DECSAI

Department of Computer Science
and Artificial Intelligence

Different Approaches to Induce Cooperation in Fuzzy Linguistic Models Under the COR Methodology

Jorge Casillas, Oscar Cordón, Francisco Herrera

Technical Report #DECSAI-01-01-08
March, 2001

E.T.S. de Ingeniería Informática. Universidad de Granada.
E-18071 Granada, Spain – Phone +34-958-244019, Fax +34-958-243317

Different Approaches to Induce Cooperation in Fuzzy Linguistic Models Under the COR Methodology*

Jorge Casillas, Oscar Cordón, Francisco Herrera

Department of Computer Science and Artificial Intelligence
E.T.S. de Ingeniería Informática, Avda. Andalucía 38
University of Granada, E-18071 Granada, Spain

{casillas,ocordon,herrera}@decsai.ugr.es

Abstract

Nowadays, Linguistic Modeling is considered to be one of the most important areas of application for Fuzzy Logic. It is accomplished by linguistic Fuzzy Rule-Based Systems, whose most interesting feature is the interpolative reasoning developed. This characteristic plays a key role in their high performance and is a consequence of the cooperation among the involved fuzzy rules.

A new approach that makes good use of this aspect inducing cooperation among rules is introduced in this contribution: the Cooperative Rules methodology. One of its interesting advantages is its flexibility allowing it to be used with different combinatorial search techniques. Thus, four specific metaheuristics are considered: simulated annealing, tabu search, genetic algorithms and ant colony optimization. Their good performance is shown when solving a real-world problem.

1 Introduction

At present, system modeling is one of the main applications of fuzzy rule-based systems (FRBSs) [2, 17]. It may be considered as an approach to model a system making use of a descriptive language based on fuzzy logic with fuzzy predicates [19]. In this framework, one of the most interesting areas is *Linguistic Modeling*, where the interpretability of the obtained model is the main requirement. This task is developed by means of linguistic FRBSs, which use fuzzy rules composed of linguistic variables [22] that take values in a term set with a real-world meaning. Thus, the linguistic model consists of a set of linguistic descriptions regarding the behavior of the system being modeled [19].

Several tasks have to be performed in order to design an FRBS (linguistic model) for a concrete application. One of the most important and difficult ones is to derive an appropriate knowledge base (KB) about the problem being solved. The KB stores the available knowledge in the form of fuzzy linguistic IF-THEN rules. It consists of the rule

*This research is supported by CICYT, project PB98-1319

base (RB), constituted by the collection of rules in their symbolic forms, and the data base (DB), which contains the linguistic term sets and the membership functions defining their meanings.

In this sense, numerous methods have been proposed to automatically generate fuzzy rules from numerical data. Usually, they consider complex rule generation mechanisms based on neural networks [9, 15] or genetic algorithms (GAs) [5, 11, 18], among others.

Opposite to them, this contribution is devoted to present the Cooperative Rules (COR) learning methodology (initially proposed in [3, 4]), whose good performance is related to the consideration of cooperation among rules. The methodology simplifies the rule generation process and is capable of being used with any combinatorial search technique. Thus, four different neighborhood-based and global search metaheuristics will be selected to be applied in COR:

- simulated annealing (SA) algorithms,
- tabu search (TS) algorithms,
- GAs, and
- ant colony optimization (ACO) algorithms.

The paper is organized as follows. Section 2 introduces the methodology proposed to improve the accuracy of linguistic models by means of more cooperative rules. Section 3 shows how to learn cooperative rules with the four said metaheuristics. Section 4 analyzes the behavior of our proposals and other methods when solving a real-world problem. Finally, Sect. 5 outlines some concluding remarks.

2 The Cooperative Rules Methodology

A family of efficient and simple methods to derive fuzzy rules guided by covering criteria of the data in the example set, called *ad hoc data-driven methods*, has been proposed in the literature in the last few years [3]. Their high performance, in addition to their quickness and easy understanding, make them very suitable for learning tasks.

However, ad hoc data-driven methods usually look for the fuzzy rules with the best individual performance (e.g. [21]) and therefore the global interaction among the rules of the RB is not considered. This sometimes causes KBs with bad cooperation among the rules to be obtained, thus not being as accurate as desired. This is due to the interpolative reasoning developed by FRBSs, which is one of the most interesting features of these kinds of systems and plays a key role in their high performance, being a consequence of the cooperative action among the linguistic rules. Moreover, the fact of locally processing these rules makes these learning methods be more sensitive to noise.

