Fuzzy XCS: An Accuracy-Based Fuzzy Classifier System *

Jorge Casillas
Dept. Computer Science & Artif. Intell.
University of Granada
Granada E-18071, Spain
casillas@decsai.ugr.es

Brian Carse, Larry Bull
Fact. Comput., Eng. & Mathematical Sci.
University of the West of England
Bristol BS16 1QY, UK
Brian.Carse, Larry.Bull@uwe.ac.uk

Abstract

The issue of rule generalization has received a great deal of attention in the discrete-valued learning classifier system field. Within it, the accuracy-based XCS system is currently the main reference.

The same issue does not appear to have received a similar level of attention in the case of the fuzzy classifier system. It may be due to the difficulty to extend the discrete-valued system behavior to the continuous case.

The intention of this contribution is to propose a theoretical approach to properly develop a fuzzy XCS system, the called F-XCS.

Keywords: learning classifier systems, XCS, fuzzy systems, fuzzy implication operators, evolutionary algorithm.

1 Introduction

The fuzzy classifier system is a machine learning system which employs linguistic rules and fuzzy sets in its representation and an evolutionary algorithm (EA) for rule discovery. It therefore combines an easily understood representation (as opposed to, for example, neural networks approaches) with a general purpose search method. In order to exploit the fuzzy representing to the full, the ability to learn generalization is of great importance.

Generalized rules allow more compact rule bases, scalability to higher dimensional spaces, faster inference, and better linguistic interpretability. The issue of rule generalization, and the interplay between general and specific rules in the same evolving population, has received a great deal attention in the discrete-valued classifier system research community (e.g. [11]). The same issue does not appear to have received a similar level of attention in the case of fuzzy classifier systems [1, 2, 3, 4, 6, 7, 8, 9, 10].

Traditional Michigan-style classifier systems have been "strength-based" in the sense that a classifier accrues strength during interaction with the environment (through rewards and/or penalties). This strength can then be used for two purposes: resolving conflicts between simultaneously matched classifiers during learning episodes; and as the basis of fitness for the EA. A completely different approach can be taken in which a classifier's fitness, from the point of view of the EA, is based on its "accuracy," i.e. how well a classifier predicts payoff whenever it fires. Such an accuracy based approach offers a number of advantages such as avoiding over-general classifiers, obtaining optimally general classifiers, and learning of a complete "covering map." This accuracy-based classifier system, called XCS, was proposed in [11] and it is currently the main reference to the research community in this field.

This work aims at proposing a new theoretical approach to achieve accuracy-based Michigan-style classifier systems. The proposal, fuzzy XCS (F-XCS), is based on XCS but properly adapted to fuzzy systems. The fuzzy inference mechanism and the reinforcement component are changed to consider a different, competitive interaction among rules that allow an accuracy-based behavior.

The paper is organized as follows. Section 2 introduces some difficulties to develop an accuracy-based fuzzy classifier system. Section 3 describes a competitive inference mechanism. Section 4 introduces the proposed F-XCS system. Section 5 concludes.

* Part of this research has been developed during a research stay of J. Casillas at the Faculty of Computing, Engineering and Mathematical Sciences of the University of the West of England (Bristol, UK) supported by the *Secretaría General de Universidades e Investigación de la Junta de Andalucía* (Spain)

2 Difficulties in Accuracy-Based Fuzzy Classifier Systems

Most of the difficulty in accuracy-based fuzzy classifier system is the fuzzy inference process, therefore, it is briefly explained in next subsection. Then, the problems when using strength-based Michigan fuzzy classifier systems, the advantages of considering accuracy-based fitness, and the difficulties in doing that are introduced.

2.1 The Most Frequently Fuzzy Inference Used: Cooperation

Linguistic (or Mamdani-type) fuzzy rule-based systems are formed by linguistic fuzzy rules with the following structure:

...
also
R^{i-1}: **IF** X_1 is A_1^{i-1} and ... and X_n is A_n^{i-1}
THEN Y_1 is B_1^{i-1} and ... and Y_m is B_m^{i-1} ,
also
R^i: **IF** X_1 is A_1^i and ... and X_n is A_n^i
THEN Y_1 is B_1^i and ... and Y_m is B_m^i ,
also
...

with X_i and Y_j being input and output linguistic variables respectively, and with A_i and B_j being linguistic labels with associated fuzzy sets defining their meaning. These linguistic labels are in a global semantic defining the set of possible fuzzy sets used for each variable.

