Comments on “FuzzyShell: A Large-Scale Expert System Shell Using Fuzzy Logic for Uncertainty Reasoning”

Zenon A. Sosnowski

Abstract—The above paper presents a rather complicated mechanism allowing fuzzy evidence to be aggregated when fuzzy inferences are made about the same fuzzy variable by different rules. The purpose of this comment is to present that the FuzzyCLIPS address this problem in simpler way.

Index Terms—Expert systems, fuzzyCLIPS, Rete network.

I. INTRODUCTION

I would like to congratulate Dr. Pan et al. for their paper, which provides a very comprehensive, clear, and detailed explanation of the Rete network. Nevertheless, I do not agree with the following statement on p. 563.

“Because of challenges unique to fuzzy expert systems design, especially when multiple levels of inference are involved, there is always the question of whether the developers of a shell or an expert system took any liberty with the logic itself in order to get the entire software system to work. The special difficulties that fuzzy reasoning presents—difficulties that do not exist for the case of crisp reasoning—are twofold.

1) All the rules that contain the same linguistic variable on the consequent side must be fired and evidence aggregated for that linguistic variable in accordance with the method of inference chosen before a fact asserted by one such rule is allowed to trigger another rule. Said another way, an asserted fact about a linguistic variable cannot be allowed to trigger another rule if there are other rules yet to be fired that would assert further facts about the same linguistic variable.

2) Since fuzzy terms often have overlapping membership functions, a fact such as (robot speed fast) should be able to match with a rule antecedent such as (robot speed medium), if there exists an overlap between the membership functions for the fuzzy terms fast and medium.

While it is clear that all the fuzzy expert system shells, FuzzyCLIPS being a case in point are able to handle the second difficulty, the same cannot be said for the first difficulty. In fact, we are not aware of any fuzzy expert system shell that handles cleanly the first difficulty."

I think that the postulate mentioned in the first difficulty can be valid only for the fuzzy logic controller type expert systems but it is not true, in general, as it would be inconsistent with other important issues of expert systems such as rule priority and rule confidence factors. In the first case, preventing to trigger another rule before firing of all rules that contain the same linguistic variable on the consequent side would violate one of the fundamental principle of expert system (especially real time expert system) namely rule priority. In the later case, there is a standard practice in expert systems that allow rules to have a certainty factors attached, to fire rules with the higher confidence factor and not to fire at all rules with certainty factors below a certain threshold value.

When FuzzyCLIPS [1] has to work as a fuzzy logic controller the first difficulty is overcome by a special procedure that we call a “global contribution” [2]. As an example of the importance of the global contribution, consider the implementation a fuzzy logic controller. In this case, the user has to ensure the firing all rules that contribute to the control action to be performed, before any other rule (usually defuzzification) fires.

II. GLOBAL CONTRIBUTION

In a crisp expert system there is never any need to reassess the facts in the system—once they exist, they exist (unless certainty factors are being used, then certainty factors are modified to account the new evidence and checked if they are larger then a threshold value). But in a fuzzy expert system, refinement of a fuzzy fact may be possible. Consequently, the fact concerning fuzzy variable is potentially useful for every rule whose left-hand side (LHS) contains this fuzzy variable and every value of a fuzzy variable may be repeatedly reevaluated. Thus, in the case where the fact concerning a linguistic variable is asserted as a result of performing the right-hand side (RHS) action of a rule, this fact is treated as giving contributing evidence about the fuzzy variable (it contributes globally). If information about that fuzzy variable has already been asserted then this new evidence (or information) about the fuzzy variable is combined with the existing information about the fuzzy variable. There are many readily identifiable methods of combining evidence. In FuzzyCLIPS, the new value of the fuzzy fact is reassessed in accordance with the formula

\[ \mu_{\tilde{A}} = \max \left( \mu_{\tilde{A}'} \cdot \mu_{\tilde{A}''} \right) \]

where

- \( \mu_{\tilde{A}} \) new value of the linguistic variable;
- \( \mu_{\tilde{A}'} \) membership function of the existing value of the linguistic variable;
- \( \mu_{\tilde{A}''} \) membership function of the linguistic variable to be asserted.

