Available at
www.ComputerScienceWeb.com
POWERED BY SCIENCE @ DIRECT®

ELSEVIER International Journal of Approximate Reasoning 32 (2003) 187–215

INTERNATIONAL JOURNAL OF APPROXIMATE REASONING

www.elsevier.com/locate/ijar

Linguistic modeling with hierarchical systems of weighted linguistic rules

Rafael Alcalá [a], Jose Ramón Cano [b], Oscar Cordón [c,*], Francisco Herrera [c], Pedro Villar [d], Igor Zwir [e]

[a] *Department of Computer Science, University of Jaén, 23071 Jaén, Spain*
[b] *Department of Electronic Engineering, Computer Systems and Automatics, University of Huelva, 21071 Huelva, Spain*
[c] *Department of Computer Science and A.I., University of Granada, 18071 Granada, Spain*
[d] *Department of Computer Science, University of Vigo, 32004 Ourense, Spain*
[e] *Department of Computer Science, University of Buenos Aires, 1428 Buenos Aires, Argentina*

Received 1 January 2002; accepted 1 April 2002

Abstract

Recently, many different possibilities to extend the Linguistic Fuzzy Modeling have been considered in the specialized literature with the aim of introducing a trade-off between accuracy and interpretability. These approaches are not isolated and can be combined among them when they have complementary characteristics, such as the hierarchical linguistic rule learning and the weighted linguistic rule learning. In this paper, we propose the hybridization of both techniques to derive Hierarchical Systems of Weighted Linguistic Rules. To do so, an evolutionary optimization process jointly performing a rule selection and the rule weight derivation has been developed. The proposal has been tested with two real-world problems achieving good results.

Keywords: Linguistic Fuzzy Modeling; Hierarchical fuzzy systems; Weighted linguistic rules; Genetic algorithms

* Corresponding author. Tel.: +34-958-246-143.
E-mail addresses: alcala@ujaen.es (R. Alcalá), jose.cano@diesia.uhu.es (J. Ramón Cano), ocordon@decsai.ugr.es (O. Cordón), herrera@decsai.ugr.es (F. Herrera), pvillar@uvigo.es (P. Villar), zwir@dc.uba.ar (I. Zwir).

0888-613X/02/$ - see front matter
PII: S0888-613X(02)00083-X

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]. A possible way to improve the linguistic fuzzy model performance without losing interpretability to a high degree is to extend its usual structure making it more flexible.

Many different possibilities to improve the Linguistic Fuzzy Modeling have been considered in the specialized literature. 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 between the rules in the Knowledge Base. When these approaches have complementary characteristics, they can be combined improving even more the system performance. This is the case of the weighted [4,18,26,33] and the hierarchical [10] linguistic rule learning approaches, when the two-level Hierarchical System of Linguistic Rules Learning Methodology (HSLR-LM) is considered.

The hierarchical methodology was proposed as a first strategy to improve *simple linguistic fuzzy models*, [1] preserving their structure and descriptive power, and only reinforcing the modeling of those problem subspaces with more difficulties by a hierarchical treatment of the rules generated in these zones. Therefore, this approach was devoted to produce a more general and well defined structure, the Hierarchical Knowledge Base (HKB) [10]. To obtain the initial simple linguistic fuzzy models, the authors considered two simple inductive Linguistic Rule Generation methods (LRG-method), the Wang and Mendel's algorithm [30] and the Thrift's algorithm [29], but any other method can be considered within the proposed approach.

On the other hand, since the said HKB is generated from an operation mode guided by local error measures, repeated and/or multiple consequent rules could be obtained. The use of rule weights as a local tuning of linguistic rules enables the linguistic fuzzy models to cope with these kinds of rules in a better way and thereby enhances the robustness, flexibility and system modeling capability [26]. In this way, the ability of this technique to indicate the interaction level of each rule with the remaining ones is considered.

In this work, we propose the hybridization of the hierarchical scheme with the use of rule weights by extending the two-level HSLR-LM proposed in [10]. The resulting Hierarchical System of Weighted Linguistic Rules (HSWLR), presents a model structure which is extended by permitting the use of weighted hierarchical linguistic rules. Besides, the summarization component – which has the aim of selecting the subset of rules best cooperating among the rules gen-

[1] Generated from any other learning method or from expert knowledge.

erated to obtain the final HKB – is modified by allowing it to jointly perform the rule selection and the rule weight derivation. A Genetic Algorithm (GA) [22] performing the rule selection together with the derivation of rule weights has been developed for this task. Hence, this extended methodology is intended as a meta-method over any other LRG-method, developed to improve simple linguistic fuzzy models by only reinforcing the modeling of those problem subspaces with more difficulties while the use of rule weights improves the way in which they interact.

This paper is organized as follows. In the next section, different possibilities to improve the Linguistic Fuzzy Modeling are briefly presented. In Section 3, the two-level HSLR-LM will be reviewed. This methodology will be extended in Section 4 in order to obtain HSWLRs. Section 5 presents the GA 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, a table with the used acronyms is presented in Appendix A.

2. Preliminaries: improving the linguistic fuzzy models

Linguistic Fuzzy Modeling has certain inflexibility due to the use of a global semantic that gives a general meaning to the used fuzzy sets [2]. However, it is possible to make some considerations to face this drawback. Many different possibilities to improve the Linguistic Fuzzy Modeling have been considered in the specialized literature. 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 Knowledge Base. This rule cooperation may be induced acting on three different model components:

- *Approaches acting on the Data Base* (*DB*):
  - Linguistic fuzzy partition granularity learning [8,11]: This approach is devoted to determine the optimal number of linguistic terms used in the variable fuzzy partitions, i.e., the granularity.
  - Linguistic fuzzy partition membership function learning [9,17,27]: It involves extracting the DB by induction from the available data set. This process is usually performed by non-supervised clustering techniques, or using GAs.
  - Membership function tuning [13,14]: This approach, usually called DB tuning, involves refining the membership function shapes from a previous definition once the remaining components have been obtained.
  - Linguistic modifier learning [12,20]: A linguistic modifier is an operator that alters the membership functions of the fuzzy sets associated to the linguistic labels, giving a more or less precise definition as a result depending on the case.

- *Approaches acting on the Rule Base* (*RB*):
  - Rule selection [16,19]: It involves obtaining an optimized subset of rules from a previous RB by selecting some of them.
  - Multiple rule consequent learning [6,25]: This approach allows the RB to present rules where each combination of antecedents may have two or more consequents associated when it is necessary.
  - Weighted linguistic rule learning [4,18,26,33]: This approach considers 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 [5]: This approach follows the primary objective of inducing a better cooperation among the linguistic rules. To do so, the RB design is made using global criteria that jointly consider the action of the different rules.
- *Approaches acting on the whole Knowledge Base*:
  - Knowledge Base derivation [21,28]: In this case, the process of designing the DB is jointly developed with the derivation of the RB in a simultaneous procedure.
  - Hierarchical linguistic rule learning [10,15]: This approach is devoted to produce a more general and well defined structure, the HKB. In this way, to improve the system accuracy, fuzzy rules consider linguistic terms that are defined in linguistic fuzzy partitions with different granularity levels.

In this work, we will consider the use of rule weights together with a hierarchical approach. In the following sections, these two approaches to relax the model structure will be analyzed.

3. Hierarchical systems of linguistic rules

This approach, proposed in [10], is based on the HKB. This structure, composed of a Hierarchical Data Base (HDB) and a Hierarchical Rule Base (HRB), should be flexible enough to allow a wide variety of linguistic fuzzy models, from very accurate to properly interpretable ones.

In this case, the main purpose of HSLRs is to preserve the descriptive abilities of the usual linguistic fuzzy models, increasing their accuracy by the use of different hierarchical levels. All of this is done by simplifying the inference mechanism adopted by other previous hierarchical approaches [15,31], independently activating each rule as it is done in the conventional fuzzy inference mechanism.

The description of the HKB and the two-level HSLR-LM process [10] to generate two-layer HKBs will be introduced in the following sections. The methodology will be extended in Section 4 in order to obtain HSWLRs (such extension will allow the rule selection process to simultaneously address the rule selection and the weight derivation).

