Building Fuzzy Graphs: Features and Taxonomy of Learning for Non-Grid-Oriented Fuzzy Rule-Based Systems*

Rafael Alcalá[a], Jorge Casillas[b,†], Oscar Cordón[b], Francisco Herrera[b]

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

[b] *Department of Computer Science and Artificial Intelligence*
University of Granada, E-18071 Granada, Spain
{casillas,ocordon,herrera}@decsai.ugr.es

Abstract

The use of Mamdani-type fuzzy rule-based systems (FRBSs) allows us to deal with the modeling of systems building a linguistic model clearly interpretable by human beings. However, the accuracy obtained is not sometimes as good as desired. This fact relates to the restriction imposed when using linguistic variables, which forces the membership functions considered in each fuzzy linguistic rule to belong to a common set of them, i.e., to use a global grid.

To solve this problem, in the last few years a new variant has been proposed working directly with fuzzy variables in the fuzzy rules instead of linguistic terms, thus ignoring the said restriction. Therefore, these systems, which are totally equivalent to fuzzy graphs (defined by Zadeh as granular representations of functional dependencies and relations), do not consider a global grid and could be named *non-grid-oriented* (NGO) FRBSs. Of course, the main objective of these models is the accuracy of the system instead its interpretability.

Until now, NGO FRBSs have been little considered and developed in the literature. However, and due to their good accuracy, their use is increasing thus making necessary a wide analysis on the features and associated learning methods in the NGO domain. This contribution aims at analyzing the structure and framework of NGO FRBSs, as well as making a taxonomy of learning methods considering the constrains imposed on the fuzzy sets in the generation process. Some automatic learning techniques and methods proposed in the literature to build these fuzzy graphs will be also reviewed and analyzed when solving several applications of different nature.

Keywords: fuzzy graphs, fuzzy modeling, non-grid-oriented fuzzy rule-based systems, accuracy improvement, learning

*This research was supported by the Spanish CICYT Project PB98-1319

†Corresponding author. Tel.: +34 958 242376, fax: +34 958 243317, website: http://decsai.ugr.es/∼casillas/

1 Introduction

At present, one of the most important areas for the application of fuzzy set theory 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. The most usual application of FRBSs is *system modeling* [5, 76], which in this field may be considered as an approach used to model a system making use of a descriptive language based on fuzzy logic with fuzzy predicates [88].

There are basically two different kinds of FRBSs in the literature, the well-known Mamdani-Assilian (MA) [70] and Takagi-Sugeno-Kang (TSK) [87, 89] ones, depending on the expression of the consequent of the fuzzy rules. The main advantage of the MA FRBS is that it provides a natural framework to include expert knowledge in the form of fuzzy rules. On the contrary, though the TSK FRBS is easier to design, the form of the rule consequents causes the system not to constitute a natural framework to represent expert knowledge.

Although the MA FRBS presents the maximum description level, sometimes it is not as accurate as desired. To address this problem, a new variant of these kinds of systems has been recently proposed where the system accuracy is more preferable than its interpretability [5, 7, 12, 19, 22, 24, 62, 86]. These systems, that could be named as *non-grid-oriented* (NGO) FRBSs due to its structure, are characterized by working directly with fuzzy variables—instead of linguistic ones—in the fuzzy rules, i.e., each rule has its own semantic instead of considering a set of linguistic terms as MA FRBSs do.

Until now, NGO FRBSs have been little considered and developed in the literature. However, due to their good accuracy, their use is increasing thus making necessary a wide analysis on the features and associated learning methods in the NGO domain. With the purpose of formalizing such aspects, this contribution aims at analyzing the NGO FRBS features. A taxonomy of learning methods will be also performed considering different kinds of constrains imposed on the fuzzy sets in the generation process. Some automatic learning methods and techniques proposed in the specialized literature will be reviewed. Finally, their behavior will be tested under different requirements.

To do so, the paper is organized as follows. Section 2 places the scope of the contribution by introducing different approaches for modeling with FRBS and presenting the analyzed way to improve the accuracy of the obtained models. Section 3 presents the justification, rule structure, and the pros and cons of NGO FRBSs. Section 4 focuses on NGO FRBS learning approaches, categorizing them in different kinds according to the constrains imposed on the fuzzy sets during the generation process. Section 5 presents and analyzes results obtained when applying the said methods to some problems with different characteristics. Finally, Section 6 shows several concluding remarks.

2 From Interpretability to Accuracy: From Linguistic Rules to Fuzzy Rules

Fuzzy modeling (FM) (i.e., system modeling with FRBSs) usually comes with two contradictory requirements to the obtained model: the *interpretability*, capability to express the behavior of the real system in an understandable way, and the *accuracy*, capability to faithfully represent the real system.

Of course, the ideal thing would be to satisfy both criteria to a high degree but, since they are contradictory issues, it is generally not possible. In this case, more priority is given to one of them (defined by the problem nature), leaving the other one in the background.

This section introduces an extreme situation where the accuracy of fuzzy models is mainly considered. Firstly, several approaches to find different interpretability and accuracy degrees

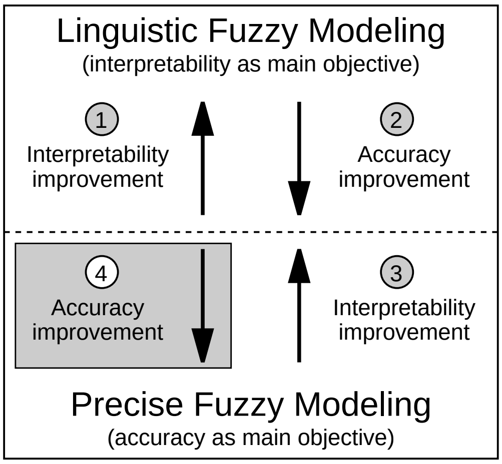


Figure 1: Improvements of interpretability and accuracy in fuzzy modeling

are shown. Then, a different, more flexible rule structure is presented as a way to improve the accuracy.

2.1 From Interpretability to Accuracy

Two FM approaches arise depending on the main objective to be considered:

- *Linguistic FM*, mainly developed by MA FRBSs [69, 70], is focused on the interpretability.
- *Precise FM*, mainly developed by TSK FRBSs [87, 89], is focused on the accuracy

Regardless of the approach, a common scheme is found in the existing literature to perform the FM:

1. Firstly, the main objective (interpretability or accuracy) is tackled defining a specific model structure to be used, thus setting the FM approach.
2. Then, the modeling components (model structure and/or modeling process) are improved by means of different mechanisms to define the desired ratio interpretability-accuracy.

As Figure 1 shows, this procedure results in four different possibilities:

1. *Linguistic FM with improved interpretability* — To achieve this approach, different methods for selecting input variables [13, 42, 54, 59, 64, 68, 84], reducing the fuzzy rule set [26, 55, 56, 58, 63, 92], using more descriptive rule expressions [18, 40, 67], or performing linguistic approximation [35, 71, 88] have been proposed.
2. *Linguistic FM with improved accuracy* — This approach has been performed by learning/tuning the membership functions by defining their shapes [11, 24, 47, 53, 59, 60, 61, 65, 66, 74], their types [83], or their context [29, 44, 45, 67, 77], learning the granularity of the fuzzy partitions [32, 36, 78, 79], or extending the model structure by using linguistic modifiers [14, 20, 41], weights [1, 17, 57, 75, 81, 96], or hierarchical architectures [33, 58], among others.
3. *Precise FM with improved interpretability* — This approach is usually developed by reducing the fuzzy rule set [73, 93, 94], reducing the number of fuzzy sets (with the subsequent merging of rules) [37, 80, 82], or exploiting the local description of the rules [9, 38, 95].

4. *Precise FM with improved accuracy* — This approach involves the use of mechanisms that allow the fuzzy model to be more accurate by making more flexible its structure.

As we can notice, recent research works are being focused on the three first approaches as an attempt to find a better balance between interpretability and accuracy [15]. On the contrary, the fourth possibility is not sufficiently considered by the current literature.

However, this approach seems to have a significant relevance in FM since it exploits the accuracy capability preserving a minimum of interpretability derived by the use of rules and fuzzy logic to express the behavior of the model. That point differs from other modeling techniques such as neural networks.

The following subsection shows a way of improving the accuracy of FM within this four possibility.

2.2 From Linguistic Rules to Fuzzy Rules

MA FRBSs are formed by a set of fuzzy rules, which in MISO (multiple-input single-output) systems take the following form:

$$R_i = \textbf{IF } X \textit{ is } \mathcal{A}_i \textbf{ THEN } Y \textit{ is } B_i$$

with $X = (X_1, \ldots, X_n)$ and Y being the input and output linguistic variables [100], and with $\mathcal{A}_i = A_{i1} \times \ldots \times A_{in}$ and B_i being respectively the cartesian product of the input linguistic labels and the output linguistic label involved in the rule R_i. Therefore, with the use of linguistic variables, each linguistic label has associated a fuzzy set defining its meaning.

This structure is simply a representation of imprecisely defined functional dependencies and relations that tries to model the behavior of a system. Therefore, as a set of fuzzy rules, it may be represented by the concept of *fuzzy graphs* introduced by Zadeh in [97] and developed by himself in [99, 101] or, more recently, in [103].

A fuzzy graph is built by a collection of *fuzzy points*—which may be interpreted as the said fuzzy rules—that represents a functional dependency f^* of Y on X and it is defined as follows:

$$f^* = \mathcal{A}_1 \times B_1 + \ldots + \mathcal{A}_r \times B_r, \text{ or, more compactly, } f^* = \sum_{i=1}^{r} \mathcal{A}_i \times B_i.$$

Nevertheless, the rule structure used in MA FRBSs has an important restriction that makes them be only a subset of the collection of possible functional dependencies covered by fuzzy graphs because of the use of linguistic variables, which forces the membership functions considered in each fuzzy rule to belong to a global set of them, i.e., a global grid is used. However, in a fuzzy graph, the individual fuzzy points (or fuzzy rules) do not depend on a common set of membership functions, instead, each fuzzy point is described through an individual set of membership functions A_{ij}. This makes fuzzy graphs without restrictions have the ability of approaching in a higher degree to the relational representation modeling the problem. Figure 2 illustrates this difference.

Therefore, basing on the fourth FM trend mentioned in the previous section, the accuracy of the models can be improved by regarding this second more flexible structure. To facilitate a clear distinction between both concepts, in this contribution we will note as *linguistic rule* those rules that are composed of linguistic variable with a set of associated global linguistic terms and as *fuzzy rule* those rules that directly contains fuzzy variables with associated independent fuzzy sets. Figure 3 clarifies this distinction.