The inference process, which obtains an output as response to a specific input, consists basically of the following steps.

For each fired (or matched) fuzzy rule, firstly the conjunction (***and***) operator is used to obtain the *matching degree* of the rule; secondly, the implication (***then***) operator is applied to scale the output fuzzy set to a degree according to the matching. Finally, the last step is to apply the aggregation (***also***) operator to reduce the combined output of all the rules acting together to a single fuzzy set. If we need a real-valued output, a last stage (defuzzification), converts the output fuzzy set to a number. Center of gravity, center of sum, or mean of maxima are some possibilities to do that.

The most widely inference scheme used in linguistic fuzzy systems is that proposed by Mamdani and Assilian in the first fuzzy controller of 1975 [5]. It involves the combination of operators (*min*,*min*,*max*), i.e., minimum for conjunction, minimum for implication, and maximum for aggregation. This inference is usually named Max-Min. Alternatively, sometimes it uses another t-norm (e.g. product) to play the role of conjunction and/or implication, or another t-conorm (e.g. bounded sum) to the also operator.

This is the most commonly used philosophy (we could say even the only one) followed in engineering problems such as fuzzy control and fuzzy modeling. It is because of, with these operators, the fuzzy rules "cooperate" to generate the output, in the sense that an Interpolative Reasoning is performed to define the output in the zones where several rules work to a medium matching degree. Indeed, the fact of using a t-conorm to aggregate the information involves that we are using a *union* and, therefore, the effect of each rule is added to the final consensual output.

2.2 Problems with "Strength-Based" Michigan Fuzzy Classifier Systems

As mentioned in the introduction, strength-based fuzzy classifier as characterized by using the same parameter to resolve conflicts between matched classifiers and to compute their fitness. A number of problems arise from this dual use of classifier strength. These include:

1. The cooperation/competition problem. High-strength, potentially cooperative classifiers go on to compete under the action of the EA.

2. Over-general rules with relatively high (but inconsistent) payoff can come to dominate the population.

3. In some environmental states, the maximum payoff achievable (by performing the best possible action for that state) may be relatively low. Although a classifier might be the best that can exist for that state, it can be eradicated from the population by other classifiers that achieve higher rewards in other states. This results in gaps in the system's "covering map."

2.3 Advantages of using "Accuracy-Based" Fitness

A completely different approach can be taken in which a classifier's fitness, from the point of view of the EA, is based on its "accuracy" i.e. how well a classifier predicts payoff whenever it fires. Such an accuracy based approach offers a number of advantages. Firstly, it can distinguish between accurate and over-general classifiers: an over-general classifier will have relatively low accuracy since payoff will vary according to the input states covered by the classifier. Indeed, it has been shown that the accuracy-based approach can lead to evolution of optimally general classifiers. Additionally it can maintain both consistently correct and consistently incorrect classifiers which allows learning of a complete "covering map." A potential drawback of the accuracy-based approach is that it is likely to require larger populations of classifiers.

2.4 Difficulties in Moving to "Accuracy-Based" Fuzzy Classifier Systems

Firstly, in a traditional fuzzy classifier system several rules fire in parallel (this is how the system achieves interpolation); credit assignment is much more difficult in the fuzzy case and it may well be that apportioning credit in proportion to a fuzzy classifier's activation level is not appropriate. A further difficulty is measuring the accuracy of a rule's predicted payoff since (particularly early in the search) a fuzzy rule will fire with many different other fuzzy rules at different time-steps, giving very different payoffs. Yet another difficulty is that the payoff a fuzzy rule receives depends on the input vector — an active fuzzy rule will receive different payoffs for different inputs. This further complicates payoff predictions used as the basis for accuracy based fitness.

3 The Competitive Fuzzy Inference

The problems explained in the previous section are mainly due to the fact that, usually, all the matched rules *cooperate* to define the final solution in a interpolative behavior as explained in Section 2.1. If it is the problem, why not to change our philosophy and look for a competitive interaction! This section introduces such a approach.

