Incremental Fuzzy CoCo: Incremental Coevolution of Fuzzy Systems

Carlos Andrés Peña-Reyes
Logic Systems Laboratory, Swiss Federal Institute of Technology in Lausanne
CH-1015 Lausanne, Switzerland.
Email: Carlos.Pena@epfl.ch.
Tel.: +41-21-693-6714, fax:+41-21-693-3705.

Abstract

Fuzzy CoCo is a cooperative coevolutionary approach to fuzzy modeling, wherein two coevolving species are defined: database (membership functions) and rule base. Fuzzy CoCo requires the user to define a maximum number of rules for a given run. However, for each problem there exists a range of ideal rule-base sizes, that is hard to determine. The user is thus obliged to find this range by trial and error. I propose in this paper an extension to Fuzzy CoCo, intended to simplify the task of finding an adequate size of the rule base. This extension, called Incremental Fuzzy CoCo, is inspired by the iterative rule learning approach. In this method, the number of rules of the sought-after system increases each time that evolution satisfies certain criteria. In this way, the search for more complex systems starts on the basis of some "good" individuals.

KEYWORDS: Incremental evolution, coevolution, evolutionary fuzzy modeling.

INTRODUCTION

Fuzzy logic is a computational paradigm that provides a mathematical tool for representing and manipulating information in a way that resembles human communication and reasoning processes [6]. Fuzzy modeling, i.e., the task of identifying the parameters of a fuzzy inference system so that a desired behavior is attained, becomes difficult when the available knowledge is incomplete or when the problem space is very large, thus motivating the use of automatic approaches to fuzzy modeling—such as evolutionary algorithms. Fuzzy CoCo is a cooperative coevolutionary approach to fuzzy modeling, wherein two coevolving species are defined: database (membership functions) and rule base [4, 5].

In Fuzzy CoCo, individuals of the first species encode values defining completely the membership functions for the variables of the system. Individuals of the second species define a set of rules of the form:

if (v_1 **is** A_1) **and** **...** (v_n **is** A_n) **then** (*output* **is** C),

where the term A_v indicates which one of the linguistic labels of fuzzy variable v is used by the rule. The evolution of the two populations is controlled by two instances of a simple genetic algorithm. They apply fitness-proportionate selection to choose the mating pool, and apply an elitist strategy with an elitism rate Er to allow a given proportion of the best individuals to survive into the next generation. Standard crossover and mutation operators are applied with probabilities P_c and P_m, respectively.

An individual undergoing fitness evaluation establishes cooperations with one or more representatives, or *cooperators*, of the other species, i.e., it is combined with individuals from the other species to construct fuzzy systems. The fitness value assigned to the individual depends on the performance of the fuzzy systems it participated in. Cooperators are selected both fitness-proportionally and randomly from the last generation since they have already been assigned a fitness value.

Applying Fuzzy CoCo requires the definition of parameters of its two main components: (1) the fuzzy system and (2) the cooperative coevolutionary algorithm. I discuss below the definition of fuzzy parameters.

The parameters of a fuzzy system can be classified into four categories: logical, structural, connective, and operational. Logical parameters (i.e., fuzzy-inference strategy and operators) are usually predefined by the designer based on experience and on problem characteristics. Connective and operational parameters, which correspond respectively to rules and membership functions, constitute the search space of Fuzzy CoCo and are, therefore, encoded into the genomes of the coevolving species.

Structural parameters define the number of rules and membership functions, thus affecting both the size of the search space and the potential performance of the fuzzy systems encoded in the population. Experience has shown that for each problem there exists a range of ideal rule-base sizes, that is hard to determine. Indeed, systems with not enough rules are unable to attain satisfactory performances as they lack of rules to cover the problem space. On the other hand, evolutionary runs with excesively large individuals (i.e., encoding too many rules) have to deal with huge search spaces and may find very hard to explore them. Given that Fuzzy CoCo requires defining a maximum number of rules for a given run, the user is obliged to find this range by trial and error leading to a large number of exploratory evolutionary runs.

I propose thus, Incremental Fuzzy CoCo, an alternative method inspired by the iterative rule learning approach [1], in which a single instance of Fuzzy CoCo is used to search for fuzzy systems whose complexity increases as evolution advances. In contrast to iterative rule learning methods, individuals in Incremental Fuzzy CoCo represent entire fuzzy systems instead of one-rule systems.

