International Journal of Approximate Reasoning 27 (2001) 61–93
www.elsevier.com/locate/ijar

Fuzzy modeling by hierarchically built fuzzy rule bases

Oscar Cordón [a], Francisco Herrera [a,*], Igor Zwir [b]

[a] *Department of Computer Science and Artificial Intelligence, ETS de Ingeniera Informatica, University of Granada, Avda. Andalucia 38, 18071 Granada, Spain*
[b] *Department of Computer Science, University of Buenos Aires, 1428 Buenos Aires, Argentina*

Received 1 August 2000; accepted 1 March 2001

Abstract

Although Mamdani-type fuzzy rule-based systems (FRBSs) became successfully performing clearly interpretable fuzzy models, they still have some lacks related to their accuracy when solving complex problems. A variant of these kinds of systems, which allows to perform a more accurate model representation, are the so-called approximate FRBSs. This alternative representation still cannot avoid the problems concerning the fuzzy rule learning methods, which as prototype identification algorithms, try to extract those approximate rules from the object problem space. In this paper we deal with the previous problems, viewing fuzzy models as a class of local modeling approaches which attempt to solve a complex problem by decomposing it into a number of simpler subproblems with smooth transitions between them. In order to develop this class of models, we first propose a common framework to characterize available approximate fuzzy rule learning methods, and later we modify it by introducing a fuzzy rule base hierarchical learning methodology (FRB-HLM). This methodology is based on the extension of the simple building process of the fuzzy rule base of FRBSs in a hierarchical way, in order to make the system more accurate. This flexibilization will allow us to have fuzzy rules with different degrees of specificity, and thus to improve the modeling of those problem subspaces where the former models have bad performance, as a refinement. This approach allows us not to have to assume a fixed number of rules and to integrate the good local behavior of the hierarchical model with the global model, ensuring a good global performance. © 2001 Elsevier Science Inc. All rights reserved.

* Corresponding author. Tel.: +34-58-24-40-19; fax: +34-58-24-33-17.
E-mail addresses: ocordon@decsai.ugr.es (O. Cordón), herrera@decsai.ugr.es (F. Herrera), zwir@dc.uba.ar (I. Zwir).

PII: S0888-613X(01)00034-2

Keywords: Fuzzy modeling; Mamdani-type fuzzy rule-based systems; Fuzzy rule base; Genetic algorithms; Hierarchical fuzzy clustering; Approximate fuzzy rules

1. Introduction

Nowadays, one of the most important areas for the application of fuzzy set theory as developed by Zadeh [35] are fuzzy rule-based systems (FRBSs). These kinds of systems constitute an extension of classical rule-based systems, because they deal with fuzzy rules instead of classical logic rules. Thanks to this, they have been successfully applied to a wide range of problems from different areas presenting uncertainty and vagueness in different ways [3,21,24,26].

There are at least two different kinds of FRBSs in the literature, the Mamdani and Takagi–Sugeno–Kang (TSK), which differ on the composition of the rule consequent. The use of one or the other depends on the fact that the main requirement is the interpretability or the accuracy of the model, respectively.

Although the Mamdani-type FRBS presents the maximum description level, it is not as accurate as desired in some cases. Therefore, at least two things could be done to improve the accuracy of this model type. On the one hand, we can preserve the linguistic representation of this model and perform successive refinements on it, improving its accuracy without losing interpretability to a high degree [13,15]. On the other hand, we can improve the model by using a more accurate representation. To do so, we focus our attention on a variant of Mamdani-type, the so-called approximate FRBSs [1,2]. These kinds of FRBSs are the ones that have fuzzy rules composed of fuzzy variables – with a fuzzy set associated defining their meaning – that do not take as a value a linguistic term, like in the case of linguistic variables [36–38], but a real fuzzy set.

Even though a great deal of research activity has focused on the development of methods to build or refine approximate FRBSs from numerical data, they still present some problems. To deal with these kinds of models and methods, in this paper we first propose a common framework to group and characterize fuzzy rule generation methods (FRG-methods), i.e., methods for learning approximate rules. Later, we introduce a modification of this framework in order to solve many of the former problems by designing a fuzzy rule base hierarchical learning methodology (FRB-HLM). The main purpose of this methodology is to automatically generate more accurate approximate fuzzy models by performing successive refinements of initial models generated by FRG-methods.

To do so, we introduce the concept of *layers*, which was previously applied to descriptive models in [13,15]. In this extension, the fuzzy rule base (FRB) is constructed by the development of set of *layers* or *FRBs*, each one containing fuzzy rules with a different specificity level, i.e. different fuzzinesses. These kinds of rules are called hierarchically generated fuzzy rules.

In order to do that, this paper is set up as follows. In Section 2, a description of the approximate FRBS model is introduced, as well as its advantages and drawbacks. In this section, we also consider some problems associated with the FRG-methods, and propose a common framework for dealing with them. In Section 3, we introduce the FRB-HLM as a solution to many of the previous problems and perform a description of the hierarchically built FRB philosophy and the relation between its components. Next, the algorithm is explained in detail. In Section 4, the fuzzy modeling process obtained from FRB-HLM and well-known inductive FRG-methods is applied to solve three different applications. Finally, in Section 5, some concluding remarks are pointed out.

2. Approximate FRBSs

In this section we first compare the approximate FRBSs with the linguistic ones, highlighting their advantages and lacks. Next, we characterize the FRG-methods, which built approximate FRBSs, providing a common framework to deal with them. Finally, the drawbacks of the FRG-methods are also discussed.

2.1. Approximate versus linguistic FRBSs

As we have said, there are at least two different forms of fuzzy modeling: Mamdani-type and TSK FRBSs. The former presents the maximum description and interpretability level, but it is not as accurate as desired in some complex problems. In opposite, the second approach performs the more accurate approximation with the drawback of losing interpretability in their consequents.

The lack of accuracy of Mamdani-type models is due to some problems related to the linguistic rule structure considered, which are a consequence of the inflexibility of the concept of linguistic variable [36–38]. A summary of these problems may be found in [1,4,8], and is briefly enumerated as follows:

- There is a lack of flexibility in the FRBS because of the rigid partitioning of the input and output spaces.
- When the system input variables are dependent themselves, it is very hard to fuzzy partition the input spaces.
- The homogenous partitioning of the input and output spaces when the input–output mapping varies in complexity within the space is inefficient and does not scale to high-dimensional spaces.
- The size of the FRB directly depends on the number of variables and linguistic terms in the system. Obtaining an accurate FRBS requires a significant

granularity amount, i.e., it needs the creation of new linguistic terms. This granularity increase causes the number of rules to rise significantly, which may take the system to lose the capability of being interpretable for human beings.

A variant of these Mamdani-type FRBS-based modeling approaches has been proposed in the last few years, the approximate FRBS [1,4]. It is based on the former approach but considers the lack of accuracy as a major drawback. While the former descriptive FRBSs have associated a knowledge base composed of a database – containing linguistic partitions – and a rule base – composed of linguistic rules which make use of these linguistic partitions –, the approximate ones only have to define an FRB. This happens because their aproximate fuzzy rules contain variables which are different locally defined fuzzy values.

In order to distinguish between the type of modeling performed to obtain Mamdani-type and approximate FRBSs, we are going to refer to the former as *linguistic modeling*, and to the latter as *fuzzy modeling*. That is, *linguistic models* are performed by descriptive Mamdani-type rules or *linguistic rules*, and *fuzzy models* are developed by approximate Mamdani-type rules or *fuzzy rules*. In the following some distinctions between both types of modeling are given:

- *Linguistic modeling* makes use of fuzzy rules composed of linguistic variables that take values in a term set with real-world meaning (linguistic rules). These kinds of models are characterized by the fact that their main requirement is the system interpretability.
- In *fuzzy modeling*, the fuzzy rules are composed of fuzzy predicates without a linguistic meaning, i.e., the variables forming the rules do not take as a value a linguistic term with a fuzzy set associated defining their meaning, but a real fuzzy set. These models pretend to be more accurate than the former ones.

The choice between how interpretable and how accurate the model must be, usually depends on the user's needs for a specific problem and will condition the kind of FRBS selected to model it. As well as that, in this paper we will focus on developing more accurate fuzzy models by an FRB-HLM, which provides approximate solutions to different problems, especially real-world problems.

2.2. Approximate FRBS features

Approximate FRBSs have some interesting advantages that get them to be very suitable for fuzzy modeling purposes in many cases [8]:

- The expressive power of the rules, that present their own specificity in terms of the fuzzy sets involved in them, thus introducing additional degrees of freedom in the system.
- The number of rules is adapted to the complexity of the problem, needing less rules in simple problems, and being able to use more rules if it is neces-

sary. This is likely to be of benefit in tackling the course of dimensionality when scaling to multidimensional systems.

These facts, which allow approximate FRBSs to be more accurate in complex problems, have unfortunately some drawbacks associated [1]:

- The FRB readability is lost because there is no global interpretation of the variables considered. In spite of this, approximate FRBSs locally describe the system behavior in a similar way to other models like neural networks, but in a more descriptive way.
- The approximation capability causes an excessive specificity with bad generalization, sometimes obtaining an unwanted overfitting.

Although fuzzy and linguistic modeling are not incompatible, but complementary, in this paper we focus our attention in the former, and consequently on the model accuracy.

