A Multiobjective Genetic Algorithm for Feature Selection and Data Base Learning in Fuzzy-Rule Based Classification Systems

O. Cordón[a], **F. Herrera**[a], **M.J. del Jesus**[b]
L. Magdalena[c], **A.M. Sánchez**[d], **P. Villar**[d] *

[a]Dept. of Computer Science and A.I.; University of Granada,
18071 - Granada, Spain. e-mail: ocordon,herrera@decsai.ugr.es

[b]Dept. of Computer Science; University of Jaén,
23071 - Jaén, Spain. e-mail: mjj@ujaen.es

[c]Dept. Mat. Aplicada; Universidad Politécnica de Madrid
28040 Madrid, Spain. e-mail: llayos@mat.upm.es

[d]Dept. Computer Science; University of Vigo
32004 - Ourense, Spain. e-mail: amlopez,pvillar@uvigo.es

Abstract

In this contribution, we propose a genetic process to select an appropiate set of features in a Fuzzy Rule-Based Classification System (FRBCS) and to automatically learn the whole Data Base definition. An ad-hoc data covering learning method is considered to obtain the Rule Base. The method uses a multiobjective genetic algorithm in order to obtain a good balance between accuracy and interpretability.

Keywords: Fuzzy Rule-Based Classification Systems, Data Base, Learning, Multiobjective Genetic Algorithms

1 Introduction

An FRBCS presents two main components: the Inference System and the Knowledge Base (KB). The KB is composed of the Rule Base (RB) constituted by the collection of fuzzy rules, and of the Data Base (DB), containing the membership functions of the fuzzy partitions associated to the linguistic variables. The composition of the KB of an FRBCS directly depends on the problem being solved. If there is no expert information about the problem under solving, an automatic learning process must be used to derive the KB from examples.

Although, there is a large quantity of RB learning methods proposed in the specialized literature [3, 4, 10, 15], among others, there is not much information about the way to

* This research has been supported by CICYT under project TIC2002-04036-C05-01

derive the DB and most of these RB learning methods need of the existence of a previous definition for it. The usual way to proceed involves choosing a number of linguistic terms (granularity) for each linguistic variable, which is normally the same for all of them, and building the fuzzy partition by a uniform partitioning of the variable domain into this number of terms. This operation mode makes the granularity and fuzzy set definitions have a significant influence on the FRBCS performance.

Moreover, high dimensionality problems present a new trouble to obtain FRBCSs with good behaviour: the large number of features, that can originate a RB with a high number of rules, thus presenting a low degree of interpretability and a possible over-fitting. This problem can be tackled from a double perspective:

- Via the compactness and reduction of the rule set, minimising the number of fuzzy rules included in it.

- Via a feature selection process that reduces the number of features used by the FRBCS.

Rule reduction methods have been formulated using different approaches (Neural Networks, clustering techniques, orthogonal transformation methods, similarity measures and Genetic Algorithms). Notice that, for high dimensional problems and problems where a high number of instances is available, it is difficult for the latter approaches to get small rule sets, and therefore the system comprehensibility and interpretability may not be as good as desired. For high dimensionality classification problems, a feature selection process, that determines the most relevant variables before or during the FRBCS inductive learning process, must be considered [2, 19]. It increases the efficiency and accuracy of the learning and classification stages.

Our objective is to develop a genetic process for feature selection and whole DB learning (granularity and membership functions for each variable) to obtain FRBCSs composed of a compact set of comprehensible fuzzy rules with high classification ability. This method uses a multiobjective GA and considers a simple generation method to derive the RB.

To carry out this task, this paper is organised as follows. In Section 2, the FRBCS components will be introduced joint with a brief description of the two main problems tackled by the learning method proposed, feature selection and DB learning. In Section 3 we will expose the characteristics of our proposal for the FRBCS design. The results obtained with Sonar data set will be shown in Section 4. In the last section, some conclusions will be pointed out.