3.1. Hierarchical Knowledge Base

The HKB [10] is composed of a set of layers, and each layer is defined by its components in the following way:

$$\text{layer}(t, n(t)) = DB(t, n(t)) + RB(t, n(t)),$$

with $n(t)$ being the number of linguistic terms in the fuzzy partitions of layer t, $DB(t, n(t))$ being the DB which contains the linguistic partitions with granularity level $n(t)$ of layer t (t-linguistic partitions), and $RB(t, n(t))$ being the RB formed by those linguistic rules whose linguistic variables take values in $DB(t, n(t))$ (t-linguistic rules). For the sake of simplicity in the descriptions, the following notation equivalences are established:

$$DB(t, n(t)) \equiv DB^t \quad \text{and} \quad RB(t, n(t)) \equiv RB^t.$$

At this point, we should note that in this work, we are using *strong fuzzy partitions* (those in which the sum of the membership degrees within the variable domain is kept to 1.0) with the same number of linguistic terms for all input-output variables, composed of symmetrical triangular-shaped membership functions (see Fig. 1). The number of linguistic terms in the t-linguistic partitions is defined in the following way:

$$n(t) = (n(1) - 1) \cdot 2^{t-1} + 1,$$

with $n(1)$ being the granularity of the initial fuzzy partitions.

Fig. 1 graphically depicts the way in which a linguistic partition in DB^1 becomes a linguistic partition in DB^2. Each term of order k from $DB(t, n(t))$, $S_k^{n(t)}$ ($S_k^{n(1)}$ in the figure) is mapped into the fuzzy set $S_{2k-1}^{2 \cdot n(t)-1}$, preserving the

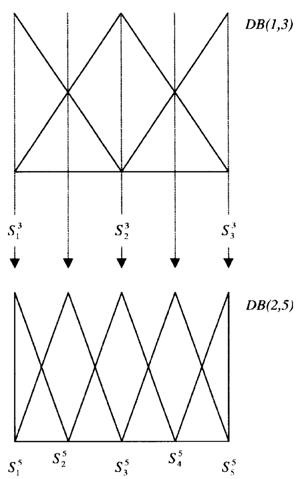


Fig. 1. Two layers of linguistic partitions which compose the HDB.

Table 1
Mapping between terms from successive DBs

$DB(t,n(t))$		$DB(t+1, 2\cdot n(t)-1)$
$S_{k-1}^{n(t)}$	$\rightarrow$	$S_{2k-3}^{2\cdot n(t)-1}$
		$S_{2k-2}^{2\cdot n(t)-1}$
$S_{k}^{n(t)}$	$\rightarrow$	$S_{2k-1}^{2\cdot n(t)-1}$
		$S_{2k}^{2\cdot n(t)-1}$
$S_{k+1}^{n(t)}$	$\rightarrow$	$S_{2k+1}^{2\cdot n(t)-1}$

former modal points, and a set of *n(t)-1* new terms is created, each one between $S_k^{n(t)}$ and $S_{k+1}^{n(t)}$ $(k = 1,\ldots,n(t)-1)$ (see Table 1).

The main purpose of developing an HRB is to model the problem space in a more accurate way. To do so, those linguistic rules from $RB(t,n(t))$ – RB^t – that model a subspace with bad performance are expanded into a set of more specific linguistic rules, which become their image in $RB(t+1, 2\cdot n(t)-1)$, RB^{t+1}. This set of rules models the same subspace that the former one and replaces it. As a consequence of the previous definitions, we could now define the HKB as the union of every layer t:

$$HKB = \bigcup_t \text{layer}(t,n(t)).$$

The proposed HKB was initially designed to improve simple linguistic fuzzy models, preserving their structure and descriptive power, and reinforcing only the modeling of those problem subspaces with more difficulties by a hierarchical treatment of the rules generated in these zones. To do so, in this contribution, we will just consider a two-layer HKB which allows us to produce a refinement of simple linguistic fuzzy models by introducing small changes to increase their accuracy.

3.2. A two-level HSLR learning methodology

In this section, we present the two-level HSLR-LM to generate two-layer HKBs [10]. To do so, we use an existing inductive LRG-method based on the existence of a set of input–output training data $E = \{e^1,\ldots,e^l,\ldots,e^q\}$ with $e^l = (ex_1^l,\ldots,ex_n^l,ey^l)$, and a previously defined DB^1. In this work, we consider the Wang and Mendel's algorithm [30] as LRG-method to obtain simple linguistic fuzzy models, *although any other technique could be used* (some experiments with the Thrift's algorithm [29] were also included in [10]). Two measures of error are used in the algorithm:

1. Global measure (used to evaluate the complete RB): The Mean Square Error (MSE) for a whole RB, calculated over E, is defined as

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

Table 2
HSLR learning methodology

HIERARCHICAL KNOWLEDGE BASE GENERATION PROCESS
Step 0. $RB(1, n(1))$ *Generation Process*
Step 1. $RB(2, 2 \cdot n(1) - 1)$ *Generation Process*
Step 2. *Summarization Process*
HIERARCHICAL RULE BASE GENETIC SELECTION PROCESS
Step 3. *HRB Selection Process*

with $s(ex^l)$ being the output value obtained from the HLSR using the current RB when the input variable values are $ex^l = (ex_1^l, \ldots, ex_n^l)$, and ey^l is the known desired value.

2. Local measure (used to determine if an individual rule is expanded): The MSE for a simple rule, [2] $R_i^{n(1)}$, calculated over E_i, is showed as follows:

$$MSE(E_i, R_i^{n(1)}) = \frac{\sum_{e^l \in E_i} (ey^l - s_i(ex^l))^2}{2 \cdot |E_i|}$$

with E_i being a set of the examples matching the ith rule antecedents to degree $\tau \in (0, 1]$ and $s_i(ex^l)$ being the output value from this rule.

The algorithm basically consists of the following steps, which are listed in Table 2:

Step 0. RB^1 *Generation*. Generate the rules from DB^1 by means of an existing LRG-method: $RB^1 = LRG\text{-}method(DB^1, E)$.

Step 1. RB^2 *Generation*. Generate RB^2 from RB^1, DB^1 and DB^2.

(a) *Calculate the error of* RB^1: $MSE(E, RB^1)$.

(b) *Calculate the error of each 1-linguistic rule*: $MSE(E_i, R_i^{n(1)})$.

(c) *Select the 1-linguistic rules with bad performance which will be expanded (the expansion factor* α *may be adapted in order to have more or less expanded rules):*

If $MSE(E_i, R_i^{n(1)}) \geqslant \alpha \cdot MSE(E, RB^1)$ *Then* $R_i^{n(1)} \in RB^1_{bad}$

Else $R_i^{n(1)} \in RB^1_{good}$.

(d) Create [3] DB^2: $DB^2_{x_j}$ and DB^2_y.

(e) *For each bad performance 1-linguistic rule to be expanded:*

(i) *Select the 2-linguistic partition terms from* DB^2 *that* δ*-intersect the ones of the bad performance 1-linguistic rules:* $I(R_i^{n(1)})\ \forall R_i^{n(1)} \in RB^1_{bad}$, where $\delta \in [0, 1]$ is a cross level of "*significant intersection*".

(ii) *Extract a candidate set of L 2-linguistic rules:*

$$CLR(R_i^{n(1)}) = LRG - method(I(R_i^{n(1)}), E_i) = \{R_{i_1}^{2 \cdot n(1)-1}, \ldots, R_{i_L}^{2 \cdot n(1)-1}\}.$$

[2] Notice that other local error measures, such as the one showed in [32] could also be considered.

[3] DB^t is referred to as $DB^t_{x_j}$ $(j = 1, \ldots, n)$, meaning that it contains the t-linguistic partition where the input variable x_j takes values, and as DB^t_y for the output variable y.

