Evolving Fuzzy Logic Controllers for Sony Legged Robots

Dongbing Gu and Huosheng Hu

Department of Computer Science, University of Essex
Wivenhoe Park, Colchester CO4 3SQ, UK
dgu@essex.ac.uk, hhu@essex.ac.uk

Abstract. This paper presents an evolutionary approach to learning a fuzzy logic controller(FLC) employed for reactive behaviour control of Sony legged robots. The learning scheme is divided into two stages. The first stage is a structure learning in which the rule base of FLC is generated by a backup updating learning. The second stage is a parameter learning in which the parameters of membership functions of fuzzy sets are learned by a genetic algorithm (GA). Simulation results are provided to show the effectiveness of the proposed learning scheme.

1 Introduction

The behaviour-based control is a strategy that decomposes complex robot tasks into a collection of sensory and action pairs. These pairs tightly couple perception and motor responses and work in a co-operative manner to complete complex tasks. There is an increasing tendency to build up the mapping of sensory and action pairs by using FLC to make robots be reactive and adaptive [8]. Problems in the design of a FLC are parameter setting of membership functions and composition of rules. They are classified into two catalogues: structure identification and parameter identification [7].

Karr first used a GA to evolve the membership functions of a FLC [4], in which a FLC is encoded as an individual. GA-based learning has been used for both structure learning and parameter learning of FLCs in robot reactive control[6]. In [1][3], the authors adopted different evolution strategies where an individual represents a rule, not a FLC. Rules in one generation compete with each other in order to be selected. The FLC was constructed by selecting a rule from each sub-population. The Actor-Critic learning is used for parameter learning in [2][5]. Their difficulty is that it needs both an actor network and a critic network to converge simultaneously.

This paper presents a reactive controller for Sony legged robots to play soccer. A behaviour-based architecture is proposed to model temporal reasoning as a deliberative component and to select a set of reactive behaviours as a reactive component. The main reactive behaviour is to move towards the ball guided by on-board sensors. There is no global visual system to provide relative positions among the robots, the ball and the goal. Only information provided from

A. Birk, S. Coradeschi, and S. Tadokoro (Eds.): RoboCup 2001, LNAI 2377, pp. 356–361, 2002.

the environment is the ball's local sensory information and the reward is not presented until the robot moves to the place close to the ball and faces to the goal. A two-stage learning process is proposed to evolve a FLC for the reactive behaviour. A rule base is first learned through a backup updating provided with initial values of the membership parameters. For Sony legged robots, output actions are discrete commands, each of which can make legged robots move a single step in different directions. The parameter identification for input membership functions is investigated in the parameter learning. A real value GA learning is adopted to optimize membership parameters. In section 2, we will formulate a FLC for Sony legged robots to move towards the ball. The generation of rule base and GA learning mechanisms of the parameter learning are addressed in section 3. The simulation results are given in section 4. Finally section 5 presents brief conclusions and future work.

2 A Fuzzy Logic Controller for the Reactive Behaviour

A reactive control scheme is adopted in the *approaching-the-ball* behaviour for Sony legged robots to play soccer. There are three state variables: the orientation relative to the ball represented by θ, the distance to the ball, and the goal's angle relative to the robot's head direction represented by α, which are important for this behaviour due to lack of the global localization. A tracking algorithm has been implemented for the head to track the ball based on color detection. It leads θ to be expressed by the head's pan angle that can be easily read from the encoder mounted on the pan motor. We use the number of image pixels(h) of the ball in a CCD camera mounted inside the head to estimate the distance. The tracking algorithm will also yield the goal's angle in the process of tracking the ball. Therefore, the input state vector is $S = [s_1, s_2, s_3]^T = [\theta, h, \alpha]^T$. The behaviour is to move the robot to approach the ball by taking action such as $MoveForward$, $LeftForward$, $RightForward$, $LeftTurn$, $RightTurn$, *Backward*, or *Stop* provided by low-level walking software.

