On Problems of Knowledge in Fuzzy Control: Extended Abstract

Gal A. Kaminka

Computer Science Department and Information Sciences Institute
University of Southern California
4676 Admiralty Way, Marina del Rey, CA 90292
(310) 822-1511 galk@isi.edu

Introduction

The field of Fuzzy Control have enjoyed tremendous success in the last decade, with both theoretical and industrial developments being introduced at an increasing rate[1]. However, fuzzy control is just one application of Soft Computing methods in general, and Fuzzy Sets theory in particular. This principled approach to approximate reasoning is not limited only to control problems, but is useful also in closely related fields, such as Artificial Intelligence (AI) in its various forms and guises, Decision Sciences, Quantitative and Qualitative Decision Theory, theoretical studies of Uncertainty, Information and Knowledge, etc. As we seek to expand the success of fuzzy control to these fields, it is very tempting to attempt use of known methods in a familiar way to solve problems in these new domains. However, a principled analysis of the goals and assumptions underlying different fields may reveal important differences from the field of control which may make current methods insufficient.

We take the position that as we look at real-world *decision problems* and *domains*, the familiar techniques of fuzzy control will be insufficient to provide adequate solutions, because these techniques are designed with the assumptions of the field of control in mind -- assumptions which do not hold in such domains. The techniques are not incorrect, but simply insufficient, and can be augmented to provide the necessary theoretical and practical infrastructure for the new domains. This short abstract will attempt to point out our initial approach to such necessary augmentations of fuzzy control techniques.

We will motivate the discussion with a very small scale decision problem, which despite its size, captures some of the underlying problems with current techniques and points the way at the necessary directions for development that will bridge the gap between fuzzy control techniques and the required technology.

Motivating Examples

We motivate our discussion by a brief presentation of experiments in fuzzy control use for a small scale problem in the domain of mobile robot navigation, described in (Kaminka 1997). A fuzzy controller based on the SAM additive fuzzy system model (Kosko 1997) was built to provide goal-directed navigation for a mobile robot using sonar range sensors. The controller directly controlled the velocity and heading of the robot based on the sensor readings. The design employed two sets of rules which attempt to balance two opposing "forces" competing for control of the robot: "pulling" the robot towards the goal, and "pushing" it away from obstacles on route. Although varying in many details, this type of robot navigation is not unlike others reported on in the literature (e.g., Saffioti et al. 1993, Yen and Pfluger 1995, Baxter and Bumby 1995). The mission of such a controller is typically to be used as the low-level navigation layer of a more elaborate system, often employing a map-based planner which can take care of problems with blocked paths as they occur. So while the controller supplies only local optimization solutions (i.e., may get stuck in place if obstacle is directly in front of goal, because contradicting output sets will result), this is sufficient in the context of the overall robot system.

However, a close examination of the behavior of the controller without the complementary support of such a planner raises two issues: The need for negative rules (negative responses), and the detection for imperfect knowledge, which we term "fuzzy impasse" (after Newell's related definition in (Newell 1990)).

Negative Rules. Conventional rules are written such that they map inputs to outputs - perceptions to actions. But in writing rules that attempt to prevent the robot from colliding into obstacles, the designer is attempting to describe to the controller what its response *should not* be. Rather than telling the robot "if the goal is in location X, head towards X", the designer is attempting to express the rule "If an obstacle is in location X, do not head towards X". These types of rules are needed not only for mobile robot navigation, but also in other domains - for instance to rule out diagnosis conclusions once better ones are found.

From an engineering point of view, this type of rules are necessary not only to enhance the expressiveness of the systems we can design, but also to increase their modularity. If we are forced to design obstacle avoidance rules such that they cannot express what direction is forbidden, but instead are forced to deal in positive terms (what direction is correct, given that an obstacle is detected at a certain location), it would be very hard to now reuse these rules in a different context, which does not necessarily attempt to head towards a specific location. Yen and Pfluger (1995), Baxter and Bumby (1995) and others describe different methods of expressing and dealing with such rules. But the issue here is not a particular solution in a specific domain, but a demonstration that the need for such rules arises even in a very small-scale constrained problem.

Imperfection of Knowledge. Regardless of whether the system is additive or non-additive, and regardless of the defuzzification method used, basically all of the works describing controllers of the type mentioned above report on the problem of having a "two-peak" output set as a result of rule firings. This output set is one in which two or more different values have the maximum membership degree, but are separated by values which have a considerably lower membership degree. Saffioti et al. (1993) call this type of response "contradictory" and suggest careful design of the rules such that this situation never arises. But as we tackle bigger and more complex problems, it becomes very difficult, if not impossible, to prevent this state from happening. We claim that although from a control point of view such an output set is problematic, from an intelligent decision system point of view this output set is very interesting, and in some sense even desired. This will be treated in the next section.

Fuzzy Impasses

An examination of the field of control in general, and fuzzy controllers especially, will reveal the following assumptions underlying the techniques:

1. Controllers have "perfect knowledge". In Fuzzy Systems' terms, a fuzzy controller has a perfect set of rules: All the necessary rules, and only the necessary rules. This assumption also implies that a controller is always *positive:* It always responds with the correct output, which is the response.
2. Controllers are designed by a "perfect designer" (human or machine) that is able to predict and design such a set of rules which encode perfect knowledge. This perfect designer is implied if we assume having perfect knowledge.
3. Controllers have infinite space and computation resources. They allow for infinite number of fuzzy inference rules to be stored and matched with the inputs instantaneously, resulting in an instantaneous output.

