The Flowering of Fuzzy CoCo: Evolving Fuzzy Iris Classifiers

Carlos Andrés Peña-Reyes and Moshe Sipper[1]

Abstract

Combining the search power of coevolutionary computation with the expressive power of fuzzy systems, we present *Fuzzy CoCo:* Fuzzy Cooperative Coevolution. We demonstrate the efficacy of our algorithm by applying it to a hard problem—flower classification—obtaining the best classification performance to date, coupled with high human-interpretability.

1. 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 [1]. *Fuzzy modeling* is the task of identifying the parameters of a fuzzy inference system so that a desired behavior is attained [1]. This task 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. In this paper we apply coevolutionary fuzzy modeling to a well-known benchmark classification problem: Fisher's iris data.

2. Fuzzy CoCo: A Cooperative Coevolutionary Approach to Fuzzy Modeling

Fuzzy CoCo is a Cooperative Coevolutionary approach to fuzzy modeling, wherein two coevolving species are defined: database (membership functions) and rule base. This approach is based primarily on the framework defined by Potter [2].

In Fuzzy CoCo, the fuzzy modeling problem is solved by two coevolving cooperative species. Individuals of the first species encode values which define completely all the membership functions for all 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 two evolutionary algorithms used to control the evolution of the two populations are instances of a simple genetic algorithm. The genetic algorithms 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 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. Representatives, or *cooperators*, are selected both fitness-proportionally and randomly from the last generation since they have already been assigned a fitness value. In Fuzzy CoCo, N_{cf} cooperators are probabilistically selected according to their fitness, usually the fittest individuals, thus favoring the exploitation of known good solutions. The other N_{cr} cooperators are selected randomly from the population to represent the diversity of the species, maintaining in this way exploration of the search space. For a more detailed exposition of Fuzzy CoCo see [3].

3. Applying Fuzzy CoCo to Fisher's iris data

Fisher's iris data, wherein iris flowers are classified according to external features, has been widely

[1]Both authors are with the Logic Systems Laboratory, Swiss Federal Institute of Technology in Lausanne, CH-1015 Lausanne, Switzerland. Emails: Carlos.Pena@epfl.ch, Moshe.Sipper@epfl.ch. Tel.: +41-21-693-2658, fax: +41-21-693-3705. Corresponding author is C. A. Peña-Reyes.

used to test classification and modeling algorithms, recently including fuzzy models [4–8]. We propose herein two types of fuzzy logic-based systems to solve the iris data classification problem: (1) fuzzy controller-type (as used by Shi *et al.* [4] and Russo [5]), and (2) fuzzy classifier-type (as used by Hong and Chen [6], Wu and Chen [7], and Hung and Lin [8]). Both types consist of a fuzzy inference subsystem whose output is fed to a selection unit.

In the fuzzy controller the fuzzy subsystem computes a single continuous value estimating the class to which the input vector belongs. Note that each class is assigned a numeric value: based on the iris data distribution, we assigned values 1, 2, and 3 to the classes *setosa*, *versicolor*, and *virginica*, respectively (such an assignment makes sense only under the assumption that *versicolor* is an intermediate species in between *setosa* and *virginica*). The selection unit approximates this value to the nearest class value using a stair function.

In the fuzzy classifier the fuzzy inference subsystem computes a continuous membership value for each of the three output classes. The selection unit chooses the most active class, provided that its membership value exceeds a given threshold (which we set to 0.5).

The two fuzzy subsystems thus differ in the number of output variables: a single output (with values {1,2,3}) for the controller-type and three outputs (with values {0,1}) for the classifier-type. In general, controller-type systems take advantage of data distribution while classifier-type systems offer higher interpretability because the output classes are independent; these latter systems are harder to design.

Fuzzy CoCo searches for four parameters: input membership-function values, relevant input variables, and antecedents and consequents of rules. The genomes of the two species are constructed as follows:

- Species 1: Membership functions. There are four input variables (SL, SW, PL, and PW), each with three parameters P_1, P_2, and P_3, defining the membership-function edges (Figure 1).