Both sensory information and actions are imprecise, incomplete, and vagueness. For example, the ball size measured in pixels from the CCD camera does not accurately correspond to the distance from the robot to the ball, but just has fuzzy concept such as *Small*(S), $Middle(M)$, or *Large*(L). The pan angle measured in degrees from the head motor encoder is accurate to some extent, but it is not the precise direction from the robot to the ball since the ball may not stay in the image center during moving. A set of fuzzy concepts $Left(L)$, *Zero*(Z), or *Right*(R) is designed for the pan angle. The goal's angle is also define as $Left(L)$, *Zero*(Z), or *Right*(R). The output of this behaviour is one of the seven one-step moving commands that can not be precisely represented in position and orientation due to non-perfect actuators or slippage on the ground.

Three fuzzy sets are designed for each of three input state variables s_n. A quadruple of (a, b, c, d) is used to represent the triangle or trapezoid membership function of a fuzzy set shown as in figure 1. The output action o is the crisp value that can be seen as fuzzy singletons $c^m (m = 1, \dots, 7)$ in a FLC. There are N

($N = 3 \times 3 \times 3$) rules in total. The membership degree of a fuzzy set F_n^i of the nth input state variable s_n in the ith rule is expressed as $\mu_{F_n^i}(s_n)$. The true value of the ith rule activated by an input state vector S is calculated by Mamdani's minimum fuzzy implication:

$$\alpha_i(S) = \min(\mu_{F_1^i}(s_1), \mu_{F_2^i}(s_2), \mu_{F_3^i}(s_3)) \quad (1)$$

The crisp output o stimulated by the input state S after the fuzzy reasoning is calculated by the center of area method (COA):

$$o(S) = \sum_{i=1}^{N} \alpha_i(S) \times c^{m_i} / \sum_{i=1}^{N} \alpha_i(S) \quad (2)$$

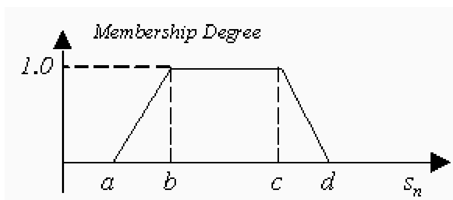

Fig. 1. Fuzzy set membership function

3 Learning FLC

The aim of the *approaching-the-ball* behaviour is to move towards the ball with an appropriate orientation to the goal. The fitness should be designed to guide the evolution of the FLCs to achieve this purpose. The fitness function is defined as:

$$fitness = w_1 \times \sum d_i/l + w_2 \times (A - l) + w_3 \times (B - \theta) + w_4 \times (C - \alpha) + w_5 \times h \quad (3)$$

where w_i ($i = 1, ..., 5$) is the weight, A, B and C are constants, and l is the number of the walking steps used in one trial. The first term in (3) indicates payoffs received during the whole movement. The second term rewards those FLCs that have fewer steps. The last three terms with last step θ, h, and α reflect the final position of the robot.

In the first stage, the rule base is learned incrementally through the backup updating. The fuzzy sets divide the input state space into difference regions and these regions overlap with each other. The points in these regions will be chosen as the learning examples where the corresponding membership degrees are equal to 1. The backing updating means the learning will start from the points where the membership degree of the fuzzy set $h = Large$ is equal to 1 to the points where the membership degree of the fuzzy set $h = Small$ is equal to 1. The robot will select as its fuzzy rule's outputs those commands with the highest fitness. The current learning examples can make use of the previous learned rules. The robot will become increasingly aware of better control rules and capable of interact with the environment via these rules during the learning process. The areas where the fuzzy sets overlap are not covered in the learning process. But they are implicitly generalized by the rules in the areas where there is no overlap among fuzzy sets. The fine-tuning within the overlapped areas will be proceeded in the next stage through the parameter learning.