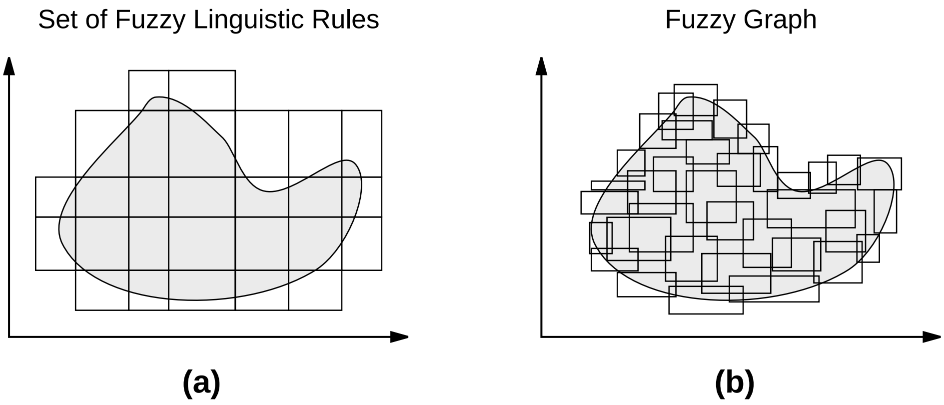


Figure 2: Approximate representation of relations: (a) considering a set of fuzzy linguistic rules, (b) considering a fuzzy graph without restrictions

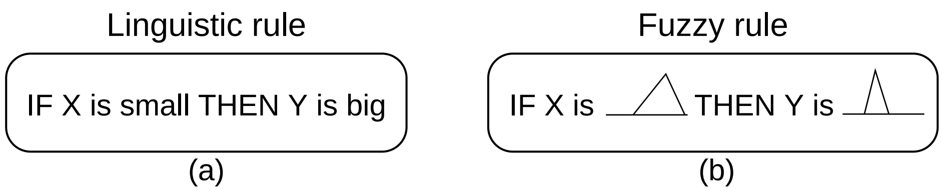


Figure 3: (a) While in a linguistic rule the variables take linguistic terms as values, (b) in a fuzzy rule the variables take fuzzy sets as values

3 Non-Grid-Oriented FRBSs

This section is devoted to justify the use of NGO FRBSs to solve the drawbacks presented by MA FRBSs. After that, the structure and the pros and cons of these new kinds of FRBSs will be introduced.

3.1 Why Non-Grid-Oriented FRBSs?

As Zadeh pointed out in his *Principle of Incompatibility* [98], "as the complexity of a system increases, our ability to make precise and yet significant statements about its behavior diminishes [. . .]." Therefore, although the use of MA FRBSs allows us to deal with the modeling of systems in which a certain degree of imprecision is involved, building models clearly interpretable by human beings, the accuracy obtained is not sometimes as good as desired. This lack of accuracy is due to some problems related to the structure of the linguistic rules considered [6, 12]:

1. There is a lack of flexibility in the FRBS because of the rigid partitioning of the input and output spaces.

2. When the system input variables are dependent themselves, it is very hard to fuzzy partition the input spaces.

3. The homogeneous 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.

4. The size (complexity) of the model directly depends on the number of variables and linguistic terms in the system. The obtaining of an accurate FRBS requires a significant granularity amount, i.e., it needs of the creation of new linguistic terms. This granularity increase causes the number of rules to rise significantly, which may make the system lose the capability of being interpretable by human beings. Moreover, in the great majority of the cases, it would be possible to obtain an equivalent FRBS having a very lesser number of rules if there would not exist that input space rigid partitioning.

To address these problems, a new variant of these kinds of systems has been recently proposed [5, 7, 12, 19, 22, 24, 62, 86]. These systems, based on the fuzzy graph concept without the consideration of global-grid restrictions, could be named as NGO (non-grid-oriented) FRBSs and they are characterized by working directly with fuzzy variables—instead of linguistic ones—in the fuzzy rules. Of course, in these systems the accuracy is preferred to their interpretability. Opposite to them, MA FRBSs could be named *linguistic* FRBSs since linguistic variables are considered. In Figure 4, we may observe the existing parallelism between linguistic and NGO FRBSs.

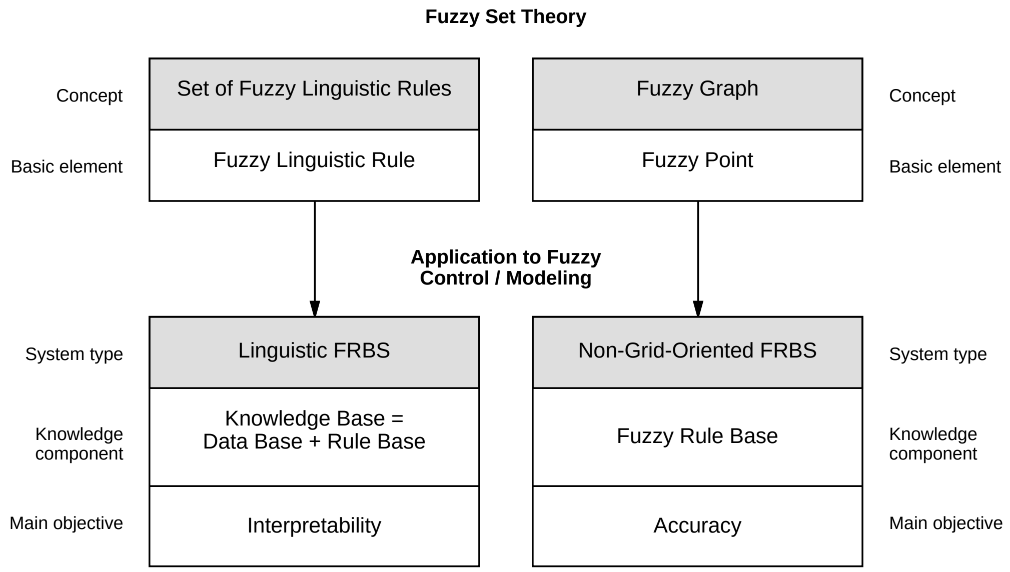


Figure 4: Parallelism between linguistic FRBS (obtained from a set of fuzzy linguistic rules) and NGO FRBS (obtained from a fuzzy graph without restrictions)

Other equivalent designations have been proposed by different authors to differentiate between these two kinds of FRBSs. Among other, for NGO or linguistic FRBSs we can respectively find

- *approximate* or *descriptive* FRBSs [22, 24] because the main objective of NGO FRBSs is the approximation while the one of linguistic FRBSs is the interpretability,
- FRBSs with *local* or *global fuzzy sets* [12] because the semantic (i.e., the definitions of membership functions) considered in NGO FRBSs is particular for each rule while a common semantic for the whole set of rules is considered by linguistic FRBSs,
- *rule-based* or *domain-based* FRBSs [19] because the membership functions are defined for each rule in NGO FRBSs while a common domain and fuzzy partition are defined in linguistic FRBSs, or

- *scatter-partitioning* or *grid-partitioning* FRBSs [39] because no homogeneous partition of the space is considered by NGO FRBSs while a partition following a grid is considered by linguistic FRBSs.

3.2 Non-Grid-Oriented FRBSs Structure

The structure of a fuzzy rule (or fuzzy point) considered in an NGO FRBS is the following:

$$R_i = \textbf{IF}\ X\ is\ \widehat{\mathcal{A}}_i\ \textbf{THEN}\ Y\ is\ \widehat{B}_i$$

with $\widehat{\mathcal{A}}_i = \widehat{A}_{i1} \times \ldots \times \widehat{A}_{in}$ and where the only change with respect to the rule structure considered in linguistic FRBSs is the fact that the inputs (X_i) and output (Y) are fuzzy variables instead of linguistic ones and, therefore, $\widehat{A}_{ij}$ and $\widehat{B}_i$ are fuzzy sets—without a direct interpretation—instead of linguistic labels.

In contrast to linguistic FBRSs, where the knowledge is stored in the data base (definitions of the fuzzy sets associated to the linguistic terms) and the rule base (formed by the fuzzy linguistic IF-THEN rules themselves), NGO FRBSs have a single fuzzy rule base (FRB) formed by a set of rules presenting the said structure, where each individual rule directly contains the meaning describing it. Figure 5 graphically shows this.

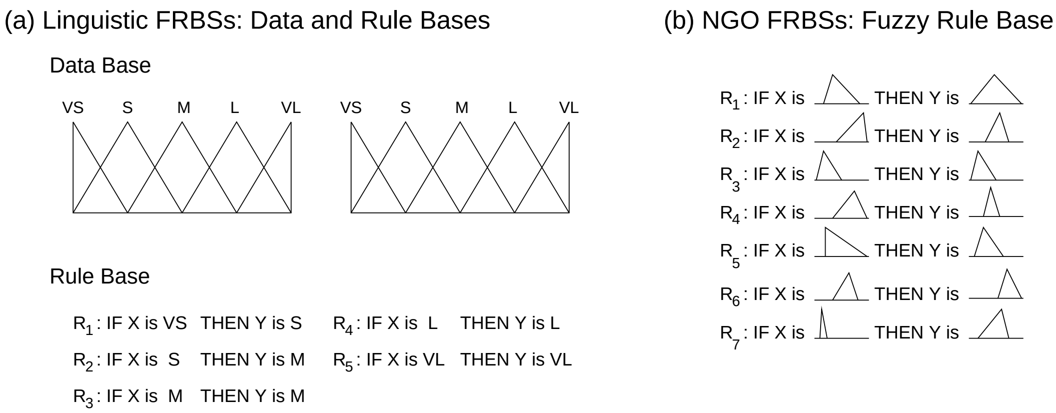


Figure 5: Graphical comparison between the structures used in a linguistic and an NGO FRBS to store the knowledge

3.3 Advantages and Drawbacks of Non-Grid-Oriented FRBSs

NGO FRBSs have some interesting advantages that make them be very suitable in many cases.

- The main advantage of the NGO approach is its expressive power to take in rules that present their own specificity in terms of the fuzzy sets involved in them, introducing thus additional degrees of freedom in the system [12].

- Another important advantage is the fact that 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 when it is necessary. This is likely to be of benefit in tackling the curse of dimensionality when scaling to multi-dimensional systems [12, 103].

These facts allow NGO FRBSs to be more accurate than linguistic ones in complex problems. Nevertheless, they have some drawbacks:

- Its main drawback with respect to the linguistic FRBS is the loss of FRB readability since there is not a global interpretation of the variables considered. However, NGO FRBSs can locally describe the system behavior, unlike other kinds of less descriptive models like neural networks (which are difficult to analyze—what makes difficult to improve their capacity—because the acquired knowledge is implicitly expressed). Therefore, we can consider that NGO FRBSs are halfway between the clear interpretability of linguistic FRBSs and the hidden reasoning of non descriptive models.

- The capability to approximate can cause an excessive specificity with bad generalization, thus obtaining an undesired overlearning.

While linguistic FRBSs are appropriated for linguistic fuzzy modeling [88], due to the fact that each rule constitutes a description of a condition-action statement that may be clearly interpreted by human beings, the main application of NGO FRBSs is the area of precise fuzzy modeling [5], where the model accuracy is the main requirement instead of its description ability.

According to this, we can assert that linguistic and NGO FRBSs are not incompatible but complementary. Therefore, depending on the problem characteristics and our intentions, we should use one or another. NGO FRBSs are recommended when we rather lose interpretability in exchange for improving the accuracy.

4 Learning Non-Grid-Oriented FRBSs

In order to design an NGO FRBS (i.e., to build a fuzzy graph) for a specific application, the main task that has to be performed is to derive an appropriate FRB about the problem being solved. The accuracy of the FRBS in solving the problem will directly depend on this task. Due to the complexity of the FRB learning, some automatic techniques have been proposed to put it into effect.

Many of these techniques are collected under the name of Soft Computing [10, 102]. Soft Computing is a new field of Computer Science that deals with the integration of problem-solving techniques such as Fuzzy Logic, Neural Networks, or GAs, among many others. Each of these techniques provides us with complementary reasoning and search methods to solve complex problems. Among all the possible combinations, we are interested on how they can help us to design an NGO FRBS by defining the FRB.

4.1 Taxonomy of the Learning Process

A taxonomy of the learning process and a brief description of the considered techniques and analyzed methods will be presented in the following.

