7915

REFINING FUZZY LOGIC CONTROLLERS WITH MACHINE LEARNING

Hamid R. Berenji
Intelligent Inference Systems Corp.
AI Research Branch, MS: 269-2
NASA Ames Research Center
Mountain View, CA 94035
berenji@ptolemy.arc.nasa.gov

ABSTRACT

In this paper, we describe GARIC (Generalized Approximate Reasoning-Based Intelligent Control) architecture which learns from past performance and modifies the labels in the fuzzy rules to improve performance. It uses fuzzy reinforcement learning which is a hybrid method of fuzzy logic and reinforcement learning. This technology can simplify and automate the application of fuzzy logic control to a variety of systems. GARIC has been applied in simulation studies of the Space Shuttle rendezvous and docking experiments. It has the potential of being applied in other aerospace systems as well as in consumer products such as appliances, cameras, and cars.

INTRODUCTION

Future generation of intelligent systems are expected to demonstrate a high degree of autonomy in their operations. For example, the future controllers for the Space Shuttle in-orbit operations should consume less fuel, be capable of performing more difficult maneuvers and rendezvous missions, and eliminate jet over-firings which can result in payload contamination. Fuzzy logic control can play an important role in development of intelligent systems for space applications [4], [6], [7], [8] where the experience of human experts can be modeled and used.

Fuzzy logic controllers generally use rules containing fuzzy terms such as small, medium, and large which can be mathematically represented using membership functions. The membership values range between zero (for non-membership) to one (for full membership). For example, a temperature reading of 85 degrees may be given a membership value of .9 in a fuzzy set *hot*.

A difficulty in design of these systems relates to fine-tuning the membership functions of the labels used in the rules. A few approaches have been recently suggested which use neural networks to define and fine-tune the membership functions (e.g., [5]). However, these have been mostly off-line and supervised learning approaches. In [1], [2], [3] the idea of using reinforcement learning for developing fuzzy membership functions has been proposed and two architectures, ARIC and GARIC, have been developed. After successful applications of these architectures to cart-pole balancing and truck backing, the performances of these algorithms in the attitude control and rendezvous docking missions of the Space Shuttle are being studied [4]. In this paper, we discuss some of the lessons learned in applying the GARIC architecture to a complex system such as the simulation of in-orbit operation of the Space Shuttle.

THE GARIC ARCHITECTURE

In some sense, GARIC emulates the way that humans learn to become experts in performing a task. For example, in learning to play tennis, a novice player first learns a number of general rules for playing this game. These rules may include how to hold the racket, how to move from a place to the next depending on the location of the opponent and the direction of the ball movement, etc. In GARIC, such a process helps in the definition of fuzzy control rules. After these general rules, which are approximate by their nature, have been learned by the novice tennis player, then he or she starts to practice. It can be argued that by practicing more, the player sharpens his or her skills in order to produce higher reinforcements (i.e., to win more games). This process in GARIC refers to refining the fuzzy control rules and in particular, changing the membership functions that are used.

