A Cooperative Coevolutionary Algorithm for Jointly Learning Fuzzy Rule Bases and Membership Functions*

Jorge Casillas
DECSAI, Univ. of Granada,
E-18071 Granada, Spain
casillas@decsai.ugr.es

Oscar Cordón
DECSAI, Univ. of Granada,
E-18071 Granada, Spain
ocordon@decsai.ugr.es

Francisco Herrera
DECSAI, Univ. of Granada,
E-18071 Granada, Spain
herrera@decsai.ugr.es

Abstract

When a whole knowledge base must be derived for a fuzzy rule-based system, learning methods usually address this task with two or more sequential stages by separately designing each of its components (mainly the rule base and the data base). Instead, we propose a simultaneous derivation process to properly consider their dependency. Since the problem complexity rises, the proposed method will be based on a cooperative coevolutionary algorithm.

Keywords: fuzzy models, learning, tuning, cooperative coevolution.

1 Introduction

Several tasks have to be performed in order to design a linguistic FRBS 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 base (RB), comprised of 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.

Numerous automatic methods have been developed to perform the derivation task. When only the derivation of the RB is addressed, methods generally operate in only one stage [10, 11]. However, methods that design both RB and DB are preferable since the automation is higher. In this case, we can distinguish between two different approaches:

- *Simultaneous derivation*: It relates to the process of directly obtaining the whole KB (RB and DB) from the available data in a simultaneous way [6]. This task is usually known as *learning process*.
- *Sequential derivation*: The task is divided into two or more stages, each of them performing a partial or complete derivation of the KB.

  Generally, one of the last stages adjusts the previously learnt/obtained DB with slight modifications to increase the system performance [4, 5]. This stage is known as *tuning process*. In most cases, a sequential process by firstly learning the RB and then tuning the DB is considered [2].

When the RB and the DB are simultaneously derived, the strong dependency of both components is properly addressed. However, the derivation process becomes significantly more complex because the search space rises and the selection of an appropriate search technique is crucial.

Recently, the coevolutionary paradigm [7] has shown an increasing interest thanks to its high ability to manage with huge search spaces and decomposable problems. The direct decomposition of the KB derivation process (thus obtaining two interdependent components, learning of the RB and DB) makes cooperative coevolutionary algorithms [9] very useful for this purpose.

*Supported by the Spanish CICYT, project PB98-1319

We propose a KB derivation method within this novel evolutionary paradigm. Actually, a method has been already proposed by Peña-Reyes and Sipper with this cooperative coevolutionary philosophy [8]. However, ur proposal performs a more sophisticated learning of the RB based on the COR methodology [1], whose good performance relates to the consideration of cooperation among rules.

In the following sections, the proposed KB derivation method, some experimental results, conclusions, and further work are shown.

2 A Cooperative Coevolutionary Algorithm for Jointly Learning Fuzzy Rule Bases and Membership Functions

Within the evolutionary computation field, a new paradigm has been recently proposed, coevolutionary algorithms [7]. They involve two or more species (populations) that permanently interact among them by a coupled fitness. Thereby, in spite of each species has its own coding scheme and reproduction operators, when an individual must be evaluated, its goodness will be calculated considering some individuals of the other species. This coevolution makes easier to find solutions to complex problems.

Intuitively, we may decompose the problem of deriving a proper KB for an FRBS into two subtasks: to obtain linguistic rule symbolic representations (learning the RB) and to define membership function shapes (learning the DB). Therefore, our algorithm consists of two species that cooperate to build the whole solution.

In the following subsections, a formulation for both learning tasks and the cooperative coevolutionary algorithm are introduced.

2.1 The Knowledge Base Derivation Process

The RB learning task is based on the COR methodology proposed in [1]. Let E be the input-output data set, $e_l = (x_1^l, \ldots, x_n^l, y^l)$ one of its elements (example), and n be the number of input variables. Let $\mathcal{A}_i$ be the set of linguistic terms of the i-th input variable and $\mathcal{B}$ the set of linguistic terms of the output variable. Its operation mode is the following:

1. Define a set of fuzzy input subspaces, $\{S_s \mid s \in \{1, \ldots, N_S\}\}$, with the antecedent combinations containing at least a positive example, i.e., $S_s = (A_1^s, \ldots, A_i^s, \ldots, A_n^s) \in \mathcal{A}_1 \times \ldots \times \mathcal{A}_n$ such that $E_s' \neq \emptyset$ — with $A_i^s \in \mathcal{A}_i$, E_s' being the set of positive examples of the subspace S_s, and N_S the number of subspaces with positive examples —.