3.1 Why Use a Competitive Inference?

Michigan-style learning classifier systems in general, and XCS in particular, do not consider the interaction of the different rules as a cooperative action, instead they consider that each rule competes with the rest to be the best one for a particular input vector. It involves that the action is only due to a set of rules that were the winners among the rules composing the match set M. Thus, we should think if in fuzzy modeling it makes sense to deal with competitive fuzzy rules and, if so, how to work with them.

In off-line fuzzy system learning, it seems that cooperative inference has some clear advantages that make it be more suitable for this problem. In this case, we known the whole data set *a priori* and our objective is to build a landscape of the output data. To do that, it seems a good approach to fill the gaps between data with interpolation, and it is the reason way cooperative inference is successful.

However, in on-line fuzzy system learning, why is it interesting to interpolate the actions of the rules? The objective in this problem should not be to define a landscape of the output, but a landscape of the reward. It means to take the best action in each state. For this purpose, a competitive action makes more sense since our objective is to optimize the rules individually to have the best reward.

With competitive fuzzy inference we understand a decision making process where the matched rules compete among themselves, and only one is the winner. The output finally obtained is mainly due to the winner rule. That is, loser rules have not an important influence on the output. Nevertheless, competition does not involves an interaction loss. There is an interaction, but with a selfish objective. To perform competitive inference we only need to change the roles of the inference operators.

3.2 Fuzzy Operators for a Competitive Inference

First of all, since each rule competes with the rest, we should use an intersection (t-norm) as *also* operator, instead of the union approach followed by the cooperative inference. We use the *minimum* in this paper. However, this change does not allow us to use the same kind of implication operator.

Analyzing the classical Boolean implication $p \Rightarrow q$, equivalent to $\neg p \vee q$, it is true when p is false. That is, when we have not information about the certainty of the fact (antecedent), we assume the "optimistic" view of thinking that the implication is true.

However, in the cooperative inference, a t-norm (intersection) is used as implication. It is a "pessimistic" view that assumes the falseness of the implication when the fact is false. It works if we use a union (t-conorm) to aggregate the effect of the different rules. This behavior could be interpreted that the belief of the rules resides in the set of them, instead of in each of them.

To develop a competitive inference where a intersection is considered as aggregation, the implication should follow the classical Boolean if-then view assuming that each rule has all the knowledge about the relationship between fact and consequence. Thus, the belief develops for each rule. This family of implications are known as *logical* fuzzy implications, instead of the *engineering* fuzzy implications based on the t-norm.

Table 1 shows the behavior of each fuzzy implication type in extreme values (0 is equivalent to false and 1 equivalent to true), while Table 2 includes some of the best known *logical* fuzzy implications.

Table 1: Implications (0=False, 1=True)

		Boolean	**Fuzzy**	
p	q	$\Rightarrow$	*Engineering* implications	*Logical* implications
0	0	1	0	1
0	1	1	0	1
1	0	0	0	0
1	1	1	1	1

Table 2: Some *logical* fuzzy implications

Operator	$\mu_R(x, y)$
Kleene-Dienes	$max\{1 - \mu_A(x), \mu_B(y)\}$
Zadeh	$max\{min\{\mu_A(x), \mu_B(y)\}, 1 - \mu_A(x)\}$
Łukasiewicz	$min\{1, 1 - \mu_A(x) + \mu_B(y)\}$
Dubois-Prade	$\begin{cases} 1 - \mu_A(x) & \text{if } \mu_B(x) = 0 \\ \mu_B(x) & \text{if } \mu_A(x) = 1 \\ 1 & \text{otherwise} \end{cases}$
Gödel	$\begin{cases} 1 & \text{if } \mu_A(x) \leq \mu_B(x) \\ \mu_B(x) & \text{otherwise} \end{cases}$
Goguen	$\begin{cases} 1 & \text{if } \mu_A(x) < \mu_B(x) \\ \frac{\mu_B(x)}{\mu_A(x)} & \text{otherwise} \end{cases}$

To understand better the different behavior between cooperative and competitive inference, Figure 1 shows the output generated by two fuzzy systems that only differ in the inference mechanism. In the cooperative inference, minimum, minimum, and maximum are used as conjunction, implication, and aggregation operators, respectively. In the competitive inference, minimum, Łukasiewicz, and minimum are used as conjunction, implication, and aggregation operators, respectively. The center of gravity is used as defuzzification process in both cases. Both systems consist of two input variables and one output variable, with 5 triangular fuzzy sets uniformly distributed in the universe of discourse [0,1] for each variable. Among the 25 fuzzy rules considered, only 7 have a consequent different from the medium linguistic term (with vertex at 0.5). As we can see, both approaches generate a smooth output but, while the cooperative inference fill the gaps by interpolating, the competitive inference isolates the effect of each rule.

