Combining Rule Weight Learning and Rule Selection to Obtain Simpler and More Accurate Linguistic Fuzzy Models

Rafael Alcalá[1], Oscar Cordón[2], and Francisco Herrera[2]

[1] Department of Computer Science, University of Jaén
E-23071 Jaén, Spain
alcala@ujaen.es

[2] Department of Computer Science and A.I., University of Granada
E-18071 Granada, Spain
{ocordon,herrera}@decsai.ugr.es

Abstract. In complex multidimensional problems with a highly nonlinear input-output relation, inconsistent or redundant rules can be found in the fuzzy model rule base, which can result in a loss of accuracy and interpretability. Moreover, the rules could not cooperate in the best possible way.

It is known that the use of rule weights as a local tuning of linguistic rules, enables the linguistic fuzzy models to cope with inefficient and/or redundant rules and thereby enhances the robustness, flexibility and system modeling capability. On the other hand, rule selection performs a simplification of the previously identified fuzzy rule base, removing inefficient and/or redundant rules in order to improve the cooperation among them. Since both approaches are not isolated and they have complementary characteristics, they could be combined among them. In this work, we analyze the hybridization of both techniques to obtain simpler and more accurate linguistic fuzzy models.

1 Introduction

One of the problems associated with Linguistic Fuzzy Modeling is its lack of accuracy when modeling some complex systems. It is due to the inflexibility of the concept of linguistic variable, which imposes hard restrictions to the fuzzy rule structure [2]. Therefore, in this kind of modeling the *accuracy* and the *interpretability* of the obtained model are contradictory properties directly depending on the learning process and/or the model structure.

Furthermore, in complex multidimensional problems with highly nonlinear input-output relations many redundant, inconsistent and conflicting rules are usually found in the obtained rule base, which is detrimental to the linguistic fuzzy model performance and interpretability. In any case, these rules could not cooperate appropriately.

To overcome these drawbacks, many different possibilities to improve the Linguistic Fuzzy Modeling have been considered in the specialized literature [4].

J. Lawry, J. Shanahan, A. Ralescu (Eds.): Modelling with Words, LNAI 2873, pp. 44–64, 2003.

All of these approaches share the common idea of improving the way in which the linguistic fuzzy model performs the interpolative reasoning by inducing a better cooperation among the rules in the learned model. This rule cooperation may be encouraged acting on three different model components: the data base, the rule base and the whole knowledge base (KB). Focusing on the rule base, there are different ways to induce rule cooperation acting on that component:

– Rule selection [6, 15, 17, 19, 20, 22, 28, 29, 30, 34]: It involves obtaining an optimized subset of rules from a previous rule base by selecting some of them.
– Multiple rule consequent learning [10, 24]: This approach allows the rule base to present rules where each combination of antecedents may have two or more consequents associated when it is necessary.
– Weighted linguistic rule learning [7, 21, 25, 35]: It is based on including an additional parameter for each rule that indicates its importance degree in the inference process, instead of considering all rules equally important as in the usual case.
– Rule cooperation [3, 31]: This approach follows the primary objective of inducing a better cooperation among the linguistic rules. To do so, the rule base design is made using global criteria that jointly consider the action of the different rules.

Two of the previous approaches, the weighted linguistic rule learning (accuracy purposes) and the rule selection (interpretability/simplicity purposes), present complementary characteristics. On the one hand, it is known that the use of rule weights as a local tuning of linguistic rules, enables the linguistic fuzzy models to cope with inefficient and/or redundant rules and thereby enhances the robustness, flexibility and system modeling capability. On the other hand, rule selection performs a simplification of the previously identified fuzzy rule base, removing inefficient and/or redundant rules in order to improve the cooperation among them. Furthermore, reducing the model complexity is a way to improve the system readability, i.e., a compact system with few rules requires a minor effort to be interpreted. Since both approaches are not isolated and they have complementary characteristics, they could be combined.

In this work, we analyze the hybridization of both techniques to obtain simpler and more accurate linguistic fuzzy models. To select the subset of rules with the best cooperation and the weights associated to them, different search techniques could be considered [26]. In this contribution, we will consider a Genetic Algorithm (GA) [18, 23] for this purpose. The proposal has been tested with two different real-world problems achieving good results.

This contribution proposes the use of weighted fuzzy rules and rule selection to improve simple linguistic fuzzy models. *This can be intended as a meta-method over any other linguistic rule generation method, developed to obtain simpler linguistic fuzzy models by only selecting the rules with a good cooperation while the use of rule weights improves the way in which they interact.* Depending on the combination of this technique with different fuzzy rule learning methods, different learning approaches arise. In this work, we will consider the Wang and Mendel's method [32] and an extension of this method to obtain double-consequent fuzzy rules [10] as the initial linguistic rule generation methods.

This contribution is arranged in the following way. In Sections 2 and 3, the use of rule weights and rule selection is analyzed in depth, considering them as two complementary ways to improve the linguistic model performance. Sections 4 and 5 present the proposed learning strategy and the evolutionary optimization process performing the rule selection together with the rule weight derivation. Experimental results are shown in Section 6. In Section 7, some concluding remarks are pointed out. Finally, the double-consequent rule structure is presented in Appendix A.

2 Weighted Linguistic Rules

Using rule weights [7, 21, 25, 35] has been usually considered to improve the way in which the rules interact, improving the accuracy of the learned model. In this way, rule weights involve an effective extension of the conventional fuzzy reasoning that allows the tuning of the system to be developed at the rule level [7, 25]. It is clear that considering rule weights will improve the capability of the model to perform the *interpolative reasoning* and, thus, its performance. This is one of the most interesting features of fuzzy rule-based systems (FRBSs) and plays a key role in their high performance, being a consequence of the cooperative action of the linguistic rules existing in the fuzzy rule base.

Weighted linguistic models are less interpretable than the classical ones but, in any case, these kinds of FRBSs can be interpreted to a high degree, and also make use of human knowledge and a deductive process. When weights are applied to complete rules, the corresponding weight is used to modulate the firing strength of a rule in the process of computing the defuzzified value. For human beings, it is very near to consider this weight as an importance degree associated to the rule, determining how this rule interacts with its neighboring ones. We will follow this approach, since the interpretability of the system is appropriately maintained. In addition, we will only consider weight values in $[0, 1]$ since it preserves the model readability. In this way, the use of rule weights represents an ideal framework to extend the linguistic model structure when we search for a trade-off between accuracy and interpretability.

2.1 Weighted Rule Structure and Inference System

As we have said, rule weights will be applied to complete rules. In order to do so, we will follow the weighted rule structure and the Inference System proposed in [25]:

$$\textbf{IF } X_1 \textbf{ is } A_1 \textbf{ and } \ldots \textbf{ and } X_n \textbf{ is } A_n \textbf{ THEN } Y \textbf{ is } B \textit{ with } [w],$$

where X_i (Y) are the input (output) linguistic variables, A_i (B) are the linguistic labels used in the input (output) variables, w is the real-valued rule weight, and *with* is the operator modeling the weighting of a rule.