The decision problems which we intend to solve using Soft Computing techniques, and the domains in which we intend to use these techniques clearly fail to satisfy some or all of these assumptions, due to the bounded rationality of any practical system operating in such a domain, and the dynamic changes to the domain. We will focus here only on the bounded rationality problem.

Bounded Rationality. Complex, dynamic environments that are rich in detail are (at least to some degree) unpredictable, and have infinite states in which the system may be able to find itself. It would take infinitely many rules to describe all the possible states (internal and external) that the (fuzzy) system could find itself in. Obviously we cannot assume the ability to store, manipulate, and process infinite number of rules in any practical system. We therefore must assume that: (a) our system will not have all the necessary rules, nor will it have (b) only necessary rules. We also cannot assume a perfect designer, since no human or machine designer will be able to supply us with all needed rules.

Of course, the rule-explosion problem in fuzzy systems is well known, but approaches to alleviating it in the domain of fuzzy control (e.g., hierarchical systems (Raju and Zhou 1993), optimal rules (Kosko 1997)) have so far focused on structuring the system or modifying the rules in such away that the same knowledge is represented in fewer rules. However, an altogether different method is used in AI systems. In general, a planner generates actions *on-line*. Rather than pre-computing rules, a planning system is capable of generating the right responses to an unfamiliar state after a process of reasoning. In a very real sense, these systems are intelligent, as they are able to use existing knowledge to generate new knowledge which they previously did not have. To duplicate this capability, our soft computing techniques require methods of detecting when the need for new rules arise, and for computing these rules when they are needed.

The need for new rules is really a need for new knowledge. A system capable of generating such new knowledge on demand is *intelligent* according to Allan Newell's definition in his seminal work "Unified Theories of Cognition" (Newell 1990). It is a system that can detect when its knowledge is not sufficient to solve a problem (generate a coherent response), and can use existing knowledge to generate what is required. Newell introduces the notion of "impasses" to describe situations in which a system can at the very least detect imperfections in its own knowledge, and potentially start a reasoning process which will lead to a resolution.

The two-peak output set clearly shows that the controller for the robot does not have perfect knowledge, as no clear ideal response was generated by the rules. This is one type of an impasse - it can happen when two or more correct solutions exist (which is imperfection in the sense that no *one* coherent response exists), when there exists a contradiction such that both responses are incorrect by themselves, or when one response (one peak) is essentially a correct response and the other is not. It could also mean that there exists a feature in the environment which can distinguish between the two responses, but which is un-modeled by the system (i.e., there exists no variable for that feature, so while the system is responding to essentially two different situations, it cannot tell the difference). Another type of impasse pointed to by Newell is the situation when no rules fire - no rules match the inputs. This implies imperfection in the rules in terms of coverage, as clearly the system is incapable of handing a situation that is encountered. Negative rules also raise an interesting related type of impasse--an output set that is all negative--where all possible responses are forbidden.

Impasse points are the basic building block of any intelligent system, as intelligence begins where directly applicable knowledge ends, and imperfections in the knowledge are found. Impasse states are exactly those in which the system requires decision making and reasoning capabilities to be able to provide a coherent response, and cannot rely on its rule base to provide it with a directly applicable answer.

Impasse points are also decision points. They are exactly the points in which the bounded rationality and resources of the system forces making a decision; e.g., between one of the two peaks in a "contradictory" output set, how to choose between all-forbidden responses (if a response is required), etc.

The capability to detect impasses and respond to them is thus required in any system which wishes to deal with decisions. Understanding impasses, their classification and the responses to them are critical to any designer of a decision-making or decision-support system if it is to be more than just a database of rules, which is a storage of knowledge.

References

Baxter, J. W. and Bumby, J. R. 1995, Fuzzy control of a mobile robotic vehicle. *In Proceedings of the Institution of Mechanical Engineers*, Vol 209, pp. 79-91.

Kaminka, G. A. 1997. Real-World Robot Navigation Using Fuzzy Reaction and Deliberation. In *Proceedings of the International Conference on Fuzzy Logic and Applications (Fuzzy-97)*, pp. 331-337. Zichron-Yaakov, Israel.

Kosko, B. 1997. *Fuzzy Engineering*. Prentice-Hall, Inc.

Newell, A. 1990. *Unified Theories of Cognition*. Harvard University Press.

Raju G. V. S. and Zhou J. 1993. Adaptive Hierarchical Fuzzy Controller", *IEEE Transactions on Systems, Man, and Cybernetics*, Vol. 23(4) pp. 973-980.

Saffioti A.; Ruspini E. H; and Konolige, K. 1993. Blending Reactivity and Goal Directedness in a Fuzzy Controller. *IEEE Neural Nets and Fuzzy Control*, San Francisco.

Yen, J. and Pfluger, N. 1995. A Fuzzy Logic Based Extension to Payton and Rosenblatt's Command Fusion Method for Mobile Robot Navigation. In *IEEE Transactions on Systems, Man, and Cybernetics*, Vol. 25(6), pp. 971-978.