THE PROPOSED ALGORITHM

Incremental Fuzzy CoCo starts as a simple instance of Fuzzy CoCo, used to search for small fuzzy systems (i.e., with a reduced number of rules, usually one). This instance runs until evolution satisfies a given criterion. At this point, part of the evolved population is used to seed the initial population of a new instance of Fuzzy CoCo. This new instance, that is set up to search for larger systems, runs until a new criterion is satisfied and a new instance with larger individuals is launched. The process is repeated until a termination criterion is satisfied. Due to the change of complexity, the genomes must be adapted before their use in a new instance of Fuzzy CoCo. Figure 1 presents the Island Fuzzy CoCo algorithm in pseudo-code format.

begin Incremental Fuzzy CoCo
    $R = R_{min}$
    Initialize Fuzzy CoCo populations for R rules $P_R(0)$
    while not done **do**
        while increment criterion not satisfied **do**
            Run Fuzzy CoCo: $P_R = FCC(P_R(0))$
            $P_R(0) = P_R$
        end while
        $Q_R =$ Select-seed$[P_R]$
        $R' = R + R_{inc}$
        $Q'_R =$ Adapt-genome$[Q_R]$
        $P'_R(0) =$ complete Q'_R with random individuals with R' rules
        $R = R'$
    end while
end Incremental Fuzzy CoCo

Figure 1: Pseudo-code of Incremental Fuzzy CoCo. A simple Fuzzy CoCo evolves a population of systems of size R. Each time an increment criterion is satisfied, the sought-after complexity is increased and part of the population is used to seed a new instance of Fuzzy CoCo.

The criteria used to decide to increase the complexity may be one or more of the following: number of generations elapsed, average performance of the entire population or of a selected elite, stagnation of evolution, or explicit user interaction. A part of the actual population is used to seed the new, complexity-increased, population. Due to the increased complexity in the new population, the genome of the selected seed must be adapted by duplicating some active rules. Special care must be taken to ensure that fitness is preserved after this operation. The remaining individuals in the initial population are generated randomly. In this way it is possible to keep the flexibility of the search while launching the new search with known "good individuals."

APPLYING INCREMENTAL FUZZY COCO

Following, I present the application of Incremental Fuzzy CoCo to the WBCD problem. The main goal of this test is to verify the potential search capabilities of the method. Below I present briefly the WBCD problem and then describe the application of the algorithm to solve it.

THE WISCONSIN BREAST CANCER DIAGNOSIS (WBCD) PROBLEM

The Wisconsin Breast Cancer Diagnosis (WBCD) problem involves classifying a presented case as to whether it is benign or malignant. It admits a relatively high number of variables and consequently a large search space. The WBCD database [3] consists of nine visually

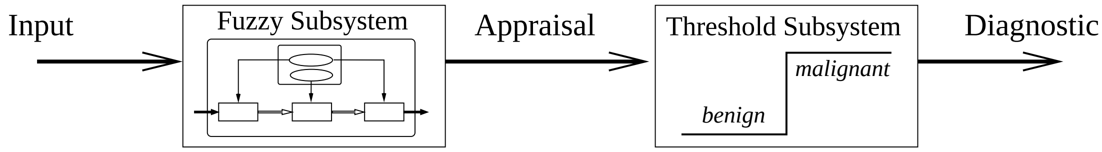

Figure 2: Proposed diagnosis system.

assessed characteristics obtained from fine needle aspirates[1] of breast masses, each of which is ultimately represented as an integer value between 1 and 10. The measured variables are as follows: (1) Clump Thickness (v_1); (2) Uniformity of Cell Size (v_2); (3) Uniformity of Cell Shape (v_3); (4) Marginal Adhesion (v_4); (5) Single Epithelial Cell Size (v_5); (6) Bare Nuclei (v_6); (7) Bland Chromatin (v_7); (8) Normal Nucleoli (v_8); and (9) Mitosis (v_9).