2.3. *Approximate FRBSs learning methods (FRG-methods) as prototype-identification algorithms*

Some automatic techniques have been proposed to learn a proper FRB for an approximate FRBS to solve a specific problem. The accuracy of the FRBS in solving this problem will depend on the intrinsic characteristics of the problem and on the mentioned learning tasks. In spite of these dependences, we will attempt to characterize these learning methods which we have labeled as FRG-methods.

Regarding [29,40], we can say that basically an FRG-method does its job as a prototype-identification algorithm, which performs the optimization of a functional $Q(F; \mathrm{Model}(\gamma))$ that measures the extent by which the parameterized model $\mathrm{Model}(\gamma)$ fits the subset F of the object being described (see Fig. 1).

From this perspective, the problem is formulated as a clustering problem in the sense that extracted subsets meet, to some extent, the requirements imposed by the model collection in the same way that elements of a clustering partition satisfy the constraint that their members be as similar as possible [29,40]. This point of view follows the original ideas of Ruspini [28], later expanded by Bezdek introducing various methods centered upon the notion of prototype [5]. The basic idea of summarizing a dataset by a number of representative prototypes – objects lying in the same space as the sample points – was later extended in many significant directions by relaxing this concept in a variety of ways, for example, line segments, ellipsoids, etc. [7]. In this paper we particularize this concept by considering these prototypes as being fuzzy rules [2,17,18].

Having these concepts in mind, FRG-methods can be seen as identification algorithms with fuzzy rule prototypes, i.e., fuzzy model builders whose main purpose is to extract the most suitable set of fuzzy rules from an object (input–output data) according to an optimization measure, which evaluates the quality of the approximation. Additionally, they organize results and

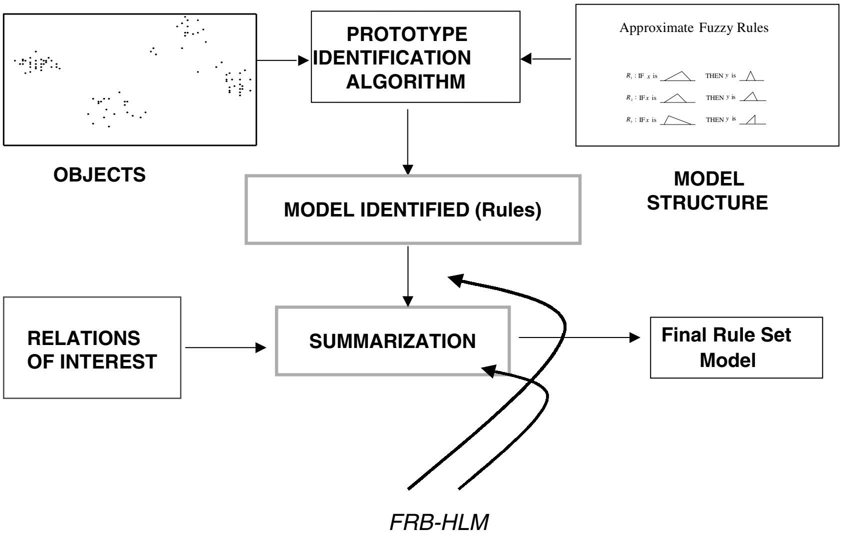

Fig. 1. FRG-methods as prototype-identification algorithms.

summarize them by an interestingness criterion, in order to provide a more compact and useful representation of the salient structures.

In order to illustrate this situation, consider for example the Weighted Counting Algorithm introduced by Bardossy and Duckstein [3] which, as can be seen in Appendix A, identifies approximate fuzzy rules from a set of input–output data (*object F*) for an approximate fuzzy model $Model(\gamma)$. The quality of identified candidate substructures (rule premises) is measured in base of its degree of fulfillment, i.e., a covering criterion ($Q(F; Model(\gamma))$). These extracted rules could also be summarized by, for example, a user-based *relation of interest* which imposes a maximum acceptable number of rules.

2.4. Drawbacks of the FRG-methods

All of these models generated by FRG-methods have the same drawbacks that prototype identification methods have, and all of them try to give their different own solutions, which become particular to the corresponding method:

- Simple formulation of the prototype-identification problem as an optimization of a functional would simply result in a large collection of very specific rules with small extent and high accuracy, but with poor generalization. Smaller rather than larger significative sets with high generalization power would be preferred.
- The determination of a complete clustering or a partition of the dataset into a fixed number of prototypes becomes a big deal for a long time.

To deal with these problems, in the following section we will present an FRB-HLM in order to build a FRB with the purpose of solving some of the above drawbacks.

3. Fuzzy rule base hierarchical learning methodology

To overcome some of the drawbacks of the approximate FRBSs (Section 2.4) and of the FRG-methods (Section 2.3), we propose a FRB-HLM which, as a meta-method, modifies the framework shown in Fig. 1 and considers the following points:

- On the one hand, we would like to implement a sort of trade-off between the extensionality and the accuracy of the models generated, having in mind that rules which perform good explanations tend to be limited in extent while those that, conversely, are capable of describing large subsets of the dataset, do it so poorly.
- On the other hand, we will adopt a more general treatment than that of a typical clustering problem, emphasizing the sequential isolation of individual clusters [23] rather than determination of a full clustering. Furthermore, we do not want to assume a priori knowledge of the total number of clusters – rule prototypes – requiring that the set of all clusters be an exhaustive partition of the complete object.

To do so, the FRB-HLM will modify the initial model identified by a FRG-method in an iterative way, performing a gradual refinement of it. Moreover, it will also modify the summarization process seen in Fig. 1, by adding a rule selection process to obtain a compact set of rules that have good cooperation between them and to remove the unnecessary ones.

3.1. Keypoints of the FRB-HLM

Our approach owes much to those clustering generalizations mentioned in Section 2.3 and to the notion of hierarchical clusters [17,19]. It is also closely related to other modeling techniques which work with fuzzy granules, fuzzy graphs, etc. [27,39]. As them, we should also answer some important questions which concern the structures (clusters, rules, granules, etc.) used by the former techniques:

How can rules, granules, clusters, etc. be partitioned?

This question is related to some aspects like the compactness of a cluster, measures inside the cluster, measures between clusters, granule perimeters, rule scopes, etc.

How many rules, granules, clusters, etc. could exist?

This is concerned with system comprehension, accuracy, overfitting, etc. In order to give an answer to these questions, and to previous problems, in the following we list some keypoints of the FRB-HLM:

- *Dynamic rule expansion or partition* depending on the FRG-method used, in order to take advantage of its intrinsic capabilities. These methods sometimes perform this task in a more static or dynamic way according to their philosophy.
- *Iterative methodology* which, as is done by hierarchical clustering techniques, emphasizes the sequential isolation of clusters rather than a full clustering. Hence, we do not have to assume an a priori fixed number of rules.
- *Gradual localized refinements* on bad modeled zones rather than in the whole problem domain, as a regulation among extensionality and precision. This task is controlled by an expansion factor, which also acts as an overfitting foreseer.
- *Summarization by rule selection*, in order to integrate the local behavior of the hierarchically built model with the global one of the whole model, ensuring a good performance.

We should note that there are many proposals in order to answer each one of the former questions or to solve the said problems [2,27,29,39]. Some of their skills will be considered as extensions of the present methodology in future works. In the following subsections, the composition of the hierarchically built FRB and the methodology will be described in detail.

3.2. Hierarchically built fuzzy rule base

In this section we present a flexible hierarchical process to define the FRB structure – based on previous keypoints – that allows us to solve some of the lacks described in Sections 2.2 and 2.4, and consequently to improve the approximate fuzzy models performance/accuracy. The hierarchical process is based on the generation of a set of layers, each one becoming an FRB which contains fuzzy rules with a different degree of specificity or extent, i.e., fuzziness

$$\text{layer}(t) = \text{FRB}^t = \bigcup_i R_i^t$$

with FRB^t being the FRB built in iteration t formed by approximate fuzzy rules R_i^t, according to the present methodology. From now on and for the sake of simplicity, we are going to refer to the components of a FRB^t as *t-fuzzy rules*.

These *t-fuzzy rules* are organized as a hierarchy, where the order is given by increasingly more specific input subspaces, i.e., the input support extent covered by the antecedents of the approximate fuzzy rules. This can be regarded as a kind of information quantization included in the fuzzy variables of the rules, i.e., their *fuzziness*. For example, given two successive layers t and $t+1$, the fuzzy input subspace covered by a *t-fuzzy rule* is more general (larger) than the ones embraced by each one of the $(t+1)$-*fuzzy rules* derived from it [25]. From this point of view, the successive approximate fuzzy rules generated can be seen as an input subspace refinement of previous layer fuzzy rules. This structure is illustrated in Fig. 2.

As is seen, the representation takes the form of a tree, where the root represents the entire problem domain space, and the nodes represent good performance *t-fuzzy rules* (light grey rectangles), which model fuzzy input subspaces that do not require further decomposition, or more specific $(t+1)$-*fuzzy rules* that model decomposed subspaces from a bad performance *t-fuzzy rule* (black rectangles). This process is performed in an iterative way. Thus, good *t-fuzzy rules* and new generated $(t+1)$-*fuzzy rules* compose the new level of the tree generated in iteration $t+1$, i.e., layer $(t+1)$.