More detailed information about the implementation of global contribution and its influence on the Fuzzy RETE network can be found in [3].

III. SPECIFIC EXAMPLE ILLUSTRATING THE FIRST DIFFICULTY

Concerning the example (p. 564) with the following three rules:

1. Rule 1 IF \((X = A)\) THEN \((Y = B)\)
2. Rule 2 IF \((X = A')\) THEN \((Y = B')\)
3. Rule 3 IF \((Y = B)\) THEN \((Z = C)\).

Assuming that the working memory contains the fact \((X = A)\) and that the compositional rule of inference is implemented as a sup-min composition it is easy to see that the final conclusions derived for the values of \(Y\) and \(Z\) do not depend on the order of rules being fired. Let us now examine how the FuzzyCLIPS carries out fuzzy reasoning. The fact \((X = A)\) enables Rule 1 and Rule 2 to be activated (put on the agenda). Of course we assume that terms \(A\) and \(A'\) associated with the

1 There is a bug in implementation of the “global contribution” mechanism in some versions of FuzzyCLIPS (e.g., FuzzyCLIPS Version 6.02a, 1994).
linguistic variable $X$ have overlapping membership functions. In the next step (does not matter which conflict resolution strategy is applied) Rule 1 or Rule 2 fires. After any of this rule fired, the fact ($Y$ is $B$) or ($Y$ is $B'$) would be asserted into the working memory. This new fact would enable Rule 3 (we assume that terms $B$ and $B'$ of the linguistic variable $Y$ have overlapping membership functions). Now we have Rule 1 (or Rule 2) and Rule 3 on the agenda. In the next steps the dictates of fuzzy logic are not violated as the final conclusions derived for the values of $Y$ and $Z$ will not depend on the order of rules being fired. Let us assume that MEA strategy of conflict resolution is applied and the Rule 3 fires first. As a result the fact concerning the value of linguistic variable $Z$ (equal to $C$ if Rule 1 had already fired) would be asserted into the working memory. Now it is time for Rule 2 (or Rule 1) to fire. After this rule fired, the fact concerning the new value of linguistic variable $Y$ refines the existing value of this variable in the working memory according to global contribution. Updating the value of linguistic variable $Y$ enables Rule 3 again. After Rule 3 fired for the second time, a “global contribution” concerning the linguistic variable $Z$ is performed. The final values of linguistic variables $Y$ and $Z$ are equal to those obtained if the order of firing rules would be one, two, and three.

IV. A NUMERICAL EXAMPLE

In what follows, we will concentrate on the example described in Section VI and show that the final conclusions do not depend on the order of rules to be fired. For the sake of our explanation here, let us recall some assumptions from that example.

Fact-1[robot (^mission obstacle-avoidance) (^sensor 2D-vision)];
Fact-2[2-D-vision (^pixel-position-x med-left) (^pixel-position-y far)]
(rule-name rule-1
 (robot (^mission ?x) (^sensor 2D vision))
 (2-D vision (^pixel-position-x med-left) (^pixel-position-y far))
 $\Rightarrow$
 (goal (^mission ?x) (^orientation NW)))
(rule-name rule-2
 (robot (^mission ?x) (^sensor 2-D vision))
 (2-D vision (^pixel-position-x med-left) (^pixel-position-y far))
 $\Rightarrow$
 (goal (^mission ?x) (^orientation NW)))
(rule-name rule-3
 (goal (^mission obstacle-avoidance) (^orientation NW)))
 $\Rightarrow$
 (^turn-angle right-20)
 (^distance-to-travel medium))
(rule-name rule-4
 (goal (^mission obstacle-avoidance) (^orientation NNW)))
 $\Rightarrow$
 (^turn-angle right-30)
 (^distance-to-travel long))