2 Preliminaries

2.1 Fuzzy Rule-Based Classification Systems

An FRBCS is an automatic classification system that uses fuzzy rules as knowledge representation tool. Two different components are distinguished within it:

1. The **KB**, composed of:

- **DB**, which contains the fuzzy set definitions related to the labels used in the fuzzy rules. So, the DB components for every variable are the number of linguistic terms (granularity) and the membership function shape of each term.
- **RB**, comprised by a set of fuzzy rules that in this work are considered to have the following structure:

$$R_k : \text{ If } X_1 \text{ is } A_1^k \text{ and } \ldots \text{ and } X_N \text{ is } A_N^k \quad \text{then} \quad Y \text{ is } C_j \text{ with } r^k$$

  where $X_1, \ldots, X_N$ are features considered in the problem and $A_1^k, \ldots, A_N^k$ are linguistic labels employed to represent the values of the variables. These kinds of fuzzy rules represent, in the antecedent part, a subspace of the complete search space by means of a linguistic label for each considered variable and, in the consequent part, a class label (C_j) and a certainty degree (r^k). This numerical value indicates the degree of certainty of the classification in that class for the examples belonging to the fuzzy subspace delimited by the antecedent part.

2. The **Fuzzy Reasoning Method (FRM)**, an inference procedure which, combining the information provided by the fuzzy rules related with the example to classify, determines the class to which it belongs to.

The majority of FRBCSs (see [3, 10] among others) use the classical FRM that classifies a new example with the consequent of the fuzzy rule having the highest degree of association. Another family of FRMs that use the information provided by all the rules compatible with the example (or a subset of them) have been developed [3, 5]. In this work, we use two different FRMs, each one belonging to one of the said groups: maximum and normalised sum.

2.2 Feature Selection and DB Learning in FRBCS design

As we mentioned before, our FRBCS learning method generates the KB by selecting an adequate feature set and by learning the appropiate DB components for each selected variable. In this section, we briefly describe these problems jointly solved in our proposal.

2.2.1 Feature Selection Process

The main objective of any feature selection process is to reduce the dimensionality of the problem for the supervised inductive learning process. This fact implies that the feature selection algorithm must determine the best features for its design.

There are two kinds of feature selection algorithms:

- *Filter feature selection algorithms* [17], which remove the irrelevant characteristics without using a learning algorithm (e.g. by means of class separability measures). They are efficient processes but, on the other hand, the feature subsets obtained by them may not be the best ones for a specific learning process because of the exclusion of the heuristic and the bias of the learning process in the selection procedure [16].

- *Wrapper feature selection algorithms* [16, 17]. This kind of feature selection algorithms selects feature subsets by means of the evaluation of each candidate subset with the precision estimation obtained by the learning algorithm. In this form, they obtain feature subsets with the best behaviour in the classifier design. Their problem is their inefficiency since the classifier has to be built for the evaluation of each candidate feature subset.

In our proposal we will use a wrapper feature selection algorithm which utilises the precision estimation provided by an efficient fuzzy rule generation process (Wang and Mendel's fuzzy rule generation process) and a GA as search algorithm. Inside the DB derivation, the granularity learning will provide us an additional way to select features when the number of linguistic labels assigned to a specific variable is only one (we will explain this in a further section).

2.2.2 DB Learning

As previously said, the derivation of the DB highly influences the FRBCS performance. In fact, some studies in Fuzzy Rule-Based Systems have shown that the system performance is much more sensitive to the choice of the semantics in the DB than to the composition of the RB [7]. Some approaches have been proposed to improve the FRBCS behaviour by means of a tuning process once the RB has been derived [4]. However, these tuning processes only adjust the shapes of the membership functions and not the number of linguistic terms in each fuzzy partition, which remains fixed from the begining of the design process.

The methods that try to learn appropiate DB components per variable usually work in collaboration with an RB derivation method. A DB generation process wraps an RB learning one working as follows: each time a DB has been obtained by the DB definition process, the RB generation method is used to derive the rules, and some type of error measure is used to validate the whole KB obtained. The works proposed in [6, 7, 8] use Simulated Annealing and GAs to learn an appropiate DB in a Fuzzy Rule-Based System. The method proposed in [14] considers a GA to design an FRBCS, working in the said way.