   In this contribution, the set of positive examples for the subspace S_s is defined as follows:

   $$E_s' = \{e_l \in E \ \mid \ \forall i \in \{1, \ldots, n\}, \\ \forall A \in \mathcal{A}_i, \ \mu_{A_i^s}(x_i^l) \geq \mu_A(x_i^l)\} \ .$$

2. For each subspace S_s, obtain a set of candidate consequents (i.e., linguistic terms of the output variable) $\mathbf{B^s}$ to build the corresponding linguistic rule.

   In this contribution, the set of candidate consequents for S_s is defined as follows:

   $$\mathbf{B^s} = \{B_k \in \mathcal{B} \mid \exists e_{l^s} \in E_s' \ where \\ \forall B_l \in \mathcal{B}, \ \mu_{B_k}(y^{l^s}) \geq \mu_{B_l}(y^{l^s})\} \ .$$

3. Perform a *combinatorial search* among these sets looking for the combination of consequents (one for each subspace) with the best global accuracy.

On the other hand, the derivation of the DB involves determining the shape of each membership function. These shapes will have a high influence in the FRBS performance. In this contribution, we will consider triangular-shaped membership functions.

2.2 The Cooperative Coevolutionary Algorithm

2.2.1 Cooperative Interaction Scheme

Let F_{ij} be the FRBS obtained by composing the subcomponents encoded in the chromosomes i and j of the species 1 (RBs) and 2 (membership functions), respectively. The objective will be to minimize the well-known *mean square error*:

$$\mathrm{MSE}_{ij} = \frac{1}{2 \cdot N} \sum_{l=1}^{N} \left(F_{ij}(x^l) - y^l \right)^2,$$

with N being the data set size, $F_{ij}(x^l)$ being the output obtained from the designed FRBS when the l-th example is considered, and y^l being the known desired output.

Each individual of species 1 or 2 is evaluated with the corresponding **fitness function** f_1 or f_2, which are defined as follows:

$$f_1(i) = \min_{j \in R_2 \cup P_2} \mathrm{MSE}_{ij}, \quad f_2(j) = \min_{i \in R_1 \cup P_1} \mathrm{MSE}_{ij},$$

with i and j being individuals of species 1 and 2 respectively, R_1 and R_2 being the set of the fittest individuals in the previous generation of the species 1 and 2 respectively, and P_1 and P_2 being individual sets selected at random from the previous generation of the species 1 and 2 respectively. The combined use of both kinds of sets makes the algorithm have a trade-off between exploitation ($R_{1|2}$) and exploration ($P_{1|2}$).

A **generational GA** scheme is followed in both species. Baker's stochastic universal sampling procedure together with an elitist mechanism that ensures to select the best individual of the previous generation are used.

The specific operators employed in each species are described in the following sections.

2.2.2 Species 1: Learning Fuzzy Rule Bases

An integer-valued vector (c) of size N_S (number of subspaces with positive examples) is employed as **coding scheme**. Each cell of the vector represents the index of the consequent used to build the rule in the corresponding subspace:

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

The **initial pool** of this species is generated building the first individual as follows

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

$$c_1[s] = \underset{k_s}{arg} \max_{B_{k_s} \in \mathbf{B^s}}$$

$$\max_{e_{l^s} \in E'_s} Min\left(\mu_{A_1^s}(x_1^{l^s}), \ldots, \mu_{A_n^s}(x_n^{l^s}), \mu_{B_{k^s}}(y^{l^s}) \right),$$

and the remaining chromosomes at random.

The standard two-point **crossover operator** is used. The **mutation operator** randomly selects a specific $s \in \{1, \ldots, N_S\}$ where $|\mathbf{B^s}| \geq 2$, and changes at random $c[s] = k^s$ by $c[s] = k^{s\prime}$ such that $B_{k^{s\prime}} \in \mathbf{B^s}$ and $k^{s\prime} \neq k^s$.

2.2.3 Species 2: Learning Fuzzy Membership Functions

As **coding scheme**, a 3-tuple of real values for each triangular membership function is used, thus being the DB encoded into a real-coded chromosome built by joining the membership functions involved in each variable fuzzy partition. A variation interval is associated to every gene to preserve meaningful fuzzy sets.

The **initial population** of this species is generated with a chromosome representing the original DB and the remaining chromosomes generated with values generated at random within the corresponding variation interval.

