Hybridization of Components to Improve the Accuracy in Linguistic Fuzzy Modeling

R. Alcalá
Dept. Computer Science,
University of Jaén
E-23071 – Jaén, Spain
alcala@ujaen.es

J. Casillas, O. Cordón, F. Herrera
Dept. Computer Science and Artificial Intelligence,
University of Granada
E-18071 – Granada, Spain
{casillas,ocordon,herrera}@decsai.ugr.es

Abstract

This contribution faces an hybridization study of soft computing techniques within the genetic fuzzy system field. It is performed by combining several accuracy improvements to the linguistic fuzzy modeling with basic genetic algorithms and cooperative coevolutionary algorithms. Thus, five learning methods that generate linguistic models with good accuracy degrees preserving the interpretability are proposed.

Keywords: fuzzy modeling, interpretability and accuracy, soft computing, cooperative coevolution.

1 Introduction

Fuzzy rule-based systems (FRBSs) constitute an extension of classical rule-based systems, because they deal with IF-THEN rules where antecedents and/or consequents are composed of fuzzy logic statements, instead of classical logic rules. The most usual application of FRBSs is *system modeling*, which in this field may be considered as an approach used to model a system making use of a descriptive language based on fuzzy logic with fuzzy predicates. Fuzzy modeling (FM) (i.e., system modeling with FRBSs) usually comes with two contradictory requirements to the obtained model:

- *interpretability*, capability to express the behavior of the real system in an understandable way, and
- *accuracy*, capability to faithfully represent the real system.

While linguistic FM (LFM) — mainly developed by linguistic (Mamdani-type) FRBSs — is focused on the interpretability, precise FM (PFM) — mainly developed by Takagi-Sugeno-Kang FRBSs — is focused on the accuracy. Since both criteria are of vital importance in system modeling, the balance between them has started to receive attention in the fuzzy community in the last few years [4].

Roughly speaking, the balance is usually attained from two different perspectives: either the LFM is extended to obtain more accurate models or the PFM is improved to obtain more interpretable model. The former approach is usually developed by improving the fuzzy rule set derivation [10], automatically defining the membership functions [12, 13, 14], or extending the model structure [5, 15, 16]. The latter approach is usually developed by reducing the fuzzy rule set [22], reducing the number of fuzzy sets (with the subsequent merging of rules) [19, 20], or exploiting the local description of the rules [1].

This paper is focused on the LFM side approach to find the balance between interpretability and accuracy. Thus, three different mechanisms to improve the accuracy of LFM are jointly considered:

- *Improving the fuzzy rule set learning* —

It is performed by inducing cooperation among the consequents composing the fuzzy rules of the model.

- *Learning the linguistic term meanings* — It is performed by considering two different ways of learning the shapes of the membership functions with linear and non linear effects.
- *Extending the fuzzy rule structure* — It is performed by using a more flexible rule structure that includes linguistic hedges.

To perform these tasks, different combinatorial or optimization tools such as evolutionary algorithms (EAs) or neural networks are usually employed. In fact, these two areas together with fuzzy logic constitute the most important partnerships of the *soft computing* field. Among the different possible hybridizations of these partnerships (neuro-genetic systems, fuzzy neural networks, fuzzy genetic algorithms, neuro-fuzzy systems...), this contribution focus on analyzing the integration of EAs with the aforementioned LFM accuracy improvement mechanisms to develop *genetic fuzzy systems* [7].

When this task is faced, we should keep in mind that, as David Goldberg stated, the integration of single methods into hybrid intelligent systems goes beyond simple combinations. For him, the future of computational intelligence "*lies in the careful integration of the best constituent technologies*" and subtle integration of the abstraction power of fuzzy systems and the innovating power of genetic systems requires a design sophistication that goes further than putting everything together [9].

This view encourages us to make a deeper study of the hybridization of components to improve the accuracy by properly exploiting the existing interdependencies. To do that, sequential and simultaneous learning process together with genetic algorithms (GAs) and cooperative coevolutionary algorithms are considered and analyzed.