The diagnostics in the WBCD database were furnished by specialists in the field. The database itself consists of 683 cases, with each entry representing the classification for a certain ensemble of measured values:

case	v_1	v_2	v_3	...	v_9	*diagnostic*
1	5	1	1	...	1	*Benign*
2	5	4	4	...	1	*Benign*
:	:	:	:		:	:
683	4	8	8	...	1	*Malignant*

The solution scheme proposed for the WBCD problem is depicted in Figure 2. It consists of a fuzzy system and a threshold unit. The fuzzy system computes a continuous appraisal value of the malignancy of a case, based on the input values. The threshold unit then outputs a *benign* or *malignant* diagnostic according to the fuzzy system's output.

Previous knowledge about the WBCD problem represents valuable information used for my choice of fuzzy parameters. When defining the setup I took into consideration the following three results concerning the composition of potential high-performance systems: (1) small number of rules; (2) small number of variables; and (3) monotonicity of the input variables [5]. Some fuzzy models forgo interpretability in the interest of improved performance. Where medical diagnosis is concerned, interpretability—also called linguistic integrity—is the major advantage of fuzzy systems. This motivated to take into account semantic and syntactic criteria, defining constraints on the fuzzy parameters [5]. I delineate below the fuzzy-system setup:

- Logical parameters: singleton-type fuzzy systems; min-max fuzzy operators; orthogonal, trapezoidal input membership functions; weighted-average defuzzification.
- Structural parameters: two input membership functions (*Low* and *High*); two output singletons (*benign* and *malignant*). The relevant variables are selected by Fuzzy CoCo. Searching for an adequate range of rule-base sizes constitute the main goal of Incremental Fuzzy CoCo.

[1]Fine needle aspiration is an outpatient procedure that involves using a small-gauge needle to extract fluid directly from a breast mass [2].

- Connective parameters: the antecedents and the consequent of the rules are searched by Fuzzy CoCo, which also searches for the consequent of a default rule playing the role of an `else` condition. All rules have unitary weight.

- Operational parameters: the input membership function values are to be found by Fuzzy CoCo. For the output singletons we used the values 2 and 4, which are the values used in the WBCD database for designing *benign* and *malignant*, respectively.

EVOLUTIONARY SETUP

Setting up Incremental Fuzzy CoCo requires the definition of the following parameters:

- Range of the number of rules. The algorithm starts with one-rule systems and works its way up to ten-rule systems.

- Increment criterion. Complexity is increased after a given number of generations. The number of generations allowed for each instance depends on the number of rules: it starts at 200 generations for one-rule systems and goes up to 400 generations for ten-rule systems.

- Fuzzy CoCo setup. All the instances of Fuzzy CoCo use the same setup. Table 1 delineates the values used for the Fuzzy CoCo parameters (see [4, 5] for more details on setting up Fuzzy CoCo).

Table 1: Fuzzy CoCo set-up for all the instances in Incremental Fuzzy CoCo for the WBCD problem.

Parameter	Value
Population size $\|Ps\|$	100
Crossover probability P_c	1
Mutation probability P_m	0.15
Elitism rate E_r	0.1
"Fit" cooperators N_{cf}	1
Random cooperators N_{cr}	1

- Number of seeding individuals. The best five percent of the evolved population is used to seed the new initial population. For the rule species, the genomes are adapted to the new size by adding a new rule. Each individual of the selected seed is used to generate three new individuals: the first is obtained by duplicating one of the active rules, the other two by adding a random rule. The remaining individuals (i.e., 85% of the population) are randomly initialized. In this way, the first 5% are known to perform well, the following 10% explore promising regions of the new enlarged search space, while the remaining 85% explore new regions.

RESULTS

Thirty-two runs were performed, all but one of which found systems whose classification performance exceeds 98.0%. In particular, considering the best individual per run (i.e., the evolved system with the highest classification success rate), 20 runs led to a fuzzy system whose performance exceeds 98.5%, and of these, 2 runs found systems whose performance exceeds 98.7%.; these results are summarized in Figure 3. The best system found obtains an overall classification rate of 98.83%. The average performance over the 32 runs is 98.50%.

