Correspondence

Comments on the Benchmarks in "A Proposal for Improving the Accuracy of Linguistic Modeling" and Related Articles

Johannes A. (Hans) Roubos and Robert Babuška

Abstract—In the above paper, the so-called accurate linguistic modeling (ALM) method was proposed to improve the accuracy of linguistic fuzzy models. A number of examples are given to demonstrate the benefits of the approach. We show that: 1) these examples are not suitable as benchmarks or demonstrators of nonlinear modeling techniques and 2) better results can be obtained by using both standard regression tools as well as other fuzzy modeling techniques. We argue that benchmark examples that are used in articles to demonstrate the effectiveness of fuzzy modeling techniques should be selected with great care. Critical analysis of the results should be made and linear models should be regarded as a lower bound on the acceptable performance.

Index Terms—Linguistic fuzzy model, rice data, spline model, Takagi–Sugeno (TS) fuzzy model.

I. INTRODUCTION

Fuzzy models differ from nonsymbolic methods such as neural networks mainly in that they can represent knowledge in a transparent manner using IF–THEN rules. Linguistic interpretability and transparency are therefore important aspects in fuzzy modeling [2]–[7]. Recently, we have witnessed a strong emphasis on approaches focusing on the prediction performance and accuracy of fuzzy systems. New construction techniques are being introduced and their performance is usually assessed by using simulation examples and real-world data. The paper by Cordón and Herrera [1] is an example of one such article. The authors acknowledge the fact that linguistic models with fixed and evenly spaced membership functions result in models with poor approximation properties. They propose the *accurate linguistic modeling* (ALM) method which allows consequents with two fuzzy sets and thereby improves the accuracy.

Three examples are given to demonstrate the benefits of the proposed technique: The "rice data," a bivariate function, and a modeling problem encountered in electricity distribution. In this paper, we discuss the first two examples (the data set of the third example was not available to us). We show that these examples are not suitable as benchmarks or demonstrators of nonlinear modeling techniques and that better results can be obtained by using both standard regression tools as well as other fuzzy modeling techniques. As the rice data set was also used by other researchers, we have included their results in our discussion, too.

II. RICE TASTE EVALUATION

The "rice data" problem was originally introduced in [8]. Later, it was used as a benchmark by several other authors [1], [9]–[12]. In [1],

the data set is described as follows: "The modeling of this problem becomes very complex due to the large quantity of relevant variables and to the fact that the problem-solving goal is not only to obtain an accurate model but also a user-interpretable model representing the nonlinear relationships existing in the problem as well as putting some light on the reasoning process performed by human experts." In the following, we show that this modeling problem is not that complex and that standard regression techniques yield accurate and interpretable models.

Rice overall evaluation was done by experts on the basis of five characterics: flavor, appearance, taste, stickiness, and toughness. These characteristics are used as real-valued inputs to the model. The output is the overall evaluation given on a real scale (positive and negative values). The data set contains 105 samples of different rice kinds and all the variables are normalized in the interval $[0, 1]$. For training and evaluation purposes, the data set is randomly partitioned into training data (75 samples) and test data (30 samples). Repeated training and validation runs are performed to obtain statistically relevant results [1], [9], [10].

A. Linear Regression

Without having any prior knowledge about the problem, one usually first analyzes the data through correlation analysis. Fig. 1 shows strong linear correlations between some of the inputs and the output. This indicates that a linear regression model may already be adequate for this data.

Given the small number of potential regressors, any standard input-selection technique can be used, such as forward or backward selection, stepwise regression and even exhaustive search. By using exhaustive search, we found that a linear model using inputs x_1 to x_4 and an offset (five parameters in total) gives the best cross-validation mean-squared prediction error $\mathrm{MSE} = 0.0014$, averaged over ten runs; see Table I:[1]

$$y = 0.1490x_1 + 0.1615x_2 + 0.5464x_3 + 0.2561x_4 - 0.0631. \quad (1)$$

In terms of prediction accuracy, this linear model outperforms the fuzzy models from [1] and other related literature. Also, the interpretation of this model is quite straightforward. All the considered features positively contribute to the overall rice evaluation and that the largest parameter is associated with input x_3 (taste).