The paper is organized as follows: Section 2 explains the three LFM accuracy improvements considered, Section 3 shows different proposals to hybridize the components, Section 4 performs an experimental study over four different modeling applications, and finally, Section 5 points out some conclusions and future work.

2 Accuracy Improvements to Linguistic Fuzzy Modeling

This section describes the different LFM accuracy improvements considered in the paper.

2.1 Improvement C: Rule Base Learning with COR

This improvement arises as an effort to exploit the accuracy ability of linguistic FRBSs by exclusively focusing on the fuzzy rule set design [3, 10]. In this case, the membership functions and the model structure keep invariable, thus resulting in the highest interpretability.

The method COR (cooperative rules) proposed in [3] follows the primary objective of inducing a better cooperation among the linguistic rules. To do that, the rule base design is made using global criteria that consider the action of the different rules jointly. It is attained by means of a strong, smart reduction of the search space. The main advantages of the COR methodology are its capability to include heuristic information, its flexibility to be used with different metaheuristics, and its easy integration within other derivation processes.

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. The operation mode of COR is as follows:

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 $E_s^l \neq \emptyset$ (with A_i^s being a label of the i-th input variable, E_s^l being the set of positive examples of the subspace S_s, and N_S the number of subspaces with positive examples).

In this contribution, we will define the set of positive examples for the subspace S_s as follows:

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

with A_{ij} being a label of the i-th input variable and μ_T the membership function of the label T.

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 fuzzy rule.

   In this contribution, we will define the set of candidate consequents for the subspace S_s as follows:

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

   with B_k being a label of the output variable.

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

For example, from the subspace $S_s = (high, low)$ and the candidate consequent set in such a subspace $\mathbf{B}^s = \{small, medium, large\}$, we will obtain the fuzzy rule:

$R_s =$ **IF** X_1 is *high* and X_2 is *low* **THEN** Y is B_s,

with $B_s \in \mathbf{B}^s$ being the label selected by the combinatorial search to represent to the subspace S_s.

2.2 Improvement M: Learning of Membership Function Parameters and Non-Linear Scaling Factors

Basic LFM methods are exclusively focused on determining the set of fuzzy rules composing the RB of the model. In these cases, the membership functions are usually obtained from expert information (if available) or by a normalization process and it remains fixed during the rule base derivation process.

However, the automatic design of the membership functions has shown to be a very suitable mechanism to increase the approximation capability of the linguistic models. Generally speaking, the procedure involves either defining the most appropriate shapes for the membership functions that give meaning to the fuzzy sets associated to the considered linguistic terms or determining the optimum number of linguistic terms used in the variable fuzzy partitions, i.e., the granularity.

In this contribution, we will focus on learning the membership functions by defining their parameters and using non-linear scaling factors to vary their shapes (these shapes will have a high influence in the FRBS performance):

- *Learning/tuning the membership function parameters* — The most common way to derive the membership functions is to change their definition parameters [12, 13]. For example, if the following triangular-shape membership function is considered:

$$\mu(x) = \begin{cases} \frac{x-a}{b-a}, & \text{if } a \leq x < b \\ \frac{c-x}{c-b}, & \text{if } b \leq x \leq c \\ 0, & \text{otherwise} \end{cases},$$

  changing the basic parameters — a, b, and c — will vary the shape of the fuzzy set associated to the membership function, thus influencing the FRBS performance. The same yields for other shapes of membership functions (trapezoidal, gaussian, sigmoid, etc.).

- *Using non-linear scaling factors* — Another way to define the membership function shapes of the DB is to use more flexible alternative expressions for the membership functions to vary the compatibility degrees to the fuzzy sets [5, 14]. For example, a new membership function can be obtained raising the membership value to the power of α, i.e.,

$$\mu'(x) = \mu(x)^{\alpha}, \;\; 0 < \alpha.$$

By changing the α value we may define different membership function shapes.

