Abstract—This paper presents an algorithm to extract rules relating input/output data and including prior knowledge. The rules are created in the environment of fuzzy systems. The fuzzy sets describing the system are constructed within a framework of *linguistic integrity* to guarantee its interpretability in the linguistic context. Two algorithms are presented in this paper. The main algorithm is the autonomous fuzzy rule extractor with linguistic integrity (AFRELI). This algorithm is complemented with the use of the FuZion algorithm created to merge consecutive membership functions, while guaranteeing the distinguish ability between fuzzy sets. Comparisons with other proposed methods show a good tradeoff between accuracy and interpretability.

Index Terms—Data mining, function approximation, fuzzy modeling, knowledge extraction.

I. INTRODUCTION

MODELS are key elements for humans or human behavior. Models enable human beings to predict the impact of their actions. Causality is an intrinsic assumption that makes models useful. Causality is reflected in language as IF-THEN rules (IF **cause-happens** THEN **a consequence is foreseen**). A set of these IF-THEN rules is nothing more than the linguistic representation of the mental model created inside the brain about a certain system by means of experience. The development of new systems to gather massive amounts of data (instrumentation and computers) has expanded our capacity to interact with nature beyond our five senses. But in the other hand our brain and senses are unable to extract "knowledge" out of massive amounts of numerical data. This situation motivates the development of computer techniques that can extract the knowledge and represent it in a linguistic way using IF-THEN rules.

There is a well-known tradeoff between numerical accuracy and linguistic interpretability. This tradeoff is the consequence

of a well-known limitation of the human brain to represent a limited number of categories on a given domain. A clear consequence of this limitation is reflected in language where the number of linguistic labels that a human being can generate to represent some categories, normally will not exceed nine and it will typically be seven [1].

On the other hand, the numerical accuracy is very important in the implementation of policies and control actions oriented to obtain a desired result from the system. This issue of accuracy is very critical when the models are used in a dynamic way, where the predicted value is fed back and the small errors will be propagated and reflected as errors in the long term prediction.

Classical algorithms for rule extraction have been proposed. Initially the so-called neuro-fuzzy algorithms [2], [3] have been oriented to minimize the numerical error, but parameters such as initial description and number of membership functions, must be provided by the designer. These algorithms are optimized using gradient descent techniques. The use of these techniques sometimes generates fuzzy sets with "too much" or "absolutely no" overlap, thereby making the interpretation of the model cumbersome.

To overcome the drawback of the initial selection of the fuzzy sets, several methods have been proposed, some of them based on local error approximation [4], [5] and some others based on clustering techniques [6]–[8]. These methods generate multidimensional fuzzy sets and project them into the input spaces. The projections again exhibit unsatisfactory overlap making the interpretation and the labeling of the fuzzy sets a difficult task.

This paper presents the algorithm autonomous fuzzy rule extractor with linguistic integrity (AFRELI); the algorithm is able to fit input/output data while maintaining the semantic integrity (interpretability) of the rule base. The issue of interpretability was mentioned by Jang *et al.* in [9, end ch. 12], where they only mention some basic ideas about how to constrain the optimization of the adaptive network based fuzzy inference system (ANFIS) scheme to preserve the interpretability. In [10], Valente presents a formulation of some constraints to guarantee semantic integrity. This formulation assumes a number of sets on each input domain and then constrains the optimization of the membership functions to guarantee the semantic integrity of the resulting fuzzy sets. In [11], the tradeoff between precision and transparency is mentioned and some examples are presented where the consequences of the rules use two terms, one numeric and one linguistic term. In [12], an algorithm to reduce the complexity of fuzzy models was presented. This al-

gorithm used a similarity measure as a criteria to reduce the number of fuzzy sets in the antecedents of the rules, but no other constraints were imposed to the fuzzy sets to guarantee their semantic integrity. The AFRELI algorithm presented in this paper is a two-step approach. The first step uses clustering and projection techniques to find a good initial position for the fuzzy sets in the input domains. The second step reduces the complexity of the model using the concept of semantic integrity as a framework. The FuZion algorithm is introduced as a tool to carry out the complexity reduction process. A rule base is constructed using the reduced representation of the fuzzy sets and the consequences are initialized and calculated with a method that improves generalization and avoids the problems due to the lack of excitation in some rules. Finally, the consequences of the rules are represented by two fuzzy sets with different strength. The number of terms in the consequences of the fuzzy rules is reduced again using the FuZion algorithm.

The paper is structured as follows. Section II presents the structure of the fuzzy model using the concept of semantic integrity to justify the selection of the parameters, Section III introduces the AFRELI algorithm, Section IV presents the FuZion algorithm to preserve the semantic integrity of a given domain, Section V shows some application examples, and, finally, Section VI gives the conclusions.

II. STRUCTURE OF THE FUZZY MODEL

A fuzzy inference system has many degrees of freedom (shape and number of membership functions, T-norms, aggregation methods, etc.). This fact gives high flexibility to the fuzzy system, but also demands systematic criteria to make these choices. For the present case, some choices are made by taking into account the concept of optimal interface design [13] and semantic integrity [14].

- Optimal interface design
- *Error-free reconstruction*: In a fuzzy system, a numerical value is converted into a linguistic value by means of fuzzification. A defuzzification method should guarantee that this linguistic value can be reconstructed into the same numerical value

$$\forall x \in [a, b]: \quad \mathcal{L}^{-1}[\mathcal{L}(x)] = x \qquad (1)$$

where the interval $[a, b]$ is the universe of discourse. The use of triangular membership functions with overlap $1/2$ and centroid defuzzification will satisfy this requirement (see proof: [13]).

- Semantic integrity: This property guarantees that the membership functions will represent a linguistic concept. The main conditions for semantic integrity are the following.
- *Distinguishability*: Each linguistic label should have semantic meaning and the fuzzy set should clearly define a range in the universe of discourse. Therefore, the membership functions should be clearly different. The assumption of the overlap equal to $1/2$ makes sure that the support of each fuzzy set will be different. The distance between the modal values of the membership functions is also very important to make sure that the membership functions can be

distinguished. The defined modal value of a membership function is defined as the α-cut with $\alpha = 1$

$$\mu_{i(\alpha=1)}(x), \quad i = 1, \ldots, N. \qquad (2)$$

- *Justifiable number of elements*: The number of sets should be compatible with the number of "quantifiers" that a human being can handle. This number should not exceed the limit of 7 ± 2 distinct terms [1]. The simple choice of the shape of the membership functions does not guarantee this property. To assure that this requirement is satisfied the FuZion algorithm is presented further in this paper. This algorithm reduces the number of sets present on each input or output domain.
- *Coverage*: Any element from the universe of discourse should belong to at least one of the fuzzy sets. This concept is also mentioned in [3] as ϵ completeness.
- *Normalization*: Due to the fact that each linguistic label has semantic meaning, at least one of the values in the universe of discourse should have a membership degree equal to one. In other words, all the fuzzy sets should be normal.