Interestingly, a linear model using x_3 only gives $\mathrm{MSE} = 0.0021$ on the test data, which is a result as good as the best fuzzy model in [1] and related works of the authors

$$y = 1.0879x_3. \quad (2)$$

As the singleton fuzzy model reported in [8] achieves a slightly better result, a question arises whether the performance of the linear regression model can be improved by using nonlinear regression (including fuzzy models).

B. Polynomial Regression

By using a range of input-selection techniques, polynomial regression models can be constructed. A simple exhaustive search over model

---

[1]Some of the performance indexes have been converted from the original ones to the index used in [1] (e.g., the error in [8] and [9] was the summed quadratic error divided by two).
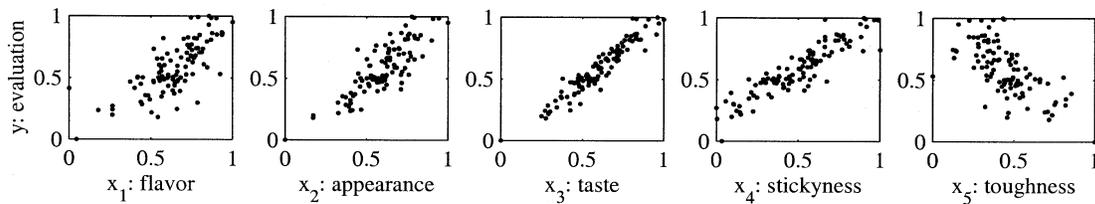
Fig. 1.   Scatter plots of the rice data.

TABLE I

RESULTS OBTAINED BY VARIOUS AUTHORS FOR THE RICE DATA PROBLEM. THE MSE VALUES ARE AVERAGES OVER 7–12 CROSS-VALIDATION RUNS (* = NOT GIVEN IN THE REFERENCE)

Method	# Rules	# Labels	0.5 MSE_{train}	0.5 MSE_{test}
Polynomial model with four inputs [this paper]	—	—	0.0011	0.0012
Linear model with four inputs [this paper]	1	—	0.0012	0.0014
TS fuzzy model with three inputs [this paper]	2	—	0.0012	0.0014
Linear model with one input [this paper]	1	—	0.0020	0.0021
D-MOGUL [10]	21	24	0.0026	0.0021
WM-method [10]	27	24	0.0026	0.0027
ALM I-based [1]	136	18	0.0014	0.0028
ALM WM-based [1]	12	18	0.0019	0.0029
ALM WM-based [1]	5	12	0.0034	0.0040
SA+CH-method [10]	*	52	0.0003	0.0108
Singleton model, gradient [8]	243	15	0.0006	0.0013
Singleton model, heuristic + gradient [8]	32	10	0.0009	0.0016
Heuristic method [9]	>32	30	0.0007	0.0029
Singleton model, heuristic + gradient [8]	3125	25	0.0001	0.0033
Heuristic method [9]	32	15	0.0029	0.0040

structures including the original inputs and products of two input variables (bilinear models) results in the following model:

$$y = 0.1371x_2 + 0.5804x_3 + 0.3332x_1x_4 + 0.1941x_4x_5. \quad (3)$$

Again, the individual parameters should not be difficult to interpret by experts. The product terms represent combined effects of the attributes involved. One can see in Table I that this model is marginally better than the best fuzzy model from the literature, while using only four parameters (compared to 243 parameters used in [8]). However, this does not suggest that fuzzy models are inferior to other techniques, as shown in Section II-C.

C. Fuzzy TS Model

By using an automated technique based on Gustafson–Kessel (GK) fuzzy clustering [13], a two-rule Takagi–Sugeno (TS) fuzzy model was constructed from the data. Two rules represent the maximum reasonable complexity, as by adding more rules the cross-validation error increases quite rapidly (Fig. 2). This confirms the observations made in Section II-A on the remarkable accuracy of the linear model.

In the TS model, input x_4 is partitioned into two fuzzy subsets labeled as 'Low' and 'High'. The corresponding rules are as follows.

