hana

— Development of an Agent with Fuzzy Reinforcement Learning —

Tomoharu Nakashima, Masayo Udo, and Hisao Ishibuchi

Department of Industrial Engineering
Osaka Prefecture University
Gakuen-cho 1-1, Sakai, Osaka 599-8531, JAPAN
{nakashi, udo, hisaoi}@ie.osakafu-u.ac.jp

1 Introduction

Designing an agent that has a learning ability is one of the most important topics in the field of artificial intelligence. For example, with a number of learning agents, complex tasks that cannot be solved by a single agent could be solved by some collective behavior. Developing a learning scheme for agents can lead to further research on the autonomous distributed multi-agent area.

In this paper, we provide a detailed description on our RoboCup simulation soccer team called *hana*. A fuzzy reinforcement learning scheme is used for acquiring optimal behavior of playing soccer. Usually, discretizing procedures of an input space are needed before implementing a reinforcement learning in order to define states of the input space. It is difficult to discretize the input space when continuous values are used for dimensions of the input space. In this paper, we show how to tackle with this issue by incorporating fuzzy systems into the reinforcement learning. That is, we discretize the input space by discretizing the input space into fuzzy states for our fuzzy reinforcement learning.

Our fuzzy reinforcement learning is used for selecting optimal actions for each fuzzy state. As a reinforcement learning scheme, we employ fuzzy Q-learning. In the fuzzy Q-learning, fuzzy if-then rules are used for suggesting a weight value for taking each action from several candidate actions depending on the position of the ball, the nearest opponent, the nearest teammate, and the player itself.

First, we introduce the concept of fuzzy Q-learning. Then, experimental setting of our succer agents are shown. Finally we show conclusions and provide future research left for our research on designing multi-agent systems.

2 Fuzzy Q-Learning

Fuzzy Q-learning is an extended version of the original Q-learning [1, 2]. The main advantage of the fuzzy Q-learning over the non-fuzzy Q-learning is that a continuous input space can be easily discretized into fuzzy states. Better generalization ability is another advantage of the fuzzy Q-learning.

2.1 Fuzzy If-Then Rule

Let us assume that our problem is to select the optimal action from C candidate actions under the condition that is described by n variables. This problem can be viewed as an n-dimensional C-class pattern classification problem. In order to deal with such problems, we use fuzzy if-then rules of the following type for our fuzzy system:

$$\text{Rule } R_j\text{: If } x_1 \text{ is } A_{j1} \text{ and } \ldots \text{ and } x_n \text{ then } \mathbf{q}_j = (q_{j1}, \ldots, q_{jC}), \quad j = 1, 2, \ldots, N, \tag{1}$$

where R_j is the rule label of the j-th fuzzy if-then rule, $A_{j1}, \ldots, A_{jn}$ are antecedent fuzzy sets, $\mathbf{q}_j = (q_{j1}, \ldots, q_{jC})$ is a weight vector, and N is the number of fuzzy if-then rules in our fuzzy system.

2.2 Calculating Q-Values

An agent (i.e., a soccer player) can receive some visual information over time. From the visual information, we can calculate detailed situation such as the absolute position of the players and ball and which player is handling the ball at that time. Suppose that a player has obtained such detailed information $\mathbf{x} = (x_1, \ldots, x_n)$ from his visual information. This information $\mathbf{x}$ is used as input for our fuzzy system. Then the agent with our fuzzy Q-learning scheme calculates a Q-value for taking each candidate action as follows:

$$Q_c = \frac{\sum_{j=1}^{N} q_{jc} \cdot \mu_j(\mathbf{x})}{\sum_{j=1}^{N} \mu_j(\mathbf{x})}, \qquad c = 1, \ldots, C, \tag{2}$$

where $\mu_j(\mathbf{x})$ is the compatibility grade of the input vector with the j-th fuzzy if-then rule R_j and is calculated by the multiplication operator as follows:

$$\mu_j(\mathbf{x}) = \mu_{j1}(x_1) \cdot \mu_{j2}(x_2) \ldots \cdot \mu_{jn}(x_n), \tag{3}$$

where $\mu_{ji}(x_i)$, $i = 1, \ldots, n$ is a membership function of the fuzzy set A_{ji}.

That is, from n-dimensional input vector $\mathbf{x}$, C-dimensional output vector $\mathbf{Q} = (Q_1, \ldots, Q_C)$ is generated by our fuzzy system. Each element Q_c of the output vector $\mathbf{Q}$ can be regarded as the degree of support for taking the c-th action from C candidate actions. A single action is selected according to the output vector $\mathbf{Q}$ by using the following proportional selection scheme;

$$\Pr_c = Q_c \Big/ \sum_{c'=1}^{C} Q_{c'}, \tag{4}$$

where $\Pr_c$ is the probability of selecting the c-th action.