Step 2. *Summarization. Obtain a Joined set of Candidate Linguistic Rules (JCLR),* performing the union of the group of the new generated *2-linguistic rules* and the former good performance *1-linguistic rules:*

$$JCLR = RB^1_{\text{good}} \cup \left(\bigcup_i CLR(R_i^{n(1)}) \right), \quad R_i^{n(1)} \in RB^1_{\text{bad}}.$$

In the following, we show an example of the whole expansion process. Let us consider $n(1) = 3$ and the following linguistic partitions:

$$DB_{x_1}(1,3) = DB_{x_2}(1,3) = DB_y(1,3) = \{S^3, M^3, L^3\},$$
$$DB_{x_1}(2,5) = DB_{x_2}(2,5) = DB_y(2,5) = \{VS^5, S^5, M^5, L^5, VL^5\},$$

where S stands for Small, M for Medium, L for Large, and V for Very. Let us consider the following bad performance *1-linguistic rule* to be expanded (see Fig. 2):

$$R_i^3 : IF \ x_1 \ is \ S_{i1}^3 \ and \ x_2 \ is \ S_{i2}^3 \quad THEN \ y \ is \ B_i^3,$$

where the linguistic terms are: $S_{i1}^3 = S^3, S_{i2}^3 = S^3, B_i^3 = S^3$, and the resulting sets I with $\delta = 0.5$ are

$$I(S_{i1}^3) = \{VS^5, S^5\}, \quad I(S_{i2}^3) = \{VS^5, S^5\}, \quad I(B_i^3) = F(\cdot) \subseteq D_y(2,5),$$
$$I(R_i^3) = I(S_{i1}^3) \times I(S_{i2}^3) \times I(B_i^3).$$

Therefore, it is possible to obtain at most four *2-linguistic rules* generated by the LRG-method from the expanded R_i^3:

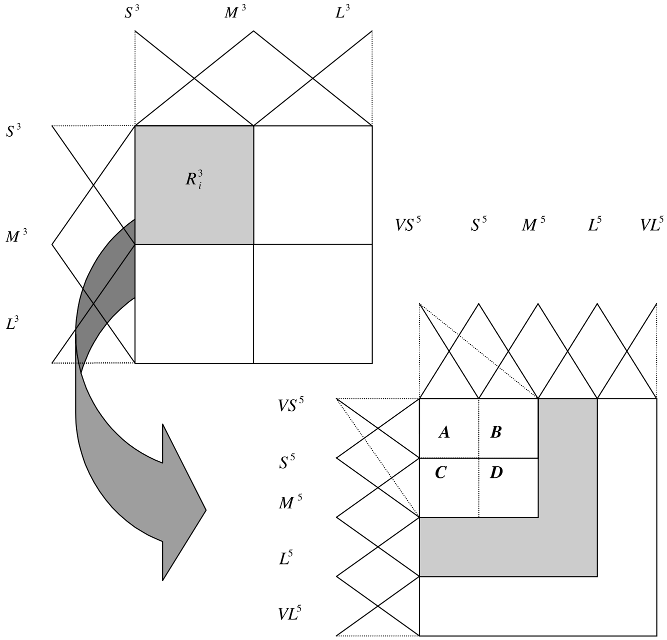
$$LRG(I(R_i^3), E_i) = \{R_{i1}^5, R_{i2}^5, R_{i3}^5, R_{i4}^5\}.$$

This example is graphically showed in Fig. 2. In the same way, other bad performance neighbor rules could be expanded simultaneously.

Step 3. *HRB selection. Simplify the set JCLR by removing the unnecessary rules from it and generating an HRB with good cooperation.* In the *JCLR* – where rules of different hierarchical layers coexist – it may happen that a complete set of *2-linguistic rules* which replaces an expanded *1-linguistic rule* does not produce good results. However, a subset of this set of *2-linguistic rules* may work properly. A genetic process is considered to put this task into effect, but any other optimization technique could be considered:

$$HRB = Selection\ Process(JCLR).$$

It is based on a binary coded GA where each gene indicates whether a rule is selected or not (alleles '1' or '0', respectively). The stochastic universal sampling procedure [3] together with an elitist selection scheme (considering the MSE as fitness) and the two-point crossover together with the uniform mutation operators are used. In this way, considering the m rules contained in *JCLR*, $C = (c_1, \ldots, c_m)$ represents a subset of rules for the HRB, such that:



$R_i^3 = IF\ x_1\ is\ S^3\ AND\ x_2\ is\ S^3\ THEN\ y\ is\ S^3$

$R_{i1}^5 = IF\ x_1\ is\ VS^5\ AND\ x_2\ is\ VS^5\ THEN\ y\ is\ F(A)$

$R_{i2}^5 = IF\ x_1\ is\ VS^5\ AND\ x_2\ is\ S^5\ THEN\ y\ is\ F(A \cup B)$

$R_{i3}^5 = IF\ x_1\ is\ S^5\ AND\ x_2\ is\ VS^5\ THEN\ y\ is\ F(A \cup C)$

$R_{i4}^5 = IF\ x_1\ is\ S^5\ AND\ x_2\ is\ S^5\ THEN\ y\ is\ F(A \cup B \cup C \cup D)$

$F(A), F(A \cup B), F(A \cup C),$
$F(A \cup B \cup C \cup D) \in D_y(2,5)$

Fig. 2. Example of the HRB Generation Process.

$$IF\ \ c_i = 1 \quad THEN\ \ (R_i \in HRB) \quad ELSE\ \ (R_i \notin HRB).$$

After applying this algorithm, the HKB is obtained as

$$HKB = HDB + HRB.$$

Remark 1. As a consequence of the previous DB^2 generation policy, which is based on selecting those terms in DB^2 which significantly intersect the ones of the bad rule, at least two different kinds of special linguistic rules can appear from the HKB Generation Process. On the one hand, *repeated* 2-*linguistic rules* can be generated as a consequence of the expansion of adjacent bad *t*-*linguistic rules*. On the other hand, *double*-*consequent* 2-*linguistic rules* (those with the same antecedent and different consequents) can be derived because of the same reason. Both of them can be intended as a local reinforcement of the problem space zones with a high complexity.

In the following section, we will be back on these kinds of rules showing the way in which they can be interpreted and how the use of rule weights can – positively – affect to them (see Sections 4.2.1 and 4.2.2).

3.3. Trade-off between accuracy and complexity

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 system with a minor number of rules requires a minor effort to be interpreted. In the following, we introduce a modification of the fitness function of the GA (see Step 3 of the algorithm) as a trade-off Accuracy-Complexity-oriented policy (AC-oriented selection policy) of the hierarchical model [16] in order to add another interestingness relation to enrich the modeling process.

Let us consider the following function $F'(C^j)$ which penalizes those RBs with a high number of rules in the following way:

$$F'(C^j) = w_1 \cdot F(C^j) + w_2 \cdot N^j_{\text{rules}}$$

with $F(C^j)$ being the former fitness function based on the error produced by the current HRB encoded in the chromosome C^j, N^j_{rules} being the number of rules of that HRB, and with w_1 and w_2 being weights defining the relative importance of each objective. In the present experiments, these coefficients are initialized as follows [9]:

$$w_1 = 1.0, \qquad w_2 = 0.1 \cdot \frac{MSE_{\text{initial}}}{N_{\text{initial rules}}}$$

with MSE_{initial} and $N_{\text{initial rules}}$ respectively being the error and the amount of rules of the original HRB to be summarized.

This policy is also viewed as a kind of post-pruning [23] which, in the methodology context, does not only consider the quality of the approximation performed by each rule but also the global cooperation among them for selecting rules in order to increase the generalization power of the system modeled.

4. Introducing weights in the HSLR-LM

It is known that the use of rule weights as a local tuning of linguistic rules enables the linguistic fuzzy models to cope with redundant or inconsistent rules and thereby enhances the robustness, flexibility and system modeling capability [26]. Hence the ability of this technique to indicate the interaction level of each rule with the remaining ones is considered, improving the global cooperation.

The hybridization of weighted rules and the two-level HSLR-LM could result in important improvements of the system accuracy, maintaining the interpretability to an acceptable level. To do so, the two-level HSLR-LM will be extended in order to obtain two-layer HSWLRs. This extension of the learning methodology will be named two-level HSWLR Learning Methodology (HSWLR-LM) and consists of two modifications:

- *Modification of the HRB structure and Inference System*, in order to consider the use of weights, obtaining Weighted HKBs (WHKBs).
- *Modification of the rule selection process* (*Step* 3 *of the two-level HSLR-LM algorithm*) to consider the derivation of rule weights.