3 Genetic Algorithm for Feature Selection and DB Learning

In this section, we propose a new learning approach to automatically generate the KB of an FRBCS composed of two methods with different goals:

- A genetic learning process for the DB that allows us to define:
  - The relevant variables for the classification process (feature selection).
  - The number of labels for each variable (granularity learning).

  - The form of each fuzzy membership function in non-uniform fuzzy partitions, using a non-linear scaling function that defines different areas in the variable working range where the FRBCS has a higher or a lower relative sensibility.

- A quick *ad hoc data-driven method* that derives the fuzzy classification rules considering the DB previously obtained. In this work we use the extension of Wang and Mendel's fuzzy rule generation method [21] for classification problems [3], but other efficient generation methods can be considered.

We should note that the granularity learning allows us another way of feature selection: if a variable is assigned only to one label, it has no influence in the RB, so it will not be considered as a relevant variable. A similar double-level feature selection process has been previously considered in genetic learning processes of FRBCSs such as SLAVE [11].

All the components of the DB will be adapted throughout a genetic process. Since it is interesting to reduce the dimensionality of the search space for that process, the use of non-linear scaling functions is conditioned by the necessity of using parameterized functions with a reduced number of parameters. We consider the scaling funtion proposed in [6], that has a single sensibility parameter called *a* ($a \in \mathbb{R}$). The function used is ($f : [-1, 1] \rightarrow [-1, 1]$)

$$f(x) = sign(x) \cdot |x|^a\,, \quad with\;\; a > 0$$

The final result is a value in $[-1, 1]$ where the parameter *a* produces uniform sensibility ($a = 1$), higher sensibility for center values ($a > 1$), or higher sensibility for extreme values ($a < 1$). In this paper, triangular membership functions are considered due to their simplicity. So, the non-linear scaling function will only be applied on the three definition points of the membership function (which is equal to transform the scaling function in a continuous piece-wise linear function), in order to make easier the structure of the generated DB and to simplify the defuzzification process. Figure 1 shows a graphical representation of the three possibilities of fuzzy partition depending on the value of parameter a.

We should note that the previous scaling function is recommended to be used with symmetrical variables since it causes symetrical effects around the center point of the interval. For example, it can not produce higher sensibility in only one of the working range extents. In the method presented in this paper, we add a new parameter (called *S*) to the non-linear scaling function as described also in [6]. S is a parameter in $\{0, 1\}$ to distinguish between non-linearities with symmetric shape (lower sensibility for middle or for extreme values, Figure 1) and asymmetric shape (lower sensibility for the lowest or for the highest values, Figure 2).

Furthermore, the main purpose of our KB design process is to obtain FRBCSs with good accuracy and high interpretability. Unfortunately, it is not easy to achieve these two objectives at the same time. Normally, FRBCSs with good performance have a high number of selected variables and also a high number of rules, thus presenting a low degree

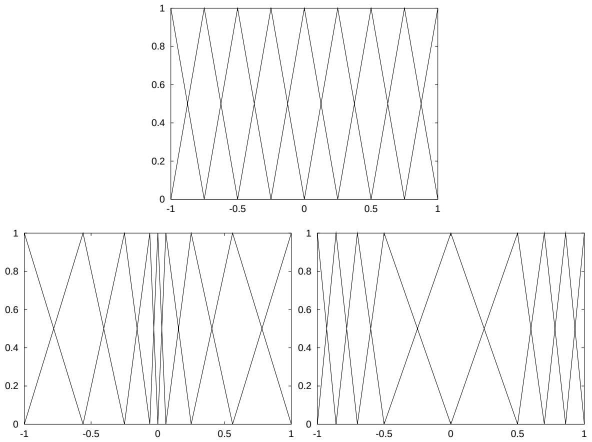

Figure 1: Fuzzy partitions with $a = 1$ (top), $a > 1$ (down left), and $a < 1$ (down right)

of readability. On the other hand, the KB design methods sometimes lead to a certain overfitting to the training data set used for the learning process.

To avoid these problems, our genetic process uses a multiobjective GA with two goals:

- Minimise the classification error percentage over the training data set.
- Design a compact and interpretable KB. This objective is performed by penalising FRBCSs with a large number of selected features and high granularity.

