Axiom 2012 Team Description Paper

Mohammad Ghazanfari, S. Omid Shirkhorshidi, Alireza Beydaghi, Farbod Samsamipour, Hossein Rahmatizade, Mohammad Mahdavi, Mostafa Zamanipour, Payam Mohajeri, S. Mohammad H. Mirhashemi

Robotics Scientific Association and Multi-Agent System Lab.
Department of Computer Engineering
Iran University of Science and Technology
Narmak, Tehran, Iran, 1684613114

Abstract. Axiom is a 2D soccer simulation team, which is continuant of AxiomOfChoice team. In our efforts we adopted A.I. techniques in order to enhance agents' performance, and especially our orientation is to develop and utilize new techniques in Machine Learning, and particularly in the scope of Abstraction in Reinforcement Learning to build a team of agents with a full A.I. based control.

1 Introduction

Axiom is a team consisting of undergraduate and graduate students of Iran University of Science and Technology (IUST). Axiom established in January 2011 with the name of AxiomOfChoice. Now Axiom is a member of IUST Robotics Scientific Association and has a close cooperation with IUST Multi Agent Systems Laboratory. Our successes include fifth place at AUTCup 2011 and the third place of IranOpen 2011.

Our team is based on Agent2D base developed by H. Akiyama [1].

This paper is organized as follows: in section 2 describes our Genetic Algorithm based Shoot, section 3 we describe our Neural Network based Pass, section 4 presents our through pass skill, section 5 expresses our recently steps to using Reinforcement Learning Abstraction and finally in section 6 we summarize and conclude our work.

2 Genetic Algorithm Based Shoot Evaluator

We had developed a good shoot skill and therefore expected to have good shoots and more goal scores as result. However, after many examinations we got believed in the fact that good shoots more than how to shoot, depends on evaluation of shoot situation. Hence, we decided to make a shoot evaluator that can tell the success probability given the situation parameters.

We considered the problem of shoot evaluator as an optimization problem described below. When the problem is optimization, one of the great choices is Evolutionary algorithms and in particular, we chose Genetic Algorithm (G.A.) [3].

Our shoot evaluator is a simple function (F) of some parameters of situation the shooter is in. This function is in form of a fraction and each parameter is either in nominator or denominator, in addition, each parameter has a constant coefficient. The aim of genetic algorithm is to optimize these coefficients to achieve a suitable shoot evaluator.

Using human expertise as well as some examinations led to following parameters:

- a: the greater angle between angles created by each two shooter-goalpost lines, with shooter-goalie line
- d: distance that the ball should traverse to goal
- O_1: distance between shooter and nearest defender
- O_2: distance between bisector of angle "a" and nearest defender

Fig. 1 illustrates these parameters.

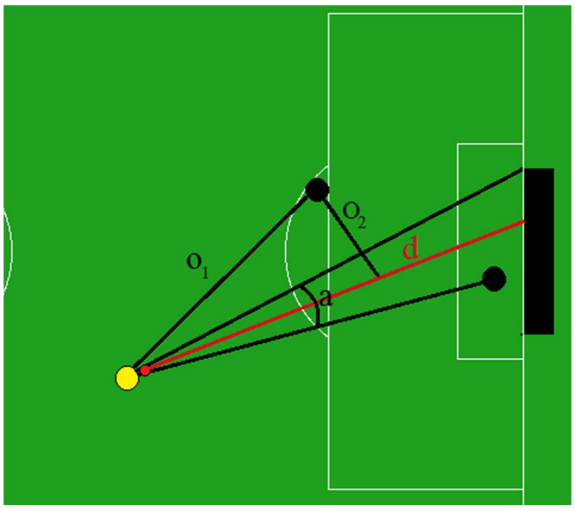

Fig. 1. Illustration of G.A. based shoot evaluator parameters

It is obvious that the following relations hold:

$$F \propto a$$

$$F \propto O_2$$

$$F \propto \frac{1}{d}$$

$$F \propto \frac{1}{O_1}$$

Therefore, the Function becomes:

$$F = \frac{w_1 a + w_2 O_2}{w_3 d + w_4 O_1}$$

Finding Coefficients is the aim of our optimization problem, so we defined each gene as one of coefficients and therefore chromosome became as follows:

w_1	w_2	w_3	w_4

Fig. 2. A Chromosome in our G.A. Algorithm

Coefficients w1, w2, w3 and w4 are integer numbers in range 1 to 10.