Based on these criteria the selected *membership functions* will be triangular and normal $(\mu_1(x), \mu_2(x), \ldots, \mu_n(x))$ with a specific overlap of $1/2$. This means that the height of the intersection of two successive fuzzy sets is

$$\mathrm{hgt}(\mu_i \cap \mu_{i\pm 1}) = \frac{1}{2}. \qquad (3)$$

The choice of the AND and the OR operation can be motivated by the need of construct a continuous and differentiable nonlinear map. This property is important if optimization of the antecedent terms is needed. In this case ANDand ORoperations using *product* and *probabilistic sum* will be preferred because their derivatives are continuous.

The aggregation method and the defuzzification method will be discussed in the next sections.

III. AFRELI ALGORITHM

The AFRELI is an algorithm designed to obtain a good compromise between numerical approximation and linguistic meaning. This particular tradeoff has been referenced for long time in science (for a compilation of remarks, see [15]). The main steps of this algorithm are as follows:

- clustering;
- projection;
- reduction of terms in the antecedents (FuZion, see Section IV);
- consequence calculation;
- further antecedent optimization (optional step);
- reduction of terms in the consequences and rule modification (FuZion, see Section IV).

Fig. 1 shows a flow diagram with the description of the algorithm. The details of the AFRELI algorithm are presented in the following lines.
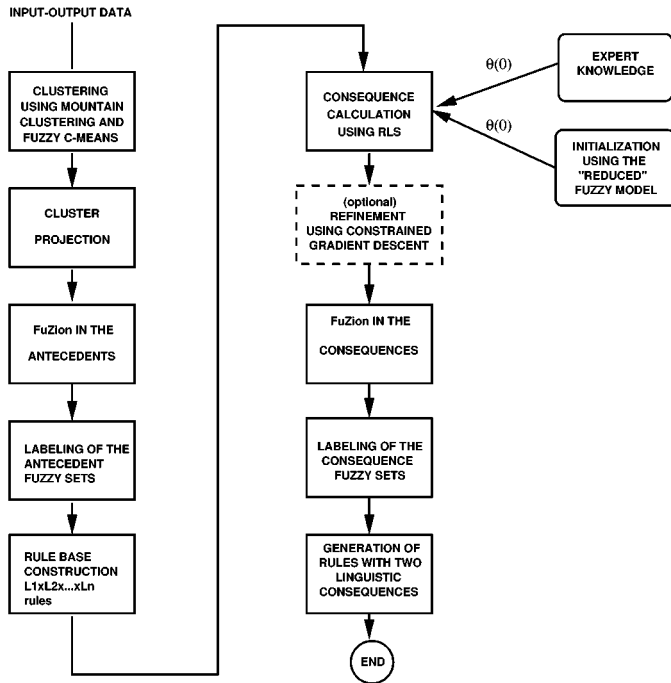
Fig. 1. Flow diagram of AFRELI algorithm.

1) Collect N points from the inputs ($U = \{u_1, \ldots, u_N\}$) and the output ($Y = \{y_1, \ldots, y_N\}$)

$$u_k = \begin{bmatrix} u_k^1 \\ \vdots \\ u_k^n \end{bmatrix} \quad (4)$$

where $u_k \in \Re^n$ and $y_k \in \Re$ represents the inputs and the output at instant k. Construct the *feature vectors* $x_k \in \Re^{n+1}$ with

$$x_k = \begin{bmatrix} u_k^1 \\ \vdots \\ u_k^n \\ y_k \end{bmatrix} . \quad (5)$$

2) Using the N feature vectors find C clusters by using mountain clustering method [7], [8] to initialize the centers and number of clusters and refine them using fuzzy C-means [16]

$$Xc = \begin{pmatrix} \tilde{x}_1^1 & \tilde{x}_2^1 & \cdots & \tilde{x}_C^1 \\ \vdots & \vdots & \ddots & \vdots \\ \tilde{x}_1^{n+1} & \tilde{x}_2^{n+1} & \cdots & \tilde{x}_C^{n+1} \end{pmatrix}$$
$$Xc \in \Re^{n+1 \times C}. \quad (6)$$

It is very important to remark that the use of *mountain clustering* will be limited to low-dimensional problems, its inherent advantage is that it can produce good initial points and number of clusters. For high-dimensional problems, the alternative is to use only fuzzy C-means and to overestimate the number of clusters. The subsequent steps (FuZion) will then reduce the number of terms.

3) Project the C prototypes of the clusters into the input spaces. Assuming that the projected value of each prototype is the modal value of a triangular membership function

$$m_i^j = \tilde{x}_i^j \quad (7)$$

where $i = 1, \ldots, C, j = 1, \ldots, n$.

4) Sort the modal values on each domain such that

$$m_i^j \le m_{i+1}^j \quad \forall j. \quad (8)$$

5) Add two more modal values for each input to guarantee full coverage of the input space

$$m_0^j = \min_{k=1,\ldots,N} u_k^j \quad (9)$$

$$m_{C+1}^j = \max_{k=1,\ldots,N} u_k^j. \quad (10)$$

6) Construct the triangular membership functions with overlap of $1/2$ as :

$$\mu_i^j(x^j) = \max\left[0, \min\left(\frac{x^j - m_{i-1}^j}{m_i^j - m_{i-1}^j}, \frac{x^j - m_{i+1}^j}{m_i^j - m_{i+1}^j} \right) \right] \quad (11)$$

where $i = 1, \ldots, C$ and the trapezoidal membership functions at the extremes of each universe of discourse

$$\mu_0^j(x^j) = \max\left[0, \min\left(\frac{x^j - m_1^j}{m_0^j - m_1^j}, 1 \right) \right] \quad (12)$$

$$\mu_{C+1}^j(x^j) = \max\left[0, \min\left(\frac{x^j - m_C^j}{m_{C+1}^j - m_C^j}, 1 \right) \right]. \quad (13)$$

7) Apply FuZion algorithm (see Section IV) to reduce the number of membership functions. This FuZion algorithm will reduce the number of fuzzy sets in each domain. This algorithm does some kind of one-dimensional clustering among the modal values of the fuzzy sets.