In the second stage, GA is used to optimize the parameters in membership functions in order to improve the reactive behavioural performance. One FLC is encoded as one individual. In each generation, a collection of the individuals is maintained to compete with each other to survive. By evolving through genetic operators, the best one will be selected as the optimal FLC for the behaviour control. For a FLC, the quadruple (a, b, c, d) of its input membership functions are real values. The real value encoding of a FLC is shown in figure 2 where the individual p represents the pth FLC in one generation.

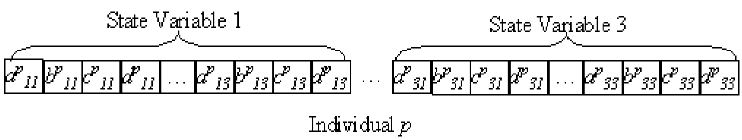

Fig. 2. An individual in a generation

There are three genetic operators employed: reproduction, crossover, and mutation. The real value mutation is proceeded by adding a certain amount of noise to new individuals to produce the offspring with a mutation probability. The learning procedure starts from random formation of initial generation of FLCs. After completing trials of one generation, the FLCs evolve into a new generation and the learning repeats until a GA terminal condition is met.

4 Experimental Results

The experiments are tested in a simulation environment where the models with the exact size of the pitch and ball are built up. The rule base is incrementally built up by the backup updating approach described in section 3. Figure 3 shows the results for the *approaching-the-ball* behaviour controlled by the FLC with the learned rule base. It can be seen that the robot can move to the ball although the robot did not exactly face the goal at final positions.

After the parameter learning by GA, the same situations as in figure 3 are tested for the robot with the evolved FLC. The results are shown in figure 5 where the improvement at final positions can be seen. In the parameter learning, the size of population in one generation is 50. The GA learning process is shown in figure 4 after evolving 300 generations. The upper curve is the maximal fitness values in each generation. The low curve is the average fitness values in each generation. It is shown that the average fitness values converge to the maximum as the generation increases.

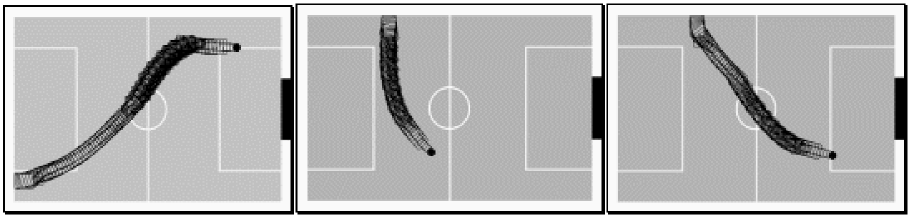

Fig. 3. Simulation results after the structure learning

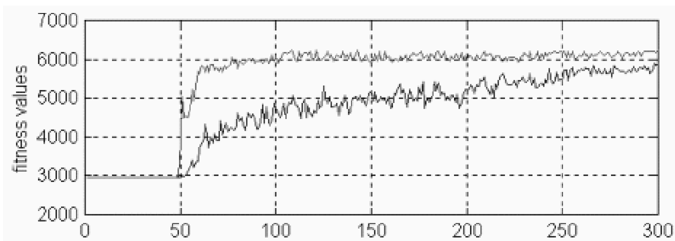

Fig. 4. Evolution process of FLCs

The quadruple (a, b, c, d) for each input fuzzy label should be constrained by their geometric shapes ($a \leq b \leq c \leq d$) during GA learning. A validation process is employed to check the constraints for each FLC before it is used. Invalid FLCs are given up. Therefore, most of FLCs are not executed due to their invalidation at early stage and the fitness values are unchanged before the $50th$ generation. Figure 6 shows the comparison made before and after the parameter learning. In the structure learning, only the areas where the input fuzzy membership degrees are equal to 1 are learned. The overlapped areas are implicitly generalized by the adjacent areas. The result shown on the left of figure 6 is the case that can not be generalized, and the result on the right shows the compensation made by the parameter learning.