Depending on whether the learning process imposes restrictions on the fuzzy sets in the generation of each fuzzy rule or not, we can distinguish between methods that perform a *constrained learning* (CL) and those that perform an *unconstrained learning* (UL):

- A learning process is *constrained* when there are restrictions that compel the fuzzy sets to lie in some concrete intervals during the learning stage. Within this kind, we can distinguish between *hard constrained learning* (HCL) and *soft constrained learning* (SCL).
  - In the former, the *hard* one, there are variation intervals that determine the region in which each point defining the membership functions may take value during the learning process.
  - In the latter, the *soft* one, the only restriction imposed on the membership function locations and shapes is to lie in a concrete interval and to be meaningful.

- On the contrary, when the are not restrictions imposed on the fuzzy sets, but they can lie in any region of the corresponding variable domain, the learning process is called *unconstrained*.

Intuitively, we can observe that the said learning kinds are related, and we can consider UL as the most extreme case of CL where the interval of performance associated to each fuzzy set corresponds to the whole domain of the system variable. In the same way, SCL can be considered a particular case of HCL where the variation intervals of the membership function points are an only common interval. To avoid confusing these terms, from now on the learning process will be classified according to the most specific definition.

Usually, the restriction information is obtained from preliminary fuzzy partitions of the problem variables. In this way, the variation intervals (for CL) can be calculated from the initial membership function parameters.

The said classification, constrained and unconstrained learning, is graphically shown in Figure 6.

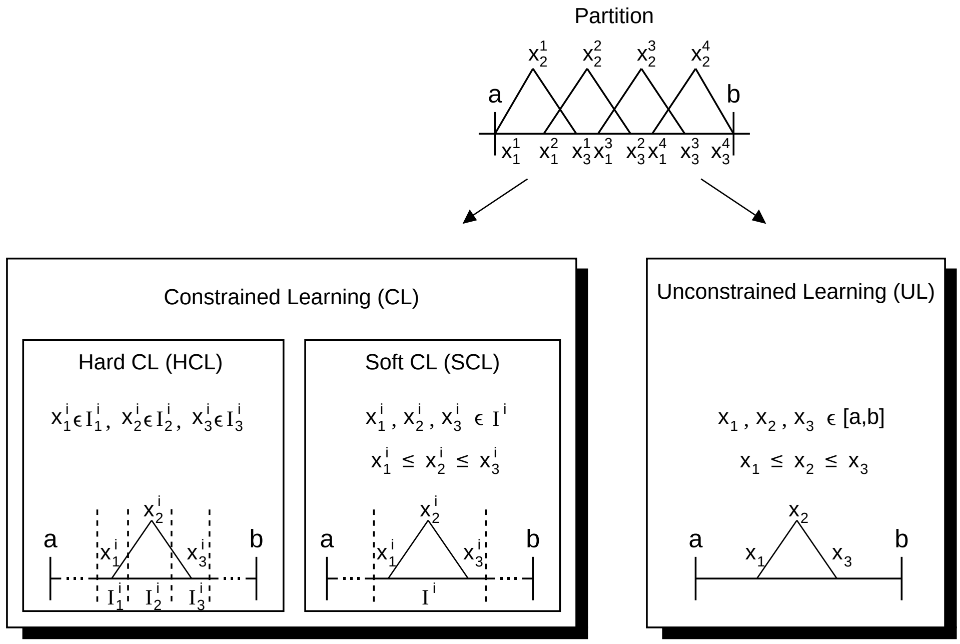


Figure 6: Constrained and unconstrained learning

The degree of freedom is the main feature that characterizes the different learning kinds introduced. While UL is totally free to determine the fuzzy sets along the whole universe of discourse, HCL generates them in confined spaces. Clearly, there is another aspect related to this property, the search space. The higher the degree of freedom is, the higher the search space is. A greater freedom allows the obtained FRBS to benefit from the high approximation capability of NGO FRBSs, however, the complexity for accomplishing the learning process grows due to the huge search space tackled. The SCL approach performs the best balance between both aspects: generation of fuzzy sets with a high degree of freedom and reduction of the search space. This trade-off could provide better accuracy degree. Figure 7 shows the different model flexibility and learning simplicity degrees attained by the linguistic FRBS and each learning approach in the NGO FRBS and the hypothetical accuracy obtained depending on them.

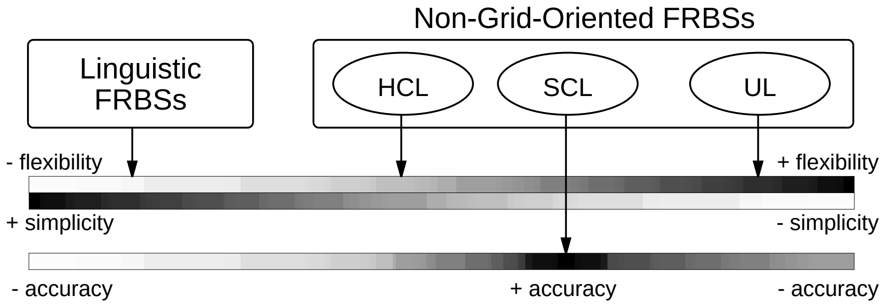


Figure 7: Hypothetical accuracy obtained by the different approaches depending on the model flexibility and learning simplicity

4.2 Learning Techniques and Methods for Non-Grid-Oriented FRBSs

This section focuses on the different approaches to the learning of NGO FRBSs considering the main techniques used and some specific methods with different characteristics. Table 1 summarizes them classified according to the said aspects.

Table 1: Analyzed Learning Methods

Ref.	**Author(s)**	**Method**	**Technique**	**Learning**
[5]	Bárdossy, Duckstein	WCA	AHDD	HCL/Ant – UL/Cons
[34, 8]	Dunn, *extended by* Bezdek	FCM	Clustering	UL
[16]	Chiu	CEB	Clustering	UL
[12]	Carse, Fogarty, Munro	P-FCS1	GA	UL
[49]	Herrera, Lozano, Verdegay	MOGUL-GLP	GA	UL
[24]	Cordón, Herrera	MOGUL-SCL	GA	SCL
[27]	Cordón, Herrera	MOGUL-HCL	GA	HCL

A HDD = Ad Hoc Data-Driven, GA = Genetic Algorithm, HCL = Hard Constrained Learning, SCL = Soft Constrained Learning, UL = Unconstrained Learning, Ant = Antecedent, Cons = Consequent

A brief description of the considered techniques and the analyzed methods will be introduced in the following subsections.

4.2.1 Ad Hoc Data-Driven Methods

Under this name, *ad hoc data-driven methods*, we are collecting those methods based on processes where the learning of the fuzzy rules is guided by covering criteria of the data in the example set. The way to address this process can not be included in any well-known optimization or searching technique but it is specifically developed for this purpose. Usually, they are characterized by being methods based on a short time-consuming iterative procedure. A specific method for NGO FRBS included in this family is introduced in the following subsection.

4.2.1.1 Bárdossy and Duckstein's Method (WCA)

The Weighted Counting Algorithm (WCA) [5] is based on the principle of firstly generating the antecedents of the rule and then obtaining the consequents taking the examples in the training data set matching the antecedents of the rule to any degree as a base.

A previous definition of the antecedent fuzzy set supports for each rule, as well as the number of rules, is required. This information can be obtained from preliminary fuzzy partitions of the antecedents. Thus, the supports used in the rule antecedents are defined by the combination of the fuzzy partition terms. Only those rules whose subspace formed by these supports contains examples will be considered.

Supposing that the universes of discourse associated to the system variables are continuous, the algorithm generates the fuzzy set shapes used for the antecedents, A_{ik}—$i = 1, \ldots, N$; $k = 1, \ldots, n$; with N being the number of rules and n being the number of input variables—, and subsequently identifies the corresponding consequents. The algorithm is the following:

1. 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.
2. Each rule $R_i = IF\ x_1\ is\ A_{i1}\ and \ldots and\ x_n\ is\ A_{in}\ THEN\ y\ is\ B_i$ is generated as follows:
   (a) 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:
   $$a_{ik}^1 = \frac{1}{\mid E_i \mid} \sum_{e^j \in E_i} ex_k^j,$$
   where $E_i = \{e^j = (ex_1^j, \ldots, ex_n^j, ey^j) \in E_p \ \mid\ ex_k^j \in (a_{ik}^-, a_{ik}^+), \forall k \in \{1, \ldots, n\}\}$.
   (b) The matching degree of the example $e^j \in E_i$, $j = 1, \ldots, \mid E_i \mid$, with the antecedent of the rule R_i is computed as $h(e^j, R_i) = T(\mu_{A_{i1}}(ex_1^j), \ldots, \mu_{A_{in}}(ex_n^j))$, with T being a t-norm and $\mu_{A_{ik}}(ex_k^j)$ being the membership degree of the value ex_k^j to the fuzzy set A_{ik}.
   (c) The shape of the rule 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 or equal to $\phi \in (0, 1]$, i.e., $E_{\phi_i} = \{e^j \in E_i \ \mid\ h(e^j, R_i) \geq \phi\}$. Such a consequent is a triangular fuzzy set defined by (b_i^-, b_i^1, b_i^+):
   $$b_i^- = \min_{e^l \in E_{\phi_i}} ey^l \ , \qquad b_i^1 = \frac{\sum_{e^l \in E_{\phi_i}} h(e_l, R_i) \cdot ey^l}{\sum_{e^l \in E_{\phi_i}} h(e_l, R_i)} \ , \qquad b_i^+ = \max_{e^l \in E_{\phi_i}} ey^l,$$

The value of ϕ has to be selected so that for each rule a sufficient number of elements of the training set is considered. The higher ϕ is, the fewer elements are used to define the consequents, and the crisper the assessed rules are.

Since the left and right points of the triangular fuzzy sets of the rule antecedent are previously defined, with the central points being calculated between both extremes, WCA follows the HCL approach in the antecedent. It is a particular case, even more strict, where $a_{ik}^- \in [a_{ik}^-, a_{ik}^-]$, $a_{ik}^+ \in [a_{ik}^+, a_{ik}^+]$, and $a_{ik}^1 \in [a_{ik}^-, a_{ik}^+]$. However, the way to learn the consequent is different, since it does not depend on any prior definition but it is calculated from the example set. Therefore, the algorithm learns the consequents according to the UL approach.

4.2.2 Clustering Fuzzy Rule-Based Systems

Fuzzy clustering is a technique used to group a set of data into clusters such that elements within the same cluster have a high degree of similarity, while elements belonging to different clusters have a high degree of dissimilarity. A good introduction to fuzzy clustering techniques can be found in [2, 8].

Fuzzy clusters can be used to determine a partition of a space of an individual variable or of a collection of them. In the first case, fuzzy sets are constructed whilst in the second one the clustering method yields fuzzy relations. In this sense, a cluster center represents a characteristic behavior of the system. Therefore, rules that describe these behaviors can be derived from cluster centers.

Hence, fuzzy clustering methods can design NGO FRBSs by directly deriving their FRB. To do so, the process is clearly divided into two totally independent stages:

- *Cluster generation*: In the first stage, the cluster centers are obtained by means of a fuzzy clustering method.
- *NGO model identification*: After that, an easier second process translates these cluster centers into NGO fuzzy rules.

There is no doubt that the cluster generation is the most important and interesting stage from the fuzzy modeling point of view, since the system description is accomplished here. Therefore, this section is focused on the first stage introducing two concrete clustering methods. These methods can be combined with any model identification algorithm.