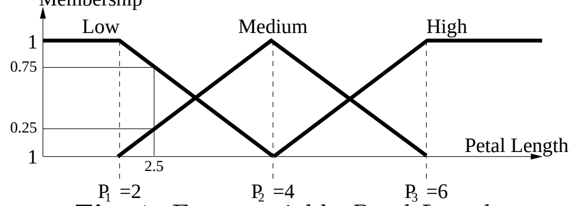

Fig. 1: Fuzzy variable *Petal Length*.

- Species 2: Rules (Controller-type systems). The i-th rule has the form:
  if (SL **is** A^i_{SL}) **and** ... **and** (PW **is** A^i_{PW}) **then** (*output* **is** C^i),
  A^i_j can take on the values: 1 (*Low*), 2 (*Medium*), 3 (*High*), or 0 (*Other*). C^i can take on the values: 1 (*setosa*), 2 (*versicolor*), or 3 (*virginica*).
- Species 2: Rules (Classifier-type systems). The i-th rule has the form:
  if (SL **is** A^i_{SL}) **and** ... **and** (PW **is** A^i_{PW}) **then** {(*setosa* **is** C^i_{set}), (*versicolor* **is** C^i_{ver}), (*virginica* **is** C^i_{vir})},
  A^i_j can take on the values: 1 (*Low*), 2 (*Medium*), 3 (*High*), or 0 (*Other*). C^i_j can take on the values: 0 (*No*), or 1 (*Yes*).

Table 1 delineates the parameter encoding for both species' genomes, which together describe an entire fuzzy system. Table 2 delineates values and ranges of values of the evolutionary parameters.

Table 1: Genome encoding.

Species 1: Membership functions

Parameter	Values	Bits	Qty	Total bits
P_i	$[V_{mn} - V_{mx}]$	5	3×4	60

Species 2: Rules (Controller-type)

Parameter	Values	Bits	Qty	Total bits
A	{0,1,2,3}	2	$4 \times N_r$	$8 \times N_r$
C	{1,2,3}	2	$N_r + 1$	$2{\times}(N_r{+}1)$
	Total Genome Length			$10 \times N_r + 2$

Species 2: Rules (Classifier-type)

Parameter	Values	Bits	Qty	Total bits
A	{0,1,2,3}	2	$4 \times N_r$	$8 \times N_r$
C	{0,1}	1	$3{\times}(N_r{+}1)$	$3{\times}(N_r{+}1)$
	Total Genome Length			$11 \times N_r + 3$

Table 2: Fuzzy CoCo set-up.

Parameter	Values
Population size N_p	{60,70}
Maximum generations G_{max}	$500 + 100 \times N_r$
Crossover probability P_c	1
Mutation probability P_m	{0.02,0.05,0.1}
Elitism rate E_r	{0.1,0.2}
"Fit" cooperators N_{cf}	1
Random cooperators N_{cr}	{1,2}

Our fitness function combines three criteria: (1) F_c: classification performance, computed as the percentage of cases correctly classified; (2) F_{mse}: a value dependent on the mean square error (mse), measured between the continuous values of the outputs and the correct classification given by the iris data set ($F_{mse} = 1 - mse$); and (3) F_v: a rule-length dependent fitness with value 0 when the average number of variables per active rule is maximal and equal to 1 in the hypothetical case of zero-variable rules. The fitness function combines these three measures:

$$F = \begin{cases} F_c \times F_{mse}^{\beta} & \text{if } F_c < 1 \\ (F_c - \alpha F_v) \times F_{mse}^{\beta} & \text{if } F_c = 1, \end{cases}$$

where $\alpha = 1/150$ and $\beta = 0.3$.

4. Results

In this section we present the fuzzy systems evolved using Fuzzy CoCo for the two setups described above. We compare our systems with those presented in recently published articles, and detail two high-performance systems obtained.

4.1. Controller-type systems

We performed a total of 145 evolutionary runs, searching for controller-type systems with 2, 3, and 4 rules, all runs of which found systems whose classification performance exceeds 97.33% (i.e., the worst system misclassifies only 4 cases). The average classification performance of these runs was 98.98%, corresponding to 1.5 misclassifications. 121 runs led to a fuzzy system misclassifying 2 or less cases, and of these, 4 runs found perfect classifiers.