2.3 Improvement L: Fuzzy Rules with Linguistic Hedges

A third possibility to increase the accuracy in LFM is to relax the rule structure by including certain operators that slightly change the meaning of the linguistic labels involved in the system when necessary [5, 11]. As Zadeh highlighted, a way to do so without losing an excessive description is to use linguistic hedges.

A linguistic hedge is an operator that alters the membership functions for the fuzzy sets associated to the linguistic labels, giving a more or less precise definition as a result depending on the case. For example, the linguistic hedges '*very*' and '*more-or-less*' performs as follows: $\mu^{very}(x) = \mu(x)^2$ and $\mu^{more-or-less}(x) = \sqrt{\mu(x)}$. An example of a rule with this structure is the following:

IF X_1 is *very* high and X_2 is low
THEN Y is *more-or-less* large .

Actually, the consideration of linguistic modifiers does not define a new meaning to the so-called *primary terms* — *high*, *low*, and *large* in our example — but they are used as generators whose meaning is defined in the context. Certainly, the fact of using fuzzy rules with linguistic modifiers will have a significative influence in the behavior of the linguistic FRBS because the matching degree of the rule antecedent as well as the output fuzzy set obtained when applied the implication operator in the inference process are changed.

3 Hybridizing Accuracy Improvements to the Linguistic Fuzzy Modeling

This section proposes different ways of hybridizing the three mentioned LFM accuracy improvements: *improvement C* (linguistic rule set learning with COR), *improvement M* (learning of the membership function parameters and non-linear scaling factors), and *improvement L* (learning of the linguistic hedges used for each linguistic variable in each linguistic rule). To develop these hybridizations, two main mechanisms are considered in this contribution.

On the one hand, we may distinguish between sequential or simultaneous learning. When several components of the FRBS are designed, we may opt to make a *sequential learning* by dividing it in two or more stages, each of them performing a partial or complete derivation of the linguistic models. Other possibility is to consider a simultaneous learning that directly obtain the whole model. With the simultaneous learning, the strong dependency of the components is properly addressed. However, the derivation process becomes significantly more complex because the search space grows.

On the other hand, we can consider the use of cooperative coevolution. Indeed, from a different point of view, the combinations of the components can be made by a basic GA or using a more sophisticated evolutionary approach such as the coevolutionary algorithms [17]. 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.

Within coevolutionary algorithms, we can mainly distinguish between two interaction schemes, depending on if the species compete with the remainder (competitive approach) or cooperate to build the problem solution (cooperative or symbiotic approach). The latter interaction is usually recommendable when the following issues arise [18]: the search space is huge, the problem may be decomposable in subcomponents, different coding schemes are used, and there is strong interdependencies among the subcomponents. Therefore, the cooperative coevolution seems to be very adequate to hybridize the different LFM accuracy improvements.

3.1 Hybridizations

Different combinations are regarded by differentiating between sequential or simultaneous learning and between basic GAs or cooperative coevolution. Among all the possible combinations we will exclusively consider those seeming to be more coherent. Thus, a good approach to perform the sequential learning would be to firstly achieve the fuzzy rule set learning (with macroscopic effects) and then adjust this rule set and the initial membership functions (with microscopic effects). As regards the cooperative coevolution, we will contemplate two criteria to decide how dividing the subcomponents into two groups: on the one hand, we will distinguish between macroscopic (C) and microscopic (ML) effects, and on the other hand, between learning of the rule set (CL) and of the membership functions (M).

Therefore, the following five learning methods are proposed as a representation of the possible hybridizations:

1. ***Method C+ML*** — Sequential learning with GAs. It is comprised of a first stage for learning the fuzzy rule set (C) and a subsequent tuning of the fuzzy rule set and membership functions (ML) with a GA.