The way to obtain NGO FRBS by clustering techniques is characterized by not imposing constraints on the fuzzy sets involved in the fuzzy rules, so that these kinds of learning algorithms follow the UL approach.

Since the second stage is common to all clustering FRBSs methods, before introducing the two said cluster generation methods, a specific algorithm for the NGO model identification will be proposed.

4.2.2.1 NGO Model Identification from Cluster Centers

Many different processes can be used as NGO model identification but perhaps the simplest and most common one to obtain triangular fuzzy sets is the following:

Let n be the number of input and output variables, $v_c = (v_{c1}, \ldots, v_{cj}, \ldots, v_{cn})$ be the c-th cluster center—with $c \in \{1, \ldots, N_c\}$, and N_c being the number of clusters—, $x_i = (x_{i1}, \ldots, x_{ij}, \ldots, x_{in})$ be the i-th example—with $i \in \{1, \ldots, N_e\}$, and N_e being the number of examples—, and $\mu_c(x_i)$ be the membership degree of the example x_i to the cluster c.

1. Let $X^c = \{x_i^c \mid \mu_c(x_i^c) \geq \mu_k(x_i^c), \forall k \in \{1, \ldots, N_c\}\}$ be the set of examples that belong to the cluster represented by v_c, i.e., those examples whose closer cluster center is v_c.
2. Let l^{cj} and r^{cj} be the patterns of $X^c \cup \{v_c\}$ with the least and greatest value in the variable j respectively:
$$l^{cj} = \begin{cases} v_c & \text{if } v_{cj} < x_{kj}^c \\ x_i^c \mid x_{ij}^c \leq x_{kj}^c & \text{otherwise} \end{cases}$$
$$r^{cj} = \begin{cases} v_c & \text{if } v_{cj} > x_{kj}^c \\ x_i^c \mid x_{ij}^c \geq x_{kj}^c & \text{otherwise} \end{cases},$$
with $j \in \{1, \ldots, n\}$, $i \in \{1, \ldots, \mid X^c \mid\}$, and $k = 1, \ldots, \mid X^c \mid$.
3. The triangular fuzzy set, (a_{cj}, b_{cj}, d_{cj}), corresponding to the j-th variable of the c-th rule is obtained as follows:
$$a_{cj} = \begin{cases} v_{cj} & \text{if } l^{cj} = v_c \\ l_j^{cj} - \frac{\mu_c(l^{cj})(v_{cj} - l_j^{cj})}{1 - \mu_c(l^{cj})} & \text{otherwise} \end{cases}$$
$$b_{cj} = v_{cj}$$
$$d_{cj} = \begin{cases} v_{cj} & \text{if } r^{cj} = v_c \\ r_j^{cj} + \frac{\mu_c(r^{cj})(r_j^{cj} - v_{cj})}{1 - \mu_c(r^{cj})} & \text{otherwise} \end{cases},$$

with l_j^{cj} and r_j^{cj} being the values of the j-th variables of the patterns l^{cj} and r^{cj} respectively. If $a_{cj} = b_{cj} = d_{cj}$ for some $j \in \{1, \ldots, n\}$, then reject the c-th rule.

We should remark that it is also possible to generate other types of fuzzy sets, for example with trapezoidal membership functions, that perhaps fit better to the generated clusters. Nevertheless, we have chosen triangular fuzzy sets to homogenize the clustering methods with the rest of methods analyzed in this contribution.

4.2.2.2 Method developed by Dunn and extended by Bezdek (FCM)

Perhaps the best known and most widely used clustering algorithm is the Fuzzy C-Means (FCM) one developed by Dunn [34] and extended by Bezdek [8]. The FCM algorithm is an iterative optimization algorithm that minimizes the cost function

$$J = \sum_{c=1}^{N_c} \sum_{i=1}^{N_e} \Big(\mu_c(x_i)^m \cdot \|x_i - v_c\|^2\Big),$$

where N_c is the number of clusters, N_e is the number of data points, x_i is the i-th data point, v_c is the c-th cluster center, $\mu_c(x_i)$ is the membership degree of the i-th data to the c-th cluster, and $m > 1$ is a constant (typically $m = 2$).

The degree of membership $\mu_c(x_i)$ is defined by

$$\mu_c(x_i) = \frac{1}{\displaystyle\sum_{k=1}^{N_c} \left(\frac{\|x_i - v_c\|}{\|x_i - v_k\|}\right)^{2/(m-1)}}.$$

Given the desired number of clusters N_c, an initial guess for each cluster center v_c ($c = 1, \ldots, N_c$), and a value for m, the FCM algorithm iteratively recomputes new cluster centers v_c using the expression

$$v_c = \frac{\displaystyle\sum_{i=1}^{N_e} (\mu_c(x_i)^m \cdot x_i)}{\displaystyle\sum_{i=1}^{N_e} \mu_c(x_i)^m},$$

and subsequently updates the membership values from these cluster centers. The iterative process stops when it has converged under some criterion, usually

$$\max_{c,i} \mid \mu_c(x_i) - \acute{\mu_c}(x_i) \mid \leq \epsilon,$$

where $\acute{\mu_c}(x_i)$ is the previous value and ϵ is a predefined accuracy level.

The FCM algorithm is based on iteratively minimizing the distance between the elements in each of the clusters while maximizing the distance between the cluster prototypes. It has been shown that the algorithm converges, even with bizarre initial classification assumptions [8].

4.2.2.3 Chiu's Method (CEB)

The Cluster Estimation-Based (CEB) algorithm [16] determines the number of clusters and does not need a previous initialization as the above presented FCM algorithm. Although the original method was proposed to generate TSK FRBSs, we will adapt it to NGO FRBSs by using the model identification method shown in Section 4.2.2.1.

Let us consider a collection of N_e data points—$x_1, \ldots, x_i, \ldots, x_{N_e}$—in an n-dimensional space. Let us assume that the data points have been normalized in each dimension so that their coordinate ranges in each dimension are equal, i.e, the data points are bounded by a hypercube.

The CEB algorithm works as follows:

1. Compute the potential measure, P_i, of each data point, x_i, by the formula

$$P_i = \sum_{j=1}^{N_e} \mathbf{e}^{-\alpha \|x_i - x_j\|^2},$$

where $\alpha = \frac{4}{r_a^2}$ and r_a is a positive constant. A data point with many neighbor data points will have a high potential value.

2. $k \leftarrow 1$. The data point with the highest potential is selected as the first cluster center. Let x_k^* be the location of the first cluster center and P_k^* be its potential value. This point is deleted from the data set.

3. Update the potential of the remaining points by the formula

$$P_i \leftarrow P_i - P_k^* \mathbf{e}^{-\beta \|x_i - x_k^*\|^2},$$

where $\beta = \frac{4}{r_b^2}$ and r_b is a positive constant.

4. $k \leftarrow k + 1$.

5. If $(P_k^* > \bar{\epsilon} \cdot P_1^*)$ then accept x_k^* as the center of the next cluster and go to 3.

   Else, if $(P_k^* < \underline{\epsilon} \cdot P_1^*)$ then reject x_k^* and end the algorithm.

   Otherwise, let d_{min} be the shortest of the distances between x_k^* and all the previously found cluster centers. If $\left(\frac{d_{min}}{r_a} + \frac{P_k^*}{P_1^*} \geq 1\right)$ then accept x_k^* as a cluster center and go to 3. Else, reject x_k^*, set P_k^* to zero, and select the data point with the next highest potential as the new x_k^* and go to 5.

The constant r_a is the radius defining a neighborhood. Data points outside this radius have a little influence on the potential. The constant r_b is the radius defining the neighborhood that will have measurable reductions in potential. To avoid obtaining closely spaced cluster centers, r_b should be greater than r_a; Chiu proposes to set $r_b = 1.5 \cdot r_a$. On the other hand, $\bar{\epsilon}$ specifies a threshold for the potential above which we will definitively accept the data point as a cluster center; $\underline{\epsilon}$ specifies a threshold below which we will definitively reject the data point. Chiu proposes $\bar{\epsilon} = 0.5$ and $\underline{\epsilon} = 0.15$. If the potential falls in the gray region, the algorithm checks if the data point provides a good trade-off between having a sufficient potential and being sufficiently far from existing cluster centers.

4.2.3 Genetic Fuzzy Rule-Based Systems

GAs are a family of computational models inspired by the evolution theory. These algorithms encode potential solutions to a specific problem on simple chromosome-like data structures forming a population. By means of crossover and mutation operators, the information is shared and changed to explore and exploit solutions. Recombination operators are applied to these structures so as to preserve critical information. Three points are the keys of a genetic process: the population of potential solutions, the pair evolution operators/code, and the performance index. For a wide introduction, refer to [72].

Although GAs are not learning algorithms, they may offer a powerful and domain-independent search method for a variety of learning tasks. In fact, there has been a good deal of interest in using GAs for machine learning problems [43]. Recently, numerous papers and applications combining fuzzy concepts and GAs have appeared, and there is an increasing concern about the integration of these two topics. In particular, a great number of publications explore the use of GAs for designing FRBSs. These approaches receive the general name of *Genetic Fuzzy Rule-Based Systems* (GFRBSs) [22, 28].

The automatic design of FRBSs can be considered in many cases as an optimization or search process on the space of potential solutions. GAs are the best known and most widely used global search technique with an ability to explore and exploit a given operating space using available performance measures. Moreover, the generic code structure and independent performance features of GAs make them suitable candidates to incorporate prior knowledge (fuzzy membership function parameters, fuzzy rules, number of rules, etc.). Over the last few years, these advantages have extended the use of GAs in the development of a wide range of approaches for designing FRBSs [12, 19, 24, 25, 27, 49].

Three alternative approaches have been proposed to apply GAs to learning processes: the *Michigan* [52], the *Pittsburgh* [85], and the *Iterative Rule Learning* (IRL) [90] approaches. In the first one, the chromosomes correspond to classifier rules that are evolved as a whole, whereas in the Pittsburgh approach, each chromosome encodes a complete set of classifiers. In the IRL approach each chromosome represents a single rule, but contrary to the first, only the best individual is considered as the solution, discarding the remaining chromosomes in the population.

Different GFRBS proposals can be found in [28, 46, 50, 51]. Some specific approaches for NGO FRBSs will be introduced in the following subsections.

3.2.3.1 Carse, Fogarty, and Munro's Method (P-FCS1)

The P-FCS1 (Pittsburgh-style Fuzzy Classifier System #1), proposed in [12], is a novel GFRBS characterized by a very interesting coding scheme and crossover operator. Its main features are the following:

- **Representation**. The rule representation employed by this method is a set of terms (x^C_{ik}, x^W_{ik}) that encode the centers and widths of fuzzy set membership functions over the range of input and output variables using a real coding scheme. A chromosome is a variable length concatenated string of such fuzzy rules. That is, r NGO fuzzy rules with n input variables and m output variables will be encoded as

$$\begin{array}{c}(x^C_{1,1}, x^W_{1,1}) \ldots (x^C_{1,n}, x^W_{1,n})(x^C_{1,n+1}, x^W_{1,n+1}) \ldots (x^C_{1,n+m}, x^W_{1,n+m})\\ \vdots \\ (x^C_{r,1}, x^W_{r,1}) \ldots (x^C_{r,n}, x^W_{r,n})(x^C_{r,n+1}, x^W_{r,n+1}) \ldots (x^C_{r,n+m}, x^W_{r,n+m}),\end{array}$$

  which involves $r \cdot (n + m)$ pairs center-width, i.e., fuzzy sets.