With the aim of addressing these drawbacks keeping the interesting advantages of ad hoc data-driven methods, a new methodology to improve the accuracy obtaining better cooperation among the rules is proposed in [3, 4]: the COR methodology. Instead of selecting the consequent with the highest performance in each subspace like ad hoc data-driven methods usually do, the COR methodology considers the possibility of using another consequent, different from the best one, when it allows the FRBS to be more accurate thanks to having a KB with better cooperation.

In this way, its operation mode consists of two stages:

1. Obtain a set of candidate consequents for each rule.
2. Perform a *combinatorial search* among these sets looking for the combination of consequents with the best global accuracy.

A wider description of the COR-based rule generation process is shown in Fig. 1, whilst an example of the operation mode for a simple problem with two input variables and three labels in the output fuzzy partition is graphically illustrated in Fig. 2.

Since the search space – Fig. 2d in the example – tackled in step 2 of the algorithm is usually large, it is necessary to use approximate search techniques. Any combinatorial search approach may be used for such a purpose. In this contribution, four different well-known techniques are proposed: SA algorithms, TS algorithms, GAs, and ACO algorithms. The following section introduces the particular aspects for applying the considered techniques to the COR methodology.

3 Different Approaches to Learn Fuzzy Linguistic Rules Inducing Cooperation Among Them

3.1 Learning with Simulated Annealing

Introduction

SA [20] is a neighborhood-based search technique based on the analogy with the physical annealing process of solids. The SA-based algorithm begins with an initial solution and generates a neighbor of this solution by means of a suitable mechanism. If the latter is better than the former, the current solution is replaced by the generated neighbor; otherwise, this replacement is accomplished with a specific probability that will be decreased during the algorithm progress. This process is iterated a large number of times.

Simulated Annealing Algorithms Applied to the COR Methodology

The proposed COR-based learning method with an SA algorithm is characterized as follows:

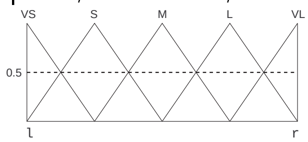
- *Representation* — An integer-valued vector (c) of size N_S is employed. Each cell of the vector represents the index of the consequent used to build the rule in the corresponding subspace:

$$\forall s \in \{1, \ldots, N_S\},\ c[s] = k_s\ s.t.\ B_{k_s} \in \mathbf{B^s}\ .$$

- *Objective function* — The said MSE function (Fig. 1) is used.
- *Cooling scheme* — The cooling scheme used is the exponential one proposed by Kirkpatrick [13] ($T_{t+1} = T_t \cdot C$).

Inputs:

- An input-output data set – $E = \{e_1, \ldots, e_l, \ldots, e_N\}$, with $e_l = (x_1^l, \ldots, x_n^l, y^l)$, $l \in \{1, \ldots, N\}$, N being the data set size, and n being the number of input variables – representing the behavior of the problem being solved.
- A fuzzy partition of the variable spaces, in our case, uniformly distributed fuzzy sets:



Let $\mathcal{A}_i$ be the set of linguistic terms of the i-th input variable – with $i \in \{1, \ldots, n\}$ –, and $\mathcal{B}$ be the set of linguistic terms of the output variable, with $|\mathcal{A}_i|$ ($|\mathcal{B}|$) being the number of linguistic terms of the i-th input (output) variable.

Algorithm:

1. *Generate candidate rules in each subspace* — For each n-dimensional fuzzy input subspace containing at least an example, $S_s = (A_1^s, \ldots, A_i^s, \ldots, A_n^s)$ such that $E'_s = \{e_l \in E \mid \mu_{A_1^s}(x_1^l) \cdot \ldots \cdot \mu_{A_n^s}(x_n^l) \neq 0\} \neq \emptyset$ – with $A_i^s \in \mathcal{A}_i$ being a label, $\mu_{A_i^s}(\cdot)$ being its membership function, $s \in \{1, \ldots, N_S\}$, and $N_S \leq \prod_{i=1}^{n} |\mathcal{A}_i|$ being the number of subspaces with examples –, do:
   (a) Let $\mathbf{B^s} = \{B_k \in \mathcal{B},\ k \in \{1, \ldots, |\mathcal{B}|\}\ s.t.\ \exists e_{l^s} \in E'_s\ with\ \mu_{B_k}(y^{l^s}) \neq 0\}$ be the set of linguistic labels in the output variable term set which contain examples belonging to E'_s, and let $|\mathbf{B^s}|$ be the number of candidate consequets in the subspace S_s.
   (b) For each linguistic label $B_{k^s}^s \in \mathbf{B^s}$ compute the covering value, CV, of the linguistic rule associated with the S_s subspace built using this term as a value in the consequent, $R_{k^s}^s$ = IF X_1 is A_1^s and ... and X_n is A_n^s THEN Y is $B_{k^s}^s$, over each example $e_{l^s} \in E'_s$ as follows:

$$CV(R_{k^s}^s, e_{l^s}) = Min(\mu_{A_1^s}(x_1^{l^s}), \ldots, \mu_{A_n^s}(x_n^{l^s}), \mu_{B_{k^s}^s}(y^{l^s})).$$

2. *Select the most cooperative rule in each subspace* — This stage is performed by running a combinatorial search algorithm to look for the combination $\{B_{k^1}^1, \ldots, B_{k^s}^s, \ldots, B_{k^{N_S}}^{N_S}\}$ with the best accuracy.
   To evaluate the quality of each solution, an index measuring the cooperation degree of the encoded rule set is considered. In our case, the algorithm uses a global error function called *mean square error* (MSE), which is defined as

$$\text{MSE} = \frac{1}{2 \cdot N} \sum_{l=1}^{N} (F(x_1^l, \ldots, x_n^l) - y^l)^2,$$

   with $F(x_1^l, \ldots, x_n^l)$ being the output obtained from the FRBS when the example e_l is used, and y^l being the known desired output. The closer to zero the measure, the greater the global performance and, thus, the better the rule cooperation.

Figure 1: Learning generic scheme followed by the COR methodology

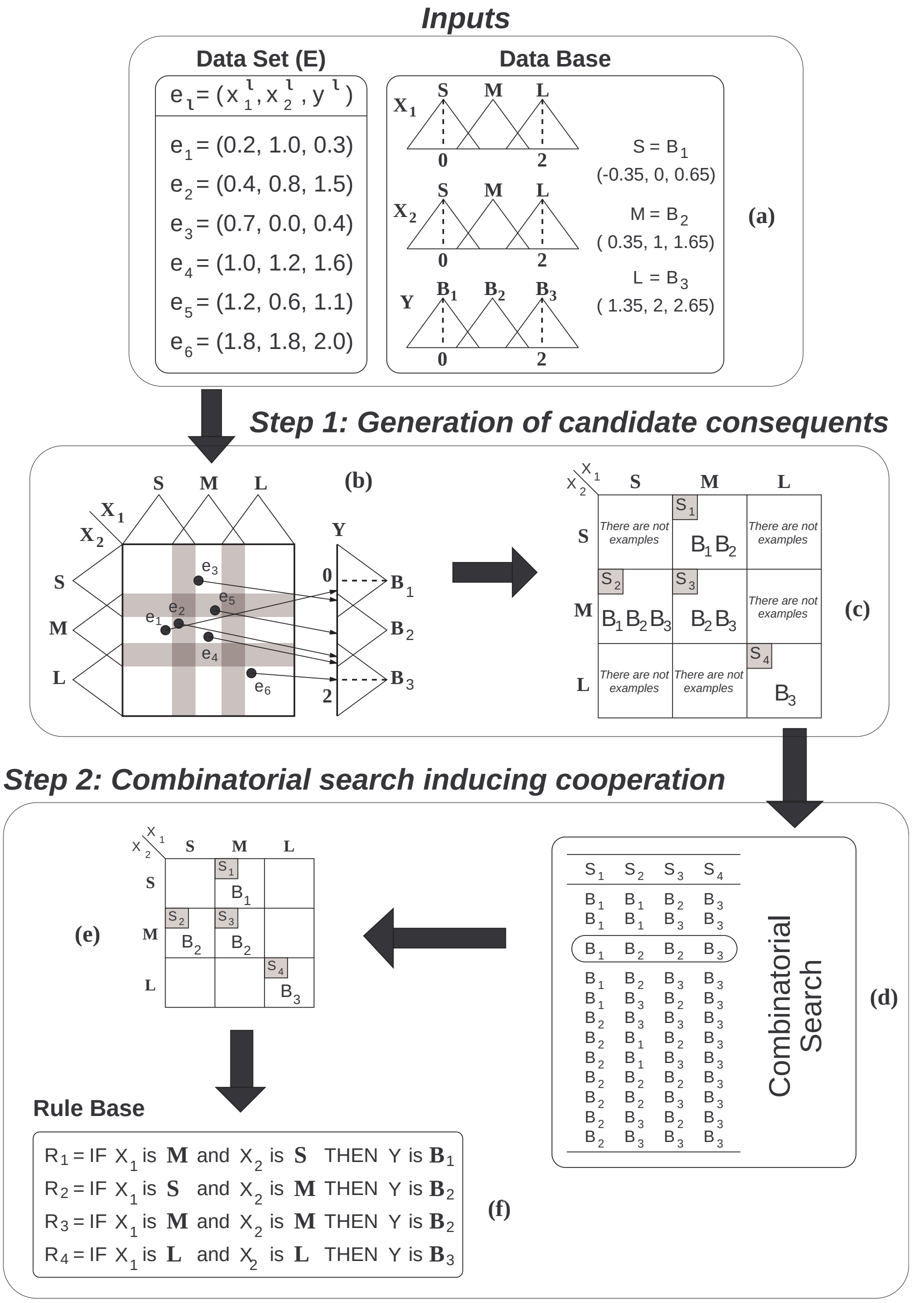


Figure 2: COR-based learning process for a simple problem with two input variables ($n = 2$) and three labels in the output fuzzy partition ($|\mathcal{B}| = 3$): (a) data set (E) and DB previously defined; (b) the six examples are located in four ($N_S = 4$) different subspaces that determine the antecedent combinations and candidate consequents of the rules; (c) set of possible consequents for each subspace; (d) combinatorial search accomplished within a space composed of twelve different combinations of consequents; (e) rule decision table for the third combination; (f) RB generated from the third combination

- *Trials per temperature* — The equilibrium at a specific temperature is achieved when a maximum number of neighbors has been generated.
- *Initial solution* — The initial solution is obtained by generating a possible combination at random.
- *Neighbor generation* — The operator considered to obtain a neighbor randomly selects a specific $s \in \{1, \ldots, N_S\}$ where $|\mathbf{B^s}| \geq 2$, and changes at random k^s by $k^{s\prime}$ such that $B^s_{k^{s\prime}} \in \mathbf{B^s}$ and $k^{s\prime} \neq k^s$.
- *Stopping criterion* — The algorithm stops when no neighbor is accepted for a specific temperature.