1)

If x_4 is low, then
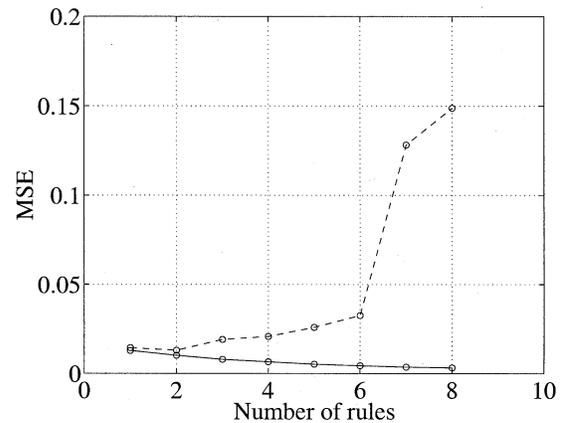$$y = 0.1243x_1 + 0.6017x_3 + 0.2757x_4 - 0.0074. \quad (4)$$

Fig. 2.   Average MSE over ten runs for the training (solid line) and test (dashed line) rice data for the TS fuzzy model with an increasing number of rules.

2) **If** x_4 is High, **then** $y = 0.2059x_1 + 0.6774x_3 + 0.2335x_4 - 0.0651$.

The performance of this model is identical to that of the linear model (1). Note, however, that the TS model only uses three input regressors, rather than four. The consequent parameters of the individual rules can be interpreted in a similar way as in the linear model. Comparing the two rules, one can see that the coefficient in front of x_1 is larger in the second rule, which means that x_1 has more influence on y when x_4 is high. This corresponds well with the interpretation of the polynomial model (3).
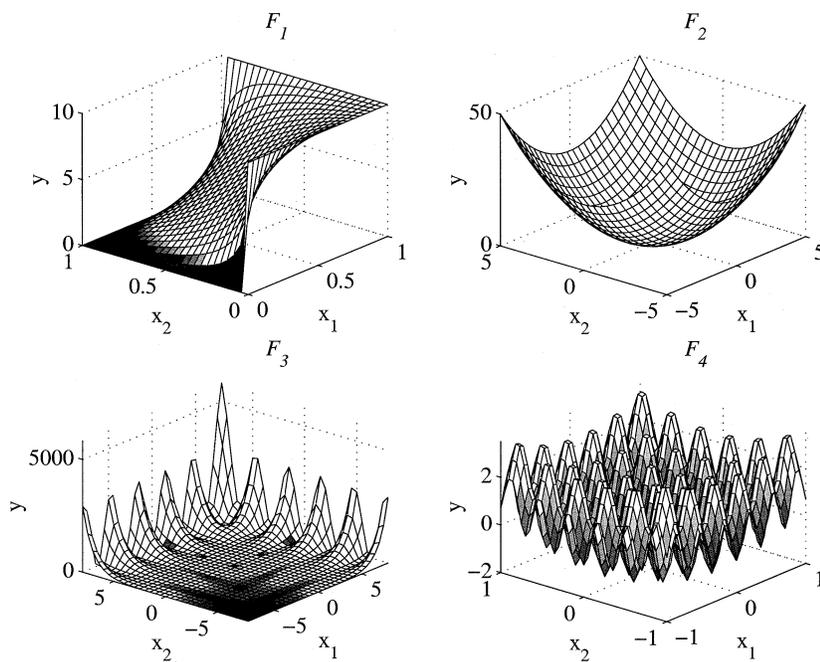
Fig. 3.   Output surfaces for functions $F1$ through $F4$.

III. APPROXIMATION OF BIVARIATE FUNCTIONS

In addition to the rice data, Cordón *et al.* used several bivariate functions to demonstrate their methods [1], [14]–[19]. Four frequently used examples are shown in Fig. 3.

Note that functions $F1$ and $F2$ are reasonably interesting for the study of linguistic fuzzy modeling as their behavior can be summarized by using a relatively small number of fuzzy IF–THEN rules. However, functions $F3$ and $F4$ are characterized by a complex behavior with a large number of local extrema and therefore are not likely to yield any useful fuzzy models. It will be shown that simple lookup tables give more accurate and transparent results than those reported in the literature.