On the other hand, Figure 2 illustrates some examples of the competition inference depending on the used implication operator. The *minimum* aggregation operator is used in all the cases. We can see how the action strength of the winner rule (the one with the highest matching degree) not only depends on its matching degree but also on the difference in the matching degree with respect to fuzzy rules with different actions

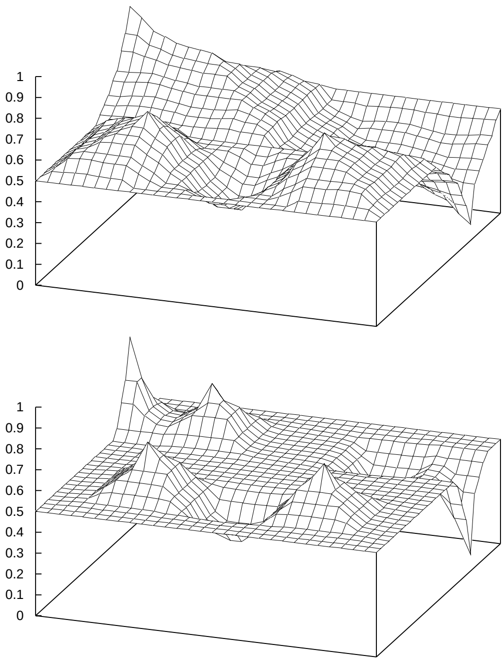

Figure 1: Output generated by cooperative (up) and competitive (bottom) inferences

(consequents), i.e., rival fuzzy rules. The more different these matching degrees, the higher the action strength of the selected rule. It involves that actually, the competition is performed among different linguistic actions. Therefore, we should say that the competitive inference leads to competitive actions, but not necessarily competitive rules.

The competition among rules with the same action is only performed when using the minimum operator as aggregation, since in this case only the best rule of that action is considered in the inference. The use of other aggregation operator such as the product will lead the inference to a cooperation among rules with the same actions. According to XCS, this approach is not desired since it aims at obtaining the $X \times A \Rightarrow P$ map with the best classifier for each situation-action combination [11].

We can characterize slightly different interactions approaches among rules with different actions (i.e. rival rules) depending on the logical fuzzy implication operator used; these are shown in Table 3.

Finally, we should say that the consideration of competitive actions seems to fit properly with the XCS aim. Other fuzzy learning classifier system proposals, like Bonarini's work [1, 2], focus on the interaction between rules in the antecedent (state) instead of the consequent (action). He proposes a competition between rules with the same antecedent but a cooperation between rules with different antecedents. More-