With this structure, the fuzzy reasoning must be extended. The classical approach is to infer with the FITA (First Infer, Then Aggregate) scheme [11] considering the matching degree of the fired rules. In this contribution, the *Center of Gravity weighted by the matching degree* will be used as the defuzzification strategy [11]:

$$y_0 = \frac{\sum_i h_i \cdot w_i \cdot P_i}{\sum_i h_i \cdot w_i},$$

with y_0 being the crisp value obtained from the defuzzification process, h_i being the matching degree of the i-th rule, w_i being the weight associated to it, and P_i being the characteristic value —*Center of Gravity*— of the output fuzzy set inferred from that rule, B'_i. On the other hand, we have selected the singleton fuzzification and the *minimum t-norm* playing the role of the implication and conjunctive operators.

A simple approximation for weighted rule learning would consists of considering an optimization technique (e.g., GAs [18, 23]) to derive the associated weights of a previously obtained set of rules.

2.2 Use of Rule Weights for Implicit Rule Selection

The use of rule weights as a local tuning of linguistic rules, enables the linguistic fuzzy models to cope with inefficient and/or redundant rules and thereby enhances the robustness, flexibility and system modeling capability [25]. Hence the ability of this technique to indicate the interaction level of each rule with the remaining ones is considered, improving the global cooperation. In this way, when we start from a previous set of rules, inefficient or redundant rules could be removed by assigning a zero weight to each of them, i.e., an *implicit rule selection* could be performed.

However, weights close to zero are usually obtained from the derivation process, practically avoiding the effects of such rules but maintaining them in the KB. It is due to the large search space tackled by this process, and can not be solved by removing these rules since in some cases they could be important rules with a low interaction level. Moreover, the obtained weights are highly dependent among them and to remove rules with low weight after applying the weighting algorithm (and then normalizing) provokes very bad accuracy. On the other hand, redundant, inconsistent and conflicting rules could be weighted as important rules if their neighbors are incorrectly weighted. In most cases, the algorithm has not the ability to avoid getting stuck at local optima removing (or weighting with low weights) these kinds of rules. Therefore, rule weighting processes could be improved considering any complementary technique that directly determines what rules should be removed.

2.3 An Example of a Learning Process for Weighted FRBSs

An example for weighted rule learning would consist of the following two steps —we will use this process in our experiments for comparison purposes—:

1. Firstly, a preliminary fuzzy rule set is derived considering a specific generation process. In this work, the generation process proposed by Wang and Mendel [32] is considered.
2. Then, a learning algorithm is used to derive the associated weights of the previously obtained rules. A real-coded GA where each gene indicates the corresponding rule weight may be considered as learning algorithm. The stochastic universal sampling procedure together with an elitist selection scheme —using the Mean Square Error (MSE) as fitness— and the max-min-arithmetical crossover [16] (see Sect. 5.3) together with the uniform mutation operator can be used.

3 Selecting Cooperative Rules

In complex multidimensional problems with highly nonlinear input-output relations many redundant, inconsistent and conflicting rules are usually found in the obtained rule base (especially in the case when they are generated by only considering expert's knowledge). On the other hand, in high-dimensional problems, the number of rules in the rule base grows exponentially as more inputs are added. A large rule set might contain many redundant, inconsistent and conflicting rules. These kinds of rules are detrimental to the model performance and interpretability.

Rule selection methods directly aggregate multiple rules and/or select a subset of rules from a given fuzzy rule set in order to minimize the number of rules while at the same time maintaining (or even improving) the system performance. Inconsistent and conflicting rules that degrade the performance are eliminated, thus obtaining a more cooperative fuzzy rule set and therefore involving a potential improvement of the system accuracy. Moreover, in many cases the accuracy is not the only requirement of the model and the interpretability becomes an important aspect. Reducing the model complexity is a way to improve the system readability, i.e., a compact system with few rules requires a minor effort to be interpreted.

Rule reduction methods have been formulated using Neural Networks, clustering techniques and orthogonal transformation methods, and algorithms based on similarity measures, among others [6, 15, 28, 29, 30, 34]. In [8], a different approach was proposed which attempted to reduce the growth of the rule base by transforming elemental fuzzy rules into DNF-form.

3.1 Considering a Genetic Approach

Using GAs to search for an optimized subset of rules is motivated in the following situations:

– the integration of an expert rule set and a set of fuzzy rules extracted by means of automated learning methods [17],
– the selection of a cooperative set of rules from a candidate fuzzy rule set [9, 10, 13, 19, 20, 22],

- the selection of rules from a given KB together with the selection of the appropriate labels for the consequent variables [5],
- the selection of rules together with the tuning of membership functions by coding all of them (rules and parameters) in a chromosome [14], and
- the derivation of compact fuzzy models through complexity reduction combining fuzzy clustering, rule reduction by orthogonal techniques, similarity driving simplification and genetic optimization [27].

Two of them are of particular interest in our case, the second and the fourth. In this work, we propose the selection of a cooperative set of rules from a candidate fuzzy rule set together with the learning of rule weights coding all of them (rules and weights) in a chromosome. This pursues the following aims:

- To improve the linguistic model accuracy selecting the set of rules best cooperating while a local tuning of rules is performed to improve the interaction among them.
- To obtain simpler, and thus easily understandable, linguistic models by removing unnecessary rules.

3.2 An Example of Rule Selection Process

A simple example to select the subset of rules best cooperating is the selection process proposed in [17] —we will use this process in our experiments for comparison purposes—. Of course, we are assuming the previous existence of a set of rules.

It is based on a binary coded GA where each gene indicates whether a rule is considered or not to belong to the final fuzzy rule base (alleles '1' or '0', respectively). The stochastic universal sampling procedure [1] together with an elitist selection scheme and the two-point crossover together with the uniform mutation operators are used, and the Mean Squared Error (MSE) is considered as fitness function. The MSE for a whole rule base RB, calculated over the example set E, is defined as:

$$MSE\,(E, RB) = \frac{\sum_{e^l \in E} (ey^l - s(ex^l))^2}{2 \cdot |E|}$$

with $s(ex^l)$ being the output value obtained considering RB when the input variable values are $ex^l = (ex^l_1, \ldots, ex^l_n)$ and ey^l is the known desired value.

In this way, considering the m rules contained in the preliminary/candidate rule set, the chromosome $C = (c_1, \ldots, c_m)$ represents a subset of rules composing the final rule base, such that:

$$IF\ c_i = 1\ THEN\ (R_i \in RB)\ ELSE\ (R_i \notin RB)\ ,$$

with R_i being the corresponding i-th rule in the candidate rule set and RB being the final rule base.

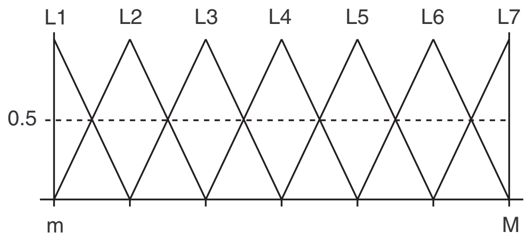


Fig. 1. Graphical representation of a possible fuzzy partition

4 Combining Rule Weight Derivation and Rule Selection

As discussed above, the hybridization of the rule weight derivation and the rule selection processes could result in important improvements of the system accuracy, obtaining simpler, and thus easily understandable, linguistic models by removing unnecessary rules. In this way, the interpretability is maintained to an acceptable level.