> How can we develop a FRB^{t+1} from a FRB^{t} in order to create the final more accurate FRB?

Each FRB^{t} is formed by a collection of approximate Mamdani-type fuzzy rules

$$R_i^t: \text{ IF } x_1 \text{ is } S_{i1}^t \text{ and } \ldots \text{ and } x_m \text{ is } S_{im}^t \text{ THEN } y \text{ is } B_i^t$$

with $x_1, \ldots, x_m$ and y being the input fuzzy variables and the output one, respectively; and with $S_{i1}^t, \ldots, S_{im}^t$, B_i^t being the fuzzy sets. As has been said, each individual fuzzy rule directly contains the meaning describing it, i.e., each variable has a fuzzy set associated as shown in Fig. 3.

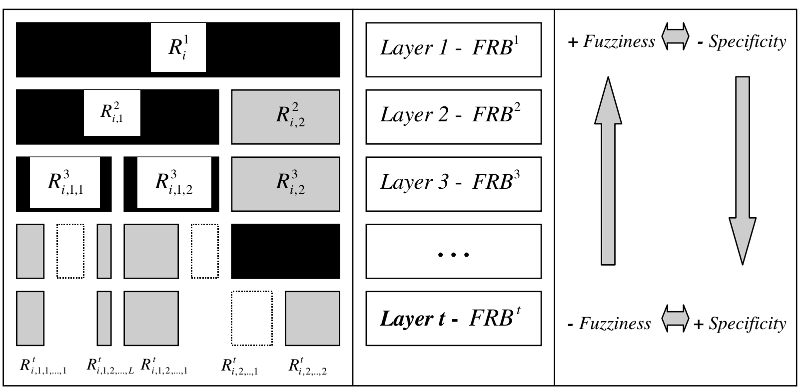

Fig. 2. Hierarchically built FRB.

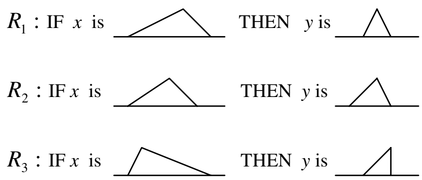

Fig. 3. Approximate fuzzy rules.

As said, the main purpose of developing a hierarchically built FRB is to model the problem space in a more accurate way. To do so, we consider the following summarizing points:

- Those *t-fuzzy rules* that model a subspace with bad performance are expanded into a set of more specific $(t+1)$-*fuzzy rules*, which become their image in *layer*$(t+1)$. This set of rules models the same subspace that the former one – but in a more local way – and replaces it

  $$R_i^t \rightarrow \{R_{i,1}^{t+1}, \ldots, R_{i,L}^{t+1}\}$$

  with L being the number of rules generated by a base FRG-method (see Fig. 2).

  We should note that not all *t-fuzzy rules* have to be expanded by the learning methodology in the present iteration t. Only those *t-fuzzy rules* which model a subspace of the problem with a significant error become the ones that are involved in this rule expansion process to build the *layer*$(t+1)$. The remaining rules preserve the same appearance that they have in *layer*(t) and are just copied to *layer*$(t+1)$.
- This expansion process is performed taking advantage of the rule construction mechanism used by the original FRG-method. Thus, the rule expansion is a dynamic process without a predefined number of rules if this condition is not imposed by the FRG-method. Although the rule construction in the expansion process is guided by the FRG-method and performed according to its own philosophy, the decision over which rule should be expanded and consequently the replacement scope is a matter of the FRB-HLM.

  An explanation for this behavior could be found in the fact that it is not always true that a set of rules with high specificity performs a better modeling of a problem than other set composed of more general fuzzy rules with more fuzziness (the chance of overfitting appears). Moreover, this is not true for all kinds of problems, and what is more, it is also not true for all fuzzy rules that model a problem [14].
- Finally, in Fig. 2 successive layers are viewed as a hierarchical clustering representation. These more specific rules, which replace the bad ones, are combined with previous good rules in a proper way, to generate a new layer. Before a new layer is finally created, a local summarization is performed in order to optimize the behavior of the corresponding FRB, leaving only

those rules (clusters, prototypes) which become nonredundant (light-grey rectangles) and with good cooperation (see Section 3.4, Step 3). Otherwise, this local treatment allows us to stop the search in any *layer*(t), which as a gradual refinement of its predecessor layer, ensures to obtain an improved even though not optimal FRB.

3.3. Fuzzy rule consequents in the FRB-HLM

As said, the FRB-HLM has been thought of as a meta-strategy to improve simple FRG-methods performance. To do so, it provides an iterative mechanism to generate a more accurate FRB. During this process, FRB-HLM generates sets of more specific $(t+1)$-*fuzzy rules* from those *t-fuzzy rules* that perform a bad modeling of the problem subspace. Let us first consider what is the meaning of a bad rule from the FRB-HLM point of view:

> a t-fuzzy rule is considered as a bad one if the performance of this rule is worst, in some measure, than the global error of the whole rule set.

As said, once a bad rule is identified, it should be replaced by more specific ones. In order to do this, we should define the scope of this hierarchical replacement process. A good point to start to find a proper replacement scope is to analyze

> what is the reason for this bad modeling performed by the bad rule?

To answer this question we will make use of previous considerations of a fuzzy rule as a prototype, which performs data (object) covering, i.e., that have some positive examples associated or belonging to it in some degree.

Having the above connection in mind, let us consider the rule expansion as a splitting process from two different points of view: the *hierarchical* and *optimization clustering* ones [19]:

- In the former approach, the points from a cluster selected to be split can go to any subclusters generated from its parent cluster in order to optimize a grouping similarity measure. In the same way, the points covered by a bad *t-fuzzy rule* could be reallocated in more specific $(t+1)$-*fuzzy rules*, which perform a better modeling of the original rule subspace. Therefore, if this replacement is caused by the need for more specificity inside the bad rule subspace, those fuzzy rules which replace it will have associated consequents restricted to the original rule subspace, i.e., constrained consequent scope (CCS).
- Otherwise, in the latter optimization approach, the points are allowed to be transferred between clusters with the only restriction of optimizing some

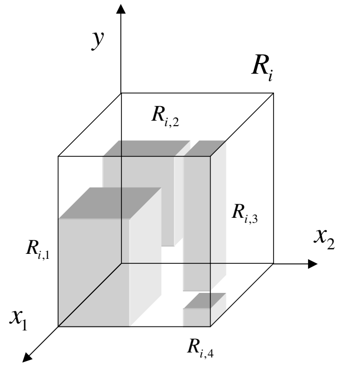

Fig. 4. Constrained consequent scope.

clustering criteria. This unrestricted approach suggests that the split rule could be better replaced by more specific rules whose consequents are not necessarily in the same subspace of the original rule, but in a neighbor or different one. Rules of this form will be called rules with an unconstrained consequent scope (UCS).

The above distinction is in agreement with the classification performed in [1] over the restrictions that the learning processes of the FRG-methods impose on the fuzzy sets in the generation of each approximate fuzzy rule:

- *Soft constrained learning*, where there are variation intervals that determine the region in which each point defining the membership functions may take value during the learning process.
- *Unconstrained learning*, where there are no restrictions imposed on the fuzzy sets, but they can lie in any region of the corresponding variable domain.

Thus, summarizing the above concepts, hierarchically built fuzzy rules can be generated in one of the following ways:

- *Constrained consequent scope*. When the scope of the membership functions of the consequent variables of the hierarchically built rules should be defined in the output subspaces of their ancestors, i.e., the consequent support sets of the expanded fuzzy rules. Fig. 4 illustrates how a *t-fuzzy rule* R_i can be replaced by its image, i.e., $(t+1)$-*fuzzy rules* $R_{i,1}, R_{i,2}, R_{i,3}, R_{i,4}$. All of these rules are completely defined in the former rule subspace.
- *Unconstrained consequent scope*. Where the only restriction imposed on the consequent membership function locations of the hierarchically built fuzzy rule is to lie in the whole domain of the system variable. Fig. 5 illustrates this scope where some of the $(t+1)$-*fuzzy rules* – $R_{i,1}, R_{i,2}$ – are defined outside the former *t-fuzzy rule* R_i output subspace, even if they are still inside its antecedent bounds.

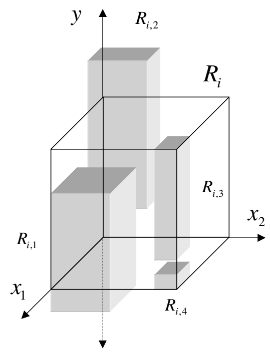

Fig. 5. Unconstrained consequent scope.

3.4. Algorithm of the basic FRB-HLM design process

In this section we present FRB-HLM as an iterative methodology to generate an FRB. To do so, we take a FRG-method as a base, which – as an inductive method – is based on the existence of a set of input–output data E_{TDS}, and a previously defined approximate fuzzy model obtained from this FRG-method. The dataset $E_{\text{TDS}} = \{e^1, \ldots, e^l, \ldots, e^q\}$ is composed of q input–output data pairs $e^l = (ex_1^l, \ldots, ex_m^l, ey^l)$, which represent the behavior of the system being modeled.

It basically consists of the following steps which may be also graphically seen in Fig. 6.