A. Function F_1

The following function was used in [1], [14], and [15]:

$$F_1(x_1, x_2) = 10 \cdot \frac{x_1 - x_1 x_2}{x_1 - 2 x_1 x_2 + x_2} \qquad (5)$$

with $x_1, x_2 \in [0, 1]$ and $F_1(x_1, x_2) \in [0, 10]$. A training data set with 676 samples[2] was obtained for $x_1, x_2 \in \{0, 0.04, \ldots, 1\}$ with $F_1(0, 0) \triangleq 10$ and $F_1(1, 1) \triangleq 0$. The test set contains 67 randomly distributed data pairs.

Note that the use of noise-free training data placed on a grid covering the entire domain makes the approximation task very easy. A more realistic approach would be to generate the training data randomly (perhaps deliberately avoiding parts of the domain) and to test the model generalization capability on the entire grid.

Many techniques can be applied to this approximation problem. Here, we show that standard cubic-spline approximation with numerical optimization of the knots results in a model of the same complexity, but of considerably higher accuracy than the best linguistic fuzzy model from the literature. We also demonstrate that the same kind of rule-based interpretation can be given as with the ALM method [1].

---

[2]Cordón et al. use 674 samples, omitting the data points at (0, 0) and (1, 1) which are not defined by (5).

TABLE II
LOOKUP TABLE APPROXIMATION OF FUNCTION F_1

$x_1 \backslash x_2$	0.000	0.035	0.192	0.439	0.818	0.966	1.000
1.000	10.0	9.8	10.0	10.0	10.0	10.0	10.0
0.966	0.7	5.0	8.7	9.6	9.9	10.0	10.0
0.818	0.0	1.3	5.0	7.7	9.5	9.9	10.0
0.439	0.0	0.4	2.3	5.0	8.5	9.7	10.0
0.192	0.0	0.1	0.5	1.5	5.0	8.6	10.0
0.035	0.0	0.0	0.1	0.3	1.4	5.0	9.2
0.000	0.0	0.0	0.0	0.0	0.0	0.2	0.0

The standard MATLAB function `griddata` was used to approximate the training data on a grid of 49 points (this is the same number of rules as the smallest fuzzy model from the literature, see Table IV). The positions of the two extreme grid points in each input domain were fixed to 0 and 1, respectively. The remaining five points were selected randomly from the uniform distribution and then optimized by at most ten iterations of the Levenberg-Marquardt algorithm (using the standard MATLAB function `lsqnonlin`). Out of ten randomly initialized runs, the following coordinates of the grid points for both x_1 and x_2 give the best fit on the training data $[0, 0.0347, 0.1918, 0.4390, 0.8175, 0.9655, 1]$. The corresponding lookup table values (model outputs) at the Cartesian product of the grid points are given in Table II.

A linguistic interpretation in terms of rules with double consequents (in the terminology of [1]) is easily obtained by choosing, for instance, five triangular membership functions for the output (very small, small, medium, large and very large) and listing the labels for which the F_1 value has nonzero membership degrees, see Table III. This is the simplest case of linguistic approximation [20]. This rule base is similar to the one given in Table IV in [1], however, the spline model is five times more accurate.

Table IV summarizes the results obtained with the spline model and several other models from the literature. Note that in terms of numerical performance, the spline model is superior to all the models, but the TS model, which has at least three times more parameters. By increasing
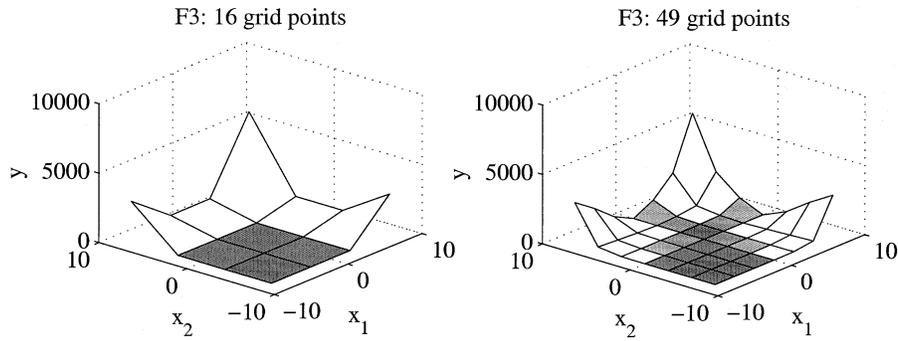
Fig. 4. F_3 surfaces by grid interpolation with 16 (left) and 49 (right) grid points.