8) Associate linguistic labels (e.g., BIG, MEDIUM, SMALL, etc.) to the resulting membership functions. This association will depend on the type of variable and the opinion of the designer. In fact the association of a fuzzy set with a label will be the result of the agreement between the fuzzy set proposed by the algorithm and the "sense" that this set creates in the mind of the user.

9) Construct the rule base with all possible antecedents (all possible permutations) using rules of the form in the equation shown at the bottom of the next page. Equivalently, the evaluation of the antecedents for each rule can be expressed in terms of operators. We use the min operator

$$\mu_l(u_k) = \min\{\mu_l^1(u_k^1), \mu_l^2(u_k^2), \ldots, \mu_l^n(u_k^n)\} \quad (14)$$

and the *product* operator

$$\mu_l(u_k) = \mu_l^1(u_k^1) \cdot \mu_l^2(u_k^2) \cdot \cdots \cdot \mu_l^n(u_k^n). \quad (15)$$

Observe that if the number of fuzzy sets for the input i is L_i and there are n inputs the number of potential rules is

$$L_1 \times L_2 \times \cdots \times L_n.$$

This description guarantees the "full" description of the system because every possible condition will be represented in the rule base. Observe that the number of rules will grow very fast as the number of inputs increases. This fact is a limitation in the sense that the comprehension of a set of rules with a large number of antecedents is difficult. Moreover, there is a storage problem since a large number terms has to be stored in the computer's memory. On the other hand, it does not represent a limitation in terms of execution time because the use of the described type of triangular membership functions will guarantee that at most 2^n rules will be evaluated during the inference process.

10) Propagate the N values of the inputs and calculate the consequences of the rules as singletons $(\bar{y}_l)$. These singletons can be calculated as the solution of a least squares problem. Observe that the output of the fuzzy system can be calculated as

$$f(u_k) = \frac{\sum_{l=1}^{L} \mu_l(u_k)\bar{y}_l}{\sum_{l=1}^{L} \mu_l(u_k)}. \qquad (16)$$

Where L is the number of rules and $\mu_l(u_k)$ can be calculated as shown in the (14) or (15) (depending on the selected AND operator). The system can then be represented as the weighted sum of the consequences

$$f(u_k) = \sum_{l=1}^{L} w_l^k(u_k)\bar{y}_l \qquad (17)$$

where

$$w_l^k(u_k) = \frac{\mu_l(u_k)}{\sum_{l=1}^{L} \mu_l(u_k)} \qquad (18)$$

expressing w_l^k as the strength of the rule l when the input is u_k. Taking all the values the problem can be seen as

$$\underbrace{\begin{bmatrix} y_1 \\ y_2 \\ \vdots \\ y_N \end{bmatrix}}_{Y} = \underbrace{\begin{bmatrix} w_1^1 & w_2^1 & \cdots & w_L^1 \\ w_1^2 & w_2^2 & \cdots & w_L^2 \\ \vdots & \vdots & \ddots & \vdots \\ w_1^N & w_2^N & \cdots & w_L^N \end{bmatrix}}_{W} \underbrace{\begin{bmatrix} \bar{y}_1 \\ \bar{y}_2 \\ \vdots \\ \bar{y}_L \end{bmatrix}}_{\theta} + \underbrace{\begin{bmatrix} e_1 \\ e_2 \\ \vdots \\ e_N \end{bmatrix}}_{E}. \qquad (19)$$

The aim here is to reduce as much as possible the norm of the error vector E. Using the quadratic norm

$$\min_{\theta} \|E\|_2 = \min_{\theta} \|Y - W\theta\|_2. \qquad (20)$$

The solution to this problem will be

$$\theta = (W^T W)^{-1} W^T Y. \qquad (21)$$

This solution will be valid if and only if

$$\text{rank}(W^T W) = \dim(\theta). \qquad (22)$$

The condition given in (22) implies that all the rules have to receive enough excitation during training. In practice, this is not always guaranteed so the application of this method with an *ill-conditioned* matrix $W^T W$ will generate catastrophic results for those rules with low excitation and a significant bias in the rules with sufficient excitation. The result will be a fuzzy system with significant errors in the points present in the training set and very poor generalization for points outside the training set.

Some ideas has been proposed to avoid this problem, among others the pruning of the "unexcited" rules using the information given by orthogonal transformations (see [17]). However, the elimination of "unexcited" rules is a dangerous solution because the generalization capabilities of the model are compromised due to the fact that the system can be excited by an input corresponding to one of the pruned rules.

This problem can be solved by using an adaptive strategy that adapts only the consequence of those rules that has been excited and for the rules without excitation a good initialization value is taken. The strategy for adaptation will be recursive least squares (RLS) [18]. RLS guarantees that the adaptation will only affect the excited rules. The RLS algorithm looks like

$$\theta(k+1) = \theta(k) + \gamma(k)[y(k+1) - W_{k+1}\theta(k)] \qquad (23)$$

with $W_k = \{w_1^k, w_2^k, \ldots, w_L^k\}$ and

$$\gamma(k) = P(k+1)W_{k+1} \qquad (24)$$

$$= \frac{1}{W_{k+1}P(k)W_{k+1}^T + 1}P(k)W_{k+1} \qquad (25)$$

$$P(k+1) = [I - \gamma(k)W_{k+1}]P(k) \qquad (26)$$

with the initial value $P(0) = \alpha I$, where α is large. The initial value of $\theta(0)$ can be assigned using the prior knowledge of the expert, if available. A very interesting fact of this strategy is that if expert knowledge exists it will be preserved as long as the data does not falsify it.

$$\cdots$$
$$\text{IF } u_k^1 \text{ is } \mu_l^1 \text{ AND } u_k^2 \text{ is } \mu_l^2 \text{ AND } \ldots \text{ AND } u_k^n \text{ is } \mu_l^n \text{ THEN } \hat{y}_k = \bar{y}_l$$
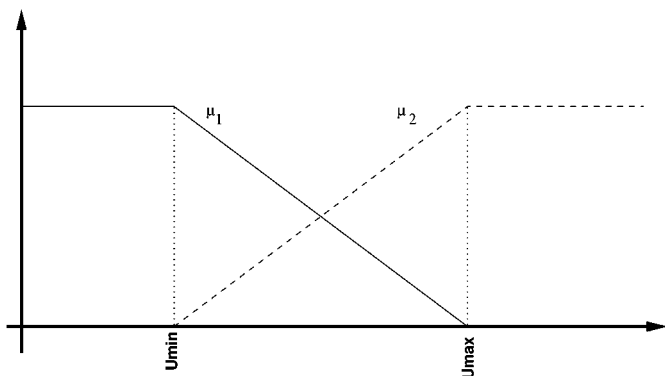$$\cdots$$

Fig. 2. Input membership functions for the smallest fuzzy model.