This section is organized as follows. First, the use of weighted linguistic rules is introduced. Then, the way to consider repeated and double-consequent rules in HSWLRs is analyzed. Finally, the WHKB structure and the two level HSWLR-LM algorithm are presented.

4.1. The use of weighted linguistic rules

Using rule weights [4,26,33] has been previously considered for two main directly related reasons:

- *To handle repeated and multiple consequent rules*. Considering these kinds of rules may be intended as a local reinforcement of the problem space zones presenting a higher complexity. The use of rule weights can deal with the modeling of these subspaces in a better way [4,33], improving the way in which they interact.
- *To improve the model accuracy*. In Linguistic Fuzzy Modeling, the tuning of any fuzzy set will influence all rules that involve it. Rule weights suppose an effective extension of the conventional fuzzy reasoning process that allows the tuning of the system to be developed at the rule level [26,4].

It is clear that both approaches improve the accuracy of the learned model since they induce a good cooperation among rules. However, they come with the drawback of a small interpretability loss which lies in the difficulty to interpret the actual action performed by each rule in the interpolative reasoning process [24].

However, from other point of view (rule level), 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 close to consider this weight as an importance degree associated to the rule, determining how this rule interacts with its neighbor 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 this preserves the model readability. In this way, the use of rule weights represents an ideal framework for extended Linguistic Fuzzy Modeling when we are searching for a trade-off between accuracy and interpretability.

4.1.1. Weighted rule structure and inference system

As 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 [26]:

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

where $X_i(Y)$ are the linguistic input (output) 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 [1] and to compute the defuzzified output as the following *weighted sum*:

$$y_0 = \frac{\sum_i m_i \cdot w_i \cdot P_i}{\sum_i m_i \cdot w_i}$$

with m_i being the matching degree of the ith rule, w_i being the weight associated to it, and P_i being the characteristic value of the output fuzzy set corresponding to that rule. In this contribution, the center of gravity will be considered as characteristic value [1].

4.1.2. An example of learning process for weighted FRBSs

A simple approximation for weighted rule learning would consist of the following two steps – we will use this process in our experiments for comparison purposes, calling it WRL:

(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 [30] 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. [4]

4.2. Handling repeated and multiple consequent rules in HSLRs using weights

As said in the previous section, rule weights present an ideal framework to consider repeated and multiple consequent rules (see Fig. 3) improving the way

[4] The stochastic universal sampling procedure together with an elitist selection scheme (considering the MSE as fitness) and the max–min-arithmetical crossover [14] (see Section 5.3) together with the uniform mutation operators are considered.

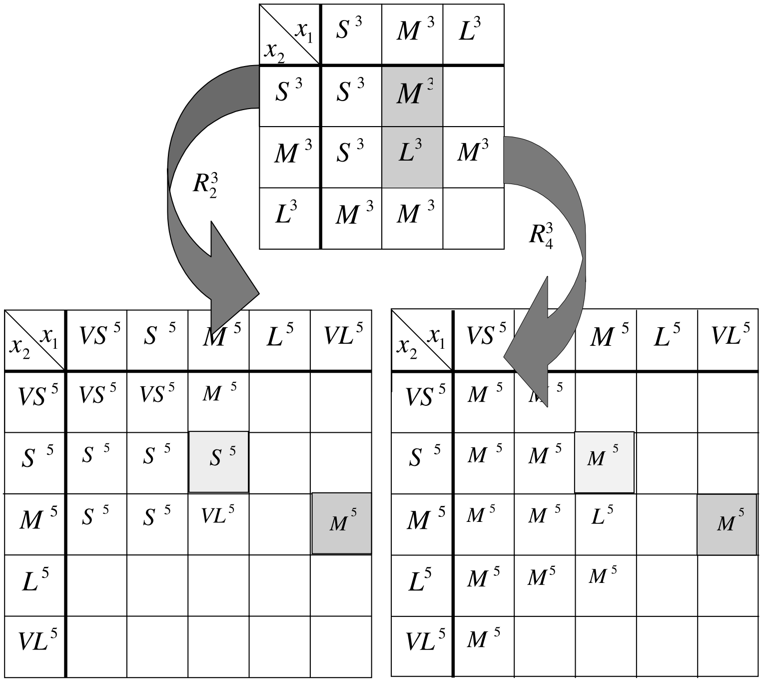


Fig. 3. Repeated and double-consequent rules.

in which these rules cooperate. In this section, we will see how these kinds of rules can be interpreted from a linguistic point of view. Besides, we will also analyze how the use of rule weights affects to them.

4.2.1. Repeated rules

More than one copy of a rule in the same layer can be produced as a consequence of the generation process (Steps 0, 1 and 2). To illustrate this situation, consider the two dark grey squares in Fig. 3 with the 2-*linguistic rule*:

$$\textbf{IF } X_1 \textbf{ is } M^5 \textbf{ and } X_2 \textbf{ is } VL^5 \quad \textbf{THEN } Y \textbf{ is } M^5,$$

which are both derived from the expansion of R_2^3 and R_4^3. The overlapping of the expanded rule images is produced by low values of the parameter δ (see Step 1.(c) i. in the algorithm). We experimentally compared the effect of the exclusion of those repeated rules in the input to the selection process of the HSLR-LM by modifying the algorithm in the following way:

$$HRB = Selection(Extract\ Repeated(JCLR))$$

However, results showed that the models obtained by the use of repeated rules performed the best approximation in training. This means that although the

Selection Process has the chance to eliminate all those redundant rules, sometimes it preserves some of them producing a sort of *reinforcement* on the whole subspace of the rule, like a global refinement action. This fact can be interpreted as a *weight* on that rule by using the extended fuzzy reasoning model presented in Section 4.1 with w_i being the number of times that the ith rule is repeated:

$$\textbf{IF } X_1 \textbf{ is } M^5 \textbf{ and } X_2 \textbf{ is } VL^5 \quad \textbf{THEN } Y \textbf{ is } M^5 \textit{ with } [w = 2],$$

In this work, we will consider this approach (which theoretically should obtain equivalent models with the same accuracy level). To do so, repeated rules are excluded of the HKB by obtaining an equivalent HSWLR without them:

$$\textit{Equivalent Weighted HRB} = (\textit{Extract Repeated}(\textit{JCLR}) + \textbf{Weights}).$$

To obtain an equivalent system without repeated rules, we maintain a single instance for each rule, with w_i being the sum of the weights of the corresponding repeated rules. Other equivalent HSWLR could be found with $w_i \in [0, 1]$ by means of a normalization process over the weights. In this way, the number of rules is decreased maintaining the same accuracy level, as we will clearly show in the experiments developed (see Section 6.2).

4.2.2. Double consequent rules

As well as we have considered the existence of repeated linguistic rules in the previous section, we can also observe that some of the learned rules have multiple consequents (see the two light grey squares in Fig. 3). This phenomenon is an extension of the usual linguistic fuzzy model structure which allows the RB to present rules where each combination of antecedents may have two or more consequents associated. The consideration of these kinds of rules may be intended as a local reinforcement of the problem space zones presenting high complexity. Therefore, as shown in [6,25], considering some rules with multiple consequents could improve the global system behavior.

In this way, we should note that these kinds of rules do not constitute an inconsistency from the Linguistic Fuzzy Modeling point of view but only a shift of the main labels making the final output of the rule lie in an intermediate zone between the most distant consequents. It is the case of the double-consequent linguistic rules [6]. For example, we can consider the specific combination of antecedents of Fig. 3, "X_1 is S^5 and X_2 is M^5", which has two different consequents associated, S^5 and M^5. The resulting double-consequent rule may be interpreted as follows:

$$\textbf{IF } X_1 \textbf{ is } S^5 \textbf{ and } X_2 \textbf{ is } M^5 \quad \textbf{THEN } Y \textbf{ is } \textit{between } S^5 \textbf{ and } M^5,$$

whose output is exactly the middle point between these two consequents,

$$y_0 = \frac{m_i \cdot d(S^5) + m_i \cdot d(M^5)}{m_i + m_i} = \frac{d(S^5) + d(M^5)}{2}.$$

Of course, notice that when these double-consequent rules interact with their neighbor ones, the output is not the middle point between both consequents but it is shifted according to the firing strengths of those neighbor rules.