Table 3 compares our best controller-type systems with the top systems obtained by two other evolutionary fuzzy modeling approaches. Shi *et al.* [4] used a simple genetic algorithm with adaptive crossover and adaptive mutation operators. Russo's FuGeNeSys method [5] combines evolutionary algorithms and neural networks to produce fuzzy systems. The main drawback of these two methods is the low interpretability of the generated systems. As they do not define constraints on the input membership-function shapes, almost none of the semantic criteria favoring interpretability are respected [9]. As evident in Table 3, the evolved fuzzy systems described in this section surpass those obtained by the two other approaches in terms of performance, while maintaining high interpretability. Our approach not only produces systems exhibiting high performance, but also ones with less rules and less antecedents per rule (which systems are thus more interpretable).

Table 3: Comparison of results. Parentheses show average number of variables per rule.

Rules per system	Shi *et al.* [4]	FuGeNeSys [5]	Fuzzy CoCo	
	best	best	average	best
2	--	--	98.71% (1.9)	99.33% (2)
3	--	--	99.10% (1.3)	**100% (1.7)**
4	98.00% (2.6)	--	99.12% (1.3)	100% (2.5)
5	--	100% (3.3)	--	--

Database

	SL	SW	PL	PW
P_1	5.68	3.16	1.19	1.55
P_2	6.45	3.16	1.77	1.65
P_3	7.10	3.45	6.03	1.74

Rule base

Rule 1	**if** (PL **is** *High*) **then** (*output* **is** *virginica*)
Rule 2	**if** (SW **is** *Low*) **and** (PW **is** *Low*) **then** (*output* **is** *virginica*)
Rule 3	**if** (SL **is** *Medium*) **and** (PW **is** *Medium*) **then** (*output* **is** *setosa*)
Default	**else** (*output* **is** *setosa*)

Fig. 2: The best evolved, controller-type system with three rules. It exhibits a classification rate of 100%, and an average of 1.7 variables per rule.

Fuzzy CoCo found controller-type systems with 3 and 4 rules exhibiting perfect performance (no misclassifications). Among these, we consider as best the system with fewest rules and variables. Figure 2 presents one such three-rule system, with an average of 1.7 variables per rule.

4.2. Classifier-type systems

We performed a total of 144 evolutionary runs, searching for controller-type systems with 2, 3, and 4 rules, all runs of which found systems whose classification performance exceeds 95.33% (i.e., the worst system misclassifies 7 cases). The average classification performance of these runs was 97.40%, corresponding to 3.9 misclassifications. 104 runs led to a fuzzy system misclassifying 5 or less cases, and of these, 13 runs found systems with a single misclassification.

Table 4 compares our best classifier-type systems with the top systems obtained by three other fuzzy

Table 4: Comparison of results. Parentheses show average number of variables per rule.

Rules per system	Hong and Chen [6]	Wu and Chen [7]	Hung and Lin [8]	Fuzzy CoCo	
	best	average	average	average	best
2	--	--	--	96.47% (2.1)	98.00% (1.5)
3	--	96.21% (4)	--	97.51% (2.4)	**99.33% (2.3)**
4	--	--	97.40% (4)	98.21% (2.3)	99.33% (2)
8	97.33% (2)	--	--	--	--

modeling approaches. Hong and Chen [6] and Wu and Chen [7] proposed sequential learning methods to progressively construct fuzzy systems. These two approaches are able to find systems with either a few [7] or simple rules [6]. They do not, however, constrain the input membership functions, thus rendering the obtained systems less interpretable. Hung and Lin [8] proposed a neuro-fuzzy hybrid approach to learn classifier-type systems. As their learning strategy hinges mainly on the adaptation of the connection weights, their systems exhibit low interpretability. The evolved fuzzy systems described herein surpass those obtained by these three approaches in terms of both performance and interpretability. As evident in Table 4, our approach not only produces systems exhibiting higher performance, but also ones with less rules and less antecedents per rule (which are thus more interpretable).