If expert knowledge is not available, the alternative will be to construct the smallest possible fuzzy model with full coverage of the working space. Such a "reduced" fuzzy model will have two membership functions on each input covering the universe of discourse, as it is shown in Fig. 2. The "reduced" model has 2^n rules where n is the number of inputs. In this model any input vector will excite all the rules making the estimation of the 2^n consequences a well conditioned problem solvable by means of the formula (21). Once this "reduced" model is obtained the modal values of the membership functions of each of the L rules of the "full" model will be propagated. The output value generated by the "reduced" model will be the *initial* consequence of the rule of the "full" model. This procedure copies the "reduced" model into the "full" model. In this way the "reduced" model is used to generate the initial values for the consequences of the rules $\theta(0)$. This method guarantees that the "full" model will be at least as good as the "reduced" model. This property is very important because the "reduced" model is the best multilinear model that can be built with the given data, so there is a guarantee that the "full" fuzzy model will be at least as good as the best multilinear model and this is, in fact, an upper bound for the modeling error.

11) (**Optional Step**) If further refinement is required to improve the approximation, constrained gradient descent methods can be applied to improve the location of the modal values of the antecedent membership functions. The main constraint applied in the optimization phase is the "distinguishability" that is represented as the minimum acceptable distance between consecutive modal values. The use of gradient descent methods will move the system parameters toward a "local minimum" close to the initial values.

Because the improvement obtained by this step will not be very significant this step is considered *optional* and will only be recommended when the numerical performance of the model does not satisfy the user requirements.

12) Convert the singletons to triangular membership functions with overlap $1/2$ and modal values equal to the position of the singleton $\bar{y}_l$. Consider the vector $\check{Y}$ whose

entries are the L consequences of the rules but sorted in such a way that

$$\tilde{y}_1 \leq \tilde{y}_2 \leq \cdots \leq \tilde{y}_L. \tag{27}$$

The triangular membership function of the ith consequence is

$$\mu_i^{\tilde{y}}(y) = \max\left[0, \min\left(\frac{y - \tilde{y}_{i-1}}{\tilde{y}_i - \tilde{y}_{i-1}}, \frac{y - \tilde{y}_{i+1}}{\tilde{y}_i - \tilde{y}_{i+1}}\right)\right] \tag{28}$$

and the two membership functions of the extremes

$$\mu_1^{\tilde{y}}(y)$$
$$= \max\left[0, \min\left(\frac{y - 2\tilde{y}_1 + \tilde{y}_2}{-\tilde{y}_1 + \tilde{y}_2}, \frac{y - \tilde{y}_2}{\tilde{y}_1 - \tilde{y}_2}\right)\right] \tag{29}$$
$$\mu_L^{\tilde{y}}(y)$$
$$= \max\left[0, \min\left(\frac{y - \tilde{y}_{L-1}}{\tilde{y}_L - \tilde{y}_{L-1}}, \frac{y - 2\tilde{y}_L + \tilde{y}_{L-1}}{-\tilde{y}_L + \tilde{y}_{L-1}}\right)\right]. \tag{30}$$

This description of the outer membership functions guarantees that their centers of gravity will be exactly on its modal value. This guarantees that the condition of error-free reconstruction for optimal interface will be achieved.

13) Apply FuZion algorithm (see Section IV) to reduce the number of membership functions in the output universe. The FuZion process reduces groups of neighboring singletons to triangular membership functions whose modal value is representative for a group of singletons and is optimal in a sense that the modal value of the "FuZioned" membership function is placed at the mean value of the neighboring singletons.

14) Associate linguistic labels to the resulting membership functions.

15) With the partition of the output universe, fuzzify the values of the singletons. Observe that each singleton will have a membership degree in at least one set and in as much as two.

16) Relate the fuzzified values with the corresponding rule. This means that each rule will have one consequence or two weighted consequences where the weights are the nonzero membership values of the fuzzified singleton. This description of the consequences of the rules using two linguistic fuzzy sets and two strength values improves the interpretability of the consequence with respect to the case where the consequence is only described by a singleton. The advantage of this description is that interpretability is improved without a loss in numerical precision. This strategy was independently proposed previously in [19] and [20].

IV. FuZion Algorithm

The FuZion algorithm is a routine that merges triangular membership functions whose modal values are "too close" to each other. This merger process is needed to preserve the
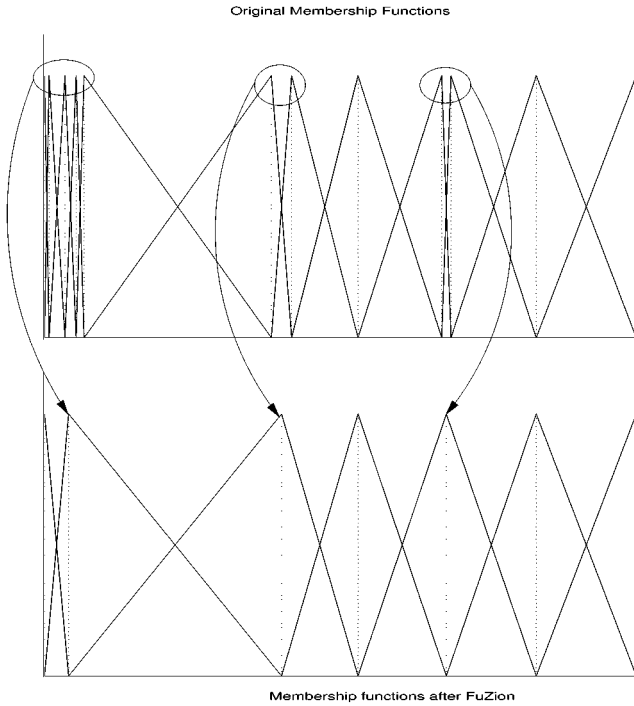
Fig. 3. Effect of the FuZion algorithm.

Fig. 4. Flow diagram of FuZion algorithm.

distinguishability and the *justifiable number of elements* on each domain to guarantee the semantic integrity. The effect of the algorithm can be observed in the Fig. 3. A fundamental parameter of this algorithm is the minimum acceptable distance between modal values and is given by M. It is important to analyze the impact of the parameter M in the characteristics of the model. As the value of M decreases, the number of acceptable membership functions per domain will increase, increasing the number of rules, increasing the complexity of the model, improving the approximation capabilities of the model, but compromising the semantic integrity of the model. On the other hand, as the value of M increases the number of membership functions per domain decreases, reducing the number of rules and increasing the approximation error.