Initialization process

Step 0. FRB^1 generation process. Generate FRB^1. A FRG-method is run in order to generate the rules of the initial layer ($t = 1$)

$$\text{FRB}^1 = \text{FRG-method}(U, V, E_{\text{TDS}})$$

with U and V being the domains where the antecedents and consequents are defined, respectively, and t being the iteration counter and the last layer generated.

Iteration process (*iteration t*)

Step 1. FRB iterative generation process. *Generate* FRB^{t+1}, where the fuzzy rules from layer $(t+1)$ – FRB^{t+1} are generated taking into account FRB^t

(a) *Bad performance t-fuzzy rule selection process.* This process performs the selection of those *t-fuzzy rules* from FRB^t which will be expanded in FRB^{t+1},

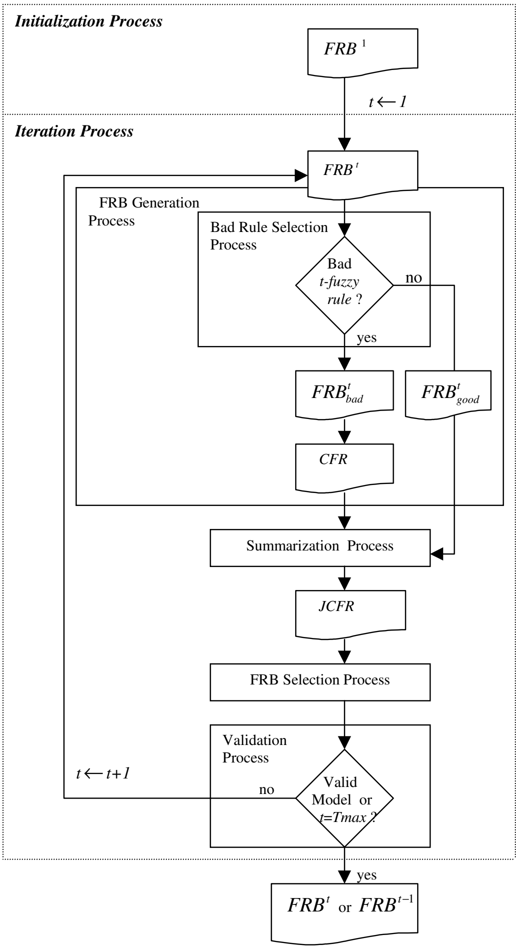

Fig. 6. FRB-HLM algorithm.

based on an error measure. This measure analyzes the accuracy of the modeling performed by each individual *t-fuzzy rule* in its definition subspace with respect to the global performance of the whole FRB. Each bad performance *t-fuzzy rule* is going to be replaced by a subset of $(t+1)$-*fuzzy rules*, which will be generated as its image. To do so, we have to follow the next steps:

(i) *Calculate the error of* FRB^t *as a whole*. Compute $\mathrm{MSE}(E_{\mathrm{TDS}}, \mathrm{FRB}^t)$. The mean square error (MSE) calculated over the training dataset E_{TDS} is the error measure used in this work. Therefore, the MSE of the entire set of *t-fuzzy rules* is represented by the following expression:

$$\mathrm{MSE}(E_{\mathrm{TDS}}, \mathrm{FRB}^t) = \frac{\sum_{e^l \in E_{\mathrm{TDS}}} (ey^l - s(ex^l))^2}{2 \cdot |E_{\mathrm{TDS}}|}$$

with $s(ex^l)$ being the output value obtained from the FRB^t, when the input variable values are $ex^l = (ex_1^l, \ldots, ex_m^l)$, and ey^l is the known desired value.

(ii) *Calculate the error of each individual t-fuzzy rule*. Compute $\mathrm{MSE}(E_i, R_i^t)$. To do so, we need to define a subset of E_{TDS}, E_i, to be used to calculate the error of the rule R_i^t. The set E_i is a set of examples matching the antecedents of the rule R_i^t to a specific degree τ:

$$E_i = \{e^l \in E_{\mathrm{TDS}} / \mathrm{Min}(\mu_{S_{i1}^t}(ex_1^l), \ldots, \mu_{S_{im}^t}(ex_m^l)) \geqslant \tau\},$$

where $\tau \in [0, 1]$. Then, we calculate the MSE of a single *t-fuzzy rule* R_i^t as

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

with $s_i(ex^l)$ being the output value obtained when inferring with R_i^t. We should note that any other local error measure can be considered with no change in our methodology, such as the one shown in [34].

(iii) *Select the t-fuzzy rules with bad performance*. Select those bad *t-fuzzy rules* which are going to be expanded, making the difference from the good ones

$$\mathrm{FRB}_{\mathrm{bad}}^t = \{R_i^t / \mathrm{MSE}(E_i, R_i^t) \geqslant \alpha \cdot \mathrm{MSE}(E_{\mathrm{TDS}}, \mathrm{FRB}^t)\},$$

$$\mathrm{FRB}_{\mathrm{good}}^t = \{R_i^t / \mathrm{MSE}(E_i, R_i^t) < \alpha \cdot \mathrm{MSE}(E_{\mathrm{TDS}}, \mathrm{FRB}^t)\}$$

with α being a threshold that represents a percentage of the error of the whole FRB, which determines the expansion of a rule. The threshold α, for example $\alpha = 1.1$, means that a *t-fuzzy rule* with an MSE a 10% higher than the MSE of the entire FRB^t should be expanded. It may be adapted in order to have more or less expanded rules. It is noteworthy that this adaptation is not linear and, as a consequence, *the expansion of more rules does not ensure the decrease of the global error of the modeled system*.

Now, for each $R_i^t \in FRB_{\mathrm{bad}}^t$:
(b) *Bad performance t-fuzzy rule expansion process*. Produce a set of L $(t+1)$-*fuzzy rules*, which are the expansion of the bad *t-fuzzy rule* R_i^t. This task is performed by a FRG-method, which takes $\mathrm{ANT}(R_i^t)$, $\mathrm{CON}(R_i^t)$ and the set of input–output data E_i as its parameters

$$\mathrm{CFR}(R_i^t) = \mathrm{FRG\text{-}method}(\mathrm{ANT}(R_i^t), \mathrm{CON}(R_i^t), E_i) = \{R_{i_1}^{t+1}, \ldots, R_{i_L}^{t+1}\}$$

with $\text{CFR}(R_i^t)$ being the image of the expanded fuzzy rule R_i^t, i.e., the *candidate fuzzy rules* to be in the FRB^{t+1} from rule R_i^t, $\text{ANT}(R_i^t)$ being the product of the support set of each antecedent fuzzy term of the rule R_i^t

$$\text{ANT}(R_i^t) = \text{supp}(S_{i1}^t) \times \cdots \times \text{supp}(S_{im}^t)$$

and $\text{CON}(R_i^t)$ being the support set of the consequents which in case of using CCS is defined as

$$\text{CON}(R_i^t) = \text{supp}(B_i^t)$$

and in case of using UCS as

$$\text{CON}(R_i^t) = V$$

with V being the domain where the consequent is defined.

Step 2. Summarization process. Obtain a joined set of candidate fuzzy rules (JCFR) performing the union of the group of the new generated $(t+1)$-*fuzzy rules* and the former good performance *t-fuzzy rules*:

$$\text{JCFR} = \text{FRB}_{\text{good}}^t \cup \left(\bigcup_i \text{CFR}(R_i^t) \right)$$

with $R_i^t \in \text{FRB}_{\text{bad}}^t$.

Step 3. FRB selection process. Simplify the set JCFR by removing the unnecessary rules from it, in order to generate an FRB^{t+1} *with good cooperation.* In this paper we consider a genetic process [11,20,22] to put this task into effect, but any other technique could be considered

$$\text{FRB}^{t+1} = \text{Selection}(\text{JCFR}).$$

In the JCFR – where there are coexisting rules generated in different layers – it may happen that a complete set of $(t+1)$-*fuzzy rules*, which replaces an expanded *t-fuzzy rule*, does not produce good results. However, a subset of this set of $(t+1)$-*fuzzy rules* may work properly, with less rules that have good cooperation between them and with the good rules from the previous layer. Thus, the JCFR set of rules generated may present redundant or unnecessary rules making the model using this FRB less accurate.

The genetic rule selection process [11,20] is based on a binary coded genetic algorithm (GA) in which the selection of the individuals is performed using the stochastic universal sampling procedure together with an elitist selection scheme, and the generation of the offspring population is put into effect by using the classical binary multipoint crossover (performed at two points) and uniform mutation operators.

The coding scheme generates fixed-length chromosomes. Considering the rules contained in JCFR counted from 1 to z, an z-bit string $C = (c_1, \ldots, c_z)$ represents a subset of rules for the FRB^{t+1}, such that

$$\text{IF } c_i = 1 \text{ THEN } (R_i \in \mathrm{FRB}^{t+1}) \text{ ELSE } (R_i \notin \mathrm{FRB}^{t+1}).$$

The initial population is generated by introducing a chromosome representing the complete previously obtained rule set, i.e., with all $c_i = 1$. The remaining chromosomes are selected at random.

As regards the fitness function $F(C_j)$, it is based on a global error measure that determines the accuracy of the FRBS encoded in the chromosome, which depends on the cooperation level of the rules existing in the JCFR. We usually work with the MSE over a training data set, as was previously defined, although other measures may be used.