To generate linguistic models combining both approaches, we may follow an operation mode similar to the learning approach proposed in Section 2.3, by including the rule selection in the second step. Therefore, after performing the first step, where an initial set of numerous promising rules is generated, the two following tasks must be performed:

– Genetic selection of a subset of rules with good cooperation.
– Genetic derivation of the weights associated to these rules.

We will consider symmetrical fuzzy partitions of triangular-shaped membership functions (see Figure 1) to derive a candidate linguistic rule set (first step). With this aim, two different but similar approaches will be considered depending on the desired rule structure of the obtained models:

– *Based on the classical rule structure*. A preliminary fuzzy rule set based on linguistic rules with the usual structure is derived considering a specific generation process. In this work, the well-known *ad hoc data-driven generation method*[1] proposed by Wang and Mendel [32] is considered.
– *Based on the double-consequent rule structure*. Taking the first step of the Accurate Linguistic Modeling (ALM) methodology [10] and considering the generation process proposed by Wang and Mendel [32], the process involves dividing the input and output spaces into different fuzzy regions, generating the rule best covering each example, and finally selecting the two rules with the highest covering degree for each fuzzy input subspace (if there is more than a single rule on it). The double-consequent rule structure is presented in Appendix A. Notice that the preliminary rule base including

[1] A family of efficient and simple methods guided by covering criteria of the data in the example set

double-consequent rules will be preprocessed before giving it as input to the selection and rule weight derivation process. In this way, each double-consequent rule will be decomposed into two simple rules in order to allow the later process to assign rule weights to each consequent and to select the consequent/s best cooperating with the remaining rules. Thus, if one of the two simple rules obtained from decomposing a double-consequent rule is removed by the selection process, the corresponding fuzzy input subspace will have just a single consequent associated.

To select the subset of rules with the best cooperation and the weights associated to them (second step), we will consider a GA coding all of them (rules and weights) in a chromosome. The proposed algorithm is presented in the following section.

5 Genetic Weight Derivation and Rule Selection Process

The proposed GA must consider the use of binary (rule selection) and real values (weight derivation) in the same coding scheme. As we will see, a double coding scheme will be considered using integer and real coded genes, and therefore appropriate genetic operators for each part of the chromosome are considered.

In the following, the main characteristics of this genetic approach are presented.

5.1 Coding Scheme and Initial Gene Pool

A double coding scheme ($C = C_1 + C_2$) for both *rule selection* and *weight derivation* is used:

- For the C_1 part, the coding scheme generates binary-coded strings of length m (the number of single fuzzy rules in the previously derived rule set). Depending on whether a rule is selected or not, the alleles '1' or '0' will be respectively assigned to the corresponding gene. Thus, the corresponding part C_1^p for the p-th chromosome will be a binary vector representing the subset of rules finally obtained.
- For the C_2 part, the coding scheme generates real-coded strings of length m. The value of each gene indicates the weight used in the corresponding rule. They may take any value in the interval $[0, 1]$. Now, the corresponding part C_2^p for the p-th chromosome will be a real-valued vector representing the weights associated to the fuzzy rules considered.

Hence, a chromosome C^p is coded in the following way:

$$\begin{aligned}
&C_1^p = (c_{11}^p, \ldots, c_{1m}^p) \mid c_{1i}^p \in \{0, 1\}, \\
&C_2^p = (c_{21}^p, \ldots, c_{2m}^p) \mid c_{2i}^p \in [0, 1], \\
&C^p = C_1^p C_2^p .
\end{aligned}$$

The initial pool is obtained with an individual having all genes with value '1' in both parts, and the remaining individuals generated at random:

$$\forall k \in \{1, \ldots, m\},\ c^1_{1k} = 1 \text{ and } c^1_{2k} = 1.0\ .$$

5.2 Evaluating the Chromosome

To evaluate the p-th chromosome, we will follow an accuracy oriented policy by using the following fitness function, $F(C^p)$::

$$F(C^p) = MSE\,(E, RB(C^p)) = \frac{\sum_{e^l \in E}(ey^l - s(ex^l))^2}{2 \cdot |E|}.$$

with E being the set of training data and $s(ex^l)$ being the output value obtained from the rule base encoded in C^p when the input $ex^l = (ex^l_1, \ldots, ex^l_n)$ is presented, and ey^l being the known desired output. In this case, $s(ex^l)$ will be computed following the extended fuzzy reasoning process in order to consider the rule weight influence.

5.3 Genetic Components

The different components of the GA are introduced as follows:

Selection and Reproduction

The selection probability calculation follows linear ranking [1]. Chromosomes are sorted in order of raw fitness, and then the selection probability of each chromosome, $p_s(C^p)$, is computed according to its rank, $rank(C^p)$ —where $rank(C^{best}) = 1$—, by using the following non-increasing assignment function:

$$p_s(C^p) = \frac{1}{N_C} \cdot (\eta_{max} - (\eta_{max} - \eta_{min}) \cdot \frac{rank(C^p) - 1}{N_C - 1}),$$

where N_C is the number of chromosomes in the population and $\eta_{min} \in [0, 1]$ specifies the expected number of copies for the worst chromosome (the best one has $\eta_{max} = 2 - \eta_{min}$ expected copies). In the experiments developed in this paper, $\eta_{min} = 0.75$.

The classical **generational** [23] scheme has been used in this algorithm. In this way, linear ranking is performed along with *stochastic universal sampling* [1]. This procedure guarantees that the number of copies of any chromosome is bounded by the floor and by the ceiling of its expected number of copies. Together with the Baker's stochastic universal sampling procedure, an elitist mechanism (that ensures to maintain the best individual of the previous generation) has been considered.

Genetic Operators: Crossover and Mutation

Due to the different nature of the chromosomes involved in the rule base definition process, different operators working on each part, C_1 and C_2, are required. Taking into account this aspect, the following operators are considered.

The crossover operator will depend on the chromosome part where it is applied: in the C_1 part, the standard two-point crossover is used, whilst in the C_2 part, the max-min-arithmetical crossover [16] is considered.

The two-point crossover involves interchanging the fragments of the parents contained between two points selected at random (resulting two descendents). On the other hand, using the max-min-arithmetical crossover in the second parts, if $C_2^v = (c_{21}^v, \ldots, c_{2k}^v, \ldots, c_{2m}^v)$ and $C_2^w = (c_{21}^w, \ldots, c_{2k}^w, \ldots, c_{2m}^w)$ are going to be crossed, the resulting descendents are the two best of the next four offspring:

$$
\begin{aligned}
O_2^1 &= aC_2^w + (1-a)C_2^v,\\
O_2^2 &= aC_2^v + (1-a)C_2^w,\\
O_2^3 &\text{ with } c_{2k}^3 = \min\{c_{2k}^v, c_{2k}^w\},\\
O_2^4 &\text{ with } c_{2k}^4 = \max\{c_{2k}^v, c_{2k}^w\},
\end{aligned}
$$

with $a \in [0,1]$ being a constant parameter chosen by the GA designer. The max-min-arithmetical crossover was proposed to be used in real-coded spaces aiming to obtain a good balance between exploration and exploitation. This crossover operator obtains four well distributed descendents, one with the higher values of both parents, one with the lower values of both parents and two between the values of both parents (one nearest of the first parent and one nearest of the second parent). The two best are selected to replace the parents performing a good exploration/exploitation of the search space.