The next subsections describe the main components of the genetic learning process.

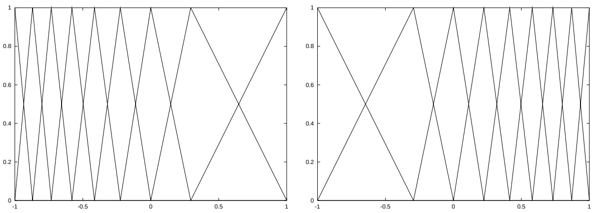

Figure 2: Fuzzy partitions with $S = 1$ (left with $a > 1$ and right with $a < 1$)

3.1 Encoding the DB

The two main DB components of the selected variables are the number of linguistic terms and the membership functions that define their semantics. Therefore, each chromosome will be composed of three parts:

- Relevant variables (C_1): For a classification problem with *N* variables, the selected features are stored into a binary coded array of length N. In this array, an 1 indicates that the corresponding variable is selected for the FRBCS.

- Number of labels (C_2): The number of labels per variable is stored into an integer array of length *N*. In this contribution, the possible values considered are taken from the set $\{1, \ldots, 5\}$.

- Sensibility parameters (C_3): An array of lenght $N \times 2$, where the sensibility parameters (*a*,*S*) are stored for each variable. In our case, the range considered for the parameter *a* is the interval $(0, 8]$.

If v_i is the bit that represents whether the variable i is selected and l_i is the granularity of variable i, a graphical representation of the chromosome is shown next:

$$C_1 = (v_1, \ldots, v_N) \quad C_2 = (l_1, \ldots, l_N)$$

$$C_3 = (a_1, \ldots, a_N, S_1, \ldots, S_N)$$

$$C = C_1 C_2 C_3$$

3.2 Evaluating the chromosome

There are three steps that must be done to evaluate each chromosome:

1. Generate the fuzzy partitions using the information contained in the chromosome. Obviously, this process is only applied to the selected variables ($v_i = 1$ and $l_i > 1$). First, each variable is linearly mapped from its working range to $[-1, 1]$. In a second step, uniform fuzzy partitions for all the variables are created considering the number of labels per variable (l_i). Finally, the non-linear scaling function with its sensibility parameters (a_i, S_i) is applied to the definition points of the membership functions obtained in the previous step, obtaining the whole DB definition.

2. Generate the RB by running a fuzzy rule learning method considering the DB obtained in the previous step.

3. Calculate the two values of the evaluation function:

   - CPE: classification percentage error over the training set.

   - $SV \cdot AL$: with SV being the number of selected variables and AL being the averaged granularity of the selected variables.

3.3 Genetic operators

The initial population is selected considering several groups of chromosomes, each one of them with different percentage for the selected variables (randomly chosen).The remaining values of the chromosome are chosen at random. For the rest of GA components, the following operators are considered.

3.3.1 Selection

We have used the selection mechanism of MOGA [9], which is based on the definition of Pareto-optimality. It is said that a solution dominates another when the former achieves better or equal values than the latter in all but one objective, where the former outperforms the latter. Hence, the pareto is composed of all the non-dominated solutions.

Taking this idea as a base, MOGA assigns the same selection probability to all non-dominated solutions in the current population. The method involves dividing the population into several classes depending on the number of individuals dominating the members of each class. Therefore, the selection scheme of our multiobjective GA involves the following five steps:

1. Each individual is assigned a rank equal to the number of individuals dominating it plus one (chromosomes encoding non-dominated solutions receive rank 1).

2. The population is increasingly sorted according to that rank.

3. Each individual is assigned a selection probability which depends on its ranking in the population, with lower ranking receiving lesser probabilities.

4. The selection probability of each equivalence class (group of chromosomes with the same rank, i.e., which are non-dominated among them) is averaged.