2. ***Method C+M-L*** — Sequential learning with cooperative coevolution. It is comprised of a first stage for learning the fuzzy rule set (C) and a subsequent tuning with cooperative coevolution, a species for the membership functions (M) and another one for the linguistic hedges of the fuzzy rules (L).

3. ***Method CML*** — Simultaneous learning with GAs. It involves a process to learn both fuzzy rules and membership functions by including in a unique chromosome the three improvement mechanisms.

4. ***Method C-ML*** — Simultaneous learning with cooperative coevolution. The fuzzy rules are learnt in a species (C) while the membership functions and linguistic hedges (ML) are learnt in another one.

5. ***Method CL-M*** — Simultaneous learning with cooperative coevolution. The fuzzy rules and their associated linguistic hedges are learnt in a species (CL) while the membership functions are learnt in another one (M).

3.2 Description of the Evolutionary Algorithms

This section shows some details related with the developed evolutionary methods. All of them have some common aspects that are described in the following.

A generational scheme is followed. 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 fitness function will be to minimize the well-known *mean square error* (MSE):

$$\mathrm{MSE} = \frac{1}{2 \cdot N} \sum_{l=1}^{N} (F(x^l) - y^l)^2,$$

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

The following subsections describe the specific coding scheme and evolutionary operators used in the different learning methods. They are grouped according to the accuracy improvement where they are used. The interaction scheme followed in the cooperative coevolutionary algorithms is also explained.

3.2.1 Subcomponent C

An integer-valued vector of size equal to the 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 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 $k^{s\prime} \neq k^s$.

3.2.2 Subcomponent M

This component is encoded with two strings, one for the parameters of the membership functions, an other for their scaling factors.

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

- The latter string consists of a real-coded chromosome that encodes the value of the non-linear scaling factor associated to each membership function. Each gene can take any value in the interval $[-1, 1]$ with the following mapping between alleles and actual value:

  $$\begin{array}{ccccccc} c_{ij} & \in & [-1,0] & \longleftrightarrow & \alpha & \in & [0,1], \\ c_{ij} & \in & [0,1] & \longleftrightarrow & \alpha & \in & [1,5], \end{array}$$

  with c_{ij} being the gene associated to the membership function for the j-th linguistic term of the i-th variable.

The max-min-arithmetical crossover [6] is considered. With respect to the mutation operator, it simply involves changing the value of the selected gene by other value obtained at random within the corresponding variation interval.

3.2.3 Subcomponent L

The coding scheme of this subcomponent generates integer-coded strings of length $m \cdot (n+1)$ (with m being the number of rules and n being the number of input variables). Each gene can take any value in the set $\{0, 1, 2\}$ with the following correspondence to the linguistic hedge used:

$$\begin{array}{lcl} c_{ij} = 0 & \longleftrightarrow & \text{“very”}, \\ c_{ij} = 1 & \longleftrightarrow & \text{no linguistic hedge}, \\ c_{ij} = 2 & \longleftrightarrow & \text{“more-or-less”}. \end{array}$$

with c_{ij} being the gene associated to the linguistic term used in the j-th variable of the i-th rule.

The standard two-point crossover is used. The mutation operator changes the gene to the allele 1 when a gene with alleles 0 or 2 must be mutated, and randomly to 0 or 2 when a gene with allele 1 must be mutated.

3.2.4 Cooperative Interaction Scheme for the Cooperative Coevolutionary Algorithms

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

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

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.

Whilst the sets $R_{1|2}$ allow the best individuals to influence in the process guiding the search towards good solutions, the sets $P_{1|2}$ introduce diversity in the search. The combined use of both kinds of sets makes the algorithm have a trade-off between exploitation ($R_{1|2}$) and exploration ($P_{1|2}$). The cardinalities of the sets $R_{1|2}$ and $P_{1|2}$ are previously defined by the designer.