In this case, eight offspring are generated by combining the two from the C_1 part (two-point crossover) with the four ones from the C_2 part (max-min-arithmetical crossover). The two best offspring so obtained replace the two corresponding parents in the population.

As regards the mutation operator, it flips the gene value in the C_1 part and takes a value at random within the interval $[0,1]$ for the corresponding gene in the C_2 part.

Fig. 2 shows the application scope of these operators.

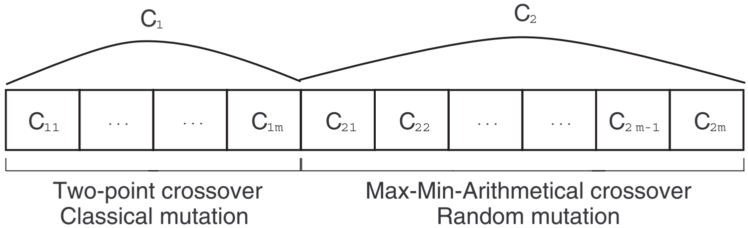


Fig. 2. Genetic representation and operators' application scope

Table 1. Methods considered for comparison

Ref.	Method	Description
[32]	**WM**	A well-known ad hoc data-driven method to obtain simple rules
[10]	**DC**	A method to obtain double-consequent rules (first step of ALM)
[10, 17]	**S**	Rule selection GA (second step of ALM or the WS C_1 part)
—	**W**	Weighted rule derivation GA (the WS C_2 part)
—	**WS**	The proposed GA performing weight derivation and rule selection

6 Experiments

In this section, we will analyze the performance of the linguistic fuzzy models generated from the proposed genetic weight derivation and rule selection process (see Section 5), when solving two different real-world problems [12]. The first presents *noise and strong nonlinearities* and the second presents four input variables, and therefore *a large search space*.

Two different approaches have been considered to obtain the initial set of candidate rules to be weighted and/or selected (see Section 4): the Wang and Mendel's method (WM) [32] and an extension of this method to obtain double-consequent fuzzy rules (DC) based on the ALM methodology [10]. In order to see the advantages of the combined action of the rule weight derivation and the rule selection, three different studies have been performed from both approaches: only considering rule selection (S), only considering rule weights (W) and considering both together, rule weights and rule selection (WS) —the algorithm proposed in this work—. Table 1 presents a short description of the methods considered for this study.

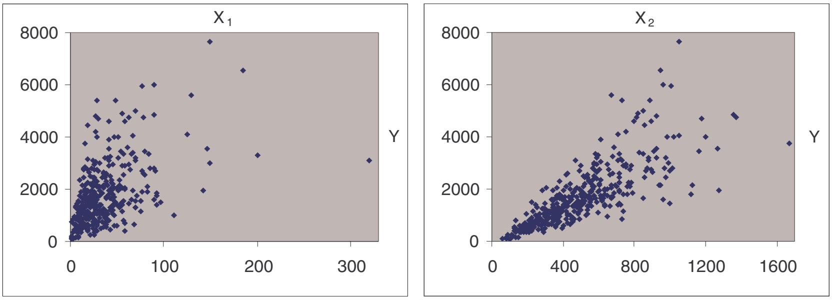
With respect to the fuzzy reasoning method used, we have selected the *minimum t-norm* playing the role of the implication and conjunctive operators, and the *center of gravity weighted by the matching* strategy acting as the defuzzification operator [11].

The values of the parameters used in all of these experiments are presented as follows[2]: 61 individuals, 1,000 generations, 0.6 as crossover probability, 0.2 as mutation probability per chromosome, and 0.35 for the a factor in the max-min-arithmetical crossover.

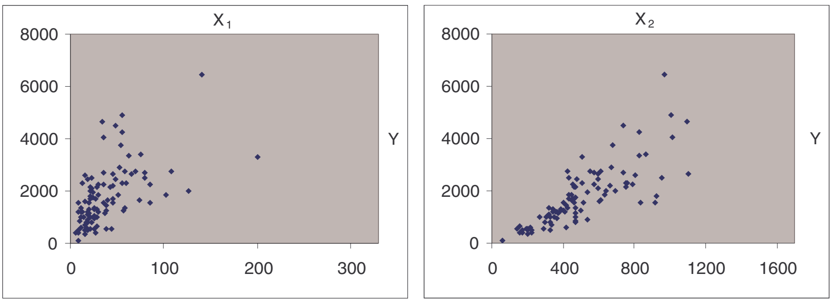
6.1 Estimating the Length of Low Voltage Lines

For an electric company, it may be of interest to measure the maintenance costs of its own electricity lines. These estimations could be useful to allow them to justify their expenses. However, in some cases these costs can not be directly calculated. The problem comes when trying to compute the maintenance costs of low voltage

[2] With these values we have tried to ease the comparisons selecting standard values for the common parameters that work well in most cases instead of searching very specific values for each specific method



(a) Training data



(b) Test data

Fig. 3. (a) (X_1,Y) and (X_2,Y) dependency in the training data; (b) (X_1,Y) and (X_2,Y) dependency in the test data

lines and it is due to the following reasons. Although maintenance costs depend on the total length of the electrical line, the length of low voltage lines would be very difficult and expensive to be measured since they are contained in little villages and rural nuclei. The installation of these kinds of lines is often very intricate and, in some cases, one company can serve to more than 10,000 rural nuclei.

Due to this reason, the length of low voltage lines can not be directly computed. Therefore, it must be estimated by means of indirect models. The problem involves relating *the length of low voltage line of a certain village* with the following two variables: *the radius of the village* and *the number of users in the village* [12]. We were provided with the measured line length, the number of inhabitants and the mean distance from the center of the town to the three furthest clients in a sample of 495 rural nuclei.

In order to evaluate the models obtained from the different methods considered in this paper, this sample has been randomly divided into two subsets, the training set with 396 elements and the test set with 99 elements, the 80% and the 20% respectively. The existing dependency of the two input variables with the output variable in the training and test data sets is shown in Fig. 3 (notice that they present strong non-linearities). Both data sets considered are

Table 2. Results obtained in the length of low voltage lines estimation problem

Method	#R	← (SC+DC)	MSE$_{tra}$	MSE$_{tst}$
WM	24	–	222,654	239,962
WM-S	**17**	–	214,177	265,179
WM-W	24	–	191,577	221,583
WM-WS	20	–	**191,565**	**219,370**
Considering double consequent rules				
DC	24	(14+10)	231,132	259,973
DC-S (ALM)	**17**	(14+ 3)	155,898	178,534
DC-W	24	(14+10)	144,983	191,053
DC-WS	18	(15+ 3)	**144,656**	**177,897**

SC = Single Consequent, DC = Double Consequent.

available at `http://decsai.ugr.es/∼casillas/fmlib/`. The linguistic partitions considered are comprised by seven linguistic terms with triangular-shaped fuzzy sets giving meaning to them (see Figure 1). The corresponding labels, $\{l_1, l_2, l_3, l_4, l_5, l_6, l_7\}$, stand for extremely small, very small, small, medium, large, very large, and extremely large, respectively.