M is the parameter that must be used to balance the tradeoff between interpretability and precision. Typical values for M will be between 5–25% of the coverage of the universe of discourse to guarantee the semantic integrity.

The FuZion algorithm is described in the following lines and Fig. 4 shows a flow diagram of the following algorithm.

1) Take the triangular membership functions $\mu_1(x), \mu_2(x), \ldots, \mu_N(x)$ with $1/2$ overlap and the modal values

$$m_i = \mu_{i(\alpha=1)}(x), \quad i = 1, \ldots, N \qquad (31)$$

with

$$m_1 \leq m_2 \leq \cdots \leq m_N. \qquad (32)$$

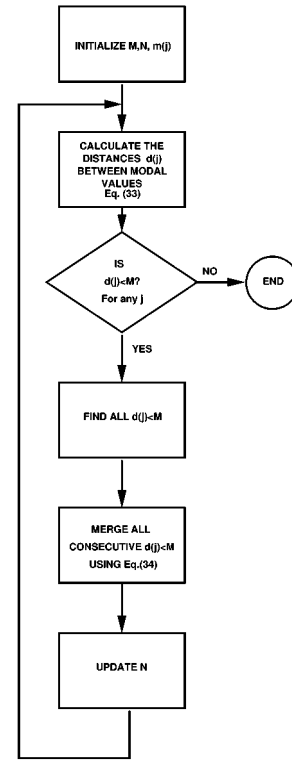2) Define the minimum distance acceptable between the modal values M.

3) Calculate the difference between successive modal values as

$$d_j = m_{j+1} - m_j, \quad j = 1, \ldots, N-1. \qquad (33)$$

4) While $\exists d_j < M$ do.

5) Find all the differences smaller than M.

6) Merge all the modal values corresponding to consecutive differences smaller than M using

$$m_{\text{new}} = \frac{\sum_{i=a}^{b} m_i}{D} \qquad (34)$$

$$D = b - a + 1 \qquad (35)$$

where a and b are, respectively, the index of the first and the last modal value of the fusioned sequence and D is the number of merged membership functions.

7) Update N.

8) Calculate the difference between the new successive modal values as

$$d_j = m_{j+1} - m_j, \quad j = 1, \ldots, N-1. \qquad (36)$$

9) end while

10) end.

V. EXAMPLES

The present section shows three examples of applications of the AFRELI and FuZion algorithm, the first two examples are approximations of nonlinear static maps and the last is the prediction of a chaotic time series.
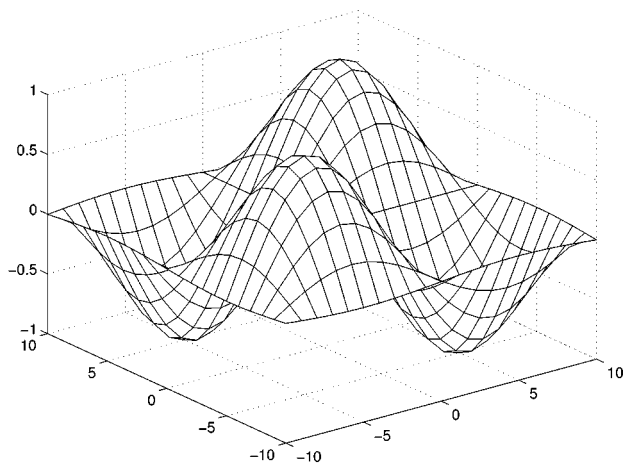
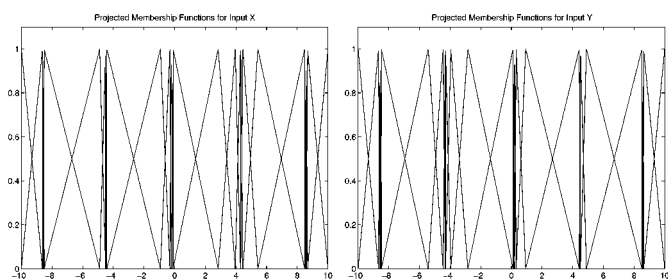Fig. 5.   Example 1: function $f(x, y) = \sin(x/\pi)\sin(y/\pi)$.
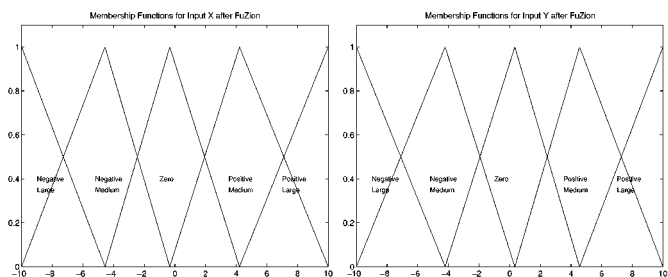

Fig. 6.   Example 1: projected membership functions.

Fig. 7.   Example 1: membership functions after FuZion with $M = 10\%$.

A. *Example 1: Modeling a Two-Input Nonlinear Function*

In this example we consider the function

$$f(x, y) = \sin\left(\frac{x}{\pi}\right)\sin\left(\frac{y}{\pi}\right) \qquad (37)$$

441 points regularly distributed were selected from the interval $[-10, 10] \times [-10, 10]$. The graph of the function is shown in Fig. 5.

Using mountain clustering and fuzzy C-means algorithm, 26 clusters were found. After the clusters were found their center values were projected into the input domains. Fig. 6 shows the projected membership functions. The Fuzion algorithm is applied with M equal to the 10% of the universe of discourse on each domain, observe that with this value of M five (5) membership functions are generated as shown in Fig. 7. When the M parameter in the FuZion algorithm is chosen equal to 7% of the
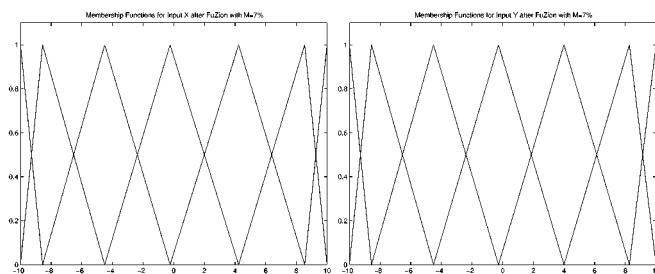

Fig. 8.   Example 1: membership functions after FuZion with $M = 7\%$.