Step 4. Model validation process. The final model is either accepted as proper for the given purpose or it is rejected generating another iteration of the process. Although many indexes can be used to measure the quality of linear or nonlinear systems after an identification loop [2,7], we consider a monotonic MSE measure on the training and test sets, combined with a previously defined maximum number of iterations *Tmax*, which is based on a trade-off between the complexity and the accuracy of the model generated.

This measure is computed as:

IF $(\mathrm{MSE}(\mathrm{FRB}^{t+1}(E_{\mathrm{TDS}})) \leqslant \mathrm{MSE}(\mathrm{FRB}^{t}(E_{\mathrm{TDS}}))$ and
$(\mathrm{MSE}(\mathrm{FRB}^{t+1}(E_{\mathrm{TST}})) \leqslant \mathrm{MSE}(\mathrm{FRB}^{t}(E_{\mathrm{TST}}))$ and $(t < T_{\max}))$
THEN $t \leftarrow t + 1$;
Goto Step 1
ELSE
$\mathrm{FRB}^{\mathrm{final}} = \mathrm{FRB}^{t}$ or FRB^{t+1}.

We should note that for the sake of simplicity in the present implementation, we only keep the last two layers of the FRB in order to allow the validation of the algorithm output model. Therefore, at last we only select as $\mathrm{FRB}^{\mathrm{final}}$ one of these two FRBs , i.e., FRB^{t} or FRB^{t+1}, the one with better performance.

4. Examples of application: experiments and analysis of results

With the aim of analyzing the behavior of the proposed methodology, two different FRG-methods aligned with the characteristics presented in this paper have been chosen. The first is the one proposed by Bardossy and Duckstein [3], Weighted Counting Algorithm (WCA), and the second is a well-known fuzzy clustering method based on Bezdek's work, the Fuzzy C-means (FCM) [5–7]. Both methods are briefly described in Appendix A.

In this section, FRB-HLM will be combined with the former FRG-methods to model three different applications: two- three-dimensional functions [9,11] and a real-world electrical engineering distribution problem in Spain [12,30,31].

In order to do this, we have organized this Section in three parts: a first part of notation and parameters, a second of experiments and a final one with an analysis of results for the experimental study.

4.1. Notation and parameters

For the sake of simplicity, in the following applications we are going to refer to those experiments produced by the FRB-HLM by the following notation:

$$\text{FRB-HLM}(\text{FRG-method}, \text{CS}, t),$$

where t is the number of layers or iterations with rule expansions performed by the methodology, and CS represents the type of consequent scope selected, i.e., CCS or UCS, e.g., FRB-HLM (FRG-method, CCS, 3).

Two FRG-methods, WCA and FCM, will be used for the experimentation. The WCA is considered with two different interval initializations, both based on the extraction of the support set from fuzzy partitions [1]. The first one, static initialization (S-WCA), is built based on a symmetrical and uniformly distributed fuzzy partition of three and five fuzzy terms for initial and subsequent iterations of the algorithms [13], respectively. The second one, dynamic initialization (D-WCA), is performed by the use of a fuzzy clustering-developed fuzzy partition. To do so, in this paper we use FCM combined with a validation index which measures the partition quality and iteratively detects a good number of clusters [32]. Both methods are briefly described in Appendix A and summarized in Table 1.

The general parameters used in all of these experiments are listed in Table 2.

Table 1
Summarization of WCA and FCM tasks

WCA		FCM
S-WCA	D-WCA	
Symmetrical, uniformly distributed fuzzy partition	Incremental application of FCM and validation index Cluster center projections on the antecedent domains	Incremental application of FCM and validation index Clusters projection on antecedent and consequent variable domains
Partition construction with a 0.5 cross level between adjacent fuzzy sets		
Support set extraction and intervals definition		
Rule construction by WCA philosophy		Rule construction

Table 2
Parameter values

Parameter	Decision
Generation	
α rule expansion factor	$0.5, 0.9, 1.1$
τ positive examples	$0.5, 0.3$
GA selection	
Number of generations	500, 1500
Population size	61, 81
Mutation probability	0.1, 0.21
Crossover probability	0.6

In this contribution, we will use the minimum t-norm in the role of conjunctive and implication operator and the *center of gravity weighted by the matching degree* [10] as defuzzification strategy.

The results obtained in the experiments developed are collected in tables where MSE_{tra} and MSE_{tst} stand for the values obtained in the MSE measure computed over the training and test data sets, respectively, and % indicates the percentage in which the FRG-method models in the table are improved by the hierarchically built FRB-based models. #R stands for the number of rules of the corresponding FRB.

4.2. Experiments

We will show results comparing the effectiveness of the hierarchical learning methodology with both original FRG-methods (WCA and FCM) on the said three problems. We should note that the hierarchically built FRBs are initialized with the FRG-methods output, in order to allow the former comparison. In all cases we show examples applying CCS and UCS consequent scope. Additionally, in the real-world electrical problem we will also compare the results obtained by FRB-HLM with other techniques: classical regressions, multilayer perceptron and a linguistic modeling hierarchical learning methodology HSLR-LM [13,16].

Finally, we will analyze the effect of using different expansion factors α, showing the robustness of the FRB-HLM and its role as an accuracy-complexity regulator and overfiting controller.

4.2.1. Fuzzy modeling of an intermediate complexity multimodal three-dimensional function (F_1)

The expression of the selected function is shown as follows, along with the universes of discourse considered for the variables [9]. Its graphical representation is shown in Fig. 7. As may be seen, F_1 is an intermediate complexity multimodal function whose expression is shown as follows:

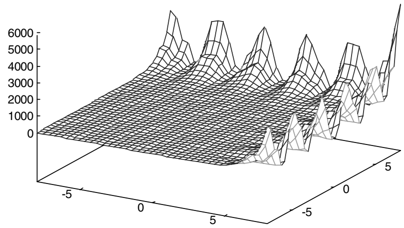

Fig. 7. Exact graphical representation of the function F_1.

$$F_1(x_1, x_2) = e^{x_1} \cdot \sin^2 x_2 + e^{x_2} \cdot \sin^2 x_1,$$
$$x_1, x_2 \in [-8, 8], \quad F_1(x_1, x_2) \in [0, 5836].$$

In order to model the F_1 function, a training data set composed of 1089 data uniformly distributed in the three-dimensional definition space has been obtained experimentally. On the other hand, another data set has been generated for its use as a test set for evaluating the performance of the design methods, avoiding any possible bias related to the data in the training set. The size of this data set is a percentage of the training set one, ten percent to be precise. The data are obtained by generating at random the state variable values in the specific universes of discourse for each one of them, and computing the associated output variable value. Hence the test set, formed by 108 data, is used to measure the accuracy of the different models designed by computing the MSE for them.

4.2.1.1. Experiments with FRG-methods. The results obtained with our FRB-HLM for WCA and FCM are shown in Table 3 and a graphical illustration of the modeling obtained can be seen in Fig. 8 (FRB-HLM (S-WCA, UCS, 3)).

Table 3
Results obtained in the fuzzy modeling of the function F_1

Method	α	$\#R$	MSE_{tra}	MSE_{tst}	$\%_{tra}$	$\%_{tst}$
S-WCA		9	244,632	257,047		
FRB-HLM (S-WCA, CCS, 3)	0.5	316	3876	6140	98.41	97.61
FRB-HLM (S-WCA, UCS, 3)	0.5	201	2406	3634	99.01	98.58
D-WCA		570	116,992	59,878		
FRB-HLM (D-WCA, CCS, 2)	0.5	718	60,142	27,947	48.59	53.32
FRB-HLM (D-WCA, UCS, 2)	0.5	838	4984	4036	95.73	93.25
FCM		6	447,584	430,713		
FRB-HLM (FCM, CCS, 1)	0.5	5	123,984	88,494	72.29	79.45
FRB-HLM (FCM, UCS, 1)	0.5	9	110,210	55,404	75.37	87.13

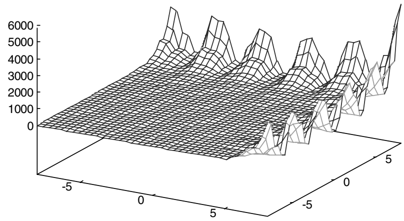

Fig. 8. F_1 modeled with 201 rules.

4.2.1.2. Experiments with different values for the expansion factor α. The results obtained with FRB-HLM with different values for the expansion factor α are shown in Table 4. We should note that we only present experiments with values of α that become representative, even though other values could have been used. Besides this, Table 4 shows experiments that have been done up to the same iterations in order to allow the comparison among different values of α. Anyway, almost all of these values can be overcome in more iterations.