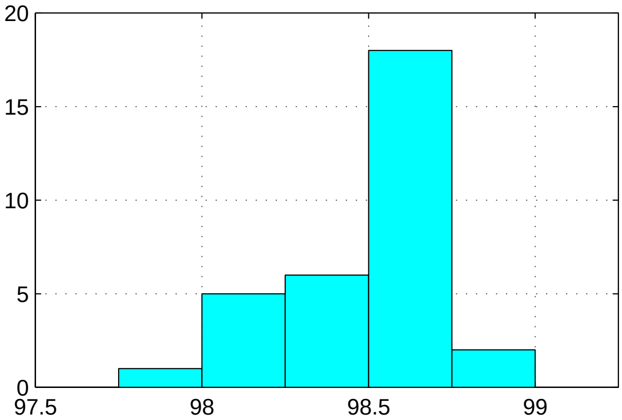

Figure 3: Summary of results of 32 evolutionary runs. The histogram depicts the number of systems exhibiting a given classification performance level at the end of the evolutionary run.

Figure 4 shows the evolution of the classification performance during Incremental Fuzzy CoCo runs.

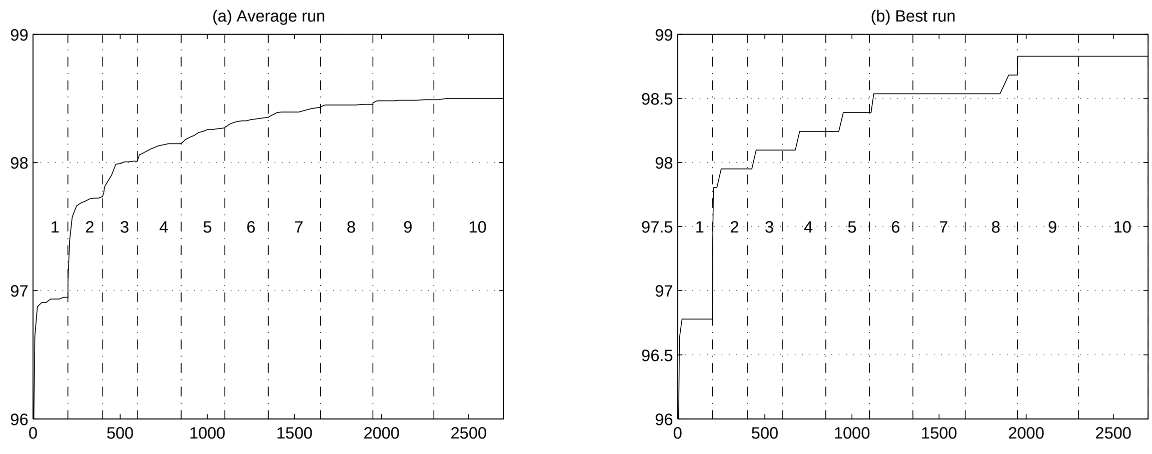

Figure 4: Evolution of classification performance in Incremental Fuzzy CoCo. The figures show the classification performance for: (a) the average over 32 runs, and (b) the best run. The abscissa represents the number of generations elapsed. The numbers 1 to 10 represent the number of rules of the systems evolved in the corresponding interval.

Taking the results obtained by Incremental Fuzzy CoCo at the end of the search for seven-rule systems (corresponding to 1650 generations), we can compare them with the results of Fuzzy CoCo when searching for seven-rule systems [4]. Table 2 presents this comparison, while Figure 5 extends the comparison to other rule base sizes. The best systems found by Incremental Fuzzy CoCo are worse than those found by Fuzzy CoCo. However, the average performance of the former is better, and its results are obtained with less generations than those of the latter.

Table 2: Comparison of seven-rule systems evolved by Incremental Fuzzy CoCo with those obtained using Fuzzy CoCo.