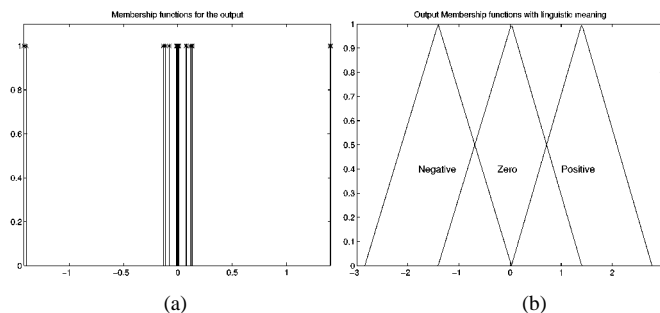(a)   (b)

Fig. 9.   Example 1: (a) Singletons. (b) Membership functions with linguistic meaning.
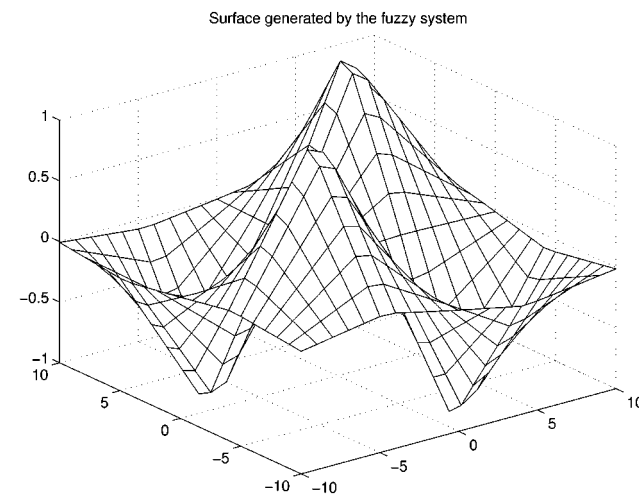

Fig. 10.   Example 1: surface generated by the fuzzy system.

universe of discourse seven (7) membership functions are obtained (see Fig. 8). The output membership functions are shown in Fig. 9. Fig. 10 shows the identified surface. In total, 25 rules were extracted. Some of the extracted rules are

IF x is Negative Large AND
    y is Negative Large
THEN z is Negative with strength
    0.01 AND Zero with strength 0.99
IF x is Negative Medium AND
    y is Negative Large
THEN z is Zero with strength 0.92
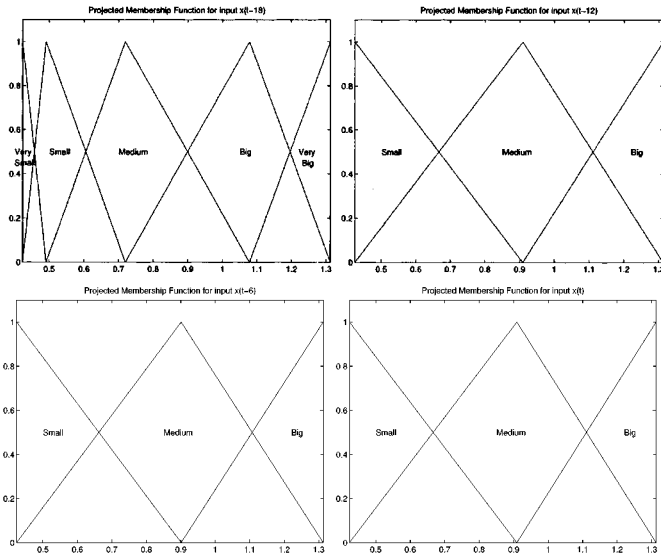   AND Positive with strength 0.08.

Fig. 11. Example 2: membership functions after projection and FuZion.

Observe that the obtained rules exhibit on each rule a clear dominance of one of the consequences, when this case appears it will be possible to eliminate the consequence with the small strength without a major impact in the numerical approximation. However, this step is a decision that must be left to the designer because it is case dependent.

B. Example 2: Predicting Chaotic Time Series

This example shows the capability of the algorithm to capture the dynamics governing the Mackey–Glass chaotic time series Figs. 13, 14. These time series were generated using the following delay differential equation:

$$\dot{x}(t) = \frac{0.2x(t-\tau)}{1 + x^{10}x(t-\tau)} - 0.1x(t) \qquad (38)$$

where $\tau = 17$. The numerical solution of this differential equation was obtained using fourth-order Runge–Kutta method, with a time step of 0.1 and initial condition $x(0) = 1.2$. The simulation was run for 2000 s and the samples were taken each second. To train and test the fuzzy system, 1000 points were extracted $t = 118$ to 1117. The first 500 points were used as training set and the remaining as validation set. First, a six step ahead predictor is constructed using past outputs as inputs of the model

$$[x(t-18)x(t-12)x(t-6)x(t)] \qquad (39)$$

and the output will be $x(t+6)$.

After applying the mountain clustering method, 57 clusters were found. Some refinement on the position of the clusters was obtained by using fuzzy C-means clustering method. After projection and FuZion the membership functions shown in Fig. 11 were obtained. A model with 135 rules was obtained.

For reasons of comparison with other methods, the prediction error was evaluated using the so called nondimensional error index (NDEI) defined as

$$\text{NDEI} = \frac{\sqrt{\frac{1}{N}\sum_{i=1}^{N}(T(i) - O(i))^2}}{\sigma(T)} \qquad (40)$$

TABLE I
EXAMPLE 2: PERFORMANCE FOR PREDICTION SIX STEPS AHEAD. THE RESULTS
FROM PREVIOUS WORKS WERE TAKEN FROM [3]

Method	Training Cases	Non-dimensional error index
AFRELI	500	0.0493
AFRELI (with optional step)	500	0.0324
ANFIS	500	0.007
AR model	500	0.19
Cascaded correlation NN	500	0.06
Backpropagation MLP	500	0.02
6th order polynomial	500	0.04

TABLE II
EXAMPLE 2: PERFORMANCE FOR PREDICTION 84 STEPS AHEAD (THE FIRST
SEVEN ROWS) AND 85 (THE LAST FOUR ROWS). RESULTS FOR THE FIRST
SEVEN METHODS ARE OBTAINED BY SIMULATION OF THE MODEL OBTAINED
FOR PREDICTION SIX STEPS AHEAD. RESULTS FOR LOCALIZED RECEPTIVE
FIELDS (LRFS) AND MULTIRESOLUTION HIERARCHIES (MRHS) ARE FOR
NEURONS TRAINED TO PREDICT 85 STEPS AHEAD. THE RESULTS FROM
PREVIOUS WORKS WERE TAKEN FROM [3]

Method	Training Cases	Non-dimensional error index
AFRELI	500	0.1544
AFRELI (with optional step)	500	0.1040
ANFIS	500	0.036
AR model	500	0.39
Cascaded correlation NN	500	0.32
Backpropagation MLP	500	0.05
6th order polynomial	500	0.85
LRF	500	0.10-0.25
MRH	500	0.05

where
$T(i)$ desired output;

$O(i)$ predicted output;

$\sigma(T)$ standard deviation of the target series.

