Evolving Fuzzy Rule Based Classifiers with GA-P: A Grammatical Approach

Santiago García, Fermín González, and Luciano Sánchez

Departamento de Informática. Universidad de Oviedo. Spain,
{carbajal, fermingm, luciano}@lsi.uniovi.es

Abstract. Genetic Programming can be used to evolve Fuzzy Rule-based classifiers [7]. Fuzzy GP depends on a grammar defining valid expressions of fuzzy classifiers, and guarantees that all individuals in the population are valid instances of it all along the evolution process. This is accomplished by restricting crossover and mutation so that they only take place at points of the derivation tree representing the same non-terminal, thus generating valid subtrees [13].
In Fuzzy GP, terminal symbols are fuzzy constants and variables that are chosen beforehand. In this work we propose a method for evolving both fuzzy membership functions of the variables and the Rule Base. Our method extends the GA-P hybrid method [6] by introducing a new grammar with two functional parts, one for the Fuzzy Rule Base (GP Part), and the other for the constants that define the shapes of the fuzzy sets involved in the Fuzzy Rule Base (GA Part). We have applied this method to some classical benchmarks taken from the collection of test data at the UCI Repository of Machine Learning Databases [9].

1 Introduction.

When applying GP to the design of fuzzy rule-based Systems the main open problems are the implementation of pure reinforcement learning, allowing rule chaining, and including the definition of the membership function in the genetic coding [7][2][12]. This last point was addressed in some GA related works (see [4] [11]), but these methods cannot be directly extrapolated to GP case.

L.M Howard and D.J. D'Angelo introduced GA-P in [6], an hybrid of traditional Genetic Algorithms and Genetic Programming able to effectively search for the values of the constants involved in GP expressions. In previous works, we adapted GA-P to imprecise, interval valued data [5] and compared it to genetically tuned fuzzy rule-based knowledge bases. Now we propose a method that we call Fuzzy GA-P to concurrently evolve the Fuzzy Rule Base and the membership functions. To test the viability of our method, we have applied it to a set of classification problems, and, in the future, will use it as a tool in some industrial applications [10].

Fuzzy GA-P describes Fuzzy Rule Based Classifiers using a grammar with two functional parts: the Rule Base (GP Part) plus the definition of the coefficients on which membership functions depend (GA Part). Both parts will evolve simultaneously by means of crossover and mutation operators designed for this

R. Poli et al. (Eds.): EuroGP'99, LNCS 1598, pp. 203–210, 1999.

representation. This way, we propose one solution to the problem of including the definition of the membership functions in the genetic coding of the individuals in Fuzzy GP.

2 Fuzzy Rule Based Classifiers based on a grammar.

In [7] Andreas Geyer Schulz introduced a grammar for deriving Fuzzy Rule Bases and combined a genetic algorithm with a context-free language to evolve classifier systems. The method was called Fuzzy GP. In Fuzzy GP, constants are regarded as terminal symbols, so their value is not affected by the learning algorithm.

A Fuzzy Rule Based Classifier can be described in terms of a BNF grammar that generates a set of IF-THEN rules that assigns the input patterns to a number of classes depending on the fuzzy values of some input variables. A generic grammar for this type of classifiers could be:

```
S := <Rule_Base>;
<Rule_Base> := <Rules> ;
<Rules> :=  <Rule> | <Rule> <Rules>  ;
<Rule> := IF <Antecedent> THEN <Consequent> ;
<Antecedent>:= ( <Assert> )
              | (  <Operator> <Antecedent>  <Antecedent> );
<Operator> := OR | AND ;
<Assert> := <Input_Variable> = <Input_Fuzzy_Value> ;
<Consequent> :=  <Output_Fuzzy_Value> ;
<Input_Fuzzy_Value> := FUZZY_SET_1  | FUZZY_SET_2|...|FUZZY_SET_K;
<Output_Fuzzy_Value>  := Class_1  | Class_2 |...| Class_m  ;
<Input_Variable> := X1 | X2 |...| XN ;
```

This grammar can be used to generate an initial population of classifiers for any problem that can be described as a set of input variables, a number of linguistic variables that partition every universe of discourse, and a number of classes to classify the input patterns. For instance:

```
IF (AND(X4= LARGE)(X1 = SMALL)) THEN CLASS_1

IF (AND
     (X6 = MEDIUM)
     (OR
       (OR (X4 = MEDIUM)(X2 = LARGE))
       (AND(X1 = LARGE)(X4 = LARGE))
     )
   ) THEN CLASS_2

IF (X2 = SMALL) THEN CLASS_3
```