The results obtained by the four methods analyzed are shown in Table 2, where $\#R$ stands for the number of rules, and MSE$_{tra}$ and MSE$_{tst}$ for the error obtained over the training and test data respectively. The best results are shown in boldface in each table. These results were obtained for a PENTIUM III with clock rate of 550 MHz and 128 MB of RAM. The run times for the different algorithms do not exceed 20 minutes.

Focusing on the WM approach, the model obtained from WM-WS presented the best performance, with improvements of a 14% in training and a 9% in test respect to the basic WM approach and, presenting a similar performance to the one obtained from WM-W. However, although accuracy and simplicity are contradictory requirements, four rules were eliminated respect to WM and WM-W (the second more accurate model), with this number representing the 17% of the candidate set of rules obtained from WM. The model obtaining the lowest number of rules was obtained from WM-S, but its performance was even worse than the original model obtained from WM.

Considering the DC approach, the proposed algorithm, DC-WS, obtains again the best performance, with improvements of about a 37% and a 31% in training and test, respectively respect to DC. In this case, there are significant differences between the results obtained by the approaches considering a single optimization (i.e., only rule selection, DC-S, or only rule weight derivation, DC-W) and our two-fold process. On the one hand, DC-S is only able to achieve a similar performance to DC-WS on the test error, but training is significantly worse. On the other hand, DC-W is only able to achieve a similar performance to DC-WS on training, but the test error is significantly worse. However, six rules were removed from the initial model obtained from DC considering WS. It represents a 25% of the total number of rules in DC. Furthermore, our model

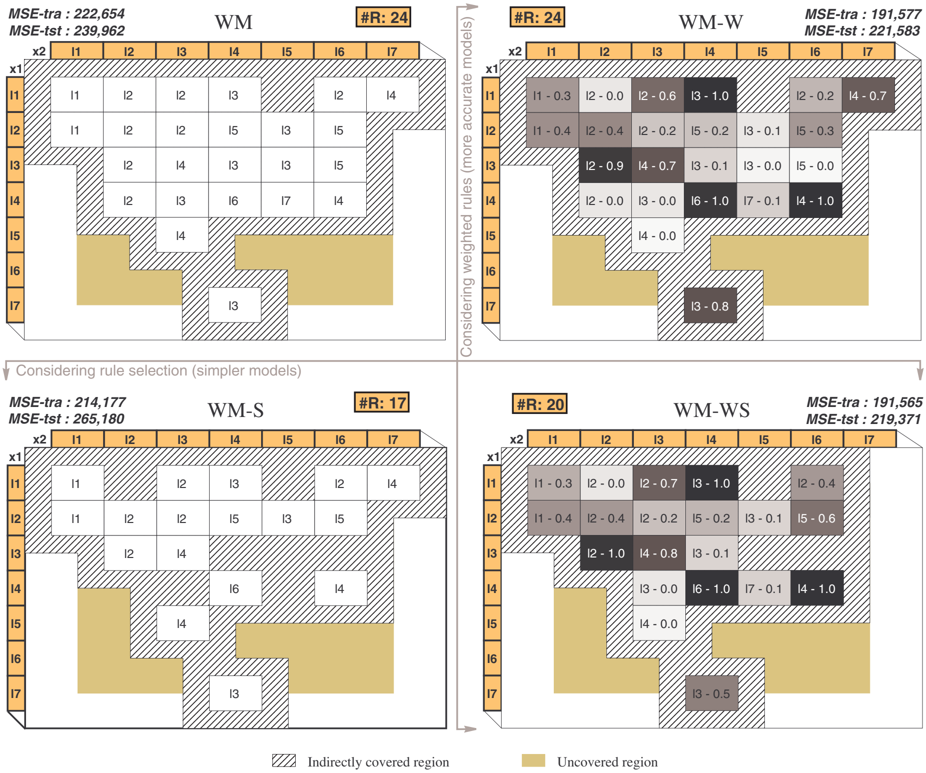


Fig. 4. Decision tables of the obtained models considering the WM approach

only presents 3 double-consequent rules respect to the 10 considered in DC and DC-W.

From the results presented in Table 2, we can say that the proposed technique is more robust than only considering weight derivation or rule selection in isolation. On the one hand, W only achieves good results by considering the WM approach. On the other hand, S only achieves good results by considering the DC approach.

The decision tables of the models obtained by the studied methods for the WM approach are presented in Figure 4. Each cell of the tables represents a fuzzy subspace/rule and contains its associated output consequent(s) —the primary and/or the secondary in importance when the DC approach is considered—, i.e., the corresponding label(s) together with its(their) respective rounded rule weight(s) when they are considered. These weights have been graphically by means of the grey colour scale, from black (1.0) to white (0.0). In this way, we can easily see the importance of a rule *with respect to their neighbors* which could help the system experts to identify important rules.