4.2.2. Fuzzy modeling of a very complex multimodal three-dimensional function (F_2)

In the following, we present a very complex multimodal three-dimensional function, F_2 [9,11]. Its graphical representation is shown in Fig. 9. Its expression, along with the universes of discourse considered for the variables, is also shown as follows:

Table 4
Results obtained in the fuzzy modeling of the function F_1 considering different values of the expansion factor α

α	CCS			UCS		
	#R	MSE_{tra}	MSE_{tst}	#R	MSE_{tra}	MSE_{tra}
(a) S-WCA						
0.5	316	3876	6140	201	2406	3634
0.9	214	5012	6248	239	3897	3123
1.1	218	3780	5979	168	3614	4036
(b) D-WCA						
0.5	718	60,142	27,944	838	4984	4036
0.9	424	73,408	79,082	668	27,476	39,463
1.1	371	94,682	69,347	571	28,771	43,508
(c) FCM						
0.5	5	123,984	88,494	9	110,210	55,404
0.9	5	123,984	88,494	9	110,210	55,404
1.1	4	112,871	90,068	9	110,210	55,404

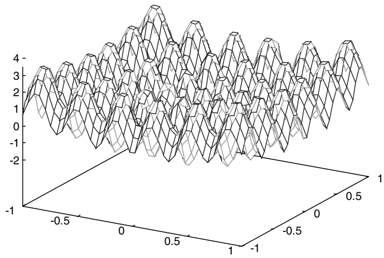

Fig. 9. Exact graphical representation of the function F_2.

$$F_2(x_1, x_2) = x_1^2 + x_2^2 - \cos(18x_1) - \cos(18x_2),$$
$$x_1, x_2 \in [-1, 1], \quad F_2(x_1, x_2) \in [-2, 3.5231].$$

The second function, F_2, has been modeled using a training data set composed of 1681 data uniformly distributed in the three-dimensional definition space. A test set of 167 data, generated in the same way that was done in function F_1, was selected for evaluating the performance of the design methods.

4.2.2.1. Experiments with FRG-methods. The results obtained with our FRB-HLM for WCA and FCM methods are shown in Table 5 and also illustrated in Fig. 10 (FRB-HLM (S-WCA, CCS, 3)).

4.2.2.2. Experiments with different values for the expansion factor α. The results obtained with our FRB-HLM with different values for the expansion factor α are shown in Tables 6 and 7. The assumptions made in the previous experiment remain for the present one. Empty boxes mean that the value of α is too high to expand rules.

Table 5
Results obtained in the fuzzy modeling of the function F_2

Method	α	$\#R$	MSE_{tra}	MSE_{tst}	$\%_{tra}$	$\%_{tst}$
S-WCA		9	0.580	0.660		
FRB-HLM (S-WCA, CCS, 3)	0.9	486	0.106	0.129	81.72	80.45
FRB-HLM (S-WCA, UCS, 3)	0.9	379	0.178	0.195	69.31	70.45
D-WCA		49	0.516	0.578		
FRB-HLM (D-WCA, CCS, 2)	0.5	1657	0.094	0.084	81.78	85.46
FRB-HLM (D-WCA, UCS, 2)	0.5	2296	0.073	0.072	85.85	87.54
FCM		7	0.553	0.579		
FRB-HLM (FCM, CCS, 4)	0.5	146	0.324	0.331	41.41	42.83
FRB-HLM (FCM, UCS, 3)	0.5	15	0.506	0.541	8.49	6.56

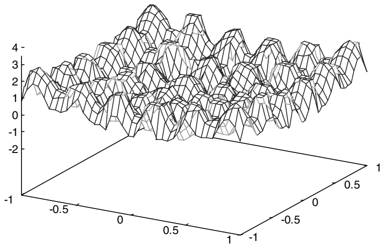

Fig. 10. F_2 modeled with 486 rules.

Table 6
Results obtained in the fuzzy modeling of the function F_2 considering different values of the expansion factor α

α	CCS			UCS		
	#R	MSE_{tra}	MSE_{tst}	#R	MSE_{tra}	MSE_{tra}
(a) S-WCA						
0.5	710	0.100	0.121	699	0.171	0.195
0.9	486	0.106	0.129	379	0.178	0.195
1.1						
(b) D-WCA						
0.5	2296	0.094	0.084	1657	0.073	0.072
0.9	697	0.163	0.169	587	0.138	0.184
1.1	213	0.297	0.299	210	0.290	0.275
(c) FCM						
0.5	127	0.374	0.388	15	0.506	0.541
0.9	66	0.385	0.421			
1.1						

Table 7
Results obtained in the fuzzy modeling of the function F_2 comparing results with two different values of α and a different number of iterations

Method	α	#R	MSE_{tra}	MSE_{tst}
FRB-HLM (D-WCA, CCS, 2)	0.5	1657	0.094	0.084
FRB-HLM (D-WCA, CCS, 4)	1.1	750	0.129	0.133

4.2.3. The electrical engineering distribution problems

Sometimes, there is a need to measure the amount of electricity lines that an electric company owns. This measurement may be useful for several aspects such us the estimation of the maintenance costs of the network, which was the main goal of the problem presented in Spain [12,31]. High and medium voltage lines can be easily measured, but low voltage line is contained in cities and

Table 8
Notation considered for the problem variables

Symbol	Meaning
x_1	Number of clients in population
x_2	Radius of ith population in the sample
y	Line length, population i

villages, and it would be very expensive to measure it. This kind of line used to be very convoluted and, in some cases, one company may serve more than 10,000 small nuclei. An indirect method to determine the length of line is needed.

Therefore, a relationship must be found between some characteristics of the population and the length of line installed on it, making use of some known data, that may be employed to predict the real length of line in any other village. We will try to solve this problem by generating different kinds of models determining the unknown relationship: fuzzy, classical regression and neural models. To do so, 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, considered as the radius of population i in a sample of 495 rural nuclei [30,31]. Our variables are named as shown in Table 8.

To design the different models we have randomly divided the sample into two sets comprising 396 and 99 samples, labeled training and test, respectively.

4.2.3.1. Experiments with different predefined types of FRG-methods. The results obtained with our FRB-HLM with the two said FRG-methods are shown in Table 9.

4.2.3.2. Experiments with different values for the expansion factor α. The results obtained with our FRB-HLM with different values for the expansion factor α

Table 9
Results obtained in the low voltage electrical application

Method	α	$\#R$	MSE_{tra}	MSE_{tst}	$\%_{tra}$	$\%_{tst}$
S-WCA		9	777,306	717,472		
FRB-HLM (S-WCA, CCS, 2)	0.5	60	154,109	184,178	80.17	74.32
FRB-HLM (S-WCA, UCS, 2)	0.5	40	158,879	186,819	79.56	73.96
D-WCA		25	192,818	202,095		
FRB-HLM (D-WCA, CCS, 1)	1.1	115	97,187	144,865	49.59	28.31
FRB-HLM (D-WCA, UCS, 1)	1.1	136	87,675	146,155	54.52	27.68
FCM		5	508,426	464,130		
FRB-HLM (FCM, CCS, 4)	0.5	27	181,196	158,312	64.36	55.01
FRB-HLM (FCM, UCS, 2)	0.9	8	208,777	171,379	58.93	63.07

Table 10
Results obtained in the low voltage electrical application considering different values of the expansion factor α

α	CCS			UCS		
	$\#R$	MSE_{tra}	MSE_{tst}	$\#R$	MSE_{tra}	MSE_{tra}
(a) S-WCA						
0.5	60	154,109	184,178	40	158,879	186,819
0.9	38	186,343	207,585	39	168,988	189,795
1.1	23	359,169	320,839	31	228,388	235,460
(b) D-WCA						
0.5	205	80,840	206,961	209	70,903	158,572
0.9	132	87,091	163,687	151	78,857	169,366
1.1	115	97,187	144,865	136	87,675	146,155
(c) FCM						
0.5	14	191,441	181,375	11	211,719	177,138
0.9	10	205,221	189,842	8	208,777	171,379
1.1	7	229,559	206,363	8	231,280	207,120

are shown in Table 10. The assumptions made in previous experiments remain for the present one.

4.2.3.3. Experiment comparing models from FRB-HLM with other techniques. Once we have analyzed the behavior of the fuzzy models designed individually, we are going to compare their accuracy with the remaining techniques considered. Table 11 shows the results obtained by them and the best ones obtained by our FRB-HLM as well. To apply classical regression, the parameters of the polynomial models were fit by Levenberg–Marquardt, while exponential and linear models were fit by linear least squares. The multilayer perceptron was trained with the QuickPropagation algorithm. The number of neurons in the hidden layer was chosen to minimize the test error [12,31]. We also compare the obtained results with a linguistic model obtained by means of a hierarchical approach described in [13,15,16], i.e., a hierarchical system of linguistic rules

Table 11
Results obtained in the low voltage electrical application compared with other techniques

Method	MSE_{tra}	MSE_{tst}	Complexity
Linear	287,775	209,656	7 nodes, 2 par.
Exponential	232,743	197,004	7 nodes, 2 par
Second-order polynomial	235,948	203,232	25 nodes, 2 par.
Third-order polynomial	235,934	202,991	49 nodes, 2 par.
Three-layer perceptron 2-25-1	169,399	167,092	102 par.
HSLR-LM	154,411	156,197	25 rules
FRB-HLM (D-WCA, UCS, 1)	97,187	144,865	115 rules
FRB-HLM (D-WCA, CCS, 1)	87,675	146,155	136 rules

learning methodology (HSLR-LM) combined with Wang and Mendel's rule generation method [33].

4.3. Analysis of results

In view of the results obtained in the above experiments, we should remark some important conclusions:

From the accuracy point of view. The different models generated from our process clearly outperform the ones of the original FRG-methods in all problems, even if they perform a bad initial approximation (see Tables 3, 5 and 9). Moreover, the most accurate FRB-HLM models outperform classical regression, neural network and hierarchical linguistic models in the approximation of both data sets, training and test, in the real-world electrical problem (see Table 11).