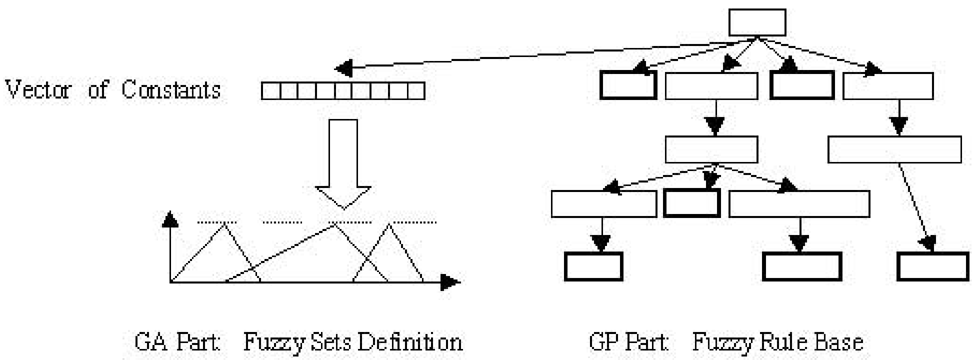

Fig. 1. A GA-P Fuzzy Rule Based Classifier. Individuals in GA-P comprise a chain of parameters and a tree. The tree defines a fuzzy rule bank. The chain of parameters codify the membership functions of the linguistic variables on which the rule bank depends.

3 A GA-P Grammar to evolve Fuzzy Rule Based Classifiers.

GA-P techniques [6] can be adapted to work with grammatically directed Fuzzy Rule Based Classifiers Evolution. To do this, we introduce a new grammar structure with two functional parts, as we mentioned in section 1. One tree describes the surface structure of the rule bank and one vector is used to codify the constants that define the fuzzy memberships of the linguistic variables on which the rules depend. For example, the grammar shown in the last section will be adapted as follows:

```
<GA-P_Rule_Base> := <Vector_Of_Constants> <Rules> ;
<Vector_Of_Constants> := <T1> ... <TM>;
<T1> := <digit> "." <digit>;
  ...
<TM> := <digit> "." <digit>;
<digit> := "0"|"1"|"2"|"3"|"4"|"5"|"6"|"7"|"8"|"9";
<Rules> :=  <Rule> |<Rule> <Rules>  ;
<Rule> := IF <Antecedent> THEN <Consequent> ;
<Antecedent>:= (<Assert> )
              | (  <Operator> <Antecedent>  <Antecedent> );
<Operator> := OR | AND ;
<Assert> := <Input_Variable> =  <Input_Fuzzy_Value>  ;
<Consequent> :=  <Output_Fuzzy_Value> ;
<Input_Fuzzy_Value> := FUZZY_SET_1  | FUZZY_SET_2 | FUZZY_SET_K ;
<Output_Fuzzy_Value>  := Class_1  | Class_2 |... | Class_m  ;
<Input_Variable> := X1 | X2 |... | XN ;
```

```
GA_PART:
(0.3 1.6 2.5) (2.1 3.9 4.3) (4.1 5.2 6.1)
GP_PART:
IF(AND(OR(AND(OR(X1 = LARGE)(X3 = MEDIUM)(X5 = MEDIUM))
(X3 = MEDIUM))(OR(AND(X4 = SMALL)(X2 = LARGE))(X3 = LARGE)))
THEN CLASS_2
IF ( X1 = SMALL ) THEN CLASS_3
IF (AND(X3 = MEDIUM)(OR(OR(X2 = MEDIUM)(X5 = LARGE))
(AND(X1 = LARGE)(X3 = LARGE))))
THEN CLASS_2
IF(AND(X1 = LARGE)(X3 = SMALL)) THEN CLASS_1
```

Fig. 2. Individual obtained as a solution of an example classification problem with fuzzy GA-P algorithms. The first part of this individual codifies the shapes of the three Fuzzy Sets SMALL, MEDIUM and LARGE, and the GP-PART reflects the derivation of the grammar for a standard Fuzzy Rule Based Classifier. All inputs were normalized between 0 and 6 so that all variables share the same fuzzy partition.