TABLE III
LINGUISTIC RULE BASE CORRESPONDING TO THE LOOKUP TABLE FOR F_1

$x_1 \backslash x_2$	ES	VS	SM	ME	LA	VL	EL
EL	VL	VL	VL	VL	VL	VL	VL
VL	VS/SM	ME	LA/VL	LA/VL	VL	VL	VL
LA	VS	VS/SM	ME	LA/VL	LA/VL	VL	VL
ME	VS	VS/SM	VS/SM	ME	LA/VL	LA/VL	VL
SM	VS	VS	VS/SM	VS/SM	ME	LA/VL	VL
VS	VS	VS	VS	VS/SM	VS/SM	ME	LA/VL
ES	VS	VS	VS	VS	VS	VS	VS

TABLE IV
RESULTS OBTAINED FOR FUNCTION F_1

Method	# Rules	$\frac{1}{2}$ MSE$_{\text{train}}$	$\frac{1}{2}$ MSE$_{\text{test}}$
spline model [this paper]	49	0.007	0.004
TS model EFS [15]	49	0.015	0.002
Genetic learning [17]	81	0.015	0.014
GR + MF [16]	80	0.025	0.017
GFS [14]	67	0.019	0.021
ALM [1]	55	0.019	0.022

the number of rules to 81, the performance of the spline model improves by a factor of six.

B. Function F_2

In [14] and [15], Mamdani and TS models were constructed for the following function:

$$F_2(x_1, x_2) = x_1^2 + x_2^2 \quad x_1, x_2 \in [-5, 5] \quad (6)$$

using 1681 training data pairs (uniformly spaced on a 41×41 grid) and 168 random test data pairs. A summary of the results is given in Table V. The Mamdani model fits the data reasonably well, however, the output surface is irregular, as shown in [14]. Better results were obtained with the TS model, but the number of rules (49) is very large for such a smooth surface.

By applying a fuzzy modeling method based on GK clustering and genetic optimization [3], a four-rule TS model was obtained whose performance is similar to the 49-rule TS model given in [15], see Table V.

Note that spline and polynomial regression models similar to those applied in Section II give for this function a zero approximation error for both the training and the test data.

TABLE V
RESULTS OBTAINED FOR FUNCTION F_2

Method	# Rules	$\frac{1}{2}$ MSE$_{\text{train}}$	$\frac{1}{2}$ MSE$_{\text{test}}$
T2 [15]	49	0.0066	0.0066
GK [this paper]	4	0.0087	0.0071
M2 [15], [14]	62	0.3358	0.2626

C. Functions F_3 and F_4

F_3 is the following bi-variate function:

$$F_3(x_1, x_2) = e^{x_1} \cdot \sin^2 x_2 + e^{x_2} \cdot \sin^2 x_1 \quad (7)$$

with $x_1, x_2 \in [-8, 8]$. In [15], Mamdani and TS models were constructed based on 1089 training and 108 test data pairs, generated similarly to F_2. The models found contain 49 to 100 rules, with 0.5 MSE in the range 28 691–68 971 for the training data and 19 838–41 000 for the validation data.[3] From the plots shown in [1] it is clear that none of the models found approximates the original surface well. Moreover, it can easily be demonstrated that the presented fuzzy models have no advantage over standard interpolation methods. A simple fixed-grid interpolation (MATLAB's function `griddata`) with 16 uniformly distributed grid points gives 0.5 MSE = 36 705 for training and 0.5 MSE = 35 781 for testing, respectively. With 49 grid points, 0.5 MSE = 13 316 is obtained for training and 0.5 MSE = 10 502 for validation (see Fig. 4).