3.2 Learning with Tabu Search Algorithms

Introduction

TS [10] is an adaptive procedure for solving combinatorial optimization problems, which guides a hill-descendent heuristic to continue exploration without becoming confounded by a lack of improving moves, and without falling back into a local optimum from which it previously emerged.

At each iteration, an admissible move is applied to the current solution, transforming it into its neighbor with the smallest cost. Solutions that increase the cost function are permitted, the reverse move is prohibited for some iterations in order to avoid cycling. The restrictions are based on a short term memory function that determines how long a tabu restriction will be enforced or, alternatively, which moves are admissible at each iteration.

Tabu Search Algorithms Applied to the COR Methodology

The proposed COR-based learning method with a TS algorithm has the following components:

- *Representation* — The same proposed for the SA approach (Sect. 3.1).
- *Objective function* — The said MSE function (Fig. 1) is used.
- *Tabu criterion* — The tabu list (or history record) contains the moves applied to obtain the most recently accepted solutions. The movement considered is the pair subspace–consequent, i.e., $(s, k^{s\prime})$.
- *Aspiration criterion* — Moves that yield solutions better than the best one obtained in the search are allowed even if they are tabu active.
- *Initial solution* — It is obtained as in the SA approach (Sect. 3.1).
- *Neighbor generation* —The same proposed in the SA approach (Sect. 3.1).

- *Intensification restart* — The current solution is replaced by the best one obtained till now when a specific *intensification convergence criterion* is meet. The tabu list size is also changed by randomly reducing or increasing it by 50%. The convergence criterion involves making a restart when the *best* and *current costs* differ during a specific number of iterations.

- *Diversification restart* — A *long term memory* is used to keep an account of the use frequency of each successful move. To do that, the array LTM is initialized to zero and whenever a move is accepted (say $(s, k^{s\prime})$), its corresponding counter is increased by one, i.e., $LTM[s][k^{s\prime}] \leftarrow LTM[s][k^{s\prime}] + 1$.