The membership functions on the fuzzy terms for the linguistic variables pixel-position-x, pixel-position-y, orientation, distance-to-travel, and turn-angle are the same as shown on p. 578, Fig. 25. Assume that the depth-first strategy is selected for conflict resolution, meaning that rule matched with the most recent facts will fire first. After Fact-1 and Fact-2 have been asserted into the working memory to start the inference engine, rule-1 and rule-2 will be activated and placed in the agenda. Assume that rule-2 is fired first. The firing of this rule cause the fact concerning the value of linguistic variable orientation to be asserted into the working memory with the appropriate value. Fig. 1 illustrates the inference process resulting from the firing of rule-2. Firing of rule-2 will cause the rule-3 and rule-4 to be enabled and placed on the agenda. Current agenda: rule-1, rule-3, rule-4. Assume rule-4 is fired next. As a result the facts concerning the value of the linguistic variables turn-angle and distance-to-travel will be updated to those present in the working memory by the global contribution procedure.
Fig. 3. Illustration of how the membership functions of the linguistic variables turn-angle and distance-to-travel are aggregated in the third cycle of the inference engine.

Fig. 4. Illustration of how the membership function for the linguistic variable orientation is aggregated in the fourth cycle of the inference engine.

Fig. 3 illustrates the inference process resulting from the firing of rule-3. Current agenda: rule-1. Now rule-1 is fired. As a result the rule-3 and rule-4 will be enabled and placed on the agenda again. This firing will not cause any changes to the crisp facts. However, the global contribution procedure will update the facts concerning the value of the linguistic variable orientation to that present in the working memory, as pictorially illustrated in Fig. 4. Current agenda: rule-3, rule-4. Assume rule-4 is fired first. As a result the facts concerning
Fig. 5. Illustration of how the membership functions for the linguistic variables turn-angle and distance-to-travel are aggregated in the fifth cycle of the inference engine.

Fig. 6. Illustration of how the membership functions for the linguistic variables turn-angle and distance-to-travel are aggregated in the sixth cycle of the inference engine.

the value of the linguistic variables turn-angle and distance-to-travel will be updated to those present in the working memory by the global contribution procedure. Fig. 5 illustrates the inference process resulting from the firing of rule-4. Current agenda: rule-3. Finally, rule-3 will be fired. Again, as a result the global contribution procedure will update the facts concerning the value of the linguistic variables turn-angle
and distance-to-travel to those present in the working memory, as pictorially illustrated in Fig. 6. Current agenda: empty. The inference process is complete. The results are equal to those obtained if the order of firing rules would be 1, 2, 3, and 4. It is easy to check, that any order of rule firing (for example 1, 3, 2, 4, 3) will give identical results.

V. CONCLUSION

In this comment, I described a way the FuzzyCLIPS addresses the issue of allowing fuzzy evidence to be aggregated when fuzzy inferences are made about the same fuzzy variable by different rules when an expert system works in the fuzzy logic controller regime. It seems to me that the much simpler way to handle the abovementioned difficulty in the fashion described in the cited paper would be allowing the agenda to work as a queue (a FIFO data structure).

REFERENCES


Authors’ Reply

Guilherme N. DeSouza and Avinash C. Kak

Index Terms—Expert systems, fuzzy expert systems, fuzzy logic, fuzzy reasoning, fuzzyshell, Rete networks.

We would like to thank Dr. Sosnowski for his interest in our work on FuzzyShell. While he has made some cogent observations about FuzzyShell, his main conclusions about our work are ill founded. We believe that his erroneous conclusions were caused by his misunderstanding the difference between rule firing and evidence aggregation in our work.

Referring to FuzzyShell, Dr. Sosnowski states:

“... preventing to trigger another rule before firing of all rules that contain the same linguistic variable in the consequent side would violate the one of the fundamental principle of expert system (. . .) namely rule priority.”