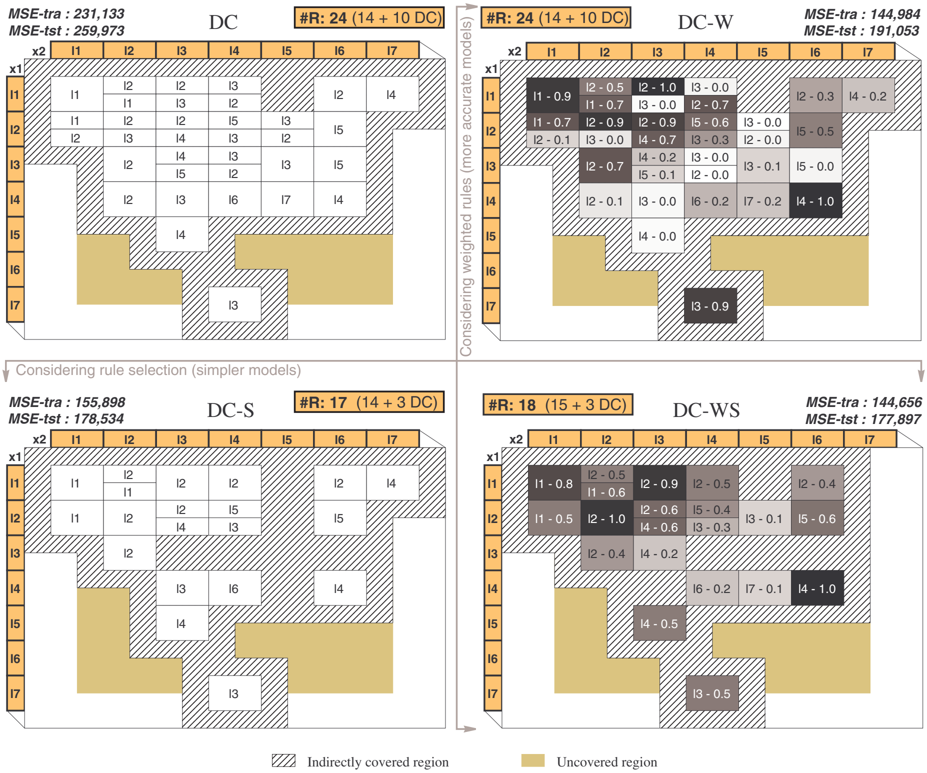


Fig. 5. Decision tables of the obtained models considering the DC approach

Notice that, the model obtained from WM-WS presents practically the same weights than that obtained from WM-W. Moreover, WM-WS and WM-S practically coincide in the selected rules to be considered in the final rule base. We can observe as some rules presenting weights very close to zero were removed in WM-WS respect to WM-W, those in the subspaces l_3-l_5, l_3-l_6 and l_4-l_2. However, as we said in Section 2.2, some rules were not removed since they were the one of their regions (that located in l_5-l_3) or since they improve the system performance interacting at low level (l_1-l_2 and l_4-l_3).

Figure 5 shows the decision tables of the models obtained by the studied methods when the DC approach is considered. Once again, there are similarities between DC-WS and DC-W (in terms of weights). However, in this case we can found some differences due to the large number of double-consequent rules considered in DC-W. On the other hand, strong similarities are found respect to the selected rules from DC-WS and DC-S. Taking into account this fact in both, WM and DC, we could say that WS inherits the accuracy characteristics of the rule weighting and the simplicity characteristics of the rule selection.

Table 3. Results obtained in the maintenance costs of medium voltage lines estimation problem

Method	#R	← (SC+DC)	$\mathbf{MSE}_{tra}$	$\mathbf{MSE}_{tst}$
WM	66	–	71,294	80,934
WM-S	**43**	–	57,025	59,942
WM-W	66	–	33,639	33,319
WM-WS	**43**	–	**32,476**	**32,638**
Considering double consequent rules				
DC	66	(49+17)	217,808	212,966
DC-S (ALM)	**47**	(44+ 3)	51,714	58,806
DC-W	66	(49+17)	26,377	28,637
DC-WS	51	(47+ 4)	**25,657**	**28,513**

SC = Single Consequent, DC = Double Consequent.

6.2 Estimating the Maintenance Costs of Medium Voltage Lines

Estimating the maintenance costs of the optimal installation of medium voltage electrical network in a town [12] is an interesting problem. Clearly, it is impossible to obtain this value by directly measuring it, since the medium voltage lines existing in a town have been installed incrementally, according to its own electrical needs in each moment. In this case, the consideration of models becomes the only possible solution. Moreover, the model must be able to explain how a specific value is computed for a certain town. These estimations allow electrical companies to justify their expenses. Our objective will be to relate the *maintenance costs of medium voltage line* with the following four variables: *sum of the lengths of all streets in the town*, *total area of the town*, *area that is occupied by buildings*, and *energy supply to the town*. We will deal with estimations of minimum maintenance costs based on a model of the optimal electrical network for a town in a sample of 1,059 towns.

To develop the different experiments in this contribution, the sample has been randomly divided into two subsets, the training and test ones, with an 80%-20% of the original size respectively. Thus, the training set contains 847 elements, whilst the test one is composed of 212 elements. These data sets used are available at `http://decsai.ugr.es/~casillas/fmlib/`. Five linguistic terms with triangular-shaped fuzzy sets giving meaning to them are considered for each variable (see Figure 1). In these case, the corresponding labels, $\{l_1, l_2, l_3, l_4, l_5\}$, stand for very small, small, medium, large, and very large, respectively.

The results obtained by the analyzed methods are shown in Table 3, where the same equivalences in Table 2 remain. Again, these results were obtained for a PENTIUM III with clock rate of 550 MHz and 128 MB of RAM. In this case the run times for the different methods do not exceed 65 minutes.

Considering the WM approach, the results obtained by WM-WS are the best in accuracy, with improvements of a 55% and a 60% in training and test, respect to the original simple linguistic model obtained by WM. Similar results were

WM-WS

X1	X2	X3	X4	Y	with
L1	L1	L1	L1	L1	[0.611]
L1	L1	L1	L2	L2	[0.529]
L1	L2	L2	L2	L2	[0.576]
L2	L1	L1	L1	L1	[0.331]
L2	L1	L2	L2	L2	[0.529]
L2	L2	L1	L1	L1	[0.280]
L2	L2	L2	L1	L2	[0.162]
L2	L2	L2	L2	L2	[0.456]
L2	L3	L2	L1	L2	[0.841]
L2	L3	L3	L2	L3	[0.945]
L3	L2	L1	L1	L1	[0.142]
L3	L2	L1	L2	L2	[0.914]
L3	L2	L1	L3	L2	[0.257]
L3	L2	L2	L3	L3	[0.308]
L3	L3	L2	L2	L2	[0.488]

X1	X2	X3	X4	Y	with
L3	L3	L2	L3	L3	[0.375]
L3	L3	L3	L2	L3	[0.859]
L3	L4	L3	L2	L3	[0.169]
L3	L4	L3	L3	L3	[0.753]
L3	L4	L4	L3	L4	[0.961]
L4	L2	L2	L1	L2	[0.208]
L4	L3	L2	L1	L2	[0.657]
L4	L3	L2	L3	L3	[0.418]
L4	L3	L2	L4	L3	[0.725]
L4	L3	L3	L3	L4	[0.294]
L4	L4	L3	L2	L3	[0.553]
L4	L4	L3	L4	L4	[0.628]
L4	L4	L4	L2	L4	[0.623]
L4	L4	L4	L3	L4	[0.342]
L4	L4	L4	L4	L5	[0.596]

X1	X2	X3	X4	Y	with
L4	L5	L4	L2	L3	[0.239]
L4	L5	L4	L3	L4	[0.583]
L4	L5	L4	L4	L5	[0.043]
L4	L5	L5	L2	L5	[0.289]
L4	L5	L5	L3	L5	[0.881]
L5	L2	L2	L2	L2	[0.248]
L5	L2	L2	L4	L3	[0.232]
L5	L2	L2	L5	L4	[0.739]
L5	L2	L3	L2	L3	[0.975]
L5	L2	L3	L5	L4	[0.899]
L5	L4	L3	L2	L3	[0.264]
L5	L4	L3	L4	L4	[0.029]
L5	L4	L3	L5	L5	[0.452]

#R: 43 *MSE-tra : 32,476* *MSE-tst : 32,638*

Considering double-consequent rules (more accurate model)