- **Crossover operator**. The crossover operator is similar to the classical two-point crossover but with an n-dimensional consideration (being n the number of input variables). First, the rules are sorted according to the centers of the input membership functions. Then, two random numbers are selected for each input variable within its range. The parameters for the i-th input variable are calculated as follows:

$$\begin{array}{l} C^1_i = MIN_i + (MAX_i - MIN_i) \cdot (R_1)^{\frac{1}{n}} \\ C^2_i = C^1_i + (MAX_i - MIN_i) \cdot (R_2)^{\frac{1}{n}} \end{array},$$

  with $[MIN_i, MAX_i]$ being the domain of the variable and with R_1 and R_2 being two random numbers uniformly generated in $[0, 1]$.

  The first offspring will contain rules from the first parent such that

$$\forall i,\ \left(\left(x^C_{ik} > C^1_i\right)\ AND\ \left(x^C_{ik} < C^2_i\right)\right)\ OR\ \left(\left(x^C_{ik} + MAX_i - MIN_i\right) < C^2_i\right),$$

as well as rules from the second parent that do not satisfy this condition, i.e.,

$$\exists i,\ \left(\left(x_{ik}^{C} \leq C_i^1\right)\ OR\ \left(x_{ik}^{C} \geq C_i^2\right)\right)\ AND\ \left(\left(x_{ik}^{C} + MAX_i - MIN_i\right) \geq C_i^2\right).$$

The second offspring will contain the remaining rules from both parents.

- **Other operators**. The authors propose a mutation operator that applies real-number creep to fuzzy set membership function centers and widths. Therefore, the mutation is used for fine tuning rather than for introducing radically different individuals into the population. They also propose operators to create and delete rules, and to ensure that all the input data is covered by at least a rule.

This method does not consider any restriction on the fuzzy set parameters since both the gene pool initialization and the genetic operators freely generate these values. Therefore, P-FCS1 performs an UL process.

3.2.3.2 Herrera, Lozano, and Verdegay's Method (MOGUL-GLP)

The Genetic Learning Process (GLP) proposed in [49] is based on the IRL approach and it is composed by three stages following the MOGUL paradigm presented in [21]. The first stage is a *genetic generation process* to obtain desirable fuzzy rules representing the complete knowledge from the set of examples. The second one is a *genetic simplification process* to combine rules and eliminate redundant rules, selecting the most cooperative set of rules. The third one is a *genetic tuning process* to adjust the membership functions of the fuzzy rules. The three stages will be briefly introduced in the following:

1. *Genetic generation process*: According to the IRL approach, this process finds the best rule in every run over the set of examples as regards the fitness function. In each iteration, the *covering method* runs the *generating method*, chooses the best chromosome (rule), assigns the relative covering value to every example, and removes the examples with a covering value greater than a previously specified threshold. The *generating method* is based on a genetic algorithm (GA) whose main characteristics are as follows:

   - **Representation**. Each chromosome contains a single fuzzy rule. To represent it, the three parameters of each triangular membership function corresponding to each variable are encoded in the chromosome (using real coding). That is, a rule with n input variables and one output variable will be encoded in the r-th chromosome as

     $$C_r = (a_{r1}, b_{r1}, c_{r1}) \ldots (a_{rn}, b_{rn}, c_{rn})(a_{rn+1}, b_{rn+1}, c_{rn+1}),$$

     which involves $3 \cdot (n+1)$ floating point numbers.
   - **Fitness function**. The fitness function is defined according to five criteria: *high-frequency value*, *high average covering degree over positive examples*, *small negative examples set*, *small membership function width*, and *high symmetrical membership functions*. Their formulations are to be shown in [49].
   - **Genetic Operators**. GLP uses the non-uniform mutation [72] to make a uniform search in the initial space at the beginning, and a very local one at later stages. The crossover operator, the max–min-arithmetical crossover [48], generates four offspring with different search properties and selects the best two of them according to the fitness function. Finally, the selection procedure considered is the stochastic universal sampling [4] together with the elitist selection.

The generating method obtains the fuzzy sets parameters in the corresponding variable domain without considering any restriction, thus following the UL approach.

2. *Genetic simplification process*: Due to the iterative nature of the generation process, an overlearning phenomenon may appear. This occurs when some examples are covered at a higher degree than the desired one and it makes the FRB obtained perform worse. In order to solve this problem and improve the NGO FRBS accuracy, it is necessary to simplify the rule set obtained from the previous process, removing the redundant rules and selecting the rule subset with best cooperation to derive the final FRB solving the problem.

   The simplification process is based on a binary-coded GA with fixed-length chromosomes. Considering the rules contained in the rule set derived from the previous step counted from 1 to m, an m-bit string $C = (c_1, ..., c_m)$ represents a subset of candidate rules to form the FRB finally obtained as this stage output, B^s, such that,

$$\textit{If } c_i = 1 \textit{ then } R_i \in B^s \textit{ else } R_i \notin B^s.$$

   The selection of the individuals is developed using the stochastic universal sampling procedure proposed by Baker in [4] together with an elitist selection scheme, and the recombination is put into effect using the classical binary multipoint crossover (performed at two points) and uniform mutation operators. The fitness function is composed of the mean square error over the training data set (to reward the similarity between the model and the data set) and a covering measure (to penalize the lack of the completeness property obtained in the previous stage).

   The initial population is generated by introducing a chromosome representing the complete previously obtained rule set, that is, $c_i = 1,\ i = \{1, \ldots, m\}$. The remaining chromosomes are selected at random.

3. *Genetic tuning process*: The genetic tuning process [47] is based on a real coding GA that adjusts the membership functions of the fuzzy rules. The fitness function is composed only of the mean square criterion. An FRB is represented as a chromosome as follows:

   A rule r_i with n input variables is represented by a piece of chromosome C_{r_i},

$$C_{r_i} = (a_{i1}, b_{i1}, c_{i1}, \ldots, a_{in}, b_{in}, c_{in}, a_{in+1}, b_{in+1}, c_{in+1}),$$

   and a base of t rules, R, is represented by the chromosome C_R,

$$C_R = C_{r_1} C_{r_2} \ldots C_{r_t}.$$

   The GA uses the non-uniform mutation [72], the max–min-arithmetical crossover [48], and the stochastic universal sampling [4] as selection procedure together with the elitist selection.

   The initial gene pool is created with the first chromosome encoding the provided FRB, and the remaining chromosomes with each membership function parameter initiated at random in a specific previously calculated interval.

3.2.3.3 Cordón and Herrera's Soft Constrained Method (MOGUL-SCL)

As the previous method, the SCL method proposed in [24] is also composed of the three said stages (generation, simplification, and tuning processes) according to the MOGUL paradigm [21].

The difference between this method and the previous one exclusively lies in the generating method of the generation process. Therefore, in this section only this difference will be introduced, considering the remaining aspects equal to the previously showed ones.

1. *Evolutionary generation process*: Although the covering method is identical to the one used in the previous method, the generating method—i.e., the way to obtain each rule—is totally different.

   A previously defined data base constituted by uniform fuzzy partitions—the number of linguistic terms must be specified by the designer—with triangular membership functions crossing at height 0.5 is considered (as shown Figure 8). Each time the generating method is run, it produces a set of candidate fuzzy rules generating the linguistic rule best covering every example from the training set. These rules are obtained taking the fuzzy partition linguistic label that best matches the example component value for each variable. After that, an (1+1)-Evolution Strategy (ES) [3] locally tunes the best linguistic rule found making it get an NGO nature.

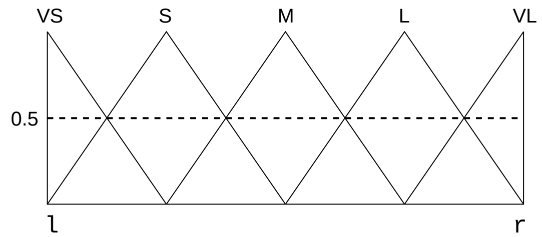


Figure 8: Graphical representation of the proposed fuzzy partition with five linguistic terms

   The method clearly follows the SCL approach in the generation stage since the (1+1)-EE adjusts the membership function parameters allowing them to take any value in the interval defined by the corresponding initial linguistic label. In this way, if $C_i = (x_0, x_1, x_2)$ is the membership function currently adapted, the associated variation interval will be $[C_i^l, C_i^r] = [x_0 - (x_1 - x_0)/2, x_2 + (x_2 - x_1)/2]$.

   The main features of this generating method are introduced in the following:

   - **Fitness function of the inductive algorithm**. The accuracy of the candidates is measured by a multicriterion fitness function. It is defined according to three features with the objective of selecting fuzzy rules covering a large number of examples (two first criteria) and verifying the consistency property. Their formulations are to be shown in [24].
   - **(1+1)-ES**. It is applied to optimize the best linguistic rule, selected from the candidate rule set, modifying the shapes of the concrete membership functions involved in it. The three main aspects of this ES are the following:
     - **Representation**. Each individual contains a fuzzy rule encoded into a real string using the three parameters of each triangular membership function corresponding to each variable.
     - **Fitness function of the ES**. It is based on the same frequentistic criteria used in the said inductive algorithm with the addition of a new one penalizing an excessive interaction among the fuzzy rules generated. To put this into effect, the ES makes use of the low niche interaction rate [23] criterion considering the individual currently being optimized and the fuzzy rules already generated. Therefore, the payoff associated to this individual will be lower when it is closer to a previously generated rule, making the rules in the FRB finally obtained to cooperate in a better way.

- **Mutation process**. The mutation scheme has two main characteristics: the definition of multiple step sizes (it must be considered the encoded membership functions are defined over different universes and require different order mutations) and the incremental optimization of the individual parameters (to obtain meaningful fuzzy sets).

2. *Genetic simplification process*: This stage was explained in the previous section.

3. *Genetic tuning process*: The tuning process is the same introduced in the previous section.

3.2.3.4 Cordón and Herrera's Hard Constrained Method (MOGUL-HCL)

This HCL method [27] is also based on the MOGUL paradigm [21] being composed of the three stages: generation, simplification, and tuning. Although it also follows the CL approach, this algorithm considers an interval where each point defining the membership functions may take value in the generation process unlike the previous method, thus performing an HCL process. It is accomplished thanks to the use of special genetic operators that maintain the constrains.

1. *Genetic generation process*: The covering method considered is again the one introduced in previous sections but a new generating method is proposed. This method is based on a GA including an (1+1)-ES as a genetic operator to locally tune the fuzzy rules and takes a set of fuzzy partitions as a base to define the variation intervals. Its main characteristics are the following:

   - **Representation**. A chromosome C encoding a candidate rule is composed of two different parts, C_1 and C_2, corresponding respectively to the composition of the fuzzy rule and to the membership functions involved in it. This distinction is necessary to maintain the constraints during the generation stage. The primary fuzzy sets belonging to each of the considered variable fuzzy partitions are numbered in order to represent the C_1 part. The C_2 part adopts the representation introduced in Section 3.2.3.2.
   - **Fitness function**. The fitness function is the same one of the (1+1)-ES introduced in the previous section.
   - **Mutation operator**. Two different operators are used depending on the chromosome part. When the mutation acts on C_1, the value of the chosen gene is randomly increased or decreased by one (only one of them if the value is an extreme). This modification makes the C_2 part change because the corresponding membership function parameters are automatically updated to the default values in the corresponding primary fuzzy partition. When the mutation is to be performed on C_2, the non-uniform mutation [72] is applied.
   - **Crossover operator**. Two different crossover operators are employed depending on the two parents' scope. If both parents encode the same rule (the C_1 parts are identical), the max–min-arithmetical crossover [48] is applied on the C_2 parts to exploit a promising space zone. Otherwise, the classical one-point crossover is applied on the C_1 parts and on the corresponding positions in the C_2 parts. This second crossover kind has higher exploration degree.
   - **(1+1)-ES operator**. Each time a GA generation is performed, the ES is applied over a percentage of the best different population individuals existing in the current genetic population. In this way, it allows us to develop again a strong exploitation over the promising space zones found in each generation by adjusting the C_2 part

values of the chromosomes located at them. This (1+1)-ES is characterized by a mutation process with definition of multiple step sizes (to consider different universes of discourse) and with an appropriate maintenance of the imposed constrains.