	Generations	Average	Best
Incremental Fuzzy CoCo	1650	98.43%	98.54%
Simple Fuzzy CoCo	1700	98.25%	98.98%

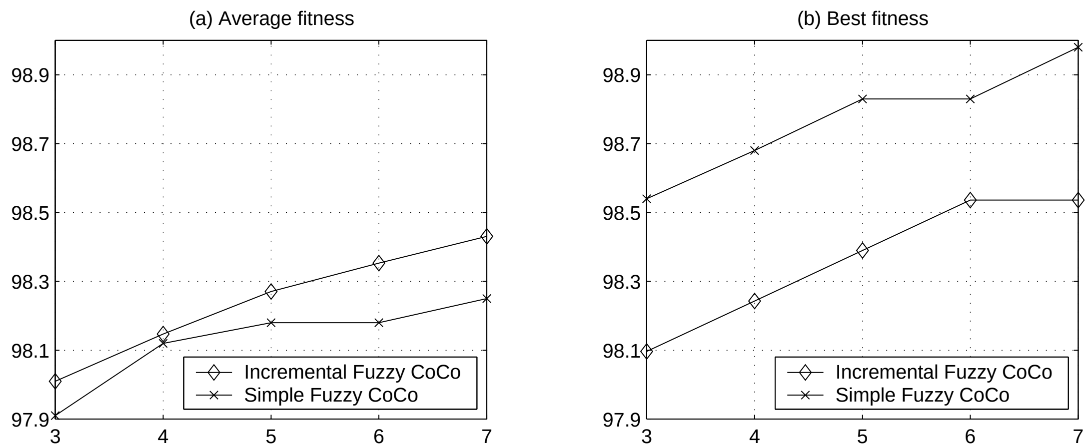

Figure 5: Comparison of systems evolved by Incremental Fuzzy CoCo with those obtained using Fuzzy CoCo. The figure shows the classification performance obtained at the end of the search for each rule-base size.

CONCLUDING REMARKS

I presented Incremental Fuzzy CoCo, an incremental-complexity extension to Fuzzy CoCo, intended to deal with the search for an adequate range of fuzzy rule-base sizes. I tested its capabilities by solving the WBCD problem. The results suggest some interesting features that deserve further exploration as explained below:

- Incremental Fuzzy CoCo seems to exhibit better repeatability than Fuzzy CoCo, as suggested by the narrow distribution of the results (see Figure 3). This is true not only for the global results but also for different rule-base sizes as shown in Figure 5. However, this approach fails to find top systems as good as those found by Fuzzy CoCo. One of the possible reasons is that all the instances of Fuzzy CoCo were identically set up.

- The use of a fixed number of generations to define the increment of the complexity should be a weak criterion as the population can either converge toward mediocre performances, if this number is too high, or be stopped before a good exploration of the search space is performed, if this number is too low. This criterion affects directly the diversity and the quality of the initial population of the following instance.

- Intuitively, the seeding strategy (5% original seed, 10% modified seed) is adequate, but the rate of seeding individuals needs to be tuned, as it should be excessive if the elite is not diverse enough.

- This method requires a deeper study to better understand the effects of the parameters (i.e., Fuzzy CoCo, increment criterion, and seeding strategy) on its performance. The goal of the study should be to find a setup that improves the quality of the best systems while preserving as much as possible the repeatability of the results.

- Even if this goal is not attained, Incremental Fuzzy CoCo can be useful, as it is, for:
  - searching automatically for an adequate range of sizes of the rule base, and
  - estimating attainable performance values for a given problem, and for different number of rules.

References

[1] F. Herrera, M. Lozano, and J. L. Verdegay. Generating fuzzy rules from examples using genetic algorithms. In B. Bouchon-Meunier, R. R. Yager, and L. A. Zadeh, editors, *Fuzzy Logic and Soft Computing*, pages 11–20. World Scientific, 1995.

[2] O. L. Mangasarian, W. N. Street, and W. H. Wolberg. Breast cancer diagnosis and prognosis via linear programming. Mathematical Programming Technical Report 94-10, University of Wisconsin, 1994.

[3] C. J. Merz and P. M. Murphy. UCI repository of machine learning databases, 1996.

[4] C. A. Peña-Reyes. *Coevolutionary Fuzzy Modeling.* PhD thesis, École Polytechnique Fédérale de Lausanne - EPFL, 2002.

[5] C. A. Peña-Reyes and M. Sipper. Fuzzy CoCo: A cooperative-coevolutionary approach to fuzzy modeling. *IEEE Transactions on Fuzzy Systems*, 9(5):727–737, October 2001.

[6] R. R. Yager and L. A. Zadeh. *Fuzzy Sets, Neural Networks, and Soft Computing.* Van Nostrand Reinhold, New York, 1994.