  Subsequently, when a specific *diversification convergence criterion* is meet, a new solution is generated by selecting a consequent in each subspace according to a probability inversely proportional to the corresponding LTM value:

$$c[s] = k^s \;\; with\ a\ probability\ of \;\; \frac{\dfrac{1}{LTM[s, k^s] + 1}}{\displaystyle\sum_{B^s_{q^s} \in \mathbf{B^s}} \dfrac{1}{LTM[s, q^s] + 1}} \,.$$

  The convergence criterion is meet when the *best cost* has not been improved during a specific number of iterations.

3.3 Learning with Genetic Algorithms

Introduction

GAs are general-purpose global search algorithms that use principles inspired by natural population genetics to evolve solutions to problems. The basic principles of the GAs were first laid down rigorously by Holland [12] and are well described in many texts as [14].

The basic idea is to maintain a population of knowledge structures that evolves over time through a process of competition and controlled variation. Each structure in the population represents a candidate solution to the specific problem and has an associated *fitness* to determine which structures are used to form new ones in the process of competition. The new individuals are created using genetic operators such as crossover and mutation. Figure 3 shows the structure of a simple GA.

Genetic Algorithms Applied to the COR Methodology

The proposed COR-based learning method with a GA is characterized as follows:

- *Coding scheme* — The same proposed for the SA approach (Sect. 3.1).

- *Fitness function* — The objective will be to minimize the said MSE function (Fig. 1).

- *Genetic approach* — An elitist generational GA with the Baker's stochastic universal sampling procedure [1].

```
Procedure Genetic Algorithm
begin
    t = 0;
    initialize P(t);
    evaluate P(t);
    while (not termination-condition) do
        t = t + 1;
        select P(t) from P(t − 1);
        cross P(t) with an specific probability;
        mutate P(t) with an specific probability;
        evaluate P(t);
    end-while
end
```

Figure 3: Basic structure of a GA

- *Initial pool* — The population is initially generated with the first individual as follows

$$\forall s \in \{1, \ldots, N_S\},$$

$$c_1[s] = arg \max_{B_{k_s} \in \mathbf{B^s}} \left\{ \max_{e_{l^s} \in E'_s} \{CV(R^s_{k^s}, e_{l^s})\} \cdot \frac{\sum_{e_{l^s} \in E'_s} CV(R^s_{k^s}, e_{l^s})}{|E'_s|} \right\},$$

and the remaining chromosomes generated at random:

$$\forall p \in \{2, \ldots, pool\_size\},\ \forall s \in \{1, \ldots, N_S\},\ c_p[s] =\ some\ k_s\ s.t.\ B_{k_s} \in \mathbf{B^s}\ .$$

- *Crossover* — The standard two-point crossover is used.
- *Mutation* — The same that the said SA neighbor generation mechanism (Sect. 3.1).

3.4 Learning with Ant Colony Optimization Algorithms

Introduction

ACO algorithms [7] constitute a new family of global search bio-inspired algorithms that has recently appeared. Since the first proposal, the Ant System algorithm [8] – applied to the Traveling Salesman Problem –, numerous models has been developed to solve a wide set of optimization problems (refer to [7] for a review of models and applications).

ACO algorithms draw inspiration from the social behavior of ants to provide food to the colony. In the food search process, consisting of the food find and the return to the nest, the ants deposit a substance called *pheromone*. The ants have the ability of sniffing the pheromone and pheromone trails guide the colony during the search. When an ant is located at a branch, it decides to take the path according to the probability defined by the pheromone existing in each trail. In this way, the depositions of pheromone terminate in

1. *Set a node for each subspace* — Use a node for each n-dimensional fuzzy input subspaces containing examples (S_s), thus having a total of N_S subspace nodes.

2. *Link the subspaces to consequents* — The subspace S_s will be linked to the consequent B_j, with $j \in \{1, \ldots, |\mathcal{B}|\}$, if and only if it meets the following condition:

$$\exists e_l \in E \textit{ such that } \mu_{A_1^s}(x_1^l) \cdot \ldots \cdot \mu_{A_n^s}(x_n^l) \cdot \mu_{B_j}(y^l) \neq 0 \ .$$

That is, if there is at least one example located in the fuzzy input subspace that is covered by such a consequent.

Figure 4: Graph construction process

constructing a path between the nest and the food that can be followed by new ants. The progressive action of the colony members involves the length of the path is progressively reduced. The shortest paths are finally the more frequently visited ones and, therefore, the pheromone concentration is higher on them. On the contrary, the longest paths are less visited and the associated pheromone trails are evaporated.