2. *Genetic simplification process*: It was showed in Section 3.2.3.2.

3. *Genetic tuning process*: It is the same presented in Section 3.2.3.2.

5 Experimental Study

To analyze the accuracy of the NGO FRBS learning methods presented, we are going to apply them in some fuzzy modeling problems. The experiments will be split into two groups. The first one is focused on comparing the linguistic and NGO FRBS modeling a relatively simple problem. In the second group of experiments, the comparative behavior of the learning methods introduced will be analyzed using two complex problems: the modeling of a three-dimensional surface and a real-world problem. Moreover, in this second problem the results obtained will be also compared with a classical approach, a neural network.

In both studies, two learning methods for linguistic FRBSs will be used to keep in mind the linguistic behavior: the well-known ad hoc data driven method proposed by Wang and Mendel [91] and the GA-based method proposed by Liska and Melsheimer [65] that simultaneously learn the data base and the rule base.

An initial data base constituted by a primary fuzzy partition for each variable is considered for the Wang-Mendel, Liska-Melsheimer, WCA (only for the input variables), MOGUL-SCL, and MOGUL-HCL methods. The partitions are formed by *seven linguistic terms* (for the first study and the tree-dimensional surface of the second study) or *five linguistic terms* (for the real-world problem of the second study) with triangular-shaped fuzzy sets giving meaning to them (as shown in Figure 8), and the appropriate scaling factors to translate the generic universe of discourse into the one associated with each problem variable.

The parameters considered by the analyzed methods have been selected in each problem to obtain the best degree of accuracy. We will consider the *mean square error* (MSE) to evaluate the quality of the results. We have defined MSE as

$$MSE = \frac{1}{2 \cdot N} \sum_{i=1}^{N} (y'_i - y_i)^2,$$

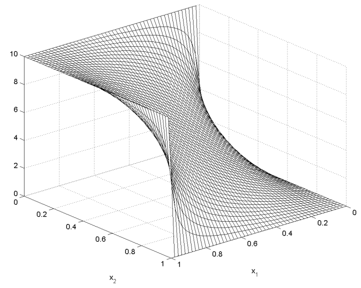
with N being the data set size, y' being the output obtained from the FRBS, and y being the known desired output. The closer to zero the measure, the greater the model accuracy.

Finally, as regards the FRBS 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 defuzzification operator [30].

5.1 First Study: Non-Grid-Oriented FRBSs Behavior in Simple Problems

In the first study, the surface generated by a two-dimensional mathematical function with a low complexity will be modeled. Its graphical representation, mathematical expression, and variable universes of discourse are shown in Figure 9. This function, F_1, is a smooth one presenting discontinuities at $(0, 0)$ and $(1, 1)$, as can be observed in its graphical representation.

A training data set uniformly distributed in the two-dimensional input space has been obtained experimentally. In this way, a set with 674 values has been generated for the function F_1 taking 26 values for each one of the two input variables considered to be uniformly distributed



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

$$x_1, x_2 \in [0, 1], \quad F_1(x_1, x_2) \in [0, 10]$$

Figure 9: Graphical representation, mathematical expression, and variable universes of discourse of the simple two-dimensional function F_1

in their intervals (is composed of 674 values instead of 676 because it is not defined in two space points).

Another data set has been generated for its use as test set to evaluate the performance of the learning method, avoiding any possible bias related to the data in the training set. The size of this data set (67) is the ten percent of the training set one. The data is obtained generating the input variable values at random in the concrete universes of discourse for each one of them, and computing the associated output variable value.

The results obtained in this experiment are collected in Table 2, where #R stands for the number of rules, #FS for the number of different fuzzy sets used in the model, and MSE_{tra} and MSE_{tst} for the values obtained by the MSE measure over the training and test data sets respectively.

Table 2: Results obtained modeling F_1

Method	**#R**	**#FS**	$\textbf{MSE}_{tra}$	$\textbf{MSE}_{tst}$
Wang-Mendel	49	14	0.194386	0.044466
Liska-Melsheimer	43	14	0.105432	0.174930
WCA	49	147	0.288179	0.138663
FCM	80	240	0.125857	0.156120
CEB	63	189	2.156918	0.445424
P-FCS1	42	126	0.044838	0.038075
MOGUL-GLP	145	435	**0.029210**	**0.034915**
MOGUL-SCL	86	258	0.094095	0.092310
MOGUL-HCL	55	165	0.160458	0.111845

Analyzing these results, the good generalization degree (MSE_{tst}) presented by the Wang-Mendel linguistic method over the NGO ones may be observed. It does not drive at *NGO FRBSs* can not obtain such a degree of accuracy in simple problems, since linguistic FRBSs may be considered a particular case of NGO FRBSs, but the *NGO learning methods* do not work fine in simple problems due to the fact that they process more complexity.

On the other hand, comparing the two linguistic methods, we may notice that the Liska-Melsheimer one does not overcome to the Wang-Mendel one in spite of performing a deep design by learning the membership functions. This fact leads us again to think that sophisticated methods are not useful in simple problems.

Only two methods, P-FCS1 and MOGUL-GLP, achieve better results than Wang-Mendel. However, this slight improvement is obtained at the expense of losing the high interpretability provided by linguistic FRBSs. Moreover, MOGUL-GLP generates too many rules for a problem as simple as this one. Therefore, we can conclude that linguistic FRBSs are preferable than

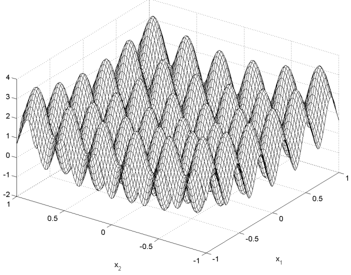
NGO ones in these kinds of problems due to their good behavior and legibility.

5.2 Second Study: Non-Grid-Oriented FRBSs Behavior in Complex Problems

In this second study we will analyze the high approximation capability of NGO FRBSs in two complex modeling problems: a very complex three-dimensional mathematical function and a real-world electrical engineering problem. In the second problem, the results obtained by a neural network will be also included to compare the NGO methods with a classical approach.

5.2.1 A Complex Three-Dimensional Surface

The *generalized Rastrigin function*, F_2, a strongly multimodal function—whose graphical representation, mathematical expression, and variable universes of discourse are shown in Figure 10—, will be considered in this section.



$$F_2(x_1, x_2) = x_1^2 + x_2^2 - cos(18x_1) - cos(18x_2)$$

$$x_1, x_2 \in [-1, 1], \quad F_2(x_1, x_2) \in [-2, 3.5231]$$

Figure 10: Graphical representation, mathematical expression, and variable universes of discourse of the complex two-dimensional function F_2

The function data is obtained like the previous case. Training and test sets, with 1,681 (taking 41 values for each of the two input variables) and 168 (ten percent) values, respectively, have been generated.

Table 3 collects the results obtained in this experiment.

Table 3: Results obtained modeling F_2

Method	**#R**	**#FS**	**MSE**$_{tra}$	**MSE**$_{tst}$
Wang-Mendel	49	14	1.824250	2.099754
Liska-Melsheimer	49	14	0.330614	0.370192
WCA	49	147	0.494001	0.536226
FCM	48	144	0.494189	0.528513
CEB	298	894	0.426659	0.529424
P-FCS1	74	222	0.220314	0.222865
MOGUL-GLP	228	684	0.267306	0.289784
MOGUL-SCL	239	717	**0.173270**	**0.168424**
MOGUL-HCL	236	708	0.177442	0.183578

From an analysis of these results, we may note the good behavior of the NGO FRBSs generated as contrasted with the linguistic Wang-Mendel method. Compared with the Liska-Melsheimer method, only the NGO FRBS learning methods based on GA obtain better accuracy results. In problems as complex as this one, the use of a local semantic introduce additional degrees of freedom that allow the NGO FRBSs obtained to be more accurate.

Within NGO FRBSs, we may also observe the clear superiority of the four GFRBSs over the rest. The evolutionary techniques generate good NGO FRBSs owing to their ability to explore and exploit given operating spaces using available performance measures.

It is also interesting to check between the two clustering methods that the automatic learning of the optimal number of clusters in complex problems is not trivial, and CEB needs to use numerous rules to obtain a similar accuracy to the NGO FRBSs designed by FCM.

Focusing on the GFRBSs, the algorithms that follow the CL approach (MOGUL-SCL and MOGUL-HCL) clearly obtain NGO FRBSs more accurate than the ones obtained via UL-based methods (P-FCS1 and MOGUL-GLP). Although the latter approach has more freedom to generate the fuzzy rules, it deals with a bigger search space that make the learning process difficult. Another important difference among the four GFRBSs is the way to apply GAs to the learning process, while P-FCS1 follows the Pittsburgh approach, MOGUL-GLP, MOGUL-SCL, and MOGUL-HCL are based on the IRL one. Following the former one, a more reduced number of rules is obtained, however, since the NGO FRBSs are oriented to problems where the accuracy is more preferable than the readability (fuzzy modeling), this aspect should not be taken into account.

Finally, comparing MOGUL-SCL and MOGUL-HCL we may observe that the use of soft restrictions obtains best results in hard problems like this, being further the FRBSs designed by MOGUL-SCL the most accurate among all the generated FRBSs.

5.2.2 The Electrical Distribution Networks Problem

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 as the estimation of the maintenance costs of the network, which was the main goal of the problem presented here in Spain [31]. High and medium voltage lines can be easily measured, but low voltage line is contained in cities and 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 for determining the length of line is needed.

The problem involves finding a model that relates the total length of low voltage line installed in a rural town with the number of inhabitants in the town and the mean of the distances from the center of the town to the three furthest clients in it [31]. This model will be used to estimate the total length of line being maintained. We will limit ourselves to the estimation of the length of line in a town, given the inputs mentioned before. Hence, our objective is to relate the first variable (line length) with the other two ones (population and radius of village).

To compare the methods, we have randomly divided the sample, composed of 495 pieces of real data obtained from direct measures in this number of villages, into two sets comprising 396 and 99 samples, labeled training and test.

The results obtained with the considered modeling methods and with a classical neural network (whose number of neurons in the hidden layer was chosen to minimize the test error) are shown in Table 4.

From the obtained results, we may again note the good behavior of NGO FRBSs. However, not all the designed NGO FRBSs overcome to the linguistic ones. WCA and MOGUL-GLP obtain worse generalization results than the Wang-Mendel and Liska-Melsheimer methods. The results of WCA show the difficulty of designing an NGO FRBSs with ad hoc data-driven algorithms in real-world problems like this. On the other hand, MOGUL-GLP designs an FRBS with many rules that bring closer to the training data performing then very bad prediction.