Figure 1 illustrates the coding of a rule bank as a GA-P individual. Assuming triangular memberships, each group of three constants is interpreted as the left, right and modal point respectively of one of the fuzzy sets that appear in the Fuzzy Rule Base.

In Figure 2 an individual obtained as a solution of a classification problem with fuzzy GA-P algorithms is shown. The first part of this individual codifies the shapes of three Fuzzy Sets (SMALL, MEDIUM and LARGE) that form a fuzzy partition of the range of all input variables (in this case, the same fuzzy partition is shared by all inputs; the modifications needed for evolving a different partition for every variable are straightforward). GP-PART section reflects the derivation of the grammar for a standard Fuzzy Rule Based Classifier, what was called by Zadeh the "surface structure" of the fuzzy rules.

Before doing the fitness calculation for each individual, the memberships that define all fuzzy partitions are set to the values stored in the GA-PART. To ensure semantic correctness the following operations are performed:

1. The values of each group of three constants are reordered after every application of the crossover operation.
2. The modal points of all fuzzy sets in every partition must also be ordered so that linguistic labels assigned to the elements of the partition (i.e. "SMALL", "MEDIUM", "LARGE") make sense.

3.1 Modified crossover and mutation to preserve grammatical correctness

3.2 Crossover operator

It is important that during the evolution process, the new individuals generated are correct and complete derivations of the grammar that describes the general

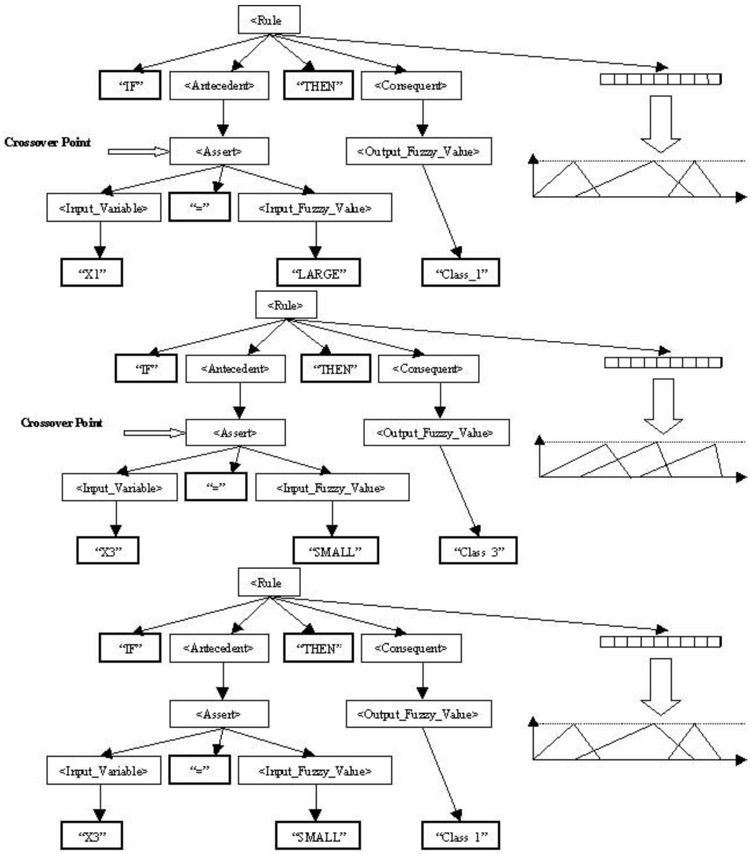

Fig. 3. Parents and offspring in GA-P crossover

structure of a Fuzzy Rule Based Classifier. So, the crossover operator must be adapted in a way that offspring produced by two syntactically valid parents will be in accordance with the grammar, too. This objective can be easily reached if we only let crossover take place in those points of the two parents that represent the same Non Terminal Symbol of the grammar. This Non Terminal Symbol will be part of the Rule Base, or part of the vector of constants. In the first case, a standard GP crossover operator is done, as shown in Figure 3. In the second case, we perform an one-point-crossover operation between the vectors of constants of the two parents [8], and the GP Part remains unchanged.