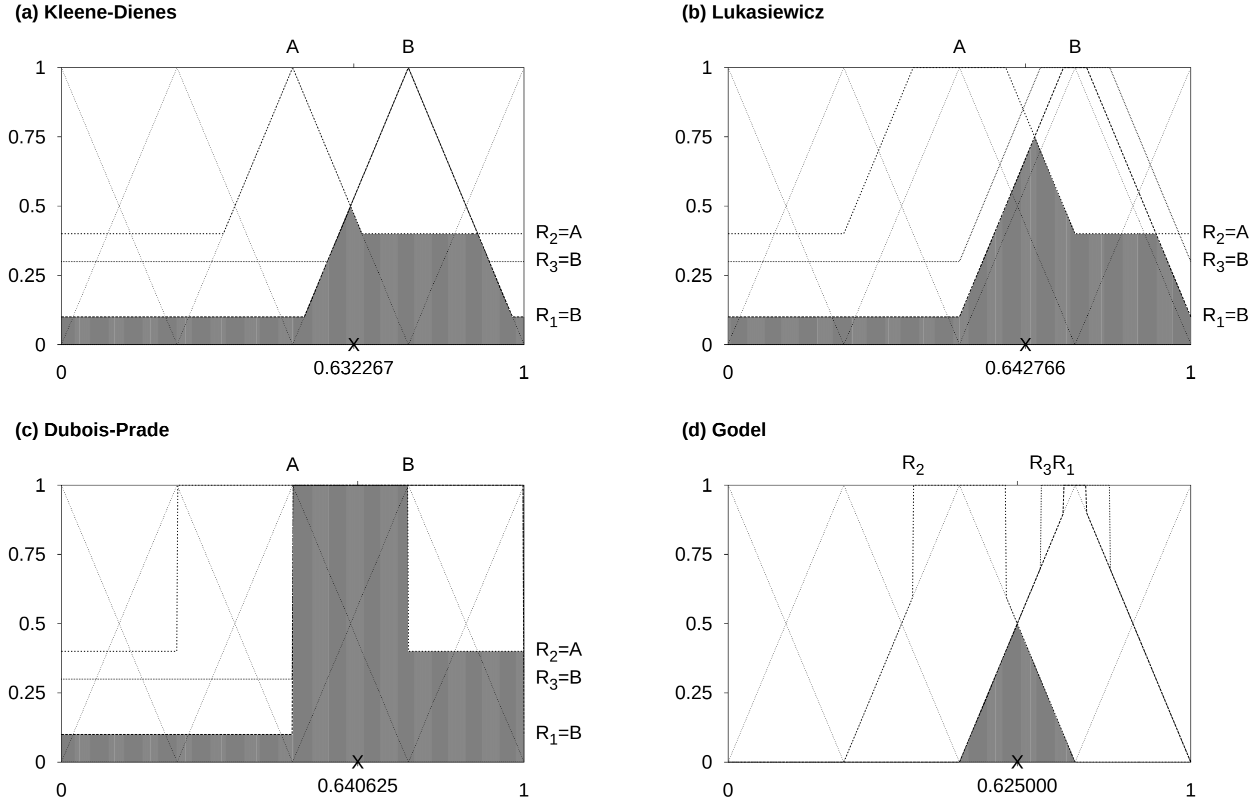

Figure 2: Results of four different fuzzy implications using minimum as *also* operator. Matching degrees: $\mu_{A_{R_1}} = 0.9$, $\mu_{A_{R_2}} = 0.6$, and $\mu_{A_{R_3}} = 0.7$

over, the implication and aggregation considered in his proposal leads implicitly to a cooperation between the consequents.

4 F-XCS

The use of competitive fuzzy rules seems to solve the problems shown by the cooperative approach (Section 2.4). Let us to see how F-XCS would work.

4.1 Generalization Representation

First of all, a representation of fuzzy classifiers to allow proper generalization must be done. We propose the use of *disjunctive normal form* (DNF) fuzzy rules with the following structure:

$$\textbf{IF } X_1 \text{ is } \widetilde{A_1} \text{ and } \ldots \text{ and } X_n \text{ is } \widetilde{A_n} \textbf{ THEN } Y \text{ is } B$$

where each input variable X_i takes as a value a set of linguistic terms $\widetilde{A_i} = \{A_{i1} \vee \ldots \vee A_{il_i}\}$, whose members are joined by a disjunctive (t-conorm) operator, whilst the output variable remains a usual linguistic variable with a single label associated. We use the *bounded sum* ($min\{1, a+b\}$) as t-conorm.

This structure uses a more compact description that allows rules with different generalization degrees. Moreover, the structure naturally supports the absence of some input variables in each rule (simply making $\widetilde{A_i}$ be the whole set of linguistic terms).

4.2 Performance Component

In XCS [11], the performance components consists of three stages: match set (M) construction, prediction array computation, and action set selection. This process has the final objective of inferring an specific action from the matched rule set.

While these stages are necessary in discrete output systems like XCS to select an action from a finite set, in real-valued output systems like F-XCS, this process should not be followed, since the output landscape must be smooth. Thus, in our F-XCS proposal we will replace the performance component by a competitive fuzzy inference, as explained in the previous section.