The remaining NGO FRBSs present a good degree of accuracy. Focusing on the two clustering methods, though FCM designs a more accurate NGO FRBS, CEB automatically learns the number of clusters (rules) obtaining a simpler FRBS without losing meaningful accuracy.

Table 4: Results obtained modeling the electrical application problem

Method	#R	#FS/Complexity	$\mathbf{MSE}_{tra}$	$\mathbf{MSE}_{tst}$
Wang-Mendel	13	10	298,450	282,029
Liska-Melsheimer	25	10	163,052	223,300
WCA	20	30	341,077	309,851
FCM	49	147	163,615	198,617
CEB	37	111	200,999	222,362
P-FCS1	71	213	141,023	197,941
MOGUL-GLP	50	150	**115,768**	365,692
MOGUL-SCL	20	60	142,109	**166,579**
MOGUL-HCL	13	39	152,684	168,254
Three layer perceptron 2-25-1	—	102 par.	169,399	167,092

Again, the genetic methods based on the CL approach obtain the best results. P-FCS1 obtains an excessive number of rules that causes a little overlearning. Both MOGUL-SCL and MOGUL-HCL stand out against all designing the best NGO FRBSs with a short difference between them.

Finally, analyzing the results obtained with the classical neural network, we may conclude that it is possible to obtain a more accurate and more interpretable model (due to its capability to locally describe the system behavior) by means of an NGO FRBS.

6 Concluding Remarks

In this paper, a study on NGO FRBSs has been accomplished. In contrast to linguistic FRBSs, the NGO ones directly work with fuzzy variables in the fuzzy rules, thus being totally equivalent to fuzzy graphs. They are suitable for problems where the accuracy of the system is preferred to its interpretability.

Some specific methods to build fuzzy graphs have been analyzed considering several forms of facing the learning process such as the used technique (ad hoc data-driven, clustering, and GAs) and a taxonomy of them based on the constrains imposed on the fuzzy sets (constrained and unconstrained learning) has been presented. Their behavior has been analyzed through the application of the reviewed methods to three different problems.

In view of the obtained results, we should remark some important conclusions. NGO FRBSs perform a good behavior in highly nonlinear and complex problems, obtaining accurate and locally interpretable models. The results obtained by the methods that make use of GAs have shown that this technique is very suitable for learning NGO FRBSs. With regard to the fact of imposing constrains on the fuzzy sets during the learning process, it seems to obtain better results. The use of soft constrains obtains NGO FRBSs very accurate thanks to perform a good balance between generation of fuzzy sets with a high degree of freedom (model flexibility) and reduction of the search space (learning simplicity).

References

[1] R. Alcalá, J. Casillas, O. Cordón, and F. Herrera. Cooperative coevolution for linguistic modeling with weighted double-consequent fuzzy rules. In *Proceedings of the 2nd International Conference in Fuzzy Logic and Technology*, pages 237–240, Leicester, UK, 2001.

[2] R. Babuška. *Fuzzy modeling for control*. Kluwer Academic, Norwell, MA, USA, 1998.

[3] T. Bäck, editor. *Evolutionary algorithms in theory and practice*. Oxford University Press, UK, 1996.

[4] J.E. Baker. Reducing bias and inefficiency in the selection algorithm. In *Proceedings of the 2nd International Conference on Genetic Algorithms*, pages 14–21, Hillsdale, NJ, USA, 1987. Lawrence Erlbaum.

[5] A. Bárdossy and L. Duckstein. *Fuzzy rule-based modeling with application to geophysical, biological and engineering systems*. CRC Press, Boca Raton, FL, USA, 1995.

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

[7] M.R. Berthold and K.-P. Huber. Constructing fuzzy graphs from examples. *Intelligent Data Analysis*, 3:37–53, 1999.

[8] J.C. Bezdek. *Pattern recognition with fuzzy objective function algorithms*. Plenum Press, 1981.

[9] M. Bikdash. A highly interpretable form of Sugeno inference systems. *IEEE Transactions on Fuzzy Systems*, 7(6):686–696, 1999.

[10] P.P. Bonissone. Soft computing: the convergence of emerging reasoning technologies. *Soft Computing. A Fusion of Foundations, Methodologies, and Applications*, 1(1):6–18, 1997.

[11] P.P. Bonissone, P.S. Khedkar, and Y. Chen. Genetic algorithms for automated tuning of fuzzy controllers: a transportation application. In *Proceedings of the 5th IEEE International Conference on Fuzzy Systems*, pages 674–680, New Orleans, LA, USA, 1996.

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

[13] J. Casillas, O. Cordón, M.J. del Jesus, and F. Herrera. Genetic feature selection in a fuzzy rule-based classification system learning process for high dimensional problems. *Information Sciences*, 136(1-4):169–191, 2001.

[14] J. Casillas, O. Cordón, M.J. del Jesus, and F. Herrera. Genetic tuning of fuzzy rule-based systems integrating linguistic hedges. In *Proceedings of the 9th IFSA World Congress and the 20th NAFIPS International Conference*, pages 1570–1574, Vancouver, Canada, 2001.

[15] J. Casillas, O. Cordón, F. Herrera, and L. Magdalena, editors. *Advance in fuzzy rule-based modeling. Part I, accuracy improvements preserving interpretability. Part II, interpretability issues*. Physica-Verlag, Heidelberg, Germany, 2002. In press.

[16] S.L. Chiu. Fuzzy model identification based on cluster estimation. *Journal of Intelligent and Fuzzy Systems*, 2:267–278, 1994.

[17] J.-S. Cho and D.-J. Park. Novel fuzzy logic control based on weighting of partially inconsistent rules using neurtal network. *Journal of Intelligent and Fuzzy Systems*, 8:99–110, 2000.

[18] W.E. Combs and J.E. Andrews. Combinatorial rule explosion eliminated by a fuzzy rule configuration. *IEEE Transactions on Fuzzy Systems*, 6(1):1–11, 1998.

[19] M.G. Cooper and J.J. Vidal. Genetic design of fuzzy controllers: the cart and jointed pole problem. In *Proceedings of the 3rd IEEE International Conference on Fuzzy Systems*, pages 1332–1337, Piscataway, NJ, USA, 1994.

[20] O. Cordón, M.J. del Jesus, and F. Herrera. Genetic learning of fuzzy rule-based classification systems cooperating with fuzzy reasoning methods. *International Journal of Intelligent Systems*, 13:1025–1053, 1998.

[21] O. Cordón, M.J. del Jesus, F. Herrera, and M. Lozano. MOGUL: A methodology to obtain genetic fuzzy rule-based systems under the iterative rule learning. *International Journal of Intelligent Systems*, 14(11):1123–1153, 1999.

[22] O. Cordón and F. Herrera. A general study on genetic fuzzy systems. In J. Periaux, G. Winter, M. Galán, and P. Cuesta, editors, *Genetic algorithms in engineering and computer science*, pages 33–57. Wiley, New York, USA, 1995.

[23] O. Cordón and F. Herrera. A hybrid genetic algorithm-evolution strategy process for learning fuzzy logic controller knowledge bases. In F. Herrera and J.L. Verdegay, editors, *Fuzzy logic and soft computing*, pages 251–278. Physica-Verlag, Heidelberg, Germany, 1996.

[24] O. Cordón and F. Herrera. A three-stage evolutionary process for learning descriptive and approximate fuzzy logic controller knowledge bases from examples. *International Journal of Approximate Reasoning*, 17(4):369–407, 1997.

[25] O. Cordón and F. Herrera. A two-stage evolutionary process for designing TSK fuzzy rule-based systems. *IEEE Transactions on Systems, Man, and Cybernetics—Part B: Cybernetics*, 29(6):703–715, 1999.

[26] O. Cordón and F. Herrera. A proposal for improving the accuracy of linguistic modeling. *IEEE Transactions on Fuzzy Systems*, 8(3):335–344, 2000.

[27] O. Cordón and F. Herrera. Hybridizing genetic algorithms with sharing scheme and evolution strategies for designing approximate fuzzy rule-based systems. *Fuzzy Sets and Systems*, 118(2):235–255, 2001.

[28] O. Cordón, F. Herrera, F. Hoffmann, and L. Magdalena. *Genetic fuzzy systems: evolutionary tuning and learning of fuzzy knowledge bases.* World Scientific, Singapore, 2001.

[29] O. Cordón, F. Herrera, L. Magdalena, and P. Villar. A genetic learning process for the scaling factors, granularity and contexts of the fuzzy rule-based system data base. *Information Sciences*, 136(1-4):85–107, 2001.

[30] O. Cordón, F. Herrera, and A. Peregrín. Applicability of the fuzzy operators in the design of fuzzy logic controllers. *Fuzzy Sets and Systems*, 86(1):15–41, 1997.

[31] O. Cordón, F. Herrera, and L. Sánchez. Solving electrical distribution problems using hybrid evolutionary data analysis techniques. *Applied Intelligence*, 10(1):5–24, 1999.

[32] O. Cordón, F. Herrera, and P. Villar. Analysis and guidelines to obtain a good fuzzy partition granularity for fuzzy rule-based systems using simulated annealing. *International Journal of Approximate Reasoning*, 25(3):187–215, 2000.

[33] O. Cordón, F. Herrera, and I. Zwir. Linguistic modeling by hierarchical systems of linguistic rules. To appear in *IEEE Transactions on Fuzzy Systems*. Draft version available at `http://decsai.ugr.es/~herrera/`.

[34] J.C. Dunn. A fuzzy relative of the ISODATA process and its use in detecting compact well separated clusters. *Journal Cybernetics*, 3(3):32–57, 1974.

[35] A. Dvořák. On linguistic approximation in the frame of fuzzy logic deduction. *Soft Computing*, 3(2):111–116, 1999.

[36] J. Espinosa and J. Vandewalle. Constructing fuzzy models with linguistic integrity from numerical data-AFRELI algorithm. *IEEE Transactions on Fuzzy Systems*, 8(5):591–600, 2000.

[37] A. Fiordaliso. Autostructuration of fuzzy systems by rules sensitivity analysis. *Fuzzy Sets and Systems*, 118(2):281–296, 2001.

[38] A. Fiordaliso. A constrained Takagi-Sugeno fuzzy system that allows for better interpretation and analysis. *Fuzzy Sets and Systems*, 118(2):307–318, 2001.

[39] B. Fritzke. Incremental neuro-fuzzy systems. In B. Bosacchi, J.C. Bezdek, and D.B. Fogel, editors, *Proceedings of the International Society for Optical Engineering: Applications of Soft Computing*, volume 3165, pages 86–97, 1997.

[40] A. González and R. Pérez. SLAVE: a genetic learning system based on an iterative approach. *IEEE Transactions on Fuzzy Systems*, 7(2):176–191, 1999.

[41] A. González and 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):257–266, 1999.

[42] A. González and R. Pérez. Selection of relevant features in a fuzzy genetic learning algorithm. *IEEE Transactions on Systems, Man, and Cybernetics—Part B: Cybernetics*, 31(3):417–425, 2001.

[43] J.J. Grefensette, editor. *Genetic algorithms for machine learning.* Kluwer Academic, Norwell, MA, USA, 1994.

[44] R.R. Gudwin and F.A.C. Gomide. Context adaptation in fuzzy processing. In *Proceedings of the 1st Brazil-Japan Joint Symposium in Fuzzy Systems*, pages 15–20, Campinas, SP, Brazil, 1994.

[45] R.R. Gudwin, F.A.C. Gomide, and W. Pedrycz. Context adaptation in fuzzy processing and genetic algorithms. *International Journal of Intelligent Systems*, 13(10-11):929–948, 1998.