4 Experimental Study

The experimental study will be devoted to analyze the behavior of the different proposed hybridizations. With this aim, we have chosen two laboratory problems (two-dimensional functions F_1 and F_2 [6]) and two real-world applications (the rice taste evaluation problem [15] and the maintenance cost estimating of an electrical network in a town [8]). Seven linguistic terms for each variable are used for problems F_1 and F_2, two labels for the rice problem, and five labels for the electrical problem.

Moreover, we will compare the results of our methods with three GA-based learning methods proposed in the literature: the method proposed by Thrift [21], the method proposed in [6] to generate linguistic models following the MOGUL methodology, and the P-FCS1 method proposed in [2] that generates fuzzy models with local semantics (approximative FRBSs). Although the latter method develops PFM instead LFM since it considers a more flexible model structure that allows it to use different membership functions for each fuzzy rule, it is interesting to include it in the comparative study to analyze the performance of our methods.

Table 1 collects the results obtained by the eight analyzed methods. In that table, #R stands for the number of fuzzy rules, and MSE_{tra} and MSE_{tst} the error obtained over the training and test data sets, respectively.

From the obtained results, we can observe the good behavior of the proposed hybridizations and the high accuracy of the generated linguistic models. Even the worst results obtained by our methods outperform to the results obtained by the comparative methods. Moreover of improving the accuracy, the interpretability of the generated models is significantly higher because a lesser number of rules is used (compared to the MOGUL method and the Thrift and P-FCS1 methods in the electrical problem) and a global semantic is considered (compared to the P-FCS1 method).

Focusing on the five proposed methods, we may observe that, usually, the simultaneous learning (methods CML, C-ML, and CL-M) attains slightly better accuracy degrees than the sequential learning (methods C+ML and C+M-L). It is because the strong dependency existing among the different components is properly considered. Only in the problem F_2 the behavior is reversed. This fact seems to be related with the problem nature. Once the discontinuities existing in this problem are addressed, the learning of the fuzzy rules with the best cooperation is as simple that the two-stage sequential approach performs better. Thus, the second phase deals with a reduced search space that allows the method to obtain good solutions.

As regards the used learning technique (basic GAs or cooperative coevolution), the obtained results are contradictory. Only in the problem F_1 the cooperative coevolution is more appropriate, while in the remaining cases the basic GAs shows a similar or even better behavior.

Finally, analyzing the linguistic models generated by our methods we may observe that the excellent accuracy degrees are obtained without losing an excessive interpretability. For example, Figure 1 depicts the model obtained by the CL-M method in the problem F_1. As may be notice, the semantics and linguistic hedges used preserve an interesting symmetry that allow us to easily interpret the behavior of the model.

5 Concluding Remarks and Further Work

This paper has broached an analysis that currently is increasing in importance: the hybridization of components based on fuzzy logic with the evolutionary computation. To do that, three mechanisms to improve the accuracy in LFM have been proposed and five different combinations have been raised.

From the performed experimental study we can obtain some interesting conclusions. Generally, the fact of simultaneously perform the different improvements increases the accuracy degree since their strong interdependencies are properly considered. Furthermore, the use of more advanced search techniques such as the cooperative coevolution to treat the hybridization does not show significative results. Although they seems to be appropriate tools for such purposes, their design becomes very complex. Finally, we have verified that if the hybridization of the accuracy improvements is

Table 1: Results obtained by the analyzed methods in the four considered applications

	F_1			F_2		
Method	**#R**	**MSE**$_{tra}$	**MSE**$_{tst}$	**#R**	**MSE**$_{tra}$	**MSE**$_{tst}$
Thrift	49	1.609890	1.193721	47	0.067518	0.032442
MOGUL	85	0.300626	0.324036	96	0.011865	0.014518
P-FCS1	**48**	1.318336	0.936980	**42**	0.044838	0.038075
C+ML	49	0.213164	0.222263	47	**0.005992**	0.010902
C+M-L	49	0.250022	0.280623	47	0.006428	0.012898
CML	49	0.244005	0.300322	45	0.006745	0.011838
C-ML	49	0.216077	0.221788	45	0.006492	**0.010727**
CL-M	49	**0.179944**	**0.203371**	45	0.008155	0.014808