To consider the importance of each fuzzy rule, the matching degrees are weighted by the corresponding

Table 3: Type of competition among rules with different actions depending on the used *implication* operator (a t-norm is used as *also* operator)

Implication	Type of competition
Kleene-Dienes	Rival rules reduce the importance of the winner rule in non-overlapped areas
Zadeh	Like Kleene-Dienes. No interaction among rules if the winner rule has a matching degree lower than 0.5
Łukasiewicz	Like Kleene-Dienes, but with higher preservation of overlapped areas than it
Dubois-Prade	Like Kleene-Dienes, but with the highest preservation of overlapped areas
Gödel	Output always in the overlapped area. No output with more than two different actions. If matchings greater than cross-point, output in the center of the overlapped area
Goguen	Like Gödel. Better with *mean of maximum* defuzzification

fitness value prior to apply the implication operator:

$$m_j = \frac{f_j}{\sum_{R_i \in M} f_i} \cdot \mu_{A_{R_j}}(x) \tag{1}$$

where m_j is the prediction degree of the classifier R_j, M is the match set, f_j is the the fitness of R_j, and $\mu_{A_{R_j}}(x)$ is its matching degree.

This inference would be like use fuzzy rules with weights, were each weight is the normalized fitness, i.e. the importance factor, of the corresponding rule.

4.3 Reinforcement Component

The p_j (prediction), ϵ_j (prediction error), and F_j (fitness) values are adjusted by the reinforcement learning standard techniques used in XCS (Q-learning, Widrow-Hoff, and MAM) for each fuzzy classifier C_j.

However, two important differences from XCS are considered in F-XCS:

- Firstly, the fitness adjustment acts on the whole match set M since in F-XCS there is not an action set.
- Secondly, the distribution among the classifiers must be made proportionally to the degree of contribution of each classifier to the obtained output. This is a crucial issue because the interaction among the classifiers (with competition actions in our case) is developed here.

The following subsections details the reinforcement distribution process and the adjustment of the parameters.

4.3.1 Distribution of the Reinforcement

The reinforcement distribution among the classifiers of the match set M is made by analyzing the contribution of each classifier to generate the aggregated output fuzzy set. Let B'_j be the scaled output fuzzy set generated by the fuzzy rule R_j:

$$B'_j = I(m_j, B_j), \tag{2}$$

where m_j is the prediction degree (see eq. (1)), I is the used logical fuzzy implication operator (see Table 2), and B_j the fuzzy set of the consequent of the rule R_j. Let $R_1 \in M$ be the winner rule, i.e., $m_1 \geq m_j, \forall R_j \in M - \{R_1\}$.

The process involves analyzing the area that the rival fuzzy rules (rules with lower m_j than R_1) "bite" into the area generated by the winner rule.

Thus, the weight of the winner rule is:

$$w_1 = \frac{\int \bigwedge_{j=1}^{|M|} B'_j \, dy}{\int B'_1 \, dy}, \tag{3}$$

with $|M|$ being the match set size and $\wedge$ the t-norm used as aggregation operator (the minimum in our case), while the weights of the rival rules is computed as follows:

$$w_j = \frac{(1 - w_1) \cdot \left(\int B'_1 \, dy - \int B'_1 \wedge B'_j \, dy\right)}{\sum_{i=2}^{N} \left(\int B'_1 \, dy - \int B'_1 \wedge B'_i \, dy\right)} \tag{4}$$

This distribution is made due to the considered competitive inference. To illustrate this behavior, we can see that, from the example shown in Figure 2(a), the competitive inference generates the weights $w_1 = 0.711$, $w_2 = 0.289$, and $w_3 = 0$, while a cooperative-based distribution proportional to the matching degrees generates the weights $w_1 = 0.409$, $w_2 = 0.318$, and $w_3 = 0.273$. Indeed, the selection pressure in the competitive inference is higher, thus allowing to discriminate between good and bad rules.

4.3.2 Adjustment of the Parameters

To adjust the parameters of each classifier, firstly the P (payoff) value is computed with the Q-learning technique as follows:

$$P = r + (\gamma \cdot m_1) \tag{5}$$

with m_1 (eq. 1) being the maximum prediction degree, r the external reward from the previous time-step, and $\gamma \in [0, 1]$ a constant decreasing factor.

Then, the following adjustment process is performed (the weights w_j computed in eqs. (3) and (4) are considered to distribute the adjustment):

1. Firstly, recalculate the fitness values F_j from the current values of ϵ_j by using the standard Widrow-Hoff delta rule with learning rate parameter β ($0 < \beta \leq 1$), i.e. $F_j \leftarrow F_j + \beta(k'_j - F_j)$, with
$$k'_j = w_j \cdot \frac{k_j}{\sum_{R_i \in M} k_i} \tag{6}$$
and
$$k_j = \begin{cases} \exp((\ln \alpha)(\epsilon_j - \epsilon_0)/\epsilon_0) & \epsilon_j > \epsilon_0 \\ 1 & \text{otherwise.} \end{cases} \tag{7}$$
The MAM (moyenne adaptive modifiee) technique is used by adjusting F_j to the average of the k'_j values, instead of the above equation, during the first $1/\beta$ times that the corresponding classifier is adjusted.