Fuzzy CoCo found classifier-type systems with 3 and 4 rules exhibiting the highest classification performance to date (i.e., 99.33%, corresponding to 1 misclassification). We consider as most interesting the system with the smallest number of conditions (i.e., the total number of variables in the rules). Figure 3 presents one such three-rule system with an average of 2.3 variables per rule, corresponding to a total of 7 conditions.

5. Concluding remarks

We presented Fuzzy CoCo, a cooperative coevolutionary approach to fuzzy modeling, and applied it to Fisher's iris data problem. Comparing our results with other fuzzy-modeling approaches, we conclude that our coevolved systems attain higher classification performance and better interpretability. These promising results have incited us to engage in further investigation, specifically: (1) application of Fuzzy CoCo to more complex problems, and (2) improving and expanding upon the methodology presented herein. Our underlying goal is to provide an approach for automatically producing high-performance, interpretable fuzzy systems for real-world problems.

Database

	SL	SW	PL	PW
P_1	4.65	2.68	4.68	0.39
P_2	4.65	3.74	5.26	1.16
P_3	5.81	4.61	6.03	2.03

Rule base

Rule 1	**if** (PW **is** *Low*) **then** {(*setosa* **is** *Yes*), (*versicolor* **is** *No*), (*virginica* **is** *No*) }
Rule 2	**if** (PL **is** *Low*) **and** (PW **is** *Medium*) **then**{(*setosa* **is** *No*), (*versicolor* **is** *Yes*), (*virginica* **is** *No*)}
Rule 3	**if** (SL **is** *High*) **and** (SW **is** *Medium*) **and** (PL **is** *Low*) **and** (PW **is** *High*) **then**{(*setosa* **is** *No*), (*versicolor* **is** *Yes*), (*virginica* **is** *No*)}
Default	**else**{(*setosa* **is** *No*), (*versicolor* **is** *No*), (*virginica* **is** *Yes*)}

Fig. 3: The best evolved, classifier-type system with three rules. It exhibits a classification rate of 99.33%, and an average of 2.3 variables per rule.

References

[1] R. R. Yager and D. P. Filev, *Essentials of Fuzzy Modeling and Control*. New York: John Wiley & Sons., 1994.

[2] M. A. Potter and K. A. DeJong, "Cooperative coevolution: An architecture for evolving coadapted subcomponents," *Evolutionary Computation*, vol. 8, pp. 1–29, spring 2000.

[3] C. A. Peña-Reyes and M. Sipper, "Applying Fuzzy CoCo to breast cancer diagnosis," in *Proceedings of the 2000 Congress on Evolutionary Computation (CEC00)*, vol. 2, pp. 1168–1175, IEEE Press, Piscataway, NJ, USA, 2000.

[4] Y. Shi, R. Eberhart, and Y. Chen, "Implementation of evolutionary fuzzy systems," *IEEE Transactions on Fuzzy Systems*, vol. 7, pp. 109–119, April 1999.

[5] M. Russo, "FuGeNeSys - A fuzzy genetic neural system for fuzzy modeling," *IEEE Transactions on Fuzzy Systems*, vol. 6, pp. 373–388, August 1998.

[6] T. P. Hong and J. B. Chen, "Processing individual fuzzy attributes for fuzzy rule induction," *Fuzzy Sets and Systems*, vol. 112, pp. 127–140, May 2000.

[7] T.-P. Wu and S.-M. Chen, "A new method for constructing membership functions and fuzzy rules from training examples," *IEEE Transactions on Systems, Man and Cybernetics, Part B: Cybernetics*, vol. 29, pp. 25–40, February 1999.

[8] C. A. Hung and S. F. Lin, "An incremental learning neural network for pattern classification," *International Journal of Pattern Recognition and Artificial Intelligence*, vol. 13, pp. 913–928, September 1999.

[9] C. A. Peña-Reyes and M. Sipper, "A fuzzy-genetic approach to breast cancer diagnosis," *Artificial Intelligence in Medicine*, vol. 17, pp. 131–155, October 1999.