Similar results can also be shown for function F_4[14], [15], [19], which is given by

$$F_4(x_1, x_2) = x_1^2 + x_2^2 - \cos(18x_1) - \cos(18x_2) \quad (8)$$

with $x_1, x_2 \in [-1, 1]$. Here, 1681 training data pairs (uniformly spaced on input grid with 41×41 values) and 168 random test data pairs were used. More accurate approximations for F_4 are given in [18], however, the number of rules becomes extremely large. Models with 200 to 3000 rules were presented for F_4, while only 1681 training data points are available. On the other hand, also models with 7–15 rules were given, which are clearly too simple to approximate F_4. The 0.5 MSE given for the training data is between 0.50 and 0.58 which is close to the performance one obtains by taking the average value of the data as a constant model (0.61). The performance obtained for the validation

---

[3]The performance on the validation data is better than the performance on the training data set because of the random choice of the former one. The uniformly spaced training data include the extrema of the function which are difficult to approximate. These extrema, however, have a low chance of being randomly selected.

data is even worse with the nine-rule model ($0.5 \, \mathrm{MSE} = 0.66$). These results are rather meaningless, which could have been discovered by using the variance accounted for (VAF) performance index instead of the MSE.

IV. DISCUSSION AND CONCLUSION

We argued that the numerical examples used in [1] and a number of related articles are not suitable as benchmarks or demonstrators of nonlinear (fuzzy) modeling methods. The rice data data very easy to model by simple linear models, whose accuracy is in fact superior to that of most fuzzy models reported in the literature (see Table I). A bilinear model and a properly tuned two-rule TS fuzzy model give a slight improvement over the linear model. All these models are transparent and easy to interpret. This data set thus does not appear suitable for demonstrating either accuracy or interpretability aspects of fuzzy modeling techniques. One can also question the approximation power of the proposed ALM method [1] and the other methods from the cited literature, as the results given for the training data are in some cases considerably worse than those achieved with the linear model (Table I). Some of the fuzzy models also seem to suffer from severe over-fitting, as up to 30 times lower accuracy has been reported for the validation data.

Next, we discussed the bivariate functions F_1 through F_4 that have been used as examples in several publications [14]–[19], [21]. In these examples, the training data are generated on a uniform grid, while the validation data are random. This choice makes the approximation task very easy. The usual approach is to generate training data randomly and to test the model generalization capability on the entire grid. This is much more realistic, as with real systems, one is often not able or allowed to design experiments to obtain data on a grid. A typical example is the identification of a dynamic system in which case the data distribution is dictated by the process dynamics and the choice of the input sequence. In addition, real data will always be corrupted by noise and other disturbances.

For training data placed on a regular grid, standard spline methods can easily be applied. The spline model obtained in this paper for function F_1 is twice as accurate as the best fuzzy linguistic model reported in the literature (with the same number of interpretable rules). Note that spline models have the same interpretation as linguistic (singleton) fuzzy models [22]. Some of the linguistic models reported for functions F_3 and F_4 perform worse than the baseline model using just the mean of the output data. Other models contain more rules (and therefore parameters) than data points in the training data set.

Summarizing, we propose that benchmark examples used to demonstrate the effectiveness of fuzzy modeling techniques should be selected with great care. The problem should not be too simple. While the use of small simple examples has a high pedagogical value, one must be aware of the fact that the results do not automatically carry over to more complex problems. At the same time, many methods can be found that will solve the simple problem well (as shown in this article). The quality of the proposed technique should be evident from a comparison with state-of-the-art regression or classification techniques. Critical analysis of the results should be made and linear models should be regarded as a lower bound on the acceptable performance. Fuzzy models certainly have the potential to outperform other techniques, but this must be clearly shown by comparisons with nonfuzzy approaches. Only in this way, fuzzy techniques can gain higher credibility outside the fuzzy community.

REFERENCES

[1] O. Cordón and F. Herrera, "A proposal for improving the accuracy of linguistic modeling," *IEEE Trans. Fuzzy Syst.*, vol. 8, pp. 335–344, June 2000.

[2] M. Setnes, R. Babuška, and H. B. Verbruggen, "Rule-based modeling: Precision and transparency," *IEEE Trans. Syst., Man, Cybern. C*, vol. 28, pp. 165–169, Feb. 1998.

[3] M. Setnes and J. A. Roubos, "GA-fuzzy modeling and classification: Complexity and performance," *IEEE Trans. Fuzzy Syst.*, vol. 8, pp. 509–522, Oct. 2000.

[4] J. Valente de Oliveira, "Semantic constraints for membership function optimization," *IEEE Trans. Fuzzy Syst.*, vol. 19, pp. 128–138, Feb. 1999.