The max-min-arithmetical **crossover** operator [4] is considered. Its formulation avoids the violation of the restrictions imposed by the variation intervals. As regards the **mutation operator**, it simply involves changing the value of the selected gene by other value obtained at random within the corresponding variation interval.

3 Experimental Results

This experimental study will be devoted to analyze the behavior of the proposed derivation method (CORMF-CC) when solving the problem of estimating the maintenance costs of the medium voltage electrical network in a town considering four different criteria (input variables) [3]. We will deal with a sample of 1,059 towns, randomly divided in two subsets, the training and test ones, with an 80%-20% of the original size respectively. *Five linguistic terms* for each variable are considered.

We will analyze linguistic models generated from the proposed process compared to the four following methods: the well-known ad hoc data-driven method proposed by Wang and Mendel (WM) [11]; a GA-based learning method following the COR methodology (COR-GA) [1]; and two sequential methods, WM+Tun and COR-GA+Tun, that firstly perform a learning of the RB with WM or COR-GA, respectively, and then adjust the membership functions with the tuning method proposed in [2].

The results obtained by the five methods analyzed are collected in Table 1, where MSE_{tra} and MSE_{tst} respectively stand for the error obtained over the training and test data sets. The best results are shown in boldface. A total of 66 fuzzy linguistic rules were obtained in all cases.

Table 1: Results obtained

Method	$\mathbf{MSE}_{tra}$	$\mathbf{MSE}_{tst}$
WM	71,294	80,934
COR-GA	67,237	69,457
WM+Tun	23,440	31,988
COR-GA+Tun	22,373	28,947
CORMF-CC	**14,237**	**20,742**

In view of the obtained results, the CORMF-CC method shows the best performance combining both approximation (MSE_{tra}) and generalization (MSE_{tst}). Analyzing the two-stage methods (WM+Tun and COR-GA+Tun), we may observe how the tuning process significantly improves the accuracy of the linguistic models generated by the WM and COR-GA learning methods. However, when the derivation process is made in only one stage with the cooperative coevolutionary approach, the linguistic model obtained overcomes the remainder thanks to the proper consideration of the dependency between the RB and the DB in the learning process.

4 Conclusions and Further Work

Performing the tasks of learning fuzzy linguistic rules and membership functions together allows our proposal to consider the tight relation between them, thus obtaining better linguistic models. To address this process, we have used a cooperative coevolutionary approach with a sophisticated rule learning component based on the cooperation among the fuzzy rules derived. The good performance of the proposed method has been shown when solving a real-world problem.

As further work, we propose to extend the components of the KB to be derived (number of labels, more flexible linguistic rules, etc.) and to coevolve other metaheuristics.

References

[1] J. Casillas, O. Cordón, F. Herrera. Different approaches to induce cooperation in fuzzy linguistic models under the COR methodology. In B. Bouchon-Meunier, J. Gutiérrez-Ríos, L. Magdalena, R.R. Yager (Eds.), *Techniques for Constructing Intelligent Systems.* Springer-Verlag, 2001. In press.

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

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

[4] F. Herrera, M. Lozano, J.L. Verdegay. Tuning fuzzy controllers by genetic algorithms. *Int. J. Approximate Reasoning*, 12:299–315, 1995.

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

[6] J. Liska, S.S. Melsheimer. Complete design of fuzzy logic systems using genetic algorithms. In *Proc. 3rd IEEE Int. Conf. on Fuzzy Syst.*, pp. 1377–1382, Orlando, 1994.

[7] J. Paredis. Coevolutionary computation. *Artificial Life*, 2:355–375, 1995.

[8] C.A. Peña-Reyes, M. Sipper. Fuzzy CoCo: a cooperative coevolutionary approach to fuzzy modeling. *IEEE Trans. Fuzzy Syst.* To appear. Draft available at `http://lslwww.epfl.ch/~penha`.

[9] M.A. Potter, K.A. De Jong. Cooperative coevolution: an architecture for evolving coadapted subcomponents. *Evolutionary Computation*, 8(1):1–29, 2000.

[10] P. Thrift. Fuzzy logic synthesis with genetic algorithms. *Proc. 4th Int. Conf. on Genetic Algorithms*, pp. 509–513, San Mateo, 1991.

[11] L.X. Wang, J.M. Mendel. Generating fuzzy rules by learning from examples. *IEEE Trans. Syst. Man Cybern.*, 22(6):1414–1427, 1992.