The shoot evaluator task is to ensure about the result of shoot, so we used two threshold values, "goal threshold" and "fail threshold". Each shoot situation with evaluated value higher than "goal threshold" will be considered as an imminent goal situation, and with lower value than "fail threshold" will be considered as imminent fail situation. Using this method with the G.A.'s outcome chromosome resulted in 90% accuracy in goal prediction and 100% accuracy in fail prediction. It is notable that for situations between two thresholds our agent uses its previous process of decision.

It may seem that our formula for shoot evaluator is too trivial, but we intentionally choose it simple to show the power of our method and our results verifies this. Also for obtaining a good chromosome, we used standard G.A. algorithm and operators, Again Our intent from using standard G.A. algorithm and not going to deeply in choosing and fine-tuning a particular G.A. algorithm is to show the suitability of our approach even without a very specialized algorithm.

3 Neural Network Based Pass Evaluator

On the issue of pass, the important point for a good pass to a teammate is that an opponent player not intercepts the ball. Predicting the result of a pass, goal teammate get the ball or no, in each situation of passing depends on many characteristics of passing moment, such as conditions of goal teammate and opponents nearby. Evaluating the pass and deciding to pass or not is a hard work and cannot be implemented with simple hardwiring decisions as if statements. Thus we decided to use a Neural Network (NN) [2] and make over this work to that.

In order to build a NN, first we should provide a training set. The good training set is that covers all situations that may occur during the real execution. For gathering the training examples, we considered the following situation: a player wants to pass to its teammate in a random distance; three opponents are randomly distributed in front of the player. All players (teammate and opponents) are equipped with intercept skill. **Fig. 3** illustrates this situation.

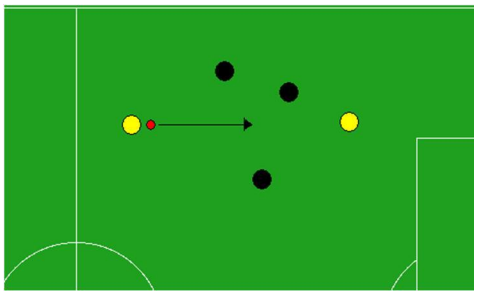

Fig. 3. Situation for training example extraction of NN

We have chosen a three Layer Multi-Layer Perceptron (MLP) as the learner for shoot evaluator. We chose MLP because of its ability to generalization by which we can expect it to have a good response for inputs that was not in its training set using its interpolation ability.

After many examinations for adjusting the network parameters we arrived to a three layer perceptron with ten, six and one perceptron in layers respectively from input to output layer. Transfer function "tansig" for input layer, "logsig" for hidden layer and "linear" for output layer is chosen. The network with these parameters achieved the accuracy of 82% for the test set. This accuracy is satisfactory for us to have a good pass evaluator.

4 Through Pass

Recently we have filled one of our team's old defects, through pass. We have implemented through pass this way: when a player perceives himself in a good position to receive a through pass, says a through pass request to his teammate whom has the ball. On the other side, the ball owner may have requests for through pass from several of his teammates, thus he will evaluate all these requests and choose the best among reliable choices. If there is not a reliable requester, he will give up and follow his other common role. Conversely, if he can find a reliable through pass receiver, he says a message to him and reports the point to which he intends to pass. However, this is not the end of through pass, now it is the receiver's turn to run and arrive to the ball. We have implemented this skill in our players.

5 Adopting an Strategy based on Reinforcement Learning

In our previous efforts, we always had a difficulty with developing a proper strategy in such complex environment, which not only considers its locality, but also observes the environment in a coarse granularity and thus make a good total decision. The most suitable solution among different techniques was Reinforcement Learning, which introduces best techniques in such stochastic and dynamic environments.