5 Conclusions and Future Work

The learning of FLC for Sony legged robots is addressed through both structure and parameter learning stages in this paper. The strategy is to make use of the heuristic experience and autonomous exploration of robot's environment to yield a good reactive controller.

The structure learning is similar to the human reasoning about rule design. The most of the state space where no overlap exists in the fuzzy sets is covered by the actual crisp conditions. It implies the overlap areas can be generalized by adjacent areas. The further adjustment of the FLC is left for the parameter learning in which the parameters in input fuzzy sets are modified. A real value GA is used to evolve the FLCs. It explores the state space by using crossover and mutation operators, while the offspring with high fitness value is exploited

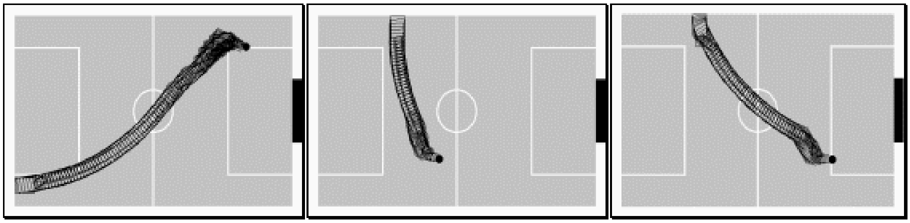

Fig. 5. Simulation results after the parameter learning

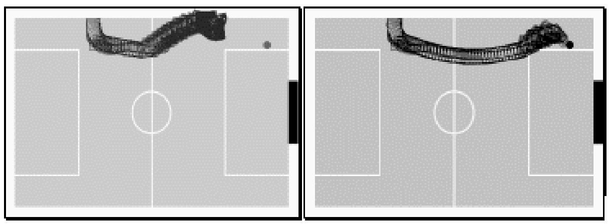

Fig. 6. The comparison between the structure learning and parameter learning

by reproduction operators. The experiment results show that the FLC can be learned by the proposed learning scheme. The proposed method will be tested on real robots in the next stage.

References

1. Bonarini, A., Evolutionary Learning of Fuzzy rules: competition and cooperation, In Pedrycz, W. (Ed.), Fuzzy Modelling: Paradigms and Practice, Kluwer Academic Press, Norwell, MA, 1996, pages 265-284.
2. Jouffe, L., Fuzzy Inference System Learning by Reinforcement Methods, IEEE Transactions On SMC-Part B, Vol. 28, No. 3, August 1998, pages 338-355.
3. Juang, C. F., Lin, J.Y., and Lin, C. T., Genetic Reinforcement learning through Symbiotic Evolution for Fuzzy Controller Design, IEEE Transactions on SMC-Part B, Vol. 30, No. 2, April 2000, pages 290-301.
4. Karr, C. L., Gentry, E. J., Fuzzy Control of pH Using Genetic Algorithms, IEEE Transactions on Fuzzy Systems, Vol. 3, Jan. 1993, pages 129-139.
5. Lin, C. T., Jou, C. P., GA-Based Fuzzy Reinforcement Learning for Control of a Magnetic Bearing System, IEEE Transactions on SMC-Part B, Vol. 30, No. 2, April 2000, pages 276-289.
6. Matellan, V., Fernandez, C., Molina, J. M., Genetic Learning of Fuzzy Reactive Controllers, Robotics and Autonomous Systems, Vol. 25, 1998, pages 33-41.
7. Mitra, S., Hayashi, Y., Neuro-Fuzzy Rule Generation: Survey in Soft Computing Framework, IEEE Transactions on Neural Networks, Vol. 11, No. 3, May 2000, pages 748-768.
8. Saffiotti, A., Ruspini, E. H., and Konolige, K., Using Fuzzy Logic for Mobile Robot Control, in Zimmermann, H. J., editor, Practical Applications of Fuzzy Technologies, Kluwer Academic Publisher, 1999, pages 185-206.