Tables I and II show some comparative results. In this example the impact of the use of the optional step of optimization can be observed. It is clear that the improvement of this optional step is small (reduction of about 30% on the NDEI) but, of course, on certain applications this value could be significant. Observe once more that the numerical performance is similar to other techniques but the obtained model is not only good from the precision point of view, but also a significant value is added with the interpretability of the obtained rule base.
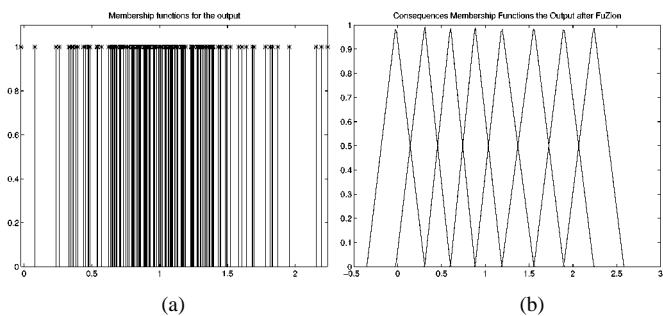
Fig. 12.   Example 2. (a) Singletons. (b) Membership functions with linguistic meaning.
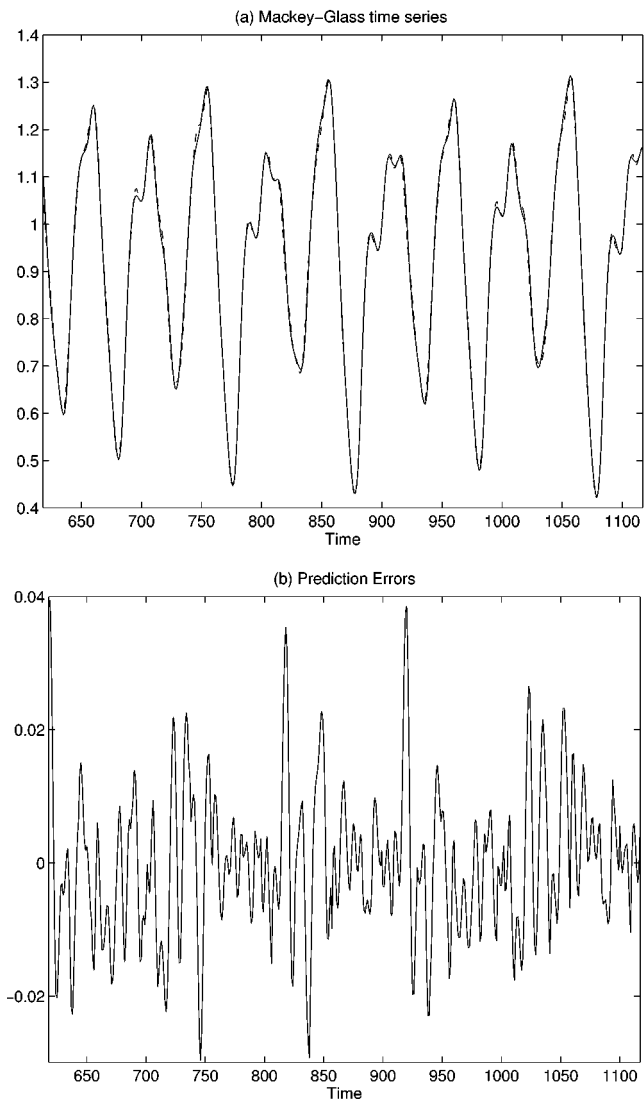

Fig. 13.   Example 2. (a) Mackey–Glass time series (solid line) from $t = 618$ to 1117 and six steps ahead prediction (dashed line). (b) Prediction errors.

Fig. 14.   Example 2. (a) Mackey–Glass time series (solid line) from $t = 118$ to 1117 and 84 steps ahead prediction (dashed line). (b) Prediction errors.

VI. CONCLUSION

The capabilities of fuzzy logic systems to approach nonlinear functions have been shown abundantly in the literature. However, the comparative advantage of fuzzy systems in comparison with other "universal approximators" is its linguistic interpretability Fig. 12. The newly proposed AFRELI algorithm in comb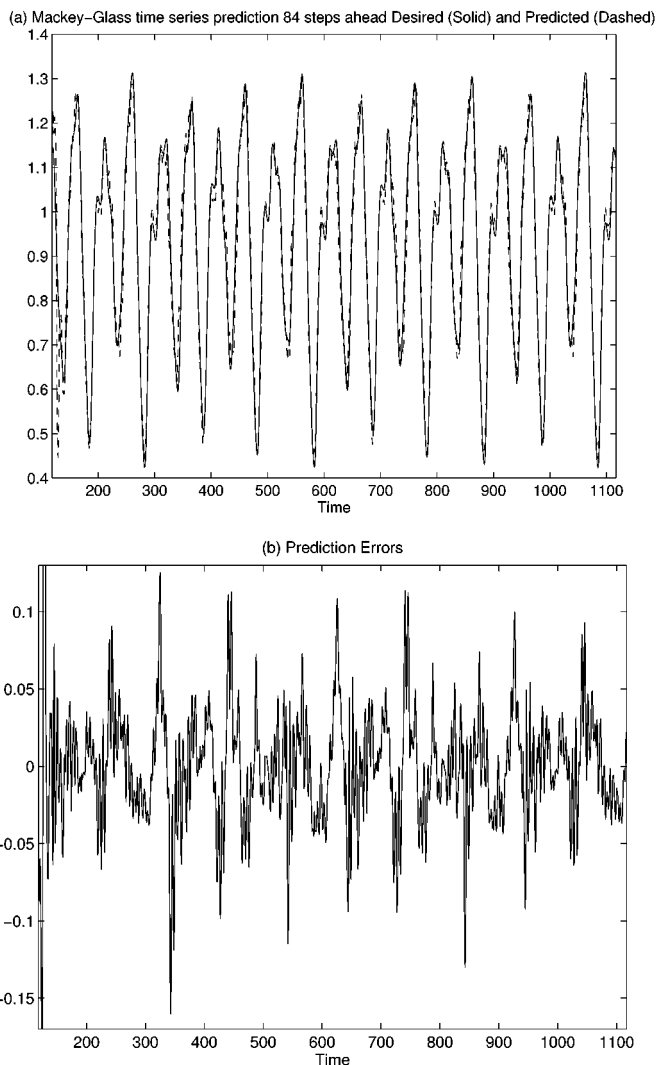ination with the FuZion algorithm guarantees a good tradeoff between numerical accuracy and interpretability. The method exploits some successful elements proposed in other methods to reduce the complexity of the model construction.