[46] F. Herrera, editor. *Special issue on genetic fuzzy systems for control and robotics*, volume 17(4) of *International Journal of Approximate Reasoning.* 1997.

[47] F. Herrera, M. Lozano, and J.L. Verdegay. Tuning fuzzy controllers by genetic algorithms. *International Journal of Approximate Reasoning*, 12:299–315, 1995.

[48] F. Herrera, M. Lozano, and J.L. Verdegay. Fuzzy connectives based crossover operators to model genetic algorithm population diversity. *Fuzzy Sets and Systems*, 92(1):21–30, 1997.

[49] F. Herrera, M. Lozano, and J.L. Verdegay. A learning process for fuzzy control rules using genetic algorithms. *Fuzzy Sets and Systems*, 100:143–158, 1998.

[50] F. Herrera and L. Magdalena, editors. *Special issue on genetic fuzzy systems*, volume 13(10) of *International Journal of Intelligent Systems.* 1998.

[51] F. Herrera and J.L. Verdegay, editors. *Genetic algorithms and soft computing*. Physica-Verlag, Heidelberg, Germany, 1996.

[52] J.H. Holland and J.S. Reitman. Cognitive systems based on adaptive algorithms. In D.A. Waterman and F. Hayes-Roth, editors, *Pattern-directed inference systems*, pages 313–329. Academic Press, San Diego, USA, 1978.

[53] A. Homaifar and E. McCormick. Simultaneous design of membership functions and rule sets for fuzzy controllers using genetic algoritms. *IEEE Transactions on Fuzzy Systems*, 3(2):129–139, 1995.

[54] T.-P. Hong and J.-B. Chen. Finding relevant attributes and membership functions. *Fuzzy Sets and Systems*, 103(3):389–404, 1999.

[55] T.-P. Hong and C.-Y. Lee. Effect of merging order on performance of fuzzy induction. *Intelligent Data Analysis*, 3(2):139–151, 1999.

[56] H. Ishibuchi, T. Murata, and I.B. Türkşen. Single-objective and two-objective genetic algorithms for selecting linguistic rules for pattern classification problems. *Fuzzy Sets and Systems*, 89(2):135–150, 1997.

[57] H. Ishibuchi and T. Nakashima. Effect of rule weights in fuzzy rule-based classification systems. *IEEE Transactions on Fuzzy Systems*, 9(4):506–515, 2001.

[58] H. Ishibuchi, K. Nozaki, N. Yamamoto, and H. Tanaka. Selecting fuzzy if-then rules for classification problems using genetic algorithms. *IEEE Transactions on Fuzzy Systems*, 3(3):260–270, 1995.

[59] Y. Jin. Fuzzy modeling of high-dimensional systems: complexity reduction and interpretability improvement. *IEEE Transactions on Fuzzy Systems*, 8(2):212–221, 2000.

[60] Y. Jin, W. von Seelen, and B. Sendhoff. On generating FC^3 fuzzy rule systems from data using evolution strategies. *IEEE Transactions on Systems, Man, and Cybernetics—Part B: Cybernetics*, 29(4):829–845, 1999.

[61] C.L. Karr. Genetic algorithms for fuzzy controllers. *AI Expert*, 6(2):26–33, 1991.

[62] L. Koczy. Fuzzy if ... then rule models and their transformation one another. *IEEE Transactions on Systems, Man, and Cybernetics*, 26(5):621–637, 1996.

[63] A. Krone and H. Taeger. Data-based fuzzy rule test for fuzzy modelling. *Fuzzy Sets and Systems*, 123(3):343–358, 2001.

[64] H.-M. Lee, C.-M. Chen, J.-M. Chen, and Y.-L. Jou. An efficient fuzzy classifier with feature selection based on fuzzy entropy. *IEEE Transactions on Systems, Man, and Cybernetics—Part B: Cybernetics*, 31(3):426–432, 2001.

[65] J. Liska and S.S. Melsheimer. Complete design of fuzzy logic systems using genetic algorithms. In *Proceedings of the 3rd IEEE International Conference on Fuzzy Systems*, pages 1377–1382, Orlando, FL, USA, 1994.

[66] B.-D. Liu, C.-Y. Chen, and 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):32–53, 2001.

[67] L. Magdalena. Adapting the gain of an FLC with genetic algorithms. *International Journal of Approximate Reasoning*, 17(4):327–349, 1997.

[68] L. Magdalena and F. Monasterio-Huelin. A fuzzy logic controller with learning through the evolution of its knowledge base. *International Journal of Approximate Reasoning*, 16(3):335–358, 1997.

[69] E.H. Mamdani. Applications of fuzzy algorithms for control a simple dynamic plant. *Proceedings of the IEE 121*, 12:1585–1588, 1974.

[70] E.H. Mamdani and S. Assilian. An experiment in linguistic systhesis with fuzzy logic controller. *International Journal of Man-Machine Studies*, 7:1–13, 1975.

[71] J.G. Marín-Blázquez, Q. Shen, and A.F. Gómez-Skarmeta. From approximative to descriptive models. In *Proceedings of the 9th IEEE International Conference on Fuzzy Systems*, pages 829–834, San Antonio, TX, USA, 2000.

[72] Z. Michalewicz. *Genetic algorithms + data structures = evolution programs.* Springer-Verlag, Heidelberg, Germany, 3rd edition, 1996.

[73] G.C. Mouzouris and J.M. Mendel. A singular-value-QR decomposition based method for training fuzzy logic systems in uncertain environments. *Journal of Intelligent and Fuzzy Systems*, 5:367–374, 1997.

[74] D. Nauck and R. Kruse. Neuro-fuzzy systems for function approximaton. *Fuzzy Sets and Systems*, 101(2):261–271, 1999.

[75] N.R. Pal and K. Pal. Handling of inconsistent rules with an extended model of fuzzy reasoning. *Journal of Intelligent and Fuzzy Systems*, 7:55–73, 1999.

[76] W. Pedrycz, editor. *Fuzzy modelling: paradigms and practice.* Kluwer Academic, Norwell, MA, USA, 1996.

[77] W. Pedrycz, R.R. Gudwin, and F.A.C. Gomide. Nonlinear context adaptation in the calibration of fuzzy sets. *Fuzzy Sets and Systems*, 88(1):91–97, 1997.

[78] H. Pomares, I. Rojas, J. Ortega, J. González, and A. Prieto. A systematic approach to a self-generating fuzzy rule-table for function approximation. *IEEE Transactions on Systems, Man, and Cybernetics—Part B: Cybernetics*, 30(3):431–447, 2000.

[79] I. Rojas, H. Pomares, J. Ortega, and A. Prieto. Self-organized fuzzy system generation from training examples. *IEEE Transactions on Fuzzy Systems*, 8(1):23–36, 2000.

[80] H. Roubos and M. Setnes. Compact and transparent fuzzy models and classifiers through iterative complexity reduction. *IEEE Transactions on Fuzzy Systems*, 9(4):516–524, 2001.

[81] L. Sánchez, J. Casillas, O. Cordón, and M.J. del Jesus. Some relationships between fuzzy and random set-based classifiers and models. To appear in *International Journal of Approximate Reasoning.*

[82] M. Setnes, R. Babuška, U. Kaymak, and H.R. van Nauta Lemke. Similarity measures in fuzzy rule base simplification. *IEEE Transactions on Systems, Man, and Cybernetics—Part B: Cybernetics*, 28(3):376–386, 1998.

[83] Y. Shi, R. Eberhart, and Y. Chen. Implementation of evolutionary fuzzy systems. *IEEE Transactions on Fuzzy Systems*, 7(2):109–119, 1999.

[84] R. Silipo and M. Berthold. Discriminative power of input features in a fuzzy model. In D. Hand, J. Kok, and M. Berthold, editors, *Advances in Intelligent Data Analysis (IDA–99)*, volume LNCS 1642, pages 87–98. Springer-Verlag, Heidelberg, Germany, 1999.

[85] S.F. Smith. A learning system based on genetic adaptive algorithms. Ph.D. Thesis. University of Pittsburgh, USA, 1980.

[86] T. Sudkamp, J. Knapp, and A. Knapp. Refine and merge: generating small bases from training data. In *Proceedings of the 9th IFSA World Congress and the 20th NAFIPS International Conference*, pages 197–202, Vancouver, Canada, 2001.

[87] M. Sugeno and G.T. Kang. Structure identification of fuzzy model. *Fuzzy Sets and Systems*, 28:15–33, 1988.

[88] M. Sugeno and T. Yasukawa. A fuzzy-logic-based approach to qualitative modeling. *IEEE Transactions on Fuzzy Systems*, 1(1):7–31, 1993.

[89] T. Takagi and M. Sugeno. Fuzzy identification of systems and its application to modeling and control. *IEEE Transactions on Systems, Man, and Cybernetics*, 15:116–132, 1985.

[90] G. Venturini. SIA: A supervised inductive algorithm with genetic search for learning attribute based concepts. In *Proceedings of the European Conference on Machine Learning*, pages 280–296, Vienna, Austria, 1993.

[91] L.-X. Wang and J.M. Mendel. Generating fuzzy rules by learning from examples. *IEEE Transactions on Systems, Man, and Cybernetics*, 22(6):1414–1427, 1992.

[92] X. Wang and J. Hong. Learning optimization in simplifying fuzzy rules. *Fuzzy Sets and Systems*, 106(3):349–356, 1999.

[93] Y. Yam, P. Baranyi, and C.-T. Yang. Reduction of fuzzy rule base via singular value decomposition. *IEEE Transactions on Fuzzy Systems*, 7(2):120–132, 1999.

[94] J. Yen and L. Wang. Simplifying fuzzy rule-based models using orthogonal transformation methods. *IEEE Transactions on Systems, Man, and Cybernetics—Part B: Cybernetics*, 29(1):13–24, 1999.

[95] J. Yen, L. Wang, and C.W. Gillespie. Improving the interpretability of TSK fuzzy models by combining global learning and local learning. *IEEE Transactions on Fuzzy Systems*, 6(4):530–537, 1998.

[96] W. Yu and Z. Bien. Design of fuzzy logic controller with inconsistent rule base. *Journal of Intelligent and Fuzzy Systems*, 2:147–159, 1994.

[97] L.A. Zadeh. Toward a theory of fuzzy systems. In *Aspects of network and system theory*, pages 469–490. Rinehart and Winston, New York, NY, USA, 1971.

[98] L.A. Zadeh. Outline of a new approach to the analysis of complex systems and desision processes. *IEEE Transactions on Systems, Man, and Cybernetics*, 3:28–44, 1973.

[99] L.A. Zadeh. On the analysis of large scale systems. In *Systems approaches and environment problems*, pages 23–37. Vandenhoeck and Ruprecht, Gottingen, Germany, 1974.

[100] L.A. Zadeh. The concept of a linguistic variable and its application to approximate reasoning. Parts I, II and III *Information Science*, 8, 8, 9:199–249, 301–357, 43–80, 1975.

[101] L.A. Zadeh. A fuzzy-algorithmic approach to the definition of complex or imprecise concepts. *International Journal on Man-Machine Studies*, 8(3):249–291, 1976.

[102] L.A. Zadeh. Soft computing and fuzzy logic. *IEEE Software*, 11(6):48–56, 1994.

[103] L.A. Zadeh. Fuzzy logic and the calculi of fuzzy rules and fuzzy graphs: a precis. *International Journal of Multiple-Valued Logic*, 1:1–38, 1996.