The use of rule weights can deal with these kinds of fuzzy rules in a better way, since it is not necessary to discard the multiple consequents when they present useful information. It improves the way in which they interact and thereby the model accuracy. In this way, *the use of weights flexibilizes the rule structure*, *whose output lies in an explicit point between the most distant consequents which is determined by the corresponding rule weights*. This is the case when double-consequent rules are presented:

$$y_0 = \frac{m_i \cdot w_i \cdot d(S^5) + m_i \cdot w_i' \cdot d(M^5)}{m_i \cdot w_i + m_i \cdot w_i'} = \frac{w_i \cdot d(S^5) + w_i' \cdot d(M^5)}{w_i + w_i'}.$$

4.3. Weighted Hierarchical Knowledge Base

In this case, only the rule structure in the HRB has to be modified. The same structure of the weighted linguistic rules will be used to form the Weighted HRB (WHRB) and then the WHKB:

$$WHKB = HDB + WHRB.$$

Therefore, the fuzzy reasoning process must be extended as in the case of weighted linguistic rules, considering the matching degree of the rules fired (see Section 4.1.1).

In this way, we can define the WHRB as a whole HRB together with their corresponding rule weights:

$$WHRB = \bigcup_t RB^t + \bigcup_t W^t.$$

with W^t being the set of weights associated to the rules from layer t. We should notice that these weights are obtained over the whole HRB (and not over the isolated layers) since they must consider the way in which all the rules interact, i.e., the weights considered in the different layers, W^t, are interdependent. Therefore, *they must be jointly derived once the whole HRB is available*.

4.4. Algorithm

The same operation mode of the two-level HSLR-LM algorithm will be followed to generate linguistic fuzzy models with this new structure, but including the weight learning. Again, we consider the Wang and Mendel's

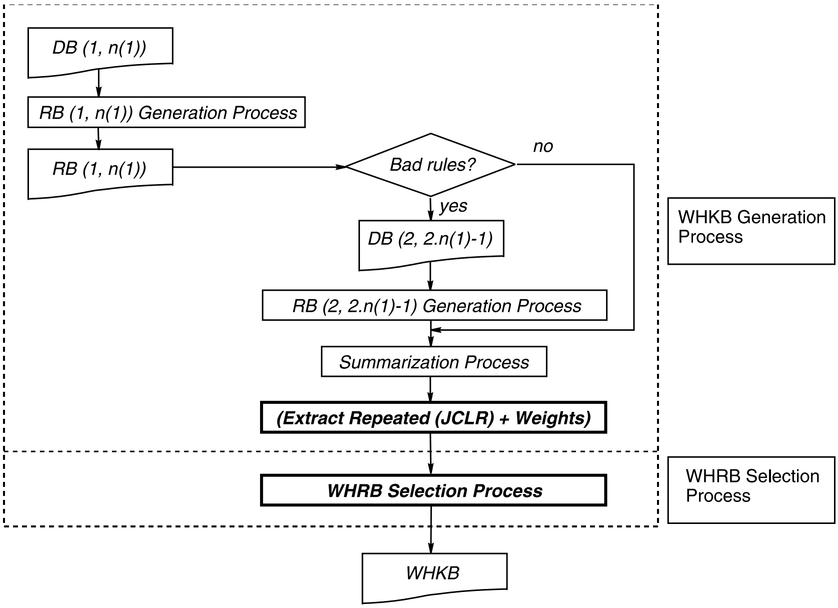


Fig. 4. HSWLR Learning Methodology.

algorithm [30] as LRG-method to obtain simple linguistic fuzzy models, although any other technique could be used. Therefore, the main steps of the extended algorithm will be the following ones:

Hierarchical Knowledge Base Generation Process

Step 0. $RB(1, n(1))$ *Generation Process*

Step 1. $RB(2, 2 \cdot n(1) - 1)$ *Generation Process*

Step 2. *Summarization Process* → (***Extract Repeated***(***JCLR***) + ***Weights***).

Hierarchical Rule Base Genetic Selection Process

Step 3. ***Genetic Weight Derivation and Rule Selection Process***.

- Genetic selection of a subset of rules presenting good cooperation.
- Genetic derivation of the weights associated to these rules.

Fig. 4 presents the flowchart of this algorithm. In the next section, the Step 3 of the two-level HSWLR-LM will be explained in deep.

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 used considering integer and real 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 pth 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 pth chromosome will be a real-valued vector representing the weights associated to the fuzzy rules considered.

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

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

Once an equivalent HSWLR without repeated rules has been considered, the initial pool is obtained with an individual having all genes with value '1' in both parts, and the remaining individuals generated at random:

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

5.2. Evaluating the chromosome

To evaluate the pth chromosome, we will follow one of the two policies proposed in Section 3.3, by using the corresponding fitness functions, $F(C^p)$ – accuracy – oriented policy – or $F'(C^p)$ – AC-oriented policy:

$$F(C^p) = MSE(E, RB(C^p)) = \frac{\sum_{e^l \in E} (ey^l - s(ex^l))^2}{2 \cdot |E|},$$
$$F'(C^p) = w_1 \cdot F(C^p) + w_2 \cdot N_{\text{rules}}^p.$$

In this case, $s(ex^l)$ – the output value obtained from the RB encoded in C^p – will be computed following the extended fuzzy reasoning process in order to consider the rule weights 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 [3]. 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, $\text{rank}(C^p)$ – with $\text{rank}(C^{best}) = 1$, by using the following non-increasing assignment function:

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

where N_C is the number of chromosomes 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, $\eta_{min} = 0.75$.

The classical *generational* [22] scheme has been considered in this algorithm. In this way, linear ranking is performed along with *stochastic universal sampling* [3]. 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 WHRB definition process, different operators working in 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 [14] 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:

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

with $a \in [0, 1]$ being a constant parameter chosen by the GA designer.

In this case, eight offspring are generated by combining the two ones from the C_1 part (two-point crossover) with the four ones from the C_2 part (max–

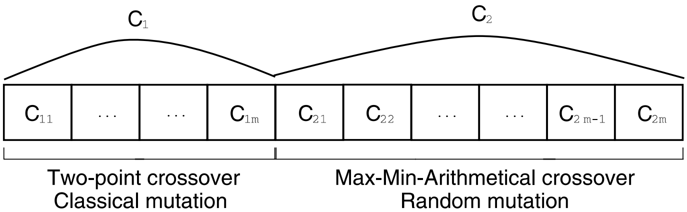


Fig. 5. Genetic representation and operators' application scope.

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. 5 shows the application scope of these operators.

6. Experiments

In this section, we will analyze the performance of the linguistic fuzzy models generated from the proposed two-level HSWLR-LM, when solving two real-world problems [7]. They will be compared to the models designed by the following methods: the well-known ad hoc data-driven method proposed by Wang and Mendel [30] (calling it as WM), the original two-level HSLR-LM method [10] and the WRL method presented in Section 4.1.2. From now on, a reference to an application of any of these methods will be represented by the following expression:

$$\text{method}(r[, q])$$

with r (and q in the case of the hierarchical-based methods) being the granularity level of the linguistic partitions used in the method.

The linguistic partitions considered are comprised by *three or five linguistic terms* with triangular-shaped fuzzy sets giving meaning to them. These linguistic terms are labeled from l_1 to l_5, for the 5-linguistic partition, standing l_1 and l_2, for very small, and small l_3, for medium, and l_4 and l_5, for large and very large, respectively. In order to ease the comparisons in the decision tables, the linguistic terms in the 3-linguistic partitions will be labeled as L_1, L_3 and L_5 whose modal points are the same that the ones for l_1, l_3 and l_5 (see Section 3.1).

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 [1].