5. The new population is created by following the Baker's stochastic universal sampling [1].

3.3.2 Crossover

Two different crossover operators are considered depending on the two parents' scope:

- *Crossover when both parents have the same selected variables and equal granularity level per variable:* If the two parents have the same values in C_1 and C_2, the genetic search has located a promising space zone that has to be adequatelly exploitated. This task is developed by applying the max-min-arithmetical (MMA) crossover operator [13] in the chromosome part based on real-coding scheme (parameters a_i) and obviously by maintaining the parent C_1 and C_2 values in the offspring. Both combinations of parameter S_i are tested and the best two chromosomes are selected.

- *Crossover when the parents encode different selected variables or granularity levels:* This second case highly recommends the use of the information encoded by the parents to explore the search space in order to discover new promising zones. So, a standard crossover operator is applied over the three parts of the chromosome. This operator performs as follows: a crossover point p is randomly generated in C_1 and the two parents are crossed at the p-th variable in all the chromosome parts, thereby producing two meaningful descendents.

3.3.3 Mutation

Three different operators are used, each one of them acting on different chromosome parts:

- *Mutation on C_1 and on the second part of C_3 (parameters S_i)*: As these parts of the chromosome are binary coded, a simple binary mutation is developed, flipping the value of the gene.
- *Mutation on C_2*: The mutation operator selected for the granularity levels is similar to the one proposed by Thrift in [20]. A local modification is developed by changing the number of labels of the variable to the immediately upper or lower value (the decision is made at random). When the value to be changed is the lowest (1) or the highest one, the only possible change is developed.
- *Mutation on first part of C_3 (parameters a_i)*: As this part is based on a real-coding scheme, Michalewicz's non-uniform mutation operator is employed [18].

4 Experimentation

We have applied the learning method to an example base with a high number of features, Sonar data set [12], which has 208 instances of a sonar objective classification problem. Each one of these instances is described by 60 features to discriminate between a sonar output corresponding to a cylindrical metal or an approximately cylindrical rock. The training set contains 104 elements and the test set contains the remaining 104 elements. Table 1 shows the parameter values considered for the experiments developed.

Table 1: Parameter values

Parameter	Value
Granularity values	$\{1, \ldots, 5\}$
Population size	100
Crossover probability	0.6
Mutation probability	0.2
Number of generations	$\{100, 500\}$

The best results obtained by our genetic learning process for the two FRMs considered are shown in Table 2. The best results found with the Wang and Mendel's RB generation method considering all the features selected and the same number of labels for each one of them are also shown in the top line of each FRM. The table contains the following columns:

- **FRM**: Fuzzy Reasoning Method used.
- **SV**: Number of selected variables.
- **AL**: Average of the number of labels considered for the selected variables.

- **NR**: The number of rules of the FRBCS.
- **% tra**: Classification percentage error obtained in the training data set.
- **% tst**: Classification percentage error obtained in the test data set.

Table 2: Best results obtained (% error)

FRM	SV	AL	NR	% tra	% tst
	60	3	104	0.9	23.1
Maximum	9	3.6	101	0.0	16.3
	8	3.6	91	2.8	13.4
	6	3.6	91	4.8	15.3
	4	4.2	72	7.7	19.2
	4	3.2	55	15.3	15.3
	3	3.6	42	19.2	19.2
	2	2.5	9	27.8	25.0
	60	3	104	2.8	25.9
Normalised Sum	9	4.1	100	0.0	18.2
	7	4.1	92	1.9	19.2
	6	3.8	93	6.7	13.4
	5	4.0	76	12.5	16.3
	4	3.7	65	15.3	19.2
	3	4.0	25	19.2	20.1
	2	4.0	16	24.1	22.1

As it can be observed, the proposed method achieves a significant reduction in the number of variables selected (about the 90% of the original number of features, or even more in some cases) even with an important increase of the generalization capability (classification rate over the test data set). Besides, many solutions present also a significant decrease in the number of rules, reducing the complexity of the KB. Therefore, our multiobjective GA provides a wide set of solutions that permit an adequate choice depending on the main goal required: good performance or high degree of interpretability.