On the one hand, we should note that a FRG-method with known or informed premises like WCA accomplishes a good covering of the examples. This fact gives later more freedom to choose subsequent hierarchical partitions without leaving uncovered examples. On the other hand, less informed techniques with an unknown structure like FCM, perform good approximations in complex problems (Tables 5 and 9), but sometimes the expansion process is early stopped (Table 3) because of the fixed condition of the factor of expansion α and the high fuzziness of the fuzzy rules identified. An adaptive α could be an option to choose in order to solve this problem.

Finally, we should not forget that these results could not escape from the initial models generated by the FRG-methods and their performance. Thus, good initial conditions usually drive faster to good solutions, as can be seen comparing different types of initial FRG-methods results and their respective hierarchically built models. Therefore, right-side percentages of the tables are better estimators of the benefits of our approach.

From the complexity point of view. This work is oriented to obtain very accurate models, thus making them to become complex in comparison with the simple initial ones, which are refined by the FRB-HLM. In spite of this, we can deal with this point by different ways.

On the one hand, the iterative character of our methodology allows us to have intermediate solutions, each one producing a hierarchically built FRB which improves the previous one but increases its complexity. On the other hand, the factor of expansion α works as a regulator among accuracy and complexity, as shown in Tables 4, 6, and 10. There, all results become improvements to the ones of the original FRG-method but with different number of rules. In this sense, we should highlight the results of Table 10 for WCA in a complex real world problem, where the best results are obtained with a high value of α, i.e., less expansion of rules. This fact prevents the system to become overfitted. In view of these results, we empirically corroborate what we previously introduced in Section 4.2, taking note that it is not always true that a fuzzy model composed of more specific fuzzy rules, and consequently with a larger number of rules, models better a problem than a less complex one [13,14].

In this real world electrical problem, the fuzzy models generated from FRB-HLM with the D-WCA FRG-methods are quite complex, but we should note that they clearly overcome the numeric models in the resulting test and training errors (see Table 11).

As we have stated previously, our main purpose is to perform gradual refinements of a system on bad modeled subspaces, trying to preserve as much as possible its original model and without causing overfitting. Table 7 is an example which shows how a combination of a high expansion factor ($\alpha = 1.1$) and more iterations (4) performs good results with less than a half of the rules generated by a lower expansion model ($\alpha = 0.5$).

Finally, let us give some final considerations about the different FRG-methods used in this work:

- We can observe that S-WCA and FCM become the less complex combination of FRG-methods and FRB-HLM. The former because of its simple initialization and minimum subsequent partitions [13], and the latter because of its projection-oriented partition.
- Those methods which perform dynamic rule identification, i.e., D-WCA, follow the data direction by clustering the space of the rule without a fixed number of initial and subsequent partitions (not a minimum number as in S-WCA). They produce more locally learned rules, i.e., more options to choose a good combination of them when bad rules have to be replaced. This better approximation could produce overfitting as seen in Table 9.
- The use of CCS and UCS gives the methods more locally freedom degrees to select the best consequent. The latter ones, which search in a wider space, increase the computational cost of the learning process. They seem to give accurate results when redundant information exists as with D-WCA, but produce rules with more fuzziness when rule components are identified by projection on the variable space, like FCM. [1] In practice, the apparent differences between CCS and UCS have been found not to be too severe. CCS usually obtains good solutions, if it is provided with a generous α.

5. Concluding remarks

In this paper, fuzzy models are viewed as a class of local modeling approaches, which attempt to solve a complex modeling problem by

[1] Sometimes this fact produces an earlier stop of the algorithm, which can be treated with an adaptive value of α.

decomposing it into a number of simpler subproblems. Fuzzy set theory offers an excellent tool for representing the uncertainty associated with the decomposition task, for providing smooth transitions between the individual local submodels, and for integrating various types of knowledge within a common framework. From this perspective, fuzzy modeling can be regarded as a search for a decomposition of a nonlinear system, which gives a desired balance between the complexity and the accuracy of the model, effectively exploring the fact that the complexity of systems is usually not uniform.

In this view, we presented FRB-HLM. Its gradual refinement, which performs localized refinements on bad modeled zones rather than in the whole problem domain, allows a regulation among extensionality and precision of the system modeled. The combination of the former tasks with an iterative process emphasizes the sequential isolation of clusters rather than a full clustering. Moreover, the combination with a dynamic rule expansion, which depends on the FRG-methods used, allows us not to have to assume a fixed number of rules. Finally, a global summarization rule selection process integrates the good local behavior of the hierarchically built model with the whole model, ensuring a good global performance.

Acknowledgements

This research has been supported by CICYT projects TIC96-0778 and PB98-1319.

Appendix A

A.1. Weighted counting algorithm

Here, we briefly describe Bárdossy and Duckstein's method, which is explained in detail in [3]. It requires a previous definition of the fuzzy set supports and the number of rules. This information in our case is obtained from preliminary fuzzy partitions of the antecedents in the case of static identification [1], and by a fuzzy clustering method FCM combined with Sugeno's validation index in the dynamic case. Only those rules whose subspace is formed by the examples contained in these supports will be considered.

- The supports (a_{ik}^-, a_{ik}^+) of the fuzzy sets A_{ik} belonging to the antecedents of the rules in the FRB are defined previously.
- The shape of the k antecedent fuzzy sets A_{ik} – which are triangular fuzzy sets defined, respectively, by the parameters $(a_{ik}^-, a_{ik}^1, a_{ik}^+)$ – is calculated with a_{ik}^1 being

$$a_{ik}^1 = \frac{1}{|E_i|} \sum_{e^l \in E_i} ex_k^l,$$

where

$$E_i = \left\{ e^l = (ex_1^l, \ldots, ex_m^l, ey^l) \in E_{\mathrm{TDS}} / ex_k^l \in (a_{ik}^-, a_{ik}^+), \quad \forall k \in \{1, \ldots, m\} \right\}.$$

- The matching degree of the example $e^l \in E_i$ with the antecedent of the rule R_i is computed as $h(e^l, R_i) = T(\mu_{A_{i1}}(ex_1^l), \ldots, \mu_{A_{im}}(ex_m^l))$, with T being a t-norm and $\mu_{A_{ik}}(ex_k^l)$ being the membership degree of the value ex_k^l to the fuzzy set A_{ik}.
- The shape of the consequent B_i is determined using the examples contained in a new subset E_{ϕ_i} formed by the examples of E_i that match the antecedents of the rule R_i to a degree greater than or equal to $\phi \in (0, 1]$ i.e., $E_{\phi_i} = \{e^j \in E_i | h(e^j, R_i) \geqslant \phi\}$. Such a consequent is a triangular fuzzy set defined by (b_i^-, b_i^1, b_i^+)

$$b_i^- = \min_{e^j \in E_{\phi_i}} ey^j, \quad b_i^1 = \frac{\sum_{e^l \in E_{\phi_i}} h(e^l, R_i) \cdot ey^l}{\sum_{e^l \in E_{\phi_i}} h(e^l, R_i)}, \quad b_i^+ = \max_{e^l \in E_{\phi_i}} ey^l.$$

A.2. Fuzzy C-means clustering algorithm

Perhaps this method developed by Bezdek [7] is the best known and most widely used fuzzy clustering algorithm. FCM is an iterative optimization algorithm that minimizes the cost function

$$J_m = \sum_{i=1}^{c} \sum_{l=1}^{n} (u_{il})^m \cdot (\|e^l - v_i\|)^2.$$

To do so it follows the next steps:

- Initialize:

  Object data: $E_{\mathrm{TDS}} = \{e^1, \ldots, e^l, \ldots, e^n\}$ composed of n input–output data pairs $e^l = (ex_1^l, \ldots, ex_r^l, ey^l)$
  Termination threshold: $\varepsilon > 0$
  Initial prototypes: $V_0 = (v_1, \ldots, v_c) \in R^{cp}$
  Number of clusters: c
  Maximum number of iterations: T
  Weighting exponent: $m > 1$
  Iterate while $(t \leqslant T)$ and $(\|V_t - V_{t-1}\| \geqslant \varepsilon)$
  ◦ Calculate U_t with V_{t-1}

$$u_{il} = \left[\sum_{j=1}^{c} \left(\frac{\|e^l - v_i\|}{\|e^l - v_j\|} \right)^{2/(m-1)} \right]^{-1} \quad \forall i, l.$$

∘ Update V_{t-1} to V_t with U_t

$$v_i = \left[\frac{\sum_{l=1}^{n} (u_{il})^m \cdot e^l}{\sum_{l=1}^{n} (u_{ik})^m} \right] \forall i.$$

∘ Get V_t.

A.3. Sugeno's model validation index

This index serves as an estimator of the cluster partition quality. The number of clusters, c, is determined so that $S(c)$ reaches a minimum as c increases (usually a local minimum)[32]:

$$S(c) = \sum_{l=1}^{n} \sum_{i=1}^{c} (\mu_{ij})^m \left(\left\| e^l = v_i \right\|^2 - \|v_i - \hat{e}\|^2 \right),$$

where n is the number of data to be clustered, c is the number of clusters $\geqslant 2$, e^l is the lth data vector, $\hat{e}$ is the average of data; v_i is the center of ith cluster, μ_{ij} is the grade of jth data belonging to the ith cluster, m is the adjustable weight, $\|\cdot\|$ is the norm.