#R: 51 (47 + 4 DC) *MSE-tra : 25,658* *MSE-tst : 28,514*

DC-WS

X1	X2	X3	X4	Y	with
L1	L1	L1	L1	L1	[0.823]
L1	L1	L1	L2	L2	[0.672]
L1	L2	L1	L1	L1	[0.407]
L1	L2	L2	L1	L2	[0.476]
L1	L2	L2	L2	L2	[0.334]
L2	L1	L1	L1	L1	[0.501]
L2	L1	L2	L2	L2	[0.511]
L2	L2	L1	L1	L1	[0.470]
L2	L2	L1	L2	L2	[0.432]
L2	L2	L2	L2	L2,L3	[0.582,0.200]
L2	L3	L2	L1	L2	[0.860]
L2	L3	L2	L2	L2	[0.439]
L2	L3	L3	L2	L3	[0.870]
L3	L2	L1	L1	L1	[0.679]
L3	L2	L1	L2	L2	[0.895]
L3	L2	L1	L3	L2	[0.437]
L3	L2	L2	L1	L2	[0.337]

X1	X2	X3	X4	Y	with
L3	L2	L2	L3	L3	[0.332]
L3	L3	L2	L2	L2	[0.507]
L3	L3	L2	L3	L3	[0.549]
L3	L3	L3	L2	L3	[0.877]
L3	L3	L3	L3	L4	[0.518]
L3	L4	L3	L2	L3	[0.224]
L3	L4	L3	L3	L3	[0.663]
L3	L4	L4	L3	L4	[0.876]
L4	L2	L2	L1	L2	[0.466]
L4	L2	L2	L2	L2,L3	[0.298,0.261]
L4	L2	L2	L3	L3,L2	[0.246,0.171]
L4	L2	L2	L4	L3	[0.349]
L4	L3	L2	L1	L2	[0.628]
L4	L3	L2	L4	L3	[0.681]
L4	L3	L3	L4	L4	[0.316]
L4	L4	L3	L2	L3	[0.391]
L4	L4	L3	L3	L3	[0.273]

X1	X2	X3	X4	Y	with
L4	L4	L3	L4	L4	[0.400]
L4	L4	L4	L2	L4	[0.579]
L4	L4	L4	L3	L4	[0.537]
L4	L4	L4	L4	L5	[0.810]
L4	L5	L4	L2	L3	[0.309]
L4	L5	L4	L3	L4	[0.800]
L4	L5	L4	L4	L5	[0.070]
L4	L5	L5	L2	L5	[0.392]
L4	L5	L5	L3	L5	[0.879]
L5	L2	L2	L2	L2,L3	[0.290,0.183]
L5	L2	L2	L4	L3	[0.084]
L5	L2	L2	L5	L4	[0.878]
L5	L2	L3	L2	L3	[0.945]
L5	L2	L3	L5	L4	[0.721]
L5	L4	L3	L2	L3	[0.267]
L5	L4	L3	L4	L4	[0.063]
L5	L4	L3	L5	L5	[0.400]

Fig. 6. Rule set of the linguistic models obtained from the proposed technique when both the WM and the DC approaches are considered

obtained by only considering rule weights, WM-W. However, the proposed algorithm presents the simplest model (in terms of the number of rules) together with WM-S, removing 23 rules (a 35%) respect to WM and WM-W, and improving WM-S about a 50% in training and test, respectively.

Focusing on the DC approach, similar results were obtained respect to DC, DC-S and DC-W. Notice that, DC-WS does not only remove 15 rules more than DC-W but it also achieves a reduction of the number of double-consequent rules, obtaining only four rules of this type.

Figure 6 represents the rule set of the linguistic models obtained from the proposed technique. In this case, each row represents a fuzzy subspace/rule and contains its associated output consequent(s) —the primary and/or the secondary in importance when the DC approach is considered—, i.e., the corresponding label(s) together with its(their) respective rounded rule weight(s). Once again, the absolute importance weight for each fuzzy rule has been graphically shown by means of the grey colour scale, from black (1.0) to white (0.0).

From the 625 (5^4) possible fuzzy rules, the obtained linguistic fuzzy models are composed of only 43 and 51, respectively. In the case of DC-WS, it only contains four double-consequent rules. Notice that, all the double-consequent rules are very near in the four-dimensional space, representing a zone with a high complexity. Moreover, rules with weights close to 1 represent groups of important rules and do not usually appear alone. As in the previous problem, some similarities can be observed between the obtained models in terms of the derived weights and the selected rules, even considering different rule structures.

7 Concluding Remarks

In this work, the use of weighted linguistic fuzzy rules together with rule selection to obtain more simple and accurate linguistic fuzzy models has been proposed. To do so, a GA coding rules and weights in each chromosome has been developed with the main aim of improving the accuracy of simple linguistic fuzzy models and maintaining their interpretabiliy to an acceptable level (i.e., to obtain compact but powerful models).

In view of the obtained results, the proposed approach seems to inherit the accuracy characteristics of the rule weighting and the simplicity characteristics of the rule selection, obtaining simple but powerful linguistic fuzzy models. This is due to the following reasons:

- The ability of rule weights to indicate the interaction level of each rule with the remainder, improving the global performance of the weighted fuzzy model.
- The complementary characteristics that the use of weights and the rule selection approach present. The ability of rule selection to reduce the search space by only choosing the rules presenting a good cooperation is combined with an improvement of the rule cooperation capability by determining the appropriate interaction levels among the selected rules by the use of weights.

References

[1] Baker, J. E.: Reducing bias and inefficiency in the selection algorithm. In: Grefenstette, J. J. (Ed.), Proceedings of the 2nd International Conference on Genetic Algorithms, Lawrence Erlbaum Associates (Hillsdale, NJ, USA, 1987) 14–21 49, 52

[2] Bastian, A.: How to handle the flexibility of linguistic variables with applications. International Journal of Uncertainty, Fuzziness and Knowlegde-Based Systems **2:4** (1994) 463–484 44

[3] Casillas, J., Cordón, O., Herrera, F.: COR: A methodology to improve ad hoc data-driven linguistic rule learning methods by inducing cooperation among rules. IEEE Transactions on Systems, Man, and Cybernetics—Part B: Cybernetics **32:4** (2002) 526–537 45

[4] Casillas, J., Cordón, O., Herrera, F., Magdalena, L.: Accuracy improvements to find the balance interpretability-accuracy in fuzzy modeling: an overview. In: Casillas, J., Cordón, O., Herrera, F., Magdalena, L. (Eds.), Fuzzy modeling and the interpretability-accuracy trade-off. Part II, accuracy improvements preserving the interpretability (Physica-Verlag, Heidelberg, Germany, 2002) 3–24 44

