Fifty-Storms: Team Description 2009

Harukazu Igarashi[1], Jun Masaki[2], Tatsuhiko Suzuki, Naoki Tonegawa, Naoto Sano,
Toshiyuki Imaizumi, Taisuke Ozawa, Kota Suzuki, Kenji Takeda,
Hiroki Shimano, and Hiroshi Fukuoka

Shibaura Institute of Technology, 3-7-5 Toyosu, Koto-ku, Tokyo 135-8548, Japan
{[1]arashi50, [2]*l*06103}@shibaura-it.ac.jp

Abstract. Fifty-Storms is a new team that is participating for the first time in the RoboCup 2D Soccer Simulation League 2009. This team is based on open-version codes of the HELIOS team, agent2d (ver.1.0.0), to which some modifications were added to enhance the offensive abilities of the forwards and defenders by hand coding. In addition, it exploits the results of reinforcement learning, where a policy gradient method derives appropriate policies for midfielder's pass selection and forward's positioning to receive a through pass.

1 Introduction

Fifty-Storms was established in 2008 for RoboCup Soccer Simulation 2D League by members of a student project at the Shibaura Institute of Technology. The members of the student project are changed every year. From 2005, teams comprised of past members had participated in domestic competitions, RoboCup Japan Open, but they failed to succeed or did not participate in any international RoboCup competition. However, the new members changed their underlying base team from UvA Trilearn 2003[1] to HELIOS[2]. After adding some new tactical skills and making modifications to optimize HELIOS's abilities, they named their team Fifty-Storms fortheir adviser, Prof. H. Igarashi, whose name means "fifty storms" in Japanese. Fifty-Storms participated in RoboCup Japan Open 2008 and narrowly lost the championship match to HELIOS.

Fifty-Storms 2009 exploits the research results of H. Fukuoka, N. Sano, and H. Igarashi who applied reinforcement learning to pass selection problems of midfielders (MFs) and positioning problems for forwards (FWs) to receive a through pass. In their work, they proposed an objective function, which is a linear combination of heuristic functions evaluating an agent's action, and determined the values of weight coefficients in the objective function by a kind of reinforcement learning called the policy gradient method. In this team description paper, we briefly describe the modifications added to HELIOS and the learning method.

2 Modifications Added to Base Team

Fifty-Storms uses HELIOS whose source codes were released by Hidehisa Akiyama and put on a URL site in [2] as agent2d (ver.1.0.0). He also released a large number of useful functions as a software library called librcsc that are necessary for making soccer agent programs. Moreover, the source codes of HELIOS and detailed descriptions of the agent's skills and the team's tactics and strategies were combined and published in 2006 (in Japanese). After this book's publication, many Japanese teams changed their base team from UvA to HELIOS as Fifty-Storms did.

In Fifty-Storms, enhancing offensive power is the main design principle to modify HELIOS. For that purpose, the following two ideas were designed and implemented. The first maximized the dribbling skill of agents in HELIOS so that a FW or a MF runs through the defense line of the opponent's defense players (DFs) and shoots as near the opponent's goal as possible. The original HELIOS had offending strategy characteristics in which a FW in a corner area in the opponent's field passes to teammates in front of the opponent's goal. That is called a side attack or an open offense. Fifty-Storms added another useful option in HELIOS's offense strategy.

The second idea takes an aggressive defense strategy by making DFs go to opponents with the ball so that DFs can take the ball from them. In addition, FWs do not return to their own field and instead stay near the opponent DFs, even if the opponent goalie catches the ball and kicks it. FWs add pressure on the opponent's DFs and try to take the ball from them.

3 Behavior Learning of Soccer Agents

3.1 Policy Gradient Approach to Soccer Agents

The RoboCup Simulation 2D League is recognized as a test bed to learn coordination in multi-agent systems because there is no need to control real robots and one can focus on learning coordinative behaviors among players. However, multi-agent learning continues to suffer from several difficult problems such as state-space explosion, concurrent learning, incomplete perception, and credit assignment. In the Robocup Simulation League games, the state-space explosion problem is the most difficult [3]. Researchers must concentrate on these four difficult problems to exploit AI and machine learning techniques in soccer agents.

As an example of multi-agent learning in a soccer match, Igarashi et al. proposed and applied a policy gradient approach to realize coordination between a kicker and a receiver in direct free kicks [4]. They dealt with a learning problem between a kicker and a receiver when a direct free kick is awarded just outside the opponent's penalty area. In such a situation, to which point should the kicker kick the ball? They proposed a function that expressed heuristics to evaluate a candidate target point for effectively sending/receiving a pass and scoring. However, they only dealt with the attacking problems of 2v2 (two attackers and two defenders), and their base team used in [4] was UvA Trilearn 2003. They applied the policy gradient approach to pass selection problems of midfielders and positioning problems for forwards to receive a through pass and implemented the learning results into Fifty-Storms.

3.2 Characteristics of Policy Gradient Method

A policy gradient method is a kind of reinforcement learning scheme that originated from Williams's REINFORCE algorithm [5]. The method locally increases and maximizes the expected reward per episode by calculating the derivatives of the expected reward function of the parameters included in a stochastic policy function. This method, which has a firm mathematical basis, is easily applied to many learning problems. One can use it for learning problems even in non-Markov Decision Processes [6][7]. It has been applied to pursuit problems where the policy function consists of state-action rules with weight coefficients that are parameters to be learned [7].