5 Conclusions

This contribution has proposed a multiobjective genetic process for jointly performing feature selection and DB components learning, which is combined with an efficient fuzzy classification rule generation method to obtain the complete KB for a descriptive FRBCS. Our method achieves an important reduction of the relevant variables selected for the final system and also adapts the fuzzy partition of each variable to the problem being solved. So, we can conclude that the proposed method allows us to significantly enhance the interpretability and accuracy of the FRBCSs generated. We have used a simple RB generation algorithm but another more accurate one can be used, having in mind its run time. Our future work will focus on improving the performance of the multiobjective

GA by using a niching technique or employing a co-evolutive GA and on comparing the results with other feature selection approaches.

References

[1] J. E. Baker. Reducing bias and inefficiency in the selection algorithms. In *Proc. of the Second International Conference on Genetic Algorithms (ICGA'87)*, pages 14–21, Hillsdale, 1987.

[2] 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):135–157, 2001.

[3] Z. Chi, H. Yan, and T. Pham. *Fuzzy algorithms with applications to image processing and pattern recognition*. World Scientific, 1996.

[4] O. Cordón, M. J. del Jesus, and F. Herrera. Genetic learning of fuzzy rule-based classification systems co-operating with fuzzy reasoning methods. *International Journal of Intelligent Systems*, 13(10/11):1025–1053, 1998.

[5] O. Cordón, M. J. del Jesus, and F. Herrera. A proposal on reasoning methods in fuzzy rule-based classification systems. *International Journal of Approximate Reasoning*, 20(1):21–45, 1999.

[6] 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.

[7] O. Cordón, F. Herrera, and P. Villar. Analysis and guidelines to obtain a good uniform fuzzy partition granularity for fuzzy rule-based systems using simulated annealing. *International Journal of Approximate Reasoning*, 25(3):187–216, 2000.

[8] O. Cordón, F. Herrera, and P. Villar. Generating the knowledge base of a fuzzy rule-based system by the genetic learning of the data base. *IEEE Transactions on Fuzzy Systems*, 9(4):667–675, 2001.

[9] C.M. Fonseca and P.J. Fleming. Genetic algorithms for multiobjective optimization: Formulation, discussion and generalization. In S. Forrest, editor, *Proc. of the Fifth International Conference on Genetic Algorithms (ICGA'93)*, pages 416–423. Morgan Kaufmann, 1993.

[10] 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.

[11] A. González and R. Pérez. Selection of relevant features in a fuzzy genetic learning algorithm. *IEEE Transctions on Systems, Man and Cybernetics - Part B*, 31, 2001.

[12] R. P. Gorman and T. J. Sejnowski. Analysis of hidden units in a layered network trained to classify sonar targets. *Neural Networks*, 1:75–89, 1988.

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

[14] H. Ishibuchi and T. Murata. A genetic-algorithm-based fuzzy partition method for pattern classification problems. In F. Herrera and J.L. Verdegay, editors, *Genetic Algorithms and Soft Computing*, pages 555–578. Physica-Verlag, 1996.

[15] H. Ishibuchi, K. Nozaki, and H. Tanaka. Construction of fuzzy classification systems with rectangular fuzzy rules using genetic algorithms. *Fuzzy Sets and Systems*, 65:237–253, 1994.

[16] R. Kohavi and G.H. John. Wrappers for feature subset selection. *Artificial Intelligence*, 97:273–324, 1997.

[17] H. Liu and H. Motoda. *Feature selection for knowledge discovery and data mining*. Kluwer Academic Publishers, 1998.

[18] Z. Michalewicz. *Genetic Algorithms + Data Structures = Evolution Program*. Springer-Verlag, 1996.

[19] T. Nakashima, T. Morisawa, and H. Ishibuchi. Input selection in fuzzy rule-based classification systems. In *Proc. of the Sixth IEEE International Conference on Fuzzy Systems (FUZZ-IEEE'97)*, pages 1457–1462, 1997.

[20] P. Thrift. Fuzzy logic synthesis with genetic algorithms. In *Proc. Fourth Int. Conference on Genetic Algorithms (ICGA'91)*, pages 509–513, 1991.

[21] L. X. Wang and J. M. Mendel. Generating fuzzy rules by learning from examples. *IEEE Transactions on Systems, Man, and Cybernetics*, 25(2):353–361, 1992.