At first, we had to convert our previous skills to adapt them to the domain of reinforcement learning. Most of our skills are multi-cycle, (i.e. they do not finish their task in a single cycle).therefore we used temporally extended actions [4], a subdomain of abstraction topic in reinforcement learning.

We used option framework [5] for formulating our skills to temporally extended actions. Formally an option is a triple $<I,\pi,\beta>$ where I is the states the option can be initiated in, π is the policy the option selects its actions by, and β is the function which returns the probability of terminating option in each state. For our skills I and β determines dynamically according to agent's immediate situation, and policy is embedded in skill itself implicitly, by selecting action for each situation. Hence, we have the ability of converting our skills such as intercept, dribble, block and pass, etc. into options. In each cycle, the agent will select appropriate option according to its policy. Now reinforcement learning algorithm fulfills the goal of finding desired strategy by learning the optimal policy.

By using reinforcement learning and option framework some issues such as "Curse of Dimensionality" occur and become crucial in simulation 2D because of the very big state-space of it the following describes our state-space formulating: consider environment as a factored-MDP and use some parameters in the agent's point of view, as state variables. We considered parameters such as:

- Self-position
- Other players position and their velocity vectors
- Ball position and its velocity vector

We use function approximation for handling this factored MDP (Markov Decision Process). By using options in MDP, even with MDP options, environment becomes SMDP (Semi-Markov Decision Process) [4]. So one of algorithms for SMDPs must be selected for solving the problem and find optimal policy). Because of our limitation in process time in each cycle and using function approximation in such a big space, new algorithms such as gradient-descent method [6] is a good choice.

As is obvious from our description, we have formulated a multi-agent environment into a non-stationary environment from a single agent's point of view. This assumption may causes learning algorithm Diverge. Although it can be troublesome, we must handle these divergences manually, or by adding some conventions to our problem.

Using reinforcement learning, particularly option framework, for learning strategies has some remarkable advantages including:

- Reinforcement learning methods learn the optimal policy themselves and free the programmer from struggling with difficulties of manually deciding in very diverse situations
- The ability to construct new options that are more complex by combining simple options, leads to have skills more similar to real human skills.
- This method could be used to learn good strategy online to play against unknown opponents.

Axiom 2D simulation team, 2012.
Iran University of Science and Technology.

The Approach Described in this section results in having efficient individual players with cooperative skills, thus the emergent overall behavior of team will be rational in most situations.

6 Summary and Conclusion

2D soccer simulation is one of the most appropriate domains for experimenting A.I. techniques because of its complexity and resemblance to real world. As described before, most of our recent efforts are centered on using A.I. techniques in this domain, and beyond that, we aim to enter into the state of the art and challenging areas of A.I. science such as abstraction in reinforcement learning, and use the 2D soccer simulation as our test field.

7 References

1. Akiyama, H., http://rctools.sourceforge.jp
2. Haykin S., "Neural Networks: A Comprehensive Foundation (2 ed.)", Prentice Hall, 1998.
3. Srinivas M., Patnik L.M., "Genetic Algorithms: A Survey", Journal of Computer, IEEE, 1994.
4. Barto A. G., Mahadevan S., "Recent advances in hierarchical reinforcement learning", Journal of Discrete Event Dynamic Systems, Springer, 2003.
5. Sutton, R.S., Precup, D., "Between MDPs and semi-MDPs: A framework for temporal abstraction in reinforcement learning", Journal of Artificial intelligence, Elsevier, 1999.
6. Sutton, R.S., Maei, H.R., Precup, D., Bhatnagar, S., Silver, D., Szepesvári, C., Wiewiora, E., "Fast gradient-descent methods for temporal-difference learning with linear function approximation", Proceedings of the 26th Annual International Conference on Machine Learning, ACM, 2009.
7. Ghazanfari M., Shirkhorshidi S. O., Beydaghi A., Samsamipour F., Rahmatizade H., Mahdavi M., Zamanipour M., Mohajeri P., Mirhashemi S. M. H., "Axiom 2D Team Description Paper", IranOpen 2012, Tehran, Iran, 2012.

Axiom 2D simulation team, 2012.
Iran University of Science and Technology.