[5] H. Pomares, I. Rojas, J. Ortega, J. Gonzalez, and A. Prieto, "A systematic approach to a self-generating fuzzy rule-table for function approximation," *IEEE Trans. Syst., Man, Cybern. B*, vol. 30, no. 3, pp. 431–447, 2000.

[6] Y. Jin, "Fuzzy modeling of high-dimensional systems: Complexity reduction and interpretability improvement," *IEEE Trans. Fuzzy Syst.*, vol. 8, pp. 212–222, Apr. 2000.

[7] T. A. Johansen, R. Shorten, and R. Murray-Smith, "On the interpretation and identification of dynamic Takagi–Sugeno fuzzy models," *IEEE Trans. Fuzzy Syst.*, vol. 8, pp. 297–313, June 2000.

[8] H. Ishibuchi, K. Nozaki, H. Tanaka, Y. Hosaka, and M. Matsuda, "Empirical study on learning in fuzzy systems by rice taste analysis," *Fuzzy Sets Syst.*, vol. 64, pp. 129–144, 1994.

[9] K. Nozaki, H. Ishibuchi, and H. Tanaka, "A simple but powerful heuristic method for generating fuzzy rules from numerical data," *Fuzzy Sets Syst.*, vol. 86, pp. 251–270, 1997.

[10] O. Cordón, F. Herrera, and P. Villar, "Analysis and guidelines to obtain a good uniform fuzzy partition granularity for fuzzy rule-based systems using simulated annealing," *Int. J. Approx. Reason.*, vol. 25, pp. 187–215, 2000.

[11] X. Z. Wang, Y. D. Wang, X. F. Xu, W. D. Ling, and D. S. Yeung, "A new approach to fuzzy rule generation: Fuzzy extension matrix," *Fuzzy Sets Syst.*, vol. 123, pp. 291–306, 2001.

[12] E. C. C. Tsang, X. Z. Wang, and D. S. Wang, "Improving learning accuracy of fuzzy decision trees by hybrid neural networks," *IEEE Trans. Fuzzy Syst.*, vol. 8, pp. 601–614, Oct. 2000.

[13] R. Babuška, *Fuzzy Modeling for Control*. Boston, MA: Kluwer, 1998.

[14] O. Cordón and F. Herrera, "A three-stage evolutionary process for learning descriptive and approximative fuzzy-logic-controller knowledge bases from examples," *Int. J. Approx. Reason.*, vol. 17, pp. 369–407, 1997.

[15] ——, "A two-stage evolutionary process for designing TSK fuzzy rule-based systems," *IEEE Trans. Syst., Man, Cybern. B*, vol. 29, pp. 703–715, Dec. 1999.

[16] O. Cordón, F. Herrera, and P. Villar, "Generating the knowledge base of a fuzzy rule-based system by the genetic learning of the data base," *IEEE Trans. Fuzzy Syst.*, vol. 9, pp. 667–674, Aug. 2001.

[17] O. Cordón, F. Herrera, L. Magdalena, and P. Villar, "A genetic learning process for the scaling of the fuzzy rule-based system data base," *Inform. Sci.*, vol. 136, pp. 85–107, 2001.

[18] O. Cordón and F. Herrera, "Fuzzy modeling by hierarchically built fuzzy rule bases," *Int. J. Approx. Reason.*, vol. 27, pp. 61–93, 2001.

[19] F. Herrera, O. Cordón, and J. L. Verdegay, "A learning process for fuzzy control rules using genetic algorithms," *Fuzzy Sets Syst.*, vol. 100, pp. 143–158, 1998.

[20] F. Esragh and E. H. Mamdani, "A general approach to linguistic approximation," *Int. J. Man Machine Studies*, vol. 11, pp. 501–519, 1979.

[21] O. Cordón, F. Herrera, and A. Peregín, "Searching for basic properties obtaining robust implication operators in fuzzy control," *Fuzzy Sets Syst.*, vol. 111, pp. 237–251, 2000.

[22] M. Brown and C. Harris, *Neurofuzzy Adaptive Modeling and Control*. Upper Saddle River, NJ: Prentice-Hall, 1994.