The basic operation mode of ACO algorithms is as follows [8]: at each iteration, a population of a specific number of ants progressively construct different tracks on the graph (i.e., solutions to the problem) according to a *probabilistic transition rule* that depends on the available information (heuristic and pheromone trails). After that, the pheromone trails are updated. This is done by first decreasing them by some constant factor (corresponding to the evaporation of the pheromone) and then reinforcing the solution attributes of the constructed solutions considering their quality. This task is developed by the *global pheromone trail update rule*.

Ant Colony Optimization Algorithms Applied to the COR Methodology

The proposed COR-based learning method with an ACO algorithm has the following main aspects:

- *Problem representation* — To use ACO algorithms in the COR methodology, it is convenient to see it as a combinatorial optimization problem with the capability of being represented on a graph. In this way, we can face the problem interpreting the COR methodology as the way of assigning consequents ($B_j \in \mathcal{B}$) – i.e., labels of the output fuzzy partition – to n-dimensional fuzzy input subspaces containing examples (S_s) with respect to an optimality criterion (MSE).

  Thus, the graph is constructed taking the steps described in Fig. 4. Following these steps, the graph corresponding to the example presented in Fig. 2 would be the one shown in Fig. 5.

- *Heuristic information* — The heuristic information on the potential preference of selecting a specific consequent, B_j, in each antecedent combination (subspace) is

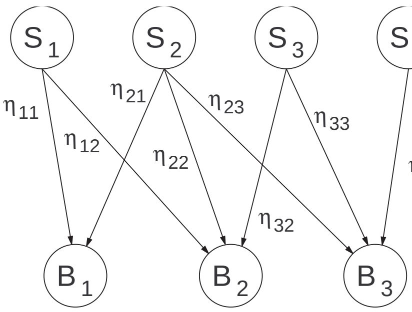


Figure 5: ACO graph corresponding to the example of Fig. 2

determined as follows:

$$\eta_{sj} = \max_{e_{l^s} \in E'_s} \{CV(R_j^s, e_{l^s})\} \cdot \frac{\sum_{e_{l^s} \in E'_s} CV(R_j^s, e_{l^s})}{|E'_s|},$$

with $R_j^s =$ IF X_1 is A_1^s and ... and X_n is A_n^s THEN Y is B_j.

- *Pheromone initialization* — The initial pheromone value of each assignment is obtained as follows:

$$\tau_0 = \frac{\sum_{s=1}^{N_S} \max_{j=1}^{|\mathcal{B}|} \eta_{sj}}{N_S}.$$

  In this way, the initial pheromone will be the mean value of the path constructed taking the best consequent in each rule according to the heuristic information (greedy assignment).

- *Fitness function* — The said MSE function (Fig. 1).

- *ACO approach* — The well-known Ant System [8] algorithm is considered.

4 Experimental Study Solving the Rice Taste Evaluation Problem

This experimental study will be devoted to analyze the behavior of the proposed COR-based methods. With this aim, we have chosen the problem of rice taste evaluation [16]. We will analyze the accuracy of the linguistic models generated from the processes introduced in the previous section compared to two well-known ad hoc data-driven methods, the ones proposed by Wang and Mendel (WM) [21] and Nozaki, Ishibuchi, and Tanaka (NIT) [16].

With respect to 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 the defuzzification operator [6].

4.1 Problem Description

Subjective qualification of food taste is a very important but difficult problem. In the case of the rice taste qualification, it is usually put into effect using a subjective evaluation called the *sensory test*. In this test, a group of experts, usually composed of 24 persons, evaluate the rice according to a set of characteristics associated with it. These factors are *flavor*, *appearance*, *taste*, *stickiness*, and *toughness* [16].

Because of the large quantity of relevant variables, the problem of rice taste analysis becomes very complex, thus requiring the design of a model representing the existing non-linear relationships. Moreover, the problem-solving goal is not only to obtain an accurate model, but to obtain a user-interpretable model as well, capable of putting some light on the reasoning process performed by the expert for evaluating a kind of rice in a specific way. Due to all these reasons, in this section we deal with obtaining a linguistic model to solve the said problem.