Finally, the values of the parameters used in all of these experiments are presented as follows:

- *Hierarchical generation*: 0.1 as δ-(2·n-1)-*linguistic partition* terms selector, 0.5 as τ – used to calculate E_i, and 1.1 as α – used to decide the expansion of rule.
- *Genetic weight derivation and rule selection process*: 61 individuals, 2000 generations, 0.6 as crossover probability, 0.2 as mutation probability per chromosome, and 0.35 for the factor a in the crossover operator. The same parameters (excluding the a factor and with 1000 generations) are considered for the selection process of the original two-level HSLR-LM method.
- *WRL*: 61 individuals, 2000 generations, 0.6 as crossover probability, 0.2 as mutation probability per chromosome, and 0.35 for the factor a 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 cannot be directly calculated. The problem comes when trying to compute the maintenance costs of low voltage 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 cannot 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* [7]. 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. 6 (notice that they present strong non-linearities). Both data sets considered are available at http://decsai.ugr.es/∼casillas/fmlib/.

The results obtained by the four methods analyzed are showed in Table 3, 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 showed in boldface in each table. These results were obtained for an AMD K7

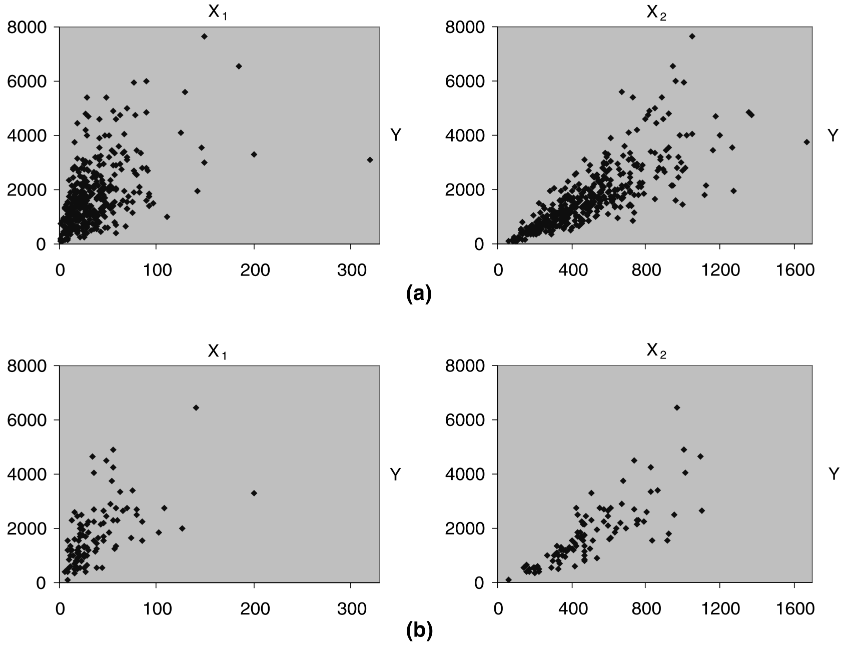


Fig. 6. (a) (X_1, Y) and (X_2, Y) dependency in the training data; (b) (X_1, Y) and (X_2, Y) dependency in the test data.

Table 3
Results obtained in the low voltage line problem

Method	#R	MSE_{tra}	MSE_{tst}
WM(3)	**7**	594,276	626,566
WM(5)	13	298,446	282,058
WRL(3)	**7**	231,917	230,035
WRL(5)	13	242,680	252,483
HSLR(3,5)	12	178,950	167,318
AC-HSLR(3,5)	11	180,111	166,210
HSWLR(3,5)	13	**161,632**	**151,259**
AC-HSWLR(3,5)	9	**163,406**	**156,434**

(Athlon) with clock rate of 1500 MHz and 256 MB of main memory. The run times for HSWLR(3,5) and AC-HSWLR(3,5) were 10 and 7 min.

The decision tables of the models obtained by the studied methods are presented in Figs. 7–10. In the left-hand side of these figures, each cell of the tables represents a fuzzy subspace and contains its associated output consequent(s), i.e., the correspondent label(s) together with its(their) respective rounded rule weight(s). The *absolute importance weight* for each fuzzy rule has

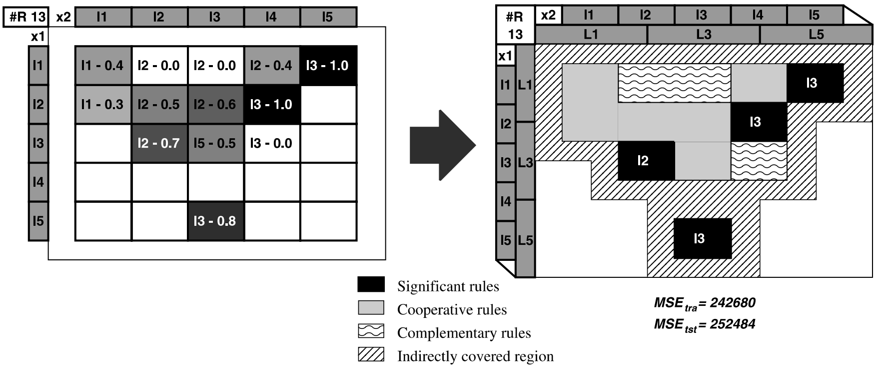


Fig. 7. Decision table of the model obtained from WRL with five labels.

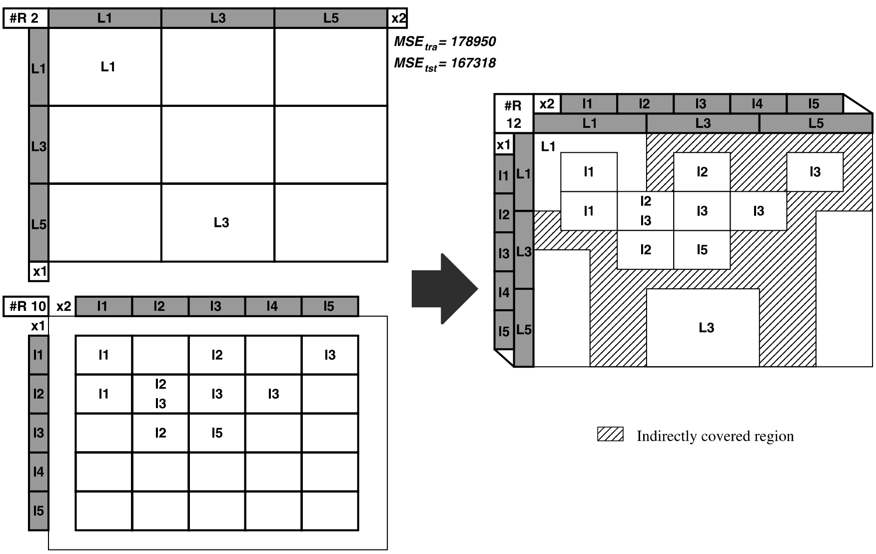


Fig. 8. Decision table of the model obtained from HSLR(3,5).

been graphically showed 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 neighbor ones* which could help the system experts to identify important rules.

On the other hand, in the right-hand side of these figures, an expert interpretation of the *relative importance* of the rules is presented as regards their influence in the modeling of the respective problem space zone. Three kinds of rules are represented in the figure:

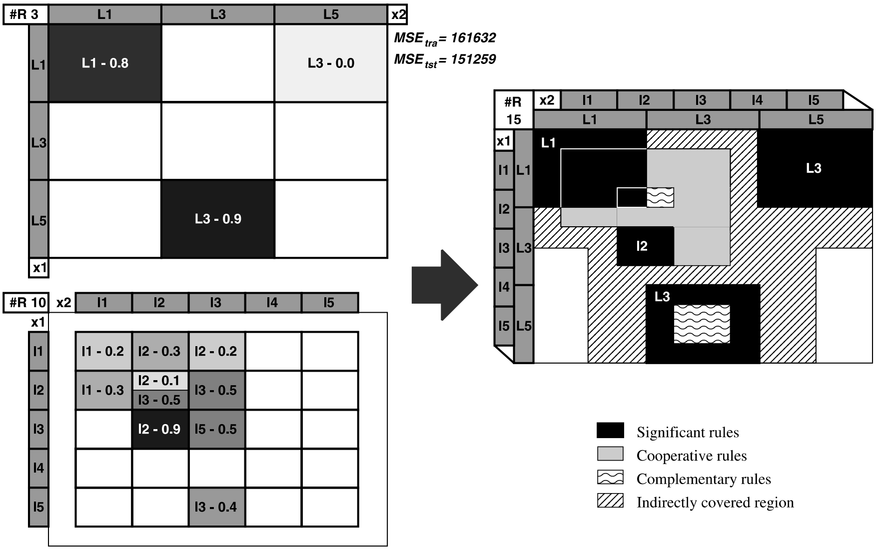


Fig. 9. Decision table of the model obtained from HSWLR(3,5).