The algorithm generates automatically the fuzzy sets and the interactive labeling process (with intervention of the designer) guarantees an agreement between the fuzzy set and the assigned label.

The method generates a rule base covering all the possible cases. This guarantees the completeness of the rule base, but the associated drawback is the exponential growth of the rule base as the number of inputs increases. However, from the computational point of view this is only a storage problem because the description of the fuzzy sets guarantees that only 2^N rules (N number of inputs) are activated on each inference making the inference process fast because only a limited number of rules is evaluated.

The completeness of the rule base plus the proposed initialization methods using "expert knowledge" or the information given by the "reduced" model improves the generalization properties of the model and guarantees a lower bound on the quality of the model. The use of RLS guarantees also that the lack of

excitation will not affect the performance of unexcited rules as does the classical LS method.

The numerical accuracy of the algorithm is directly related with the choices of the parameter M governing the FuZion algorithm and the choices in the clustering algorithm. When the number of inputs is large, the use of mountain clustering method will be limited. A practical solution is to use the fuzzy C-means algorithm with an overestimated number of clusters.

Some improvements of the numerical performance of the model can be obtained by making a "fine" tuning of the parameters of the antecedents by means of constrained gradient descent techniques.

REFERENCES

[1] D. Broadbent, "Studies in long term memory," in *The Magic Number Seven After Fifteen Years*. Reading, MA: Addison-Wesley, 1975, p. 3.

[2] L.-X. Wang, *Adaptive Fuzzy Systems and Control*. Englewood Cliffs, NJ: Prentice-Hall, 1994.

[3] J.-S. Jang, "Neuro-fuzzy modeling: Architectures, analyses, and applications," Ph.D. dissertation, Univ. California, Berkeley, 1992.

[4] J.-S. R. Jang, "Structure determination in fuzzy modeling: A fuzzy cart approach," in *Proc. IEEE Int. Conf. Fuzzy Syst.*, Orlando, FL, June 1994.

[5] S. Tan and J. Vandewalle, "An on-line structural and parametric scheme for fuzzy modeling," in *Proc. 6th Int. Fuzzy Syst. Assoc. World Congress IFSA'95*, Sao Paulo, Brazil, July 1995, pp. 189–192.

[6] M. Sugeno and T. Yasukawa, "A fuzzy logic based approach to qualitative modeling," *IEEE Trans. Fuzzy Syst.*, vol. 1, pp. 7–31, Feb. 1993.

[7] R. Yager and D. Filev, *Essentials of Fuzzy Modeling and Control*. New York: Wiley, 1994.

[8] N. Lori and P. Costa Branco, "Autonomous mountain-clustering method applied to fuzzy systems modeling," in *Intelligent Engineering Systems Through Artificial Neural Networks, Smart Engineering Systems: Fuzzy Logic and Evolutionary Programming*, C. Dagli, M. Akay, C. P. Chen, B. Fernández, and J. Ghosh, Eds. New York: ASME, 1995, vol. 5, pp. 311–316.

[9] J.-S. Jang, C.-T. Sun, and E. Mizutani, *Neuro Fuzzy and Soft Computing*. Englewood Cliffs, NJ: Prentice-Hall, 1997.

[10] J. Valente de Oliveira, "Semantic constraints for membership function optimization," *IEEE Trans. Syst., Man, Cybern.*, pt. A, vol. 29, pp. 128–138, Jan. 1999.

[11] M. Setnes, R. Babuska, and H. Verbruggen, "Rule-based modeling: Precision and transparency," *IEEE Trans. Syst., Man, Cybern.*, pt. C, vol. 28, pp. 165–170, Feb. 1998.

[12] M. Setnes, R. Babuska, U. Kaymak, and H. van Nauta Lemke, "Similarity measures in fuzzy rule base simplification," *IEEE Trans. Syst., Man, Cybern.*, pt. B, vol. 28, pp. 376–386, June 1998.

[13] W. Pedrycz, "Why triangular membership functions?," *Fuzzy Sets Syst.*, vol. 64, pp. 21–30, 1994.

[14] W. Pedrycz and J. Valente de Oliveira, "Optimization of fuzzy models," *IEEE Trans. Syst., Man, Cybern.*, pt. B, vol. 26, pp. 627–636, Aug. 1996.

[15] "Fuzzy Logic Toolbox, User's Guide V-2.0," Mathworks, 1998.

[16] J. Bezdek, "A physical interpretation of fuzzy isodata," *IEEE Trans. Syst., Man, Cybern.*, vol. 6, pp. 387–389, May 1976.

[17] J. Yen and L. Wang, "Simplifying fuzzy rule-based models using orthogonal transformation methods," *IEEE Trans. Syst., Man, Cybern.*, pt. B, vol. 29, pp. 13–24, Feb. 1999.

[18] G. Biermann, *Factorization Methods for Discrete Sequential Estimation*. New York: Academic, 1977.

[19] K. Nozaki, H. Ishibuchi, and H. Tanaka, "A simple but powerful heuristic method for generating fuzzy rules from numerical data," *Fuzzy Sets Syst.*, no. 86, pp. 251–270, 1997.

[20] J. Espinosa and J. Vandewalle, "Fuzzy modeling with linguistic integrity," in *Proc. Int. Workshop Adv. Black Box Tech. Nonlinear Modeling*, Leuven, Belgium, July 1998, pp. 197–202.

[21] ——, "Fuzzy modeling and identification, a guide for the user," in *Proc. IEEE Singapore Int. Symp. Contr. Theory Applicat.*, Singapore, Aug. 1997, pp. 437–443.

[22] ——, "Fuzzy modeling and identification, using afreli and fuzion algorithms," in *Proc. 5th Int. Conf. Soft Computing IIZUKA'98*, Iizuka, Japan, Oct. 1998, pp. 535–540.