Dr. Sosnowski is wrong. FuzzyShell does not demand that all the rules that address the same linguistic variable in their consequents be fired before any other rule with that linguistic variable in its antecedent is allowed to fire. The main paragraph in the left column on p. 578 of our paper makes clear the distinction between rule firing and evidence aggregation. While all the rules that address the same linguistic variable in their consequents are used for evidence aggregation, only one of them is fired. The choice of the rule that is fired is dictated completely by the conflict resolution strategy used.

The difference between evidence aggregation and rule firing goes to the very heart of FuzzyShell. We would like to reproduce here some of the ending sentences of the main paragraph in the left column of p. 578 of our paper.

“Therefore, as far as fuzzy inference is concerned, we may think of rule-1 and rule-2 as being bundled together. But note that bundling together does not imply that rule-2 should be fired immediately after rule-1. In general, the consequent side of a rule (such as rule-2) will include action elements that are nonfuzzy; the inference process with regard to these action elements must proceed in the traditional manner.”

To further explain this distinction between rule firing and evidence aggregation as implemented in FuzzyShell, we will use the following example:

Rule 1 IF (X is A) THEN (Y is B) and (W is D)
Rule 2 IF (X is A') THEN (Y is B')
Rule 3 IF (Y is B) THEN (Z is C).

Assuming that the conflict resolution strategy is set to depth and assuming that the facts in the working memory match the antecedents of both Rules 1 and 2, FuzzyShell will initially put Rule 1 and Rule 2 into the agenda, with Rule 2 set first for firing on account of its recency. Since the consequents of both the rules affect the antecedent of Rule 3, the fuzzy evidence corresponding to (Y is B) of Rule 1 and (Y is B') of Rule 2 is aggregated into one single linguistic term that will subsequently be used to enable Rule 3. Note, however, that this does NOT imply that Rule 1 must be fired before Rule 3. The evidence aggregation step has no bearing on the order in which the rules are fired. In this example, where we assumed the depth strategy, the rules will be fired in the order: Rule 2 → Rule 3 → Rule 1. That can be easily established by observing that for the depth conflict resolution strategy, the term (W is D) in the consequent of Rule 1 is not asserted in the working memory until after Rule 3 is fired, regardless of whether W is a fuzzy variable or a crisp one.

We will now address what Dr. Sosnowski refers to as a simpler solution for dealing with rule chaining in fuzzy expert systems. This solution relies on a refinement of fuzzy evidence through a mechanism that amounts to reassertion of facts in the working memory. What he refers to as “refining the existing value of variables in the working memory according to global contributions” is, we believe, inherently inefficient because it relies on a reassertion of facts in the working memory after the computation of their “global contribution.” This strategy entails the following: 1) rules that were previously fired be placed back in the agenda after the system calculates the global contributions and 2) fuzzy variables be temporally defuzzified, to be “reassessed” again at a later time and then defuzzified again. Placing the rules back in the agenda due to the reassertion of facts with global contribution decreases the performance of the system by unnecessarily firing the same rule repeatedly. Intermediate defuzzification and reassessment of the fuzzy variables make the system assert erroneous fuzzy facts even if it is only for a short period of time. This can be a serious problem especially for real-time applications.

As for the unnecessary firing of rules, as the example in Section IV of Dr. Sosnowski’s comment shows, a rule that depends on two other rules (Rules 3 and 4 in that example) will be fired twice instead of just once in FuzzyShell. In general, if m rules depend on n other rules, mn rules will be fired in the proposed system, as opposed to m + n rules in FuzzyShell. This difference in the number of rules fired becomes even larger if we consider longer chains of rules. For example, m rules that depend on n rules which also depend on k other rules will cause the system to fire mn + nk times, while it would fire only m + n + k times in FuzzyShell.