3.3 Mutation.

To guarantee that mutation produces syntactically valid individuals, we only let this operator take place at points of the derivation tree that represent a Non-Terminal symbol of the grammar with more than one possible derivation. Then

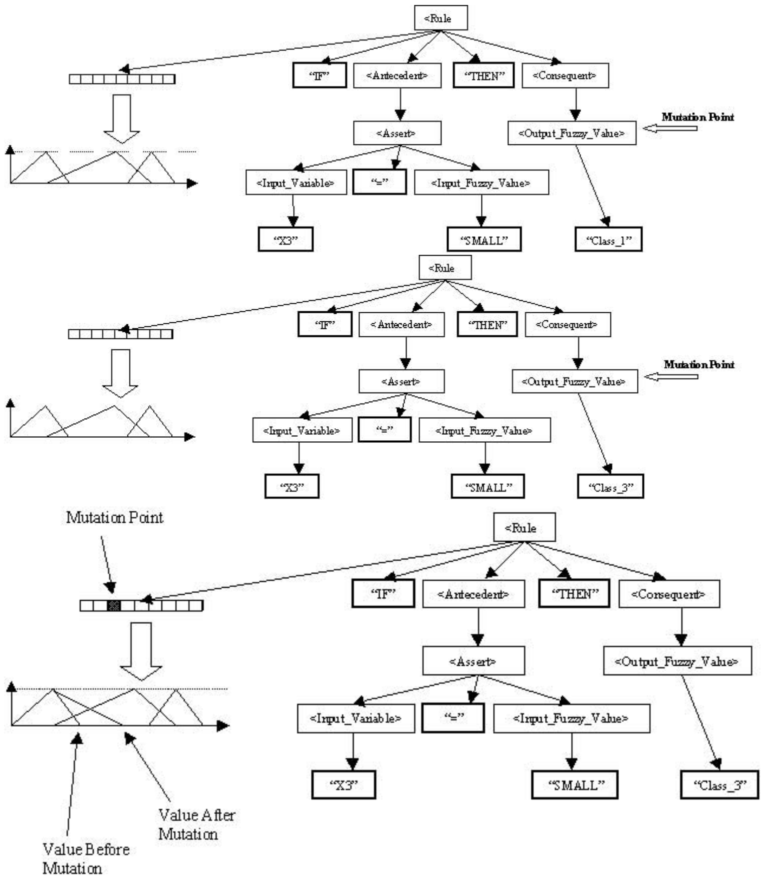

Fig. 4. Individuals before and after GP mutation (upper and center parts) and GA mutation (lower part).

mutation consists on the process of changing one derivation of a Non-Terminal for another.

Figure 4 shows how the mutation operator changes the first derivation of the Non Terminal Symbol `<Output_Fuzzy_Value>` by the third one, while ensuring that the individual is grammatically correct. In the same figure is also shown how mutation operates when the selected point lies inside the chain of parameters. In this last case, a standard GA mutation is applied.

4 Results

Seven standard classification problems were selected to compare Fuzzy GA-P with Geyer Schulz's Fuzzy GP [7]. The results of the experiments are shown

Problem	Train Fuz. GP	Train Fuz. GA-P	Test Fuz. GP	Test Fuz. GA-P
IRIS	97.63	97.45	95.6	95.1
PIMA	77.4	78.1	62.5	65.67
WINE	94	90	82.2	80
MONK1	100	97.3	98.5	95.1
MONK2	97.3	94.5	95.4	89.2
IONOSPHERE	96.3	89.3	86.9	80.2
SOYBEAN	83	91	73.1	76.9

Table 1. Comparison of numerical performance and complexity of the final model between Fuzzy GP and Fuzzy GA-P. Means of 50 runs.

Problem	Best Fuz. GP	Best Fuz. GA-P	Mean Fuz. GP	Mean Fuz. GA-P	Deviation Fuz. GP	Deviation Fuz. GA-P
IRIS	9	5	10.55	7.55	2.35	3.82
PIMA	8	6	8.45	8.70	0.73	3.34
WINE	11	7	13.1	9.80	2.34	3.38
MONK1	5	5	6.9	8.20	2.32	2.85
MONK2	11	9	12.15	10.70	2.03	2.23
IONOSPHERE	16	16	17.8	19.95	1.98	2.57
SOYBEAN	29	25	32.9	26.70	2.86	2.95

Table 2. Comparison of number of rules obtained using Fuzzy GP and Fuzzy GA-P. Means of 50 runs.