The first term is the variance of the data in a cluster and the second term is that of the clusters themselves. Therefore, the optimal clustering is considered to minimize the variance in each cluster and to maximize the variance between the clusters.

A.4. Cluster projection on the variable domains

In order to estimate the membership functions of the fuzzy variables of a rule determined by a fuzzy cluster, we project the degree of membership of its members on the variable axis [2,32]. To do so, we consider triangular membership functions and a linear regression from the cluster center (x_0, y_0) in the variable domain, with the aim of obtaining the corresponding triangular extremes. To perform this calculus only those positive examples to a certain degree (ε) are considered.

Having a set of examples: $E_{\text{TDS}} = \{e^1, \ldots, e^l, \ldots, e^n\}$ composed of n input-output data pairs $e^l = (ex_1^l, \ldots, ex_r^l, ey^l)$ and a variable A_k, we will identify its membership function from the ith cluster.

Consider a linear equation in x and y of the form:

$$y = a \cdot x + b$$

and the ith cluster center of the form (x_0, y_0), where $y_0 = \mu_i(x_0)$, then we get the constants

$$a = \frac{\sum_{l=1}^{n} ex_k^l \cdot \mu - x_0 \cdot \sum_{l=1}^{n} \mu_i(ex_k^l) - y_0 \cdot \sum_{l=1}^{n} ex_k^l + x_0 \cdot y_0 \cdot n}{\sum_{l=1}^{n} (ex_k^l)^2 - 2 \cdot x_0 \cdot \sum_{l=1}^{n} ex_k^l + n \cdot (x_0)^2},$$

$$b = y_0 - x_0 \cdot a,$$

where n represents the amount of left or right positive examples – according to the extreme which is being calculated

$$^{+}n = (\mu_i(ex^l_j) \geqslant \varepsilon) \ \& \ (ex^l_j \leqslant x_0),$$

$$n^{+} = (\mu_i(ex^l_j) \geqslant \varepsilon) \ \& \ (ex^l_j > x_0).$$

Finally, we get the extreme (left or right) of the triangular membership function as

$$x_k = -\frac{b}{a}.$$

Then the triangular membership function of variable A_k is determined by the points (x^-_k, x_0, x^+_k).

References

[1] R. Alcalá, J. Casillas, O. Cordón, F. Herrera, Approximate Mamdani-type fuzzy rule-based systems: features and taxonomy of learning methods, Technical Report #DECSAI-990117, Department of Computer Science and Artificial Intelligence, E.T.S. de Ingeniería Informática, University of Granada, Spain, 1999.

[2] R. Babuška, Fuzzy Modeling for Control, Kluwer Academic Publishers, Dordrecht, 1998.

[3] A. Bardossy, L. Duckstein, Fuzzy Rule-Based Modeling with Application to Geophysical, Biological and Engineering Systems, CRC Press, Boca Raton, 1995.

[4] 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.

[5] J.C. Bezdek, Fuzzy Mathematics in Pattern Classification, Ph.D. thesis, Cornell University, 1973.

[6] J.C. Bezdek, S. Pal (Eds.), Fuzzy Models for Pattern Recognition: Methods that Search for Structures in Data, IEEE Press, New York, 1992.

[7] J.C. Bezdek, Fuzzy clustering, in: E.H. Ruspini, P.P. Bonisone, W. Pedrycz (Eds.), Handbook of Fuzzy Computation, Institute of Physics Press, 1998.

[8] B. Carse, T.C. Fogarty, A. Munro, Evolving fuzzy rule based controllers using genetic algorithms, Fuzzy Sets and Systems 80 (1996) 273–294.

[9] O. Cordón, F. Herrera, Hybridizing genetic algorithms and evolutionary strategies to design approximate fuzzy rule-based systems, Fuzzy Sets and Systems 118 (2) (2001) 235–255.

[10] O. Cordón, F. Herrera, A. Peregrín, Applicability of the fuzzy operators in the design of fuzzy logic controllers, Fuzzy Sets and Systems 86 (1997) 15–41.

[11] O. Cordón, F. Herrera, 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.

[12] O. Cordón, F. Herrera, L. Sánchez, Solving electrical distribution problems using hybrid evolutionary data analysis techniques, Applied Intelligence 10 (1999) 5–24.

[13] O. Cordón, F. Herrera, I. Zwir, Linguistic modeling by hierarchical systems of linguistic rules, Technical Report #DECSAI-99114, Department of Computer Science and Artificial Intelligence, E.T.S. de Ingeniería Informática, University of Granada, Spain, 1999.

[14] O. Cordón, F. Herrera, P. Villar, Analysis and guidelines to obtain a good uniform fuzzy partition granularity for FRBSs using simulated annealing, International Journal of Approximate Reasoning 25 (3) (2000) 187–215.
[15] O. Cordón, F. Herrera, I. Zwir, A hierarchical knowledge-based environment for linguistic modeling: models and methodology, #DECSAI-000106, Department of Computer Science and Artificial Intelligence, University of Granada, Spain, 2000.
[16] O. Cordón, F. Herrera, I. Zwir, Hierarchical knowledge bases for fuzzy rule-based systems, in: Proceedings of the 8th Conference on Information Processing and Management of Uncertainty in Knowledge-based Systems (IPMU), Madrid, Spain, 2000, pp. 1770–1777.
[17] M. Delgado, A.F. Gómez-Skarmeta, A. Vila, On the use of hierarchical clustering in fuzzy modeling, International Journal of Approximate Reasoning 14 (1996) 237–257.
[18] M. Delgado, A.F. Gómez-Skarmeta, F. Marín, A fuzzy clustering-based rapid prototyping for fuzzy rule-based modeling, IEEE Transactions on Fuzzy Systems 5 (2) (1997) 223–232.
[19] D.J. Hand, Discrimination and Classification, Wiley, New York, 1992.
[20] F. Herrera, M. Lozano, J.L. Verdegay, A learning process for fuzzy control rules using genetic algorithms, Fuzzy Sets and Systems 100 (1998) 143–158.
[21] K. Hirota (Ed.), Industrial Applications of Fuzzy Technology, Springer, Berlin, 1993.
[22] H. Ishibuchi, K. Nozaki, N. Yamamoto, H. Tanaka, Selecting fuzzy if–then rules for classification problems using genetic algorithms, IEEE Transactions on Fuzzy Systems 3 (3) (1995) 260–270.
[23] R. Krishnapuram, J. Keller, A possibilistic approach to clustering, IEEE Transactions on Fuzzy Systems (1993) 98–110.
[24] C.T. Leondes (Ed.), Fuzzy Theory Systems, Techniques and Applications, Academic Press, New York, 2000.
[25] T. Mitchell, Machine Learning, McGraw-Hill, New York, 1997.
[26] W. Pedrycz (Ed.), Fuzzy Modelling: Paradigms and Practice, Kluwer Academic Press, Dordrecht, 1996.
[27] W. Pedrycz, A.V. Vasilakos, Linguistic models and linguistic modeling, IEEE Transactions on Systems, Man, and Cybernetics 29 (6) (1999).
[28] E.H. Ruspini, A new approach to clustering, Information and Control 15 (1) (1969) 22–32.
[29] E.H. Ruspini, I.S. Zwir, Automated qualitative description of measurements, in: Proceedings of the 16th IEEE Instrumentation and Measurement Technology Conference, Venice, Italy, 1999.
[30] L. Sánchez, Study of the Asturias rural and urban low voltage network, Technical Report, Hidroeléctrica del Cantábrico Research and Development Department (in spanish), Asturias, Spain, 1997.
[31] L. Sánchez, Interval-valued GA-P algorithms, IEEE Transactions on Evolutionary Computation 4 (1) (2000) 64–72.
[32] M. Sugeno, T. Yasukawa, A fuzzy-logic-based approach to qualitative modeling, IEEE Transactions on Fuzzy Systems 1 (1) (1993) 7–31.
[33] L.X. Wang, J.M. Mendel, Generating fuzzy rules by learning from examples, IEEE Transactions on Systems, Man, and Cybernetics 22 (1992) 1414–1427.
[34] J. Yen, L. Wang, C. Wayne 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.
[35] L.A. Zadeh, Fuzzy sets, Information and Control 8 (1965) 338–353.
[36] L.A. Zadeh, The concept of a linguistic variable and its application to approximate reasoning, Information Science Part I 8 (1975) 199–249.
[37] L.A. Zadeh, The concept of a linguistic variable and its application to approximate reasoning, Information Science (Part II) 8 (1975) 301–357.

[38] L.A. Zadeh, The concept of a linguistic variable and its application to approximate reasoning, Information Science (Part III) 9 (1975) 43–80.
[39] L.A. Zadeh, Toward a theory of information granulation and its centrality in human reasoning and fuzzy logic, Fuzzy Sets and Systems 90 (1997) 111–127.
[40] I. Zwir, E.H. Ruspini, Qualitative object description: initial reports of the exploration of the frontier, in: Proceedings of the EUROFUSE-SIC '99, Budapest, Hungary, 1999, pp. 485–490.