[5] Chin, T. C., Qi, X. M.: Genetic algorithms for learning the rule base of fuzzy logic controller. Fuzzy Sets and Systems **97:1** (1998) 1–7 49
[6] Chiu, S.: Fuzzy model identification based on cluster estimation. Journal of Intelligent and Fuzzy Systems **2** (1994) 267–278 45, 48
[7] Cho, J. S., Park, D. J.: Novel fuzzy logic control based on weighting of partially inconsistent rules using neural network. Journal of Intelligent Fuzzy Systems **8** (2000) 99–110 45, 46
[8] Combs, W. E., Andrews, J. E.: Combinatorial rule explosion eliminated by a fuzzy rule configuration. IEEE Transactions on Fuzzy Systems **6:1** (1998) 1–11 48
[9] Cordón, O., Herrera, F.: A three-stage evolutionary process for learning descriptive and approximative fuzzy logic controller knowledge bases from examples. International Journal of Approximate Reasoning **17:4** (1997) 369–407 48
[10] Cordón, O., Herrera, F.: A proposal for improving the accuracy of linguistic modeling. IEEE Transactions on Fuzzy Systems **8:4** (2000) 335–344 45, 46, 48, 50, 54, 63, 64
[11] Cordón, O., Herrera, F., Peregrín, A.: Applicability of the fuzzy operators in the design of fuzzy logic controllers. Fuzzy Sets and Systems **86:1** (1997) 15–41 47, 54, 63
[12] Cordón, O., Herrera, F., Sánchez, L.: Solving electrical distribution problems using hybrid evolutionary data analysis techniques. Applied Intelligence **10** (1999) 5–24 54, 55, 59
[13] Cordón, O., del Jesús, M. J., Herrera, F.: Genetic learning of fuzzy rule-based classification systems cooperating with fuzzy reasoning methods. International Journal of Intelligent Systems **13:10-11** (1998) 1025–1053 48
[14] Gómez-Skarmeta, A. F., Jiménez, F.: Fuzzy modeling with hybrid systems. Fuzzy Sets and Systems **104** (1999) 199–208 49
[15] Halgamuge, S., Glesner, M.: Neural networks in designing fuzzy systems for real world applications. Fuzzy Sets and Systems **65:1** (1994) 1–12 45, 48
[16] Herrera, F., Lozano, M., Verdegay, J. L.: Tuning fuzzy logic controllers by genetic algorithms. International Journal of Approximate Reasoning **12** (1995) 299–315 48, 53
[17] Herrera, F., Lozano, M., Verdegay, J. L.: A learning process for fuzzy control rules using genetic algorithms. Fuzzy Sets and Systems **100** (1998) 143–158 45, 48, 49, 54
[18] Holland, J. H.: Adaptation in natural and artificial systems. Ann arbor: The University of Michigan Press (1975), The MIT Press, London (1992) 45, 47
[19] Ishibuchi, H., Nozaki, K., Yamamoto, N., Tanaka, H.: Selecting fuzzy if-then rules for classification problems using genetic algorithms. IEEE Transactions on Fuzzy Systems **9:3** (1995) 260–270 45, 48
[20] Ishibuchi, H., Murata, T., Türksen, I. B.: Single-objective and two-objective genetic algorithms for selecting linguistic rules for pattern classification problems. Fuzzy Sets and Systems **89** (1997) 135–150 45, 48
[21] Ishibuchi, H., Takashima, T.: Effect of rule weights in fuzzy rule-based classification systems. IEEE Transactions on Fuzzy Systems **3:3** (2001) 260–270 45, 46
[22] Krone, A., Krause, H., Slawinski, T.: A new rule reduction method for finding interpretable and small rule bases in high dimensional search spaces. Proceedings of the 9th IEEE International Conference on Fuzzy Systems (San Antonio, TX, USA, 2000) 693–699 45, 48
[23] Michalewicz, Z.: Genetic algorithms + data structures = evolution programs. Springer-Verlag (1996) 45, 47, 52

[24] Nozaki, K., Ishibuchi, H., Tanaka, H.: A simple but powerful heuristic method for generating fuzzy rules from numerical data. Fuzzy Sets and Systems 86 (1997) 251–270 45, 63, 64
[25] Pal, N. R., Pal, K.: Handling of inconsistent rules with an extended model of fuzzy reasoning. Journal of Intelligent Fuzzy Systems **7** (1999) 55–73 45, 46, 47
[26] Pardalos, P. M., Resende, M. G. C.: Handbook of applied optimization. Oxford University Press, NY (2002) 45
[27] Roubos, H., Setnes, M.: Compact fuzzy models through complexity reduction and evolutionary optimization. Proceedings of the 9th IEEE International Conference on Fuzzy Systems **2** (San Antonio, Texas, USA, 2000) 762–767 49
[28] Rovatti, R., Guerrieri, R., Baccarani, G.: Fuzzy rules optimization and logic synthesis. Proceedings of the 2nd IEEE International Conference on Fuzzy Systems **2** (San Francisco, USA, 1993) 1247–1252 45, 48
[29] Setnes, M., Babuska, R., Kaymak, U., van Nauta-Lemke, H. R.: Similarity measures in fuzzy rule base simplification. IEEE Transactions on Systems, Man, and Cybernetics—Part B: Cybernetics **28** (1998) 376–386 45, 48
[30] Setnes, M., Hellendoorn, H.: Orthogonal transforms for ordering and reduction of fuzzy rules. Proceedings of the 9th IEEE International Conference on Fuzzy Systems **2** (San Antonio, Texas, USA, 2000) 700–705 45, 48
[31] Thrift, P.: Fuzzy logic synthesis with genetic algorithms. Proceedings of the 4th International Conference on Genetic Algorithms (ICGA'91) (Morgan Kaufmann, San Mateo, CA, 1991) 509–513 45
[32] Wang, L. X., Mendel, J. M.: Generating fuzzy rules by learning from examples. IEEE Transactions on Systems, Man, and Cybernetics **22** (1992) 1414–1427 46, 48, 50, 54
[33] Yam, Y., Baranyi, P., Yang, C. T.: Reduction of fuzzy rule base via singular value decomposition. IEEE Transactions on Fuzzy Systems **7** (1999) 120–132
[34] Yen, J., Wang, L.: Simplifying fuzzy rule-based models using orthogonal transformation methods. IEEE Transactions on Systems, Man, and Cybernetics—Part B: Cybernetics **29** (1999) 13–24 45, 48
[35] Yu, W., Bien, Z.: Design of fuzzy logic controller with inconsistent rule base. Journal of Intelligent Fuzzy Systems **2** (1994) 147–159 45, 46

A Double-Consequent Rule Structure

More flexible linguistic models may be obtained by allowing them to present fuzzy rules where each combination of antecedents may have two consequents associated [10, 24]:

$$\textbf{IF } X_1 \text{ is } A_1 \text{ and } \ldots \text{ and } X_n \text{ is } A_n \textbf{ THEN } Y \text{ is } \{B_1, B_2\},$$

with X_i (Y) being the linguistic input (output) variables, A_i being the linguistic label used in the i-th input variable, and B_1 and B_2 the two linguistic terms associated to the output variable.

Since each double-consequent fuzzy rule can be decomposed into two different rules with a single consequent, the usual plain fuzzy inference system can be applied. The only restriction imposed is that the defuzzification method must consider the matching degree of the rules fired, for example, the *center of gravity weighted by the matching degree* defuzzification strategy [11] may be used.

The consideration of this structure to generate advanced linguistic models was initially proposed in [24]. Another approach, according to the ALM methodology, was also introduced in [10].