To do that, we use the data set presented in [16]. This set is composed of 105 data arrays collecting subjective evaluations of the six variables in question (the five mentioned and the overall evaluation of the kind of rice), made up by experts on the number of kinds of rice grown in Japan (e.g., Sasanishiki, Akita-Komachi, etc.). The six variables are normalized, thus taking values in the real interval [0, 1].

With the aim of not biasing the learning, we have randomly obtained ten different partitions of the mentioned set, composed by 75 pieces of data in the training set and 30 in the test one, to generate 10 different linguistic models in each experiment. Two labels will be considered to partition each linguistic variable domain.

4.2 Experiments and Analysis Results

The following values have been considered for the parameters of each COR-based method:

- *COR-SA*: cooling factor (C), 0.9; trials per temperature, 32; and initial temperature, 70.
- *COR-TS*: number of iterations, 150; initial tabu list size, 32; intensification convergence criterion, 10 iterations; and diversification convergence criterion, 7 iterations;
- *COR-GA*: number of generations, 500; population size, 61; crossover probability, 0.6; and mutation probability, 0.2.
- *COR-AS*: number of ants, 32; pheromone evaporation (ρ), 0.6; pheromone trail weight (α), 1; and heuristic information weight (β), 2.

The results obtained by the six methods analyzed are collected in Table 1, where #R stands for the number of rules, MSE_{tra} and MSE_{tst} respectively for the error obtained over the training and test data sets, EBS for the number of evaluations needed to obtain the best solution, and $\bar{x}$ and σ respectively for the arithmetic mean and standard deviation values over the 10 models generated by each method. The best results are shown in boldface.

In view of the obtained results, the methods based on the COR methodology perform a good learning process generating accurate models in both approximation – MSE_{tra} – and generalization – MSE_{tst} –, overcoming the WM and NIT methods.

Table 1: Results obtained in the rice taste evaluation problem

	#R		MSE$_{tra}$		MSE$_{tst}$		EBS	
Method	$\bar{x}$	σ	$\bar{x}$	σ	$\bar{x}$	σ	$\bar{x}$	σ
WM	**15**	0.632456	0.013284	0.005987	0.013119	0.004239	–	–
NIT	64	**0.000000**	0.008626	0.000345	0.009851	0.001931	–	–
COR-SA	32	**0.000000**	0.007076	**0.000571**	0.008012	0.001766	3,963	**110**
COR-TS	32	**0.000000**	0.007213	0.000553	0.008398	0.001638	**433**	118
COR-GA	32	**0.000000**	**0.006845**	0.000574	0.007830	**0.001457**	5,123	1,632
COR-AS	32	**0.000000**	0.006943	0.000602	**0.007702**	0.001594	1,108	307

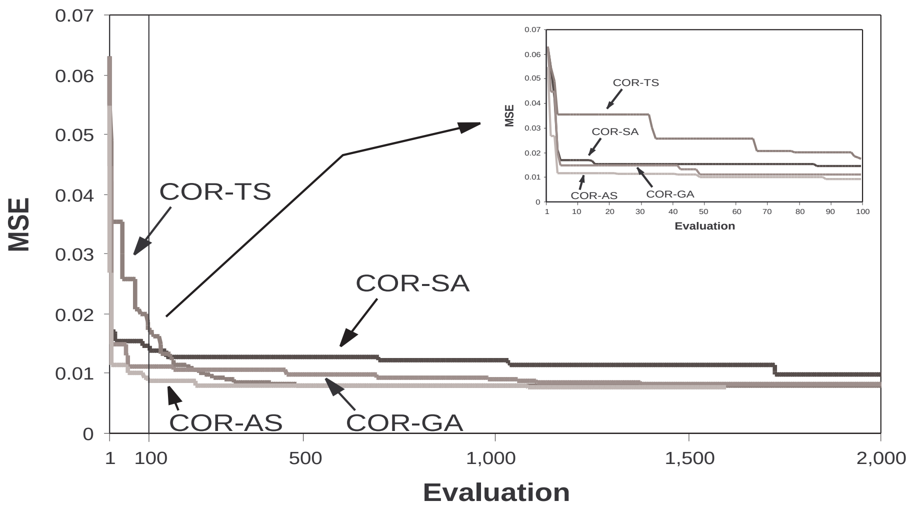


Figure 6: Evolution chart of the four COR-based methods in the first data set partition

Within the COR-based methods, the global search techniques (GAs and ACO) obtain better results than the neighborhood-based ones (SA and TS). Between the COR-GA and COR-AS methods, though they both generate models with similar accuracy, the latter only needs a fifth of the evaluations to find the solution, which is an interesting aspect to take into account. This fact seems to be related to the consideration of heuristic information made by ACO algorithms.