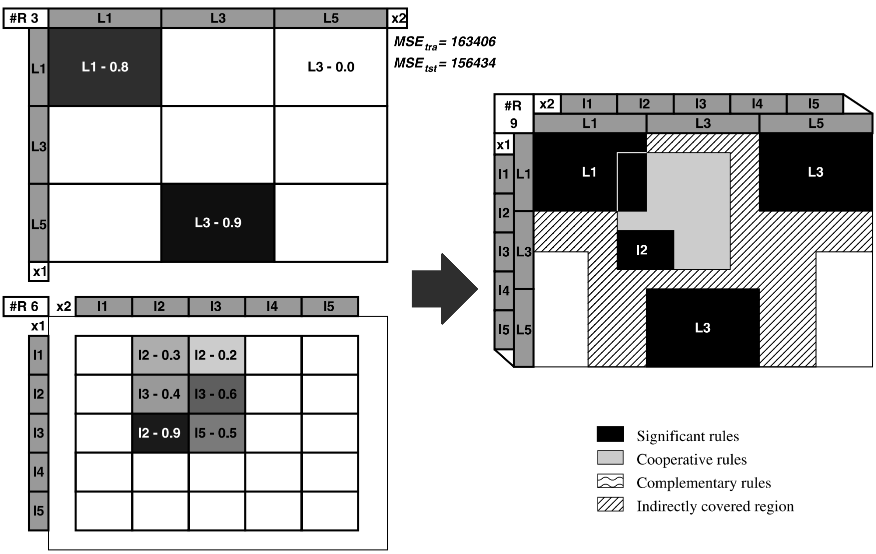


Fig. 10. Decision table of the model obtained from AC-HSWLR(3,5).

- *Significant or important rules*: Those in black, corresponding to rules that have a higher weight than their neighbors or rules that are the ones of their regions.

- *Cooperative rules*: Those in grey, representing rules that have a more or less similar weight than their neighbor ones.
- *Complementary rules*: Those in white (with waves), representing rules that have a lower weight than their neighbor ones.

Figs. 7 and 8 respectively show the decision tables of WRL and HSLR-LM, the previous approaches performing improved Linguistic Fuzzy Modeling.

In view of the results obtained in the above experiments and analyzing the presented decision tables, we will point out some important conclusions and features of the proposed methodology.

However, before continuing, we should remark that all the features of the HSLR-LM [10] remain in the new methodology. Thus, we will focus our analysis on the characteristics that the rule weight approach add to the mentioned methodology.

6.1.1. Analysis of the results from the accuracy point of view

From this point of view, we can say that the proposed hybrid methodology outperforms the best results obtained from the remaining methods, with improvements of about a 10% over the HSLR-LM and about a 73% over the classical method, with both selection policies. The similar results obtained when we use both policies evidence the robustness of the idea of including the weight learning in this methodology, since the improvements in the models directly depend on the combination of both techniques and not on other factors.

On the other hand, *the good accuracy results achieved by this new technique lie in the complementary characteristics that the use of weights and the hierarchical approach present*. The original HSLR was designed to improve simple linguistic fuzzy models reinforcing only the modeling of those problem subspaces with more difficulties while the use of rule weights improves the way in which they interact. In Fig. 9, we can see how weighted double-consequent rules are considered as a local reinforcement interacting at a low level (see rules "*IF* X_1 is l_2 and X_2 is l_2 *THEN* Y is l_2 *with* $[w = 0.1]$ and l_3 *with* $[w = 0.5]$", where the adapted weights make the second consequent more important). In many cases, a simple selection process can not address these cases.

Furthermore, there are single consequent rules with a bad interaction level which, if removed, make the system accuracy decrease. Rule weight learning can deal with these kinds of rules, decreasing their influence in the fuzzy reasoning when they interact with other rules. This is the case of the rule "*IF* X_1 is L_1 and X_2 is L_5 *THEN* Y is L_3 *with* $[w = 0.039]$", whose weight is pretty close to 0 in Figs. 9 and 10:

- These kinds of rules are important since they are the ones of their definition spaces (see that they are usually the rules in the corners).
- In these cases, the weight is so low in order to make these rules appropriately interact with their neighbor ones.

6.1.2. Analysis of the results from the complexity (interpretability) point of view

In view of the results presented in Table 3, we can conclude that the model presenting the best trade-off between accuracy and interpretability is the one obtained with AC-HSWLR(3,5) (see Fig. 10), which confirms the good behavior of the AC-oriented selection policy when rule weights are considered. The model so obtained was composed of only nine rules presenting a very similar accuracy to the most accurate result with 13.

Notice that, *useful information can be obtained if we consider rule weights as relative importance factors.* [5] If we pay attention to the decision tables of the models obtained from the HSWLR-LM (Figs. 9 and 10), we could see how practically the same important rules can be observed in all the figures. In the case of the AC selection policy-based model shown in Fig. 10, the following important rules can be obtained:

$$IF\ X_1\ is\ L_1\ and\ X_2\ is\ L_1\quad THEN\ Y\ is\ L_1,$$
$$IF\ X_1\ is\ L_1\ and\ X_2\ is\ L_5\quad THEN\ Y\ is\ L_3,$$
$$IF\ X_1\ is\ L_5\ and\ X_2\ is\ L_3\quad THEN\ Y\ is\ L_3,$$
$$IF\ X_1\ is\ l_3\ and\ X_2\ is\ l_2\quad THEN\ Y\ is\ l_2.$$

The first three rules can represent all the covered space of the corresponding HRB. These three rules determine the system behavior at a first upper level, while the remaining ones produce local refinements on them. All this information can be very useful to understand the system behavior.

As main conclusion of this study, we could say that the systems obtained with the proposed approach present a good interpretability level.

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 [7] 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. These estimations allow electrical companies to justify their expenses. Moreover, the model must be able to explain how a specific value is computed for a certain town. Our objective will be

[5] The rule weights must be considered as relative importance factors since the final output depends on the weights of their neighbor rules. Hence, notice that, $w = 0$ does not necessarily entail a *non-important rule*.

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 1059 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% and 20% of the original size respectively. Thus, the training set contains 847 elements, whilst the test one is composed by 212 elements. These data sets used are available at http://decsai.ugr.es/~casillas/fmlib/.

The results obtained by the analyzed methods are showed in Table 4, where the same equivalences than for Table 3 remain. Again, these results were obtained for an AMD K7 (Athlon) with clock rate of 1500 MHz and 256 MB of main memory. In this case the run times for HSWLR(3,5) and AC-HSWLR(3,5) were 75 and 49 min.

From these results, we can see that the WRL and HSLR methods achieve improvements over the WM method. Analyzing the model obtained by HSWLR, we can conclude that it seems to present the best performance in approximation (MSE_{tra}) and practically the same that AC-HSLR in generalization (MSE_{tst}), with improvements of about an 11% over the HSLR method and about a 260% over the classical one (WM).

Notice that by using the *AC-oriented policy* in the *Weight Derivation and Rule Selection* process, the number of rules is reduced maintaining the accuracy to a significantly good level. It is due to the elimination of the repeated rules by obtaining an equivalent system. This way, the HSWLR obtained by using the AC-oriented policy presents approximately the half number of rules than the one derived from the accuracy-oriented policy, and much less compared with the previous HSLR-LM methods.

Table 4
Results obtained in the medium voltage line problem

Method	#R	MSE_{tra}	MSE_{tst}
WM(3)	**28**	197,313	174,400
WM(5)	66	71,294	80,934
WRL(3)	**28**	78,869	74,070
WRL(5)	66	32,562	32,801
HSLR(3,5)	172	22,358	23,755
AC-HSLR(3,5)	132	23,525	**22,328**
HSWLR(3,5)	80	**19,558**	22,358
AC-HSWLR(3,5)	49	20,425	22,873

7. Concluding remarks

In this work, we have proposed the hybridization of the hierarchical scheme with the use of rule weights by extending a previous two-level HSLR-LM. To do so, the model structure is extended by allowing the use of weighted hierarchical linguistic rules and the selection component (which has the aim to select the rules best cooperating) is modified by allowing it to jointly perform the rule selection and the rule weight derivation. In this way, a GA performing both tasks has been developed.