2. Then, adjust the error values ϵ_j using the Widrow-Hoff technique toward $w_j \cdot |P - p_j|$, i.e. $\epsilon_j \leftarrow \epsilon_j + \beta(w_j \cdot |P - p_j| - \epsilon_j)$. MAM technique is also used here during the $1/\beta$ first adjustments.

3. Finally, adjust prediction values $p_j \leftarrow p_j + \beta(w_j \cdot P - p_j)$. Again, MAM technique is used at the beginning.

4.4 Discovery Component

As in XCS, the EA for F-XCS acts only on the match set M. It selects two classifiers from the match set with probabilities proportional to their fitness, applies crossover and mutation operators with probabilities χ and μ, respectively, and inserts the offspring in the population. If the population contains the maximum number of fuzzy classifiers allowed, two individuals are deleted to make room. They are randomly selected proportionally to the mean match set sizes where each fuzzy classifier was involved.

When no fuzzy rules cover the state, i.e. $M = \emptyset$, a covering mechanism is used to include a fuzzy classifier with the input linguistic term set that best matches the input, and a random consequent.

5 Concluding Remarks

The paper has presented a theoretical proposal to properly develop an accuracy-based fuzzy classifier system. It is mainly based on a different inference approach that consider the interaction among fuzzy classifiers (rules) from a competitive point of view. The reinforcement component, based on XCS, is adapted to allow this behavior. Immediate work involves experimentally investigating the proposal.

References

[1] A. Bonarini. Evolutionary learning of fuzzy rules: competition and cooperation. In W. Pedrycz (Ed.), *Fuzzy modelling: paradigms and practice*, pages 265–284. Kluwer Academic, Norwell, MA, USA, 1996.

[2] A. Bonarini and V. Trianni. Learning fuzzy classifier systems for multi-agent coordination. *Information Sciences*, 136:215–239, 2001.

[3] T. Furuhashi, K. Nakaoka, and Y. Uchikawa. Suppression of excess fuzziness using multiple fuzzy classifier systems. In *Proc. 3th IEEE Int. Conf. on Fuzzy Systems*, pp. 411–414, Piscataway, NJ, USA, 1994.

[4] H. Ishibuchi, T. Nakashima, and T. Murata. Performance evaluation of fuzzy classifier systems for multidimensional pattern classification problems. *IEEE Trans. on Syst., Man, and Cybern.—Part B: Cybern.*, 29:601–608, 1999.

[5] E.H. Mamdani and S. Assilian. An experiment in linguistic systhesis with fuzzy logic controller. *Int. J. of Man-Machine Studies*, 7:1–13, 1975.

[6] K. Nakaoka, T. Furuhashi, and T. Uchikawa. A study on apportionment of credits of fuzzy classifier systemfor knowledge acquisition in large scale systems. In *Proc. 3th IEEE Int. Conf. on Fuzzy Systems*, pp. 1797–1800, Piscataway, NJ, USA, 1994.

[7] A. Parodi and P. Bonelli. A new approach to fuzzy classifier systems. In *Proc. 5th Int. Conf. on Genetic Algorithms*, pp. 223–230, San Mateo, CA, USA, 1993. Morgan Kaufmann Publishers.

[8] M. Valenzuela-Rendón. The fuzzy classifier system: a classifier system for continuously varying variables. In *Proc. 4th Int. Conf. on Genetic Algorithms*, pp. 346–353, San Mateo, CA, USA, 1991. Morgan Kaufmann Publishers.

[9] M. Valenzuela-Rendón. Reinforcement learning in the fuzzy classifier system. *Expert Systems with Applications*, 14:237–247, 1998.

[10] J.R. Velasco. Genetic-based on-line learning for fuzzy process control. *Int. J. Intell. Syst.*, 13:891–903, 1998.

[11] S.W. Wilson. Classifier fitness based on accuracy. *Evolutionary Computation*, 3(2):149–175, 1995.