2

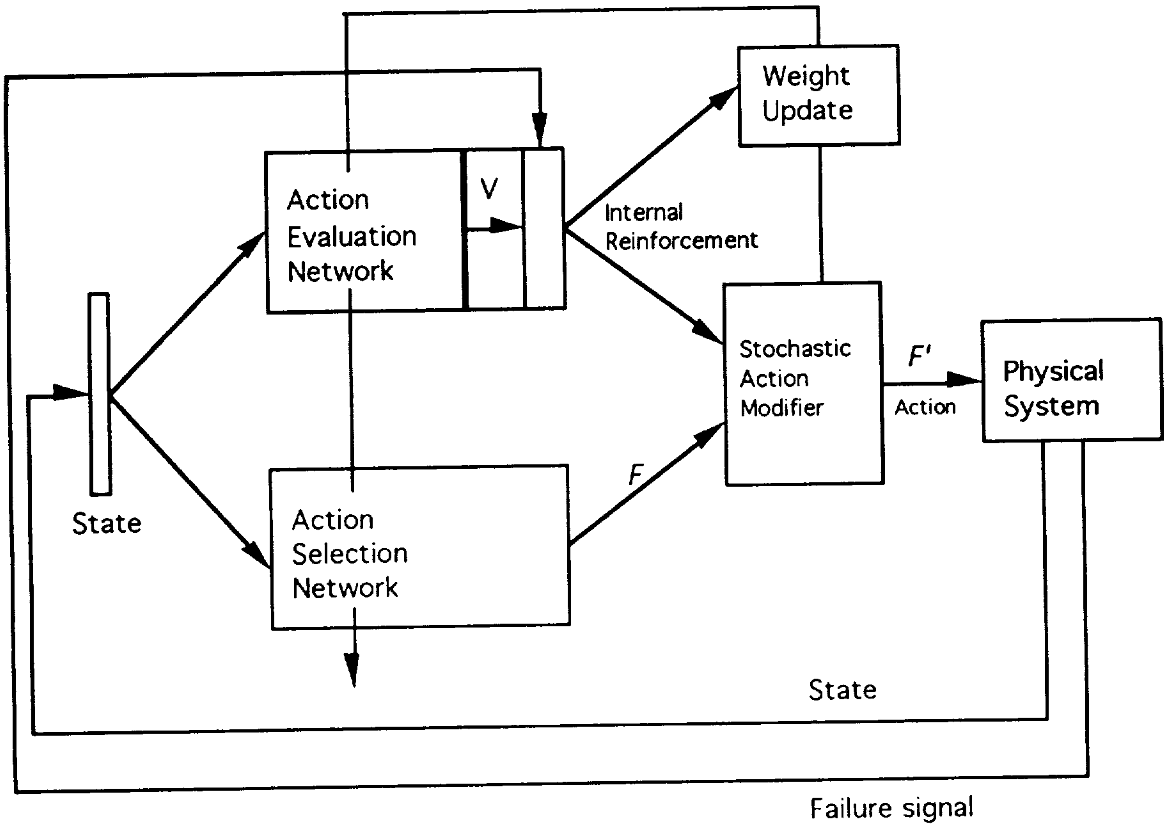

Figure 1: The GARIC Architecture

GARIC is a hybrid architecture for using fuzzy logic control and reinforcement learning. In reinforcement learning, one assumes that there is no supervisor to critically judge the chosen control action at each time step. The learning system is told indirectly about the effect of its chosen control action. GARIC uses reinforcements from the environment to refine its definition of fuzzy labels globally in all the rules and allows any type of differentiable membership function to be used in the construction of a fuzzy logic controller.

The architecture of GARIC is schematically shown in Figure 1. It has three components:

1. The Action Selection Network (ASN) which, given a situation (i.e., a state vector) and by consulting its fuzzy rules, recommends performing a control action *F*

2. The Action Evaluation Network (AEN) maps a state vector and a failure signal into a scalar score (V) which indicates state goodness. This is also used to produce internal reinforcement.

3. The Stochastic Action Modifier (SAM) uses both *F* and internal reinforcement to produce an action *F'* which is applied to the plant.

The ensuing state is fed back into the controller, along with a boolean failure signal. Learning occurs by fine-tuning of the free parameters in the two networks : in the AEN, the weights are adjusted; in the ASN, the parameters describing the fuzzy membership functions change. Further details on GARIC are described in [1].

SPACE SHUTTLE IN-ORBIT OPERATION WITH GARIC

The attitude and translational control are important parts of the Space Shuttle in-orbit operations. The attitude controller performs a variety of tasks including:

(1) attitude hold or maintaining the desired attitude within a small region of the desired value, typically known as a deadband
(2) attitude maneuver or going from one attitude to another.

Typical controllers based on the phase plane concept have angle errors and rate errors as input values. The output controller value is a command for generating a correcting torque. For the space shuttle, the rotational corrective torques are generated by thrusters by having compensating thrusters fire along a given axis to nullify the input errors. It uses two types of thrusters (two levels of jet thrusts), known as primary and vernier, and operates with two different sets of deadband values. It can perform rate maneuvers in pulse as well as discrete modes. Typical perturbations acting on the system include gravity gradient, aerodynamic torques, and translational burns.