The policy gradient method used in refs. [4] and [7] has the following technical characteristics. A Boltzmann distribution function is adopted as a stochastic policy for action decision. For the autonomous action decisions and the learning of each agent, the policy function for the entire multi-agent system was approximated by the product of each agent's policy function [8]. The Boltzmann function has an objective function defined by

$$E_\lambda\left(a_\lambda;s_\lambda,\left\{\omega_j^\lambda\right\}\right)=-\sum_j \omega_j^\lambda\cdot U_j\left(a_\lambda;s_\lambda\right), \qquad \textbf{(1)}$$

where a_λ is the action of agent λ and s_λ is the state perceived by agent λ. Function U_j $(a_\lambda;s_\lambda)$ is the j-th heuristics that evaluates action a_λ. At the end of each learning episode, common reward r is given to all agents. The derivative of expectation of reward $E[r]$ for parameter $\omega^\lambda{}_j$ can be calculated to derive the following learning rule on $\omega^\lambda{}_j$:

$$\Delta\omega_j^\lambda=\varepsilon\cdot r\cdot\frac{1}{T}\sum_{t=0}^{L-1}\left[U_j\left(a_\lambda\left(t\right)\right)-\sum_{a_\lambda}U_j\left(a_\lambda\right)\pi_\lambda\left(a_\lambda;s_\lambda\left(t\right),\left\{\omega_j^\lambda\right\}\right)\right], \qquad \textbf{(2)}$$

where $a_\lambda(t)$ is an action actually selected by policy π_λ and $s_\lambda(t)$ is a state perceived by agent λ at time t. Each agent updates $\omega^\lambda{}_j$ by the learning rule in (2) at the end of each episode [4].

4 Pass Selection Problem

By applying the policy gradient method summarized in Section 3, Fifty-Storms exploits the learning results obtained to pass selection problems of midfielders and positioning problems for forwards to receive a through pass. We only give a brief description of the former application here.

A pass is a typical cooperative play between two players in a soccer match. Determining the receiver from among teammates is crucial in a pass. The first action to consider is safely passing the ball to a receiver. However, useless iteration of backward passes should be avoided. Thus, a player must select a receiver who stands in a position to receive the pass safely and who has a relatively high possibility of scoring a goal after receiving the ball. For this purpose, we use heuristics functions that seem useful for selecting a pass receiver as $U_j\ (a_\lambda;s_\lambda)$'s in (1). We used five heuristics from U_1 to U_5 for evaluating the current position of a teammate as a desirable pass receiver. U_1, U_2, and U_3 are heuristics for passing the ball safely. U_4 is a heuristics for making an aggressive pass. U_5 is used for treating reliable information as more important knowledge than unreliable information. All U_i's are normalized between 0 and 10.

Dribbling is considered a pass from and to the passer itself. If midfielders can pass and dribble the ball safely without being intercepted, the length of time their team holds the ball will be increased. Keeping the ball for as long as possible is obviously one effective strategy for midfielders in a soccer match.

We define learning episode σ of agent λ by a history of its states and actions from the time when agent λ gets the ball to the time when the ball is taken by an opponent player. Reward r given to agent λ is defined as $r(\sigma) = -1/L(\sigma)$. Since r takes a negative real value, it is actually a penalty rather than a reward.

5 Summary

In this team description paper, we outlined the characteristics of the Fifty-Storms team that is participating in the RoboCup 2D Soccer Simulation League for the first time. This team is based on an open version of the HELIOS team, agent2d (ver.1.0.0), and to which some modifications to HELIOS were added to enhance the offensive abilities of forwards and defenders by hand coding. In addition, it exploits the results of reinforcement learning, where a policy gradient method derives appropriate policies for midfielder's pass selection and forward's positioning to receive a through pass.

References

1. UvA Trilearn 2003, http://staff.science.uva.nl/~jellekok/robocup/2003/
2. HELIOS's URL site (in Japanese), http://rctools.sourceforge.jp/pukiwiki/

3. Riedmiller, M., Gabel, T., and Strasdat, H.: Brainstormers 2D – Team Description 2005. In: Bredenfeld, A., Jacoff, A., Noda, I., Takahashi, Y. (eds.) RoboCup 2005: Robot Soccer World Cup IX, LNCS, Springer (2005).
4. Igarashi, H., Nakamura, K., and Ishihara, S.: Learning of Soccer Player Agents Using a Policy Gradient Method: Coordination between Kicker and Receiver during Free Kicks. In: 2008 International Joint Conference on Neural Networks (IJCNN 2008), Paper No. NN0040, pp. 46-52 (2008).
5. Williams, R. J.: Simple Statistical Gradient-Following Algorithms for Connectionist Reinforcement Learning. Machine Learning, vol. 8, pp. 229-256 (1992).
6. Igarashi, H., Ishihara, S., and Kimura, M.: Reinforcement Learning in Non-Markov Decision Processes-Statistical Properties of Characteristic Eligibility-. IEICE Transactions on Information and Systems, vol. J90-D, no. 9, pp. 2271-2280 (2007, in Japanese). This paper is translated into English and included in The Research Reports of Shibaura Institute of Technology, Natural Sciences and Engineering, vol. 52, no. 2, pp. 1-7 (2008). ISSN 0386-3115
7. Ishihara, S. and Igarashi, H.: Applying the Policy Gradient Method to Behavior Learning in Multiagent Systems: The Pursuit Problem. Systems and Computers in Japan, vol. 37, no. 10, pp. 101-109 (2006).
8. Peshkin, L., Kim, K. E., Meuleau, N., and Kaelbling, L. P.: Learning to Cooperate via Policy Search. In Proc. of 16th Conference on Uncertainty in Artificial Intelligence (UAI2000), pp. 489-496 (2000).