Figure 6 illustrates the behavior of the four analyzed techniques showing their evolution charts in the first data set partition. While the COR-TS method performs a gradual descent, the other three methods show a strong decrease at the beginning, although the COR-SA method becomes stabilized after several initial evaluations. As may be observed, the COR-AS method presents the best behavior quickly obtaining good solutions.

5 Concluding Remarks

A learning methodology to quickly generate accurate and simple linguistic models has been presented in this contribution: the COR methodology. It is based on considering the cooperation among the fuzzy rules in the generation process making good use of the interpolative reasoning developed by the finally designed FRBS. One of its interesting advantages is its flexibility allowing it to be used with different combinatorial search techniques.

Thus, four specific metaheuristics (SA, TS, GAs, and ACO) have been considered. Their good performance has been shown when solving a real-world problem. From this experimental study, the best results have been obtained by the ACO learning method thanks to it considers heuristic information to guide the search.

The obtained results lead us to conclude that the consideration of cooperative rules improves the performance of the linguistic models and the derivation of linguistic rules by firstly generating a candidate rule set and then searching the best combination of rules is a good way to accomplish this aspect.

References

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

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

[3] Casillas J., Cordón O., Herrera F. (2000) A methodology to improve ad hoc data-driven linguistic rule learning methods by inducing cooperation among rules. Technical Report #DECSAI-00-01-01, Dept. Computer Science and A.I., University of Granada, Spain.

[4] Casillas J., Cordón O., Herrera F. (2000) Improving the Wang and Mendel's fuzzy rule learning method by inducing cooperation among rules. Proceedings of the 8th Information Processing and Management of Uncertainty in Knowledge-Based Systems Conference, Madrid, Spain, 1682–1688.

[5] Cordón O., Herrera F., Hoffmann F., Magdalena L. (2001) Genetic fuzzy systems: evolutionary tuning and learning of fuzzy knowledge bases. World Scientific, Singapore. In press.

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

[7] Dorigo M., Di Caro G. (1999) The ant colony optimization meta-heuristic. In: Corne D., Dorigo M., Glover F. (Eds.) New Ideas in Optimization. McGraw-Hill, 11–32.

[8] Dorigo M., Maniezzo V., Colorni A. (1996) The ant system: optimization by a colony of cooperating agents. IEEE Transactions on Systems, Man, and Cybernetics. Part B: Cybernetics 26(1):29–41.

[9] Fullér R. (2000) Introduction to neuro-fuzzy systems. Springer-Verlag, Berlin/New York, Germany/USA.

[10] Glover F., Laguna M. (1997) Tabu search. Kluwer Academic, Dordrecht/Norwell, MA, USA.

[11] Herrera F., Verdegay J.L. (Eds.) (1996) Genetic algorithms and soft computing. Physica-Verlag, Heidelberg, Germany.

[12] Holland J.H. (1975) Adaptation in natural and artificial systems. Ann arbor: The University of Michigan Press.

[13] Kirkpatrick S. (1984) Optimization by simulated annealing: quantitative studies. Journal of Statistical Physics 34:975–986.

[14] Michalewicz Z. (1996) Genetic algorithms + data structures = evolution programs. Springer-Verlag, Berlin/New York, Germany/USA.

[15] Nauck D., Klawonn F., Kruse R. (1997) Fundations of neuro-fuzzy systems. John Wiley & Sons, New York, USA.

[16] Nozaki K., Ishibuchi H., Tanaka H. (1997) A simple but powerful heuristic method for generating fuzzy rules from numerical data. Fuzzy Sets and Systems 86:251–270.

[17] Pedrycz W. (Ed.) (1996) Fuzzy modelling: paradigms and practice. Kluwer Academic, Dordrecht/Norwell, MA, USA.

[18] Pedrycz W. (Ed.) (1997) Fuzzy evolutionary computation. Kluwer Academic, Dordrecht/Norwell, MA, USA.

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

[20] van Laarhoven P.J.M., Aarts E.H.L. (1987) Simulated annealing: theory and applications. Kluwer Academic, Dordrecht/Norwell, MA, USA.

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

[22] Zadeh L.A. (1975) The concept of a linguistic variable and its application to approximate reasoning. Information Science 8:199–249, 8:301–357, 9:43–80.