2.3 Updating Fuzzy If-Then Rules

In the previous subsection, a single action is selected from C candidate actions according to the output vector $\mathbf{Q}$ from our fuzzzy system. In this subsection, we show how to reinforce fuzzy if-then rules after the action selection.

If the action selection is successful (for example, suppose that an agent select to pass the ball and it resulted in success), then we reward the fuzzy if-then rules by increasing the weight value corresponding to the selected action as follows:

$$q_{jc}^{\text{new}} = (1 - \alpha') \cdot q_{jc}^{\text{old}} + \alpha' \cdot (\gamma \cdot r + \delta \cdot V), \qquad j = 1, \ldots, N, \tag{5}$$

where r is the reward, V is a value of the current fuzzy state, γ, δ are constant values, and α' is a fuzzy learning rate which is determined by the conventional learning rate α and the grade of compatibility $\mu_j(\mathbf{x})$:

$$\alpha' = \alpha \cdot \frac{\mu_j(\mathbf{x})}{\sum_{k=1}^{N} \mu_k(\mathbf{x})}. \tag{6}$$

3 Implementation

In this section, we show the implementation of our fuzzy Q-learning scheme in soccer agents. We use the following eight variables as elements of input vectors for our fuzzy system: (x_1, x_2) for the absolute position of the player, (x_3, x_4) for the absolute position of the ball, (x_5, x_6) for the absolute position of the nearest teammate, and (x_7, x_8) for the absolute position of the nearest opponent.

We partition the soccer field into fuzzy partition as in Fig. 1.

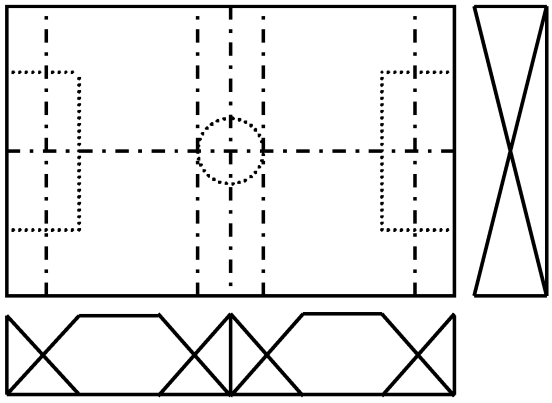

Fig. 1. Fuzzy partition of the soccer field.

We chose the following six actions as the candidate actions for our fuzzy Q-learning scheme: *kick to a teammate*, *kick to a point*, *dribble*, *move to a point*, *move to the ball*, and *do nothing*.

The base program source is YowAI [3]. We modified the program source of YowAI in order to incorporate the learning ability based on our fuzzy Q-learning. Thus the modified program has the property of both our method and YowAI's behavior. We used the parameter setting as follows: $\alpha = 0.9$, $\delta = 0.5$, and $r = 1.0$.

Before taking a log of a game, we performed several soccer games in order to fine-tune the weights of fuzzy if-then rules. Then, we recorded the final weights of the fuzzy if-then rules and used them for the next soccer game.

4 Conclusions and Future Works

In this paper, we show our learning scheme. We employ fuzzy Q-learning method where a weight value is determined by using a fuzzy inference for each candidate action. By the fuzzy Q-learning scheme, agents can obtain optimal action for various situation thus agents can make appropriate decision making.

However, there are many open problems with our method. First, we use 20736 fuzzy if-then rules for calculating a Q-value. Obviously, one of the most well-known problem in the reinforcement learning scheme called *the curse of dimensionality* occurs. It takes a lot of time for calculating the Q-value and agents cannot move during the calculation. One possible way of avoiding the curse of dimensionality is that from as many as 20736 fuzzy if-then rules we select only a small number of fuzzy if-then rules. Genetic algorithms are promising methods for doing this task [4]

Our action selection was done from roughly defined candidate selection. For example, in the case that an action *move to a point* is selected, it is not clearly specified exactly what the point is. In the implementation, we specified likely best point by hand. Future work also includes an automatic determination of such further specification of actions. This can be done by using another level of the fuzzy system.

References

1. H. Ishibuchi, C.-H. Oh and T. Nakashima, "Improving the Performance of Q-Learning by Fuzzy Logic," *Proc. of the 2nd International Symposium on Artificial Life and Robotics*, , pp.50-53, 1997.
2. C.-H. Oh, T. Nakashima, H. Ishibuchi, "Initialization of Q-values by Fuzzy Rules for Accelerating Q-Learning", *Proc. of International Joint Conference on Neural Networks*, pp.2051-2056, 1998.
3. T. Suzuki, "YowAI", *RoboCup-99: Robot Soccer World Cup III*, pp.646–648, 2000.
4. H. Ishibuchi, T. Nakashima, and T. Murata, "Peformance of Fuzzy Classifier Systems for Multi-Dimensional Pattern Classification Problems", *IEEE Trans. on System, Man, and Cybernetics*, Vol. 29, No. 5, pp.601-618, October 1999.