	Rice			*Electrical*		
Method	**#R**	**MSE**$_{tra}$	**MSE**$_{tst}$	**#R**	**MSE**$_{tra}$	**MSE**$_{tst}$
Thrift	16	0.004949	0.005991	534	34,063	42,116
MOGUL	**6**	0.001075	0.002331	57	21,709	25,290
P-FCS1	51	0.000596	0.009055	272	13,833	15,523
C+ML	17	0.000558	**0.001923**	49	8,559	12,584
C+M-L	17	0.000562	0.002052	49	9,708	11,967
CML	25	**0.000479**	0.001942	50	**4,751**	**8,422**
C-ML	25	0.000504	0.002065	**48**	7,004	11,277
CL-M	26	0.000561	0.001992	51	9,181	12,432

carefully made, the generated linguistic models shows a good degree of accuracy preserving the interpretability.

As further work, we propose to improve the cooperative coevolution by performing a better interaction scheme of the species and considering more than two species for a proper decomposition of the problem. This latter approach involves a geometric growth of the complexity that is difficult to address with the current coevolutionary proposals. Moreover, a deeper hybridization study including other combinations of soft computing partnerships is necessary.

References

[1] M. Bikdash. A highly interpretable form of Sugeno inference systems. *IEEE Trans. Fuzzy Syst.*, 7(6):686–696, 1999.

[2] B. Carse, T.C. Fogarty, A. Munro. Evolving fuzzy rule based controllers using genetic algorithms. *Fuzzy Sets Syst.*, 80:273–294, 1996.

[3] J. Casillas, O. Cordón, F. Herrera. A methodology to improve ad hoc data-driven linguistic rule learning methods by inducing cooperation among rules. Technical Report #DECSAI-00-01-01, University of Granada, Spain, 2000. Available at `http://decsai.ugr.es/~casillas/`.

[4] J. Casillas, O. Cordón, F. Herrera, L. Magdalena (Eds.) *Advance in fuzzy rule-based modeling. Part I, accuracy improvements preserving interpretability. Part II, interpretability issues.* Physica-Verlag, 2001. In press.

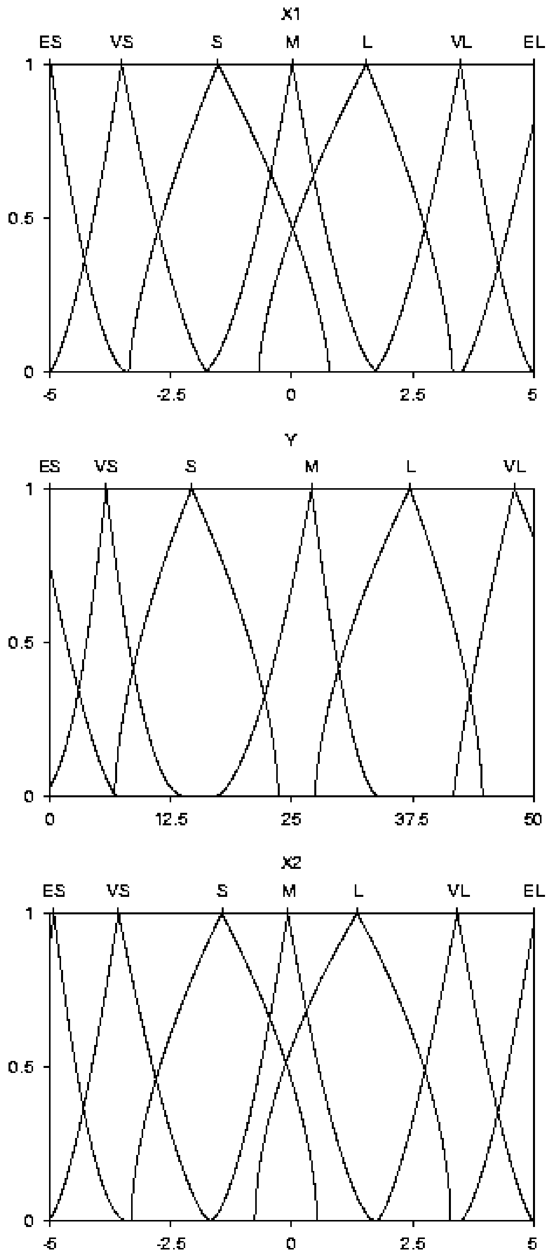
[5] O. Cordón, M.J. del Jesus, F. Herrera. Genetic learning of fuzzy rule-based classification systems cooperating with fuzzy reasoning methods. *International Journal of Intelligent Systems*, 13:1025–1053, 1998.