in tables 1 and 2. There are not significative numerical differences in the performance of both systems. Fuzzy GP and Fuzzy GA-P Results are not better than that ones obtained using pure GP of GA-tuned Fuzzy Systems. But there is a slight difference of complexity between Fuzzy GA-P and Fuzzy GP, which in turn produces models far more simple than GA-tuned Fuzzy Classifiers [4]. Since the objective of these kind of methods is to find an expression that makes sense linguistically while not being much less accurate than black-box models (i.e. neural networks, statistical classifiers or large GP expressions), we suggest that the ability of GA-P methods for searching numerical values contributes positively to the problem of finding a rule-based description of a classifier that is as short as possible.

5 Concluding remarks

Fuzzy GP is a kind of Evolutionary Fuzzy Rule-Based Systems that up to date has achieved less numerical accuracy than GA-based Fuzzy Rule Bank tuning procedures. On the other hand, the descriptions of the systems that Fuzzy GP produces are by far much simpler than GA-tuned fuzzy classifiers that use to be structured as a Mc Vicar-Gregor table. Fuzzy GP is capable to evolve models that do not depend on all input variables and its grammar can use linguistic connectives different from AND, which is the only connective allowed in most EFRBS.

In this work we have proposed one solution to an open problem in Fuzzy GP: evolving the memberships of the linguistic variables along with the structure of the rules. This evolution allows models with even simpler structure than Fuzzy GP, as our results show.

References

1. Blickle, T. "Evolving Compact Solutions in Genetic Programming: A Case Study". Parallel Problem Solving From Nature IV. Proceedings of the International Conference on Evolutionary Computation. Voigt, Ebeling, Rechenberg, Schwefel, eds. Springer-Verlag, LNCS 1141. pp 564-573, 1996.
2. Bonarini, A.: Elf: "Learning Incomplete Fuzzy rule Sets for an Autonomous Robot". Proceedings of the First European Congress on Intelligent Technologies and Soft Computing (EUFIT '93), ELITE Foundation, Aachen, D, 69-75. 1993.
3. Dietterich, T.G.: "Approximate Statistical Tests for Comparing Supervised Classifiers". Neural Computation, 1998.
4. Cordón, O., Herrera, F. "A three-stage evolutionary process for learning descriptive and approximate fuzzy logic controllers knowledge bases from examples". International Journal of Approximate Reasoning, vol 17, no. 4, pp 369-407, 1997.
5. Cordón, O. Herrera, F. , Sánchez, L. "Evolutionary Learning Processes for Data Analysis in Electrical Engineering Applicationes". Applied Intelligence, No. 10, pp. 5-24. 1999.
6. Howard L. M., D'Angelo, D. J.: "The GA-P: A Genetic Algorithm and Genetic Programming Hybrid". IEEE Expert. Vol 10, N. 3, June 1995.
7. Geyer-Schulz, A.: *Fuzzy Rule-Based Expert Systems and Genetic Machine Learning.* Studies in Fuzziness and Soft-Computing. Springer-Verlag. 1996.
8. Koza, J.: *Genetic Programming: on the Programming of Computers by Means of Natural Selection.* MIT Press, 1992.
9. Murphy, P.M,, Aha, D.W: UCI Repository of Machine Learning Databases. Mantained at the Department of Information and Computer Science, The University of California at Irvine, 1992.
10. Otero Rodríguez, J., García Carbajal, S. Sánchez Ramos, L.: "Fuzzy Control Applied to a Gas Transport Network In a Siderurgycal Environment". In Proceedings of 7th. International Conference in Information Processing and Management of Uncertainty in Knowledge-Based Systems I.P.M.U.'98. pp. 403-410. Paris. July, 1998.
11. Pérez Rodríguez, R.: *Fuzzy Rule Learning Using Genetic Algorithms.* Ph. D. Thesis. Department of Computer Science and artificial Intelligence. E.T.S. de Ingeniería Informática. University of Granada.
12. Valenzuela-Rendón, M.: "The Fuzzy Classifier System: A Classifier System for continuosly Varying Variables". Proceedings of the Fourth International Conference on Genetic Algorithms. Belew, Booker, eds. San Mateo, California. pp. 346-353. 1991.
13. Whigham, P.A.: *Grammatically Based Genetic Programming.* San Mateo, California. Morgan Kaufmann Publishers. 1991.