The translational controller also performs a variety of tasks including the R-bar or V-bar approaches (in which the Orbiter moves along the target's radius vector or velocity vector, respectively, with a sequence of "hops"), station-keeping, and flyaround operations. All testing and training is performed using the Orbital Operation Simulator (OOS) which is a high fidelity shuttle simulator and includes the space shuttle Digital Auto-Pilot (DAP) for attitude operation.

A fuzzy logic controller using 31 rules for each axis (pitch, roll, yaw) have been developed [7], [4]. For each rule, seven labels (Negative-Big, Negative-Medium, Negative-Small, Zero, Positive-Small, Positive-Medium, Positive-Big) are used for angle error and angle error rate, and five labels (NM, NS, ZE, PS, PM) are used for jet firing commands. This controller holds the error between a .5 deadband. If a tighter deadband is required, then the membership functions need to be adjusted manually. However, by using the GARIC architecture, the system learns to automatically adjust its membership functions so that the error remains within the new tighter deadband. In a learning experiment, a failure occurs when the value of a state variable goes beyond the desired deadband. Over a number of trials, and by using the fuzzy reinforcement learning, the GARIC architecture learns to control the error to stay within the new deadband. Similar experiments were also performed for translational control including the R-bar approach, V-bar approach, and fly-around operations.

A set of experiments were performed to tune our fuzzy logic controller to perform a new task of keeping the error within a .4 deadband (i.e., -.4 to +.4) for pitch, roll, and yaw. Less than 10 trials were needed to refine the triangular membership functions as used in our fuzzy rules. Once GARIC has completed its training, we take the refined labels and run the controller again with no on-line learning in order to test its behavior. These experiments showed that GARIC can learn to perform a new task within a limited number of trials in a complex environment such as the simulation of the Space Shuttle in-orbit operations. Further details about these experiments can be found in [4].

Since it is relatively simple to translate a fuzzy rule base into a 5-layer neural network as is used in ASN, then it is expected that GARIC can be applied to other domains where fuzzy logic control has been used. For example, fuzzy logic control based applications in consumer products such as appliances, automobiles, and cameras can use GARIC's method in fine-tuning their performances.

CONCLUSION

GARIC provides a general approach for developing intelligent systems. It starts with the available prior knowledge of the experts in the form of fuzzy rules and refines it using the reinforcements obtained while experimenting with the system. As such, this approach generalizes fuzzy logic control and adds an adaptive behavior to it. In this paper, we briefly discussed an application of GARIC in in-orbit operations of the Space Shuttle. However, a general learning technique as developed in GARIC, may be used in many other domains that fuzzy logic control can be used.

REFERENCES

[1] Berenji, H.R. and P. Khedkar, Learning and Tuning Fuzzy Logic Controllers Through Reinforcements, vol. 3, no.5, IEEE Transactions on Neural Networks, 1992.

[2] Berenji, H. R., An Architecture for Designing Fuzzy Controllers using Neural Networks, International Journal of Approximate Reasoning, volume 6, no. 2, pp. 267-292, Feb. 1992.

[3] Berenji, H. R., On the integration of reinforcement learning and approximate reasoning for control, American Control Conference, page 1900-1904, Brighton, England, 1991.

[4] Berenji, H.R. and R.N Lea and Y. Jani and A. Malkani and J. Hoblit, A Learning Fuzzy Logic Controller for the Space Shuttle's Orbital Operations, AI Research Branch, FIA-93-30, October 1993.

[5] Jang, J.S. Self-learning fuzzy controllers based on temporal back propagation, IEEE Transactions on Neural Networks, 3(5), 1992.

[6] Lea, R. and J. Villarreal and Y. Jani and C. Copeland, Learning Characteristics of a Space-Time Neural Network as a Tether Skiprope Observer, North American Fuzzy Information Processing Society, December 1992", pp. 154-165, Puerto Vallarta, Mexico.

[7] Lea, R. and J. Hoblit and Y. Jani, Performance Comparison Of A Fuzzy Logic Based Attitude Controller With The Shuttle On-orbit Digital Auto Pilot, North American Fuzzy Information Processing Society, May 1991, Columbia, Missouri.

[8] Lea, R. and M. Togai and J. Teichrow and Y. Jani, Fuzzy Logic Approach to Combined Translational and Rotational Control of a Spacecraft in Proximity of the Space Station, International Fuzzy Set Association Conference, pp. 23-29, 1989.