The accurate results of the proposed methodology, compared with other related approaches, have been contrasted when solving two real-world electrical distribution problems. On the other hand, the obtained models presented an acceptable interpretability level, where significant rules were identified studying the weights of the obtained rules, helping us to easily interpret the model behavior.

Appendix A. Acronyms

Acronym	Meaning
AC	Accuracy-Complexity-oriented policy
DB	Data Base
GA	Genetic Algorithm
HDB	Hierarchical Data Base
HKB	Hierarchical Knowledge Base
HRB	Hierarchical Rule Base
HSLR	Hierarchical System of Linguistic Rules
HSLR-LM	Hierarchical System of Linguistic Rules Learning Methodology
JCLR	Joined set of Candidate Linguistic Rules
LRG	Linguistic Rule Generation method
MSE	Mean Square Error
RB	Rule Base
Those considering rule weights	
HSWLR	Hierarchical System of Weighted Linguistic Rules
HSWLR-LM	Hierarchical System of Weighted Linguistic Rules Learning Methodology
WHKB	Weighted Hierarchical Knowledge Base
WHRB	Weighted Hierarchical Rule Base

References

[1] A. Bardossy, L. Duckstein, Fuzzy rule-based modeling with application to geophysical, in: Biological and Engineering Systems, CRC Press, Boca Raton, 1995.
[2] A. Bastian, How to handle the flexibility of linguistic variables with applications, International Journal of Uncertainty, Fuzziness and Knowledge-Based Systems 2 (4) (1994) 463–484.
[3] J.E. Baker, Reducing bias and inefficiency in the selection algorithm, in: J.J. Grefenstette (Ed.), Proceedings of the 2nd International Conference on Genetic Algorithms, Lawrence Erlbaum, Hillsdale, NJ, USA, 1987, pp. 14–21.
[4] J.S. Cho, D.J. Park, Novel fuzzy logic control based on weighting of partially inconsistent rules using neural network, Journal of Intelligent and Fuzzy Systems 8 (2000) 99–110.
[5] J. Casillas, O. Cordón, F. Herrera, 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.
[6] O. Cordón, F. Herrera, A proposal for improving the accuracy of linguistic modeling, IEEE Transactions on Fuzzy Systems 8 (4) (2000) 335–344.
[7] O. Cordón, F. Herrera, L. Sánchez, Solving electrical distribution problems using hybrid evolutionary data analysis techniques, Applied Intelligence 10 (1999) 5–24.
[8] O. Cordón, F. Herrera, P. Villar, Analysis and guidelines to obtain a good uniform fuzzy partition granularity for fuzzy rule-based systems using simulated annealing, International Journal of Approximate Reasoning 25 (3) (2000) 187–215.
[9] O. Cordón, F. Herrera, P. Villar, A genetic learning process for the scaling factors, granularity and contexts of the fuzzy rule-based system data base, Information Science 136 (1–4) (2001) 85–107.
[10] O. Cordón, F. Herrera, I. Zwir, Linguistic modeling by hierarchical systems of linguistic rules, IEEE Transactions on Fuzzy Systems 10 (1) (2002) 2–20.
[11] J. Espinosa, J. Vandewalle, Constructing fuzzy models with linguistic integrity from numerical data-AFRELI algorithm, IEEE Transactions on Fuzzy Systems 8 (5) (2000) 591–600.
[12] A. González, R. Pérez, A study about the inclusion of linguistic hedges in a fuzzy rule learning algorithm, International Journal of Uncertainty, Fuzziness and Knowledge-Based Systems 7 (3) (1999) 257–266.
[13] H.B. Gürocak, A genetic-algorithm-based method for tuning fuzzy-logic controllers, Fuzzy Sets and Systems 108 (1) (1999) 39–47.
[14] F. Herrera, M. Lozano, J.L. Verdegay, Tuning fuzzy logic controllers by genetic algorithms, International Journal of Approximate Reasoning 12 (1995) 299–315.
[15] H. Ishibuchi, K. Nozaki, H. Tanaka, Efficient fuzzy partition of pattern space for classification problems, Fuzzy Sets and Systems 59 (1993) 295–304.
[16] H. Ishibuchi, K. Nozaki, N. Yamamoto, H. Tanaka, Selecting fuzzy if–then rules for classification problems using genetic algorithms, IEEE Transactions on Fuzzy Systems 9 (3) (1995) 260–270.
[17] H. Ishibuchi, T. Murata, A genetic-algorithms-based fuzzy partition method for pattern classification problems, in: F. Herrera, J. Verdegay (Eds.), Genetic Algorithms and Soft Computing, Physica Verlag, Wurzburg, 1996, pp. 555–578.
[18] H. Ishibuchi, T. Takashima, Effect of rule weights in fuzzy rule-based classification systems, IEEE Transactions on Fuzzy Systems 3 (3) (2001) 260–270.
[19] A. Krone, H. Krause, T. Slawinski, A new rule reduction method for finding interpretable and small rule bases in high dimensional search spaces, in: Proceedings of the 9th IEEE International Conference on Fuzzy Systems, San Antonio, TX, USA, 2000, pp. 693–699.
[20] B.D. Liu, C.Y. Chen, J.Y. Tsao, Design of adaptive fuzzy logic controller based on linguistic-hedge concepts and genetic algorithms, IEEE Transactions on Systems, Man and Cybernetics – Part B: Cybernetics 31 (1) (2001) 32–53.

[21] L. Magdalena, F. Monasterio-Huelin, A fuzzy controller with learning through the evolution of its knowledge base, International Journal of Approximate Reasoning 16 (3) (1997) 335–358.
[22] Z. Michalewicz, Genetic Algorithms + Data Structures = Evolution Programs, Springer, Heidelberg, Germany, 1996.
[23] T. Mitchell, Machine Learning, McGraw-Hill, New York, 1997.
[24] D. Nauck, R. Kruse, How the learning of rule weights affects the interpretability of fuzzy systems, in: Proceedings of the 7th IEEE International Conference on Fuzzy Systems, Anchorage, AK, USA, 1998, pp. 1235–1240.
[25] K. Nozaki, H. Ishibuchi, H. Tanaka, A simple but powerful heuristic method for generating fuzzy rules from numerical data, Fuzzy Sets and Systems 86 (1997) 251–270.
[26] N.R. Pal, K. Pal, Handling of inconsistent rules with an extended model of fuzzy reasoning, Journal of Intelligent and Fuzzy Systems 7 (1999) 55–73.
[27] W. Pedrycz, Fuzzy equalization in the construction of fuzzy sets, Fuzzy Sets and Systems 119 (2) (2001) 329–335.
[28] C.A. Peña-Reyes, M. Sipper, Fuzzy CoCo: a cooperative coevolutionary approach to fuzzy modeling, IEEE Transactions on Fuzzy Systems 9 (5) (2001) 727–737.
[29] P. Thrift, Fuzzy logic synthesis with genetic algorithms, in: Proceedings of 4th International Conference on Genetic Algorithms, Morgan Kaufmann, San Mateo, CA, 1991, pp. 509–513.
[30] L.X. Wang, J.M. Mendel, Generating fuzzy rules by learning from examples, IEEE Transactions on Systems, Man and Cybernetics 22 (1992) 1414–1427.
[31] R.R. Yager, On the construction of hierarchical fuzzy systems model, IEEE Transactions on Systems, Man and Cybernetics 28 (1) (1998) 55–66.
[32] J. Yen, L. Wang, C.W. Gillespie, Improving the interpretability of TSK fuzzy models by combining global learning and local learning, IEEE Transactions on Fuzzy Systems 6 (4) (1998) 530–537.
[33] W. Yu, Z. Bien, Design of fuzzy logic controller with inconsistent rule base, Journal of Intelligent and Fuzzy Systems 2 (1994) 147–159.