[6] 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.

[7] O. Cordón, F. Herrera, F. Hoffmann, L. Magdalena. *Genetic fuzzy systems: evolutionary tuning and learning of fuzzy knowledge bases.* World Scientific, 2001.

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

[9] D.E. Goldberg. A meditation on the computational intelligence and its future. Technical Report Illigal 2000019, Illinois Genetic



X2 \ X1	ES	VS	S	M	L	VL	EL
ES	VL	L	very M	mol mol M	very M	L	VL
VS	L	M	S	VS	S	M	L
S	very M	S	VS	very ES	VS	S	very M
M	very M	very VS	ES	mol mol ES	ES	very VS	very M
L	very M	S	VS	ES	VS	S	very M
VL	L	M	S	very VS	S	M	L
EL	VL	L	very M	very M	very M	L	VL

Figure 1: Linguistic model generated by the CL-M method in the problem F_1 (#R=49, $MSE_{tra/tst}$ = 0.179944 / 0.203371)

Algorithms Laboratory, University of Illinois at Urbana-Champaign, 2000. Available at http://www-illigal.ge.uiuc.edu/.

[10] A. González, R. Pérez. A fuzzy theory refinement algorithm. *Inte. J. Approximate Reasoning*, 19(3-4):193 220, 1998.

[11] A. González, R. Pérez. A study about the inclusion of linguistic hedges in a fuzzy rule learning algorithm. *Int. J. Uncertainty, Fuzziness and Knowledge-Based Syst.*, 7(3):257 266, 1999.

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

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

[14] B.-D. Liu, C.-Y. Chen, J.-Y. Tsao. Design of adaptive fuzzy logic controller based on linguistic-hedge concepts and genetic algorithms. *IEEE Trans. Syst. Man Cybern. Part B: Cybern.*, 31(1):32 53, 2001.

[15] K. Nozaki, H. Ishibuchi, H. Tanaka. A simple but powerful heuristic method for generating fuzzy rules from numerical data. *Fuzzy Sets Syst.*, 86(3):251 270, 1997.

[16] N.R. Pal, K. Pal. Handling of inconsistent rules with an extended model of fuzzy reasoning. *J. Intell. Fuzzy Syst.*, 7:55 73, 1999.

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

[18] C.A. Peña-Reyes, M. Sipper. Fuzzy CoCo: a cooperative coevolutionary approach to fuzzy modeling. *IEEE Trans. Fuzzy Syst.*, 9(5):727 737, 2001.

[19] H. Roubos, M. Setnes. Compact and transparent fuzzy models and classifiers through iterative complexity reduction. *IEEE Trans. Fuzzy Syst.*, 9(4):516 524, 2001.

[20] M. Setnes, R. Babuška, U. Kaymak, H.R. van Nauta Lemke. Similarity measures in fuzzy rule base simplification. *IEEE Trans. Syst. Man Cybern. Part B: Cybern.*, 28(3):376 386, 1998.

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

[22] J. Yen, L. Wang. Simplifying fuzzy rule-based models using orthogonal transformation methods. *IEEE Trans. Syst. Man Cybern. Part B: Cybern.*, 29(1):13 24, 1999.