

# NCL09

Yasushi Nakabayashi, Masato Masuda, and Kazuaki Ochiai

Department of Computational Science and Engineering,  
Faculty of Engineering, Toyo University,  
2100 Kujirai, Kawagoe, Saitama 350-8585, Japan  
nakabaya@eng.toyo.ac.jp

**Abstract.** NCL09 is the team of RoboCup soccer simulation 2D developed by Network Computing Laboratory in Toyo University, Japan. This team is based on the HELIOS developed by Akiyama. The main research topics of our team are the optimization of formation, dynamic strategy optimization and so on. Firstly, we modified the sourcecode of HELIOS for changing the way of using formation data. Then, we used Parallel Distributed Genetic Algorithm(PDGA) for optimizing formation data. The PDGA is known as the better method for optimizing multi-peak optimization problem. Secondly, we use coach agent for dynamic strategy change. Coach agent is monitoring the game conditions such as score and positions of all agents during games. Then the coach agent change the team strategy depend on the situation. Finally, we could develop the more efficient and strong team than original team.

**Keywords:** HELIOS, Optimization of formation, Dynamic optimization, Parallel Distributed Genetic Algorithm

## 1 Introduction

The history of our team NCL is very short. We firstly attended RoboCup competition in Japan Open 2006. Then we achieved the second place, but our team was questionable. Because we used the binary of UvAtrilearn2004 without any change. We just did the optimization of formation data in the "formation.conf" file. This approach is very simple, but we believe it very important and effective for developing 2D simulation team. Because UvA trilearn is highly sophisticated team and we could not develop more clever agents. This approach has meaning only from the viewpoint of parametric optimization, but has not meaning from the view point of developing multi-agents system.

Next year(2007), we began to take a new approach to attending RoboCup competition. We use the source code of HELIOS team developed by Akiyama. Firstly, we modified the source code of HELIOS for changing the way of using formation data. Then, we used Parallel Distributed Genetic Algorithm(PDGA) for optimizing formation data. The PDGA is known as the better method for optimizing multi-peak optimization problem. Furthermore, we optimize some parameter (e.g. shoot area, dribble intension) using parametric search for making up for the high computational costs of GA.

The research topics of our team is as follows:

1. Dynamically changing formation depending on the ball position
2. Dynamically changing formation depending on the score
3. Dynamically changing parameter of behavioral selection
4. Optimization of all parameter including formation.conf

This year, we use coach agent during all game time for changing team strategy. For example, the coach agent counts the number of effective opponents in offence time. If there are less than  $N_{opp}$  effective opponent player, some player change the role, e.g. sided back to side forward.

## 2 Optimization

### 2.1 Using formation data

We use HELIOS source code. But we would like to use UvA trilearn type formation.conf data. Then we combine HELIOS source with agent2D source code, which is also developed by Akiyama and is support formation.conf data. We investigated which agents should use formation.conf data. After using some heuristic method, we decided using one formations.

### 2.2 Dynamically changing formation depending on the ball position

The default formation is 4-3-3. Our team change formation dynamically depending on the ball position. If the ball position is below the line-20, our team use 5-2-3 formation. If the ball position is between the line-20 and +20, our team use 4-3-3 formation. If the ball position is above the line +20, our team use 3-3-4 formation. Furthermore, we optimize the line, -20 and +20, using the Genetic Algorithm.

### 2.3 Dynamically changing formation depending on the score

We also dynamically change formation depending on the score. If our team has more score than competitor, we replace 5-2-3 formation by SD5-2-3 formation. This new formation is extremely defensive and optimized for not giving goals.

### 2.4 Dynamically changing parameter of behavioral selection

In the source code of HELIOS, there are some constant parameter for deciding behavior. For example, the length parameter between the agent which has ball and nearest opponent is used for deciding next behavior is pass or dribble. We changed it as variable, and optimize it using GA.

## 2.5 Optimization of all parameter including formation.conf

Finally, we optimize all parameter including formation.conf. Our team has four formations, 5-2-3, SD5-2-3, 4-3-3 and 3-3-4. We use the GA algorithm for each formation sequentially. The one set of the GA optimization is 100 generations. After 800 generations of the GA optimization, we decreased the mutation parameter for local search. Total 11,000 generation is one set of the GA optimization. We optimized formations.conf data using many set of this method.

## 2.6 Optimization of player role using coach agent

We use coach agent during all game time for changing team strategy. Coach agent count the number of effective opponents. If there are less than  $N_{opp}$  effective opponent player, some player change the role, e.g. side back to side forward. We also use GA method for this optimization. The GA parameters are 11, the role number of all players except for goalie and  $N_{opp}$ . The evaluation function is  $(\text{goal scored}) - (\text{goal allowed}) / 5$ .

## 2.7 Parametric search

We optimize some parameters (e.g. shoot area, dribble intension) using parametric search for making up for the high computational costs of GA. Generally, GA requires many computational times and evaluating the developed team also spend long time. Therefore, we used simple parametric search for optimizing some parameters. For example, we varied the parameter of shoot area from 0.0 to 9.0, then search the value which shows the best performance.

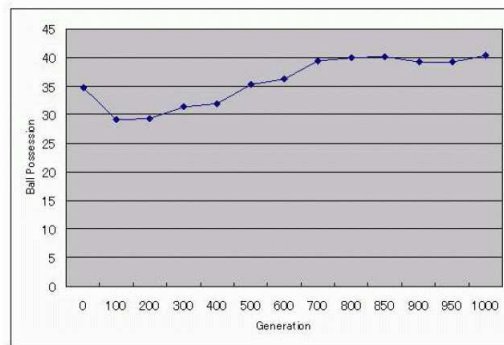
## 3 Result

Fig.1 shows the generation transition of ball possession. This result shows the efficiency of our approach. Fig.2 shows the time transition of ball possession against HELIOS and WrightEagle. This result shows that our developed team is stronger in the second half. Furthermore, Fig.3 shows the generation transition of GA's evaluation function. The x-axis means the number of generations. The y-axis means the value of evaluation function. This result shows that our team could become stronger.

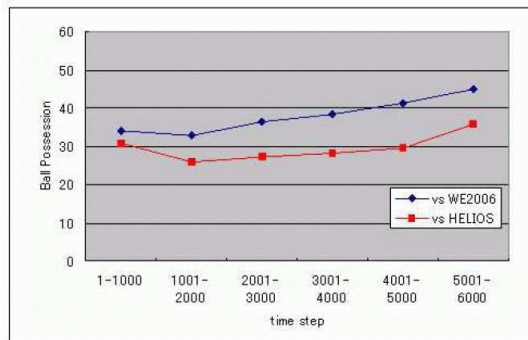
Fig.4 shows the result of parametric search of dribble parameter. We adopted 0.75 as the best value. Fig.5 shows the result of parametric search of shoot parameter. We adopted 7.0 as the best value.

## 4 Conclusions

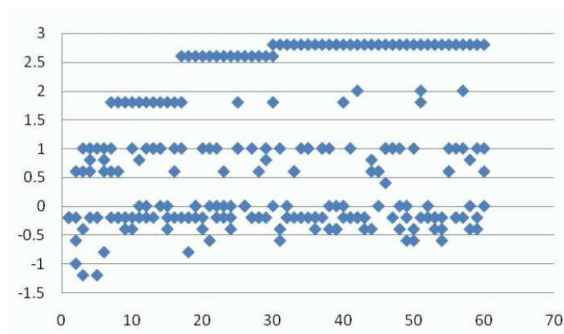
We developed RoboCup soccer simulation 2D team based on HELIOS source code. Our approach is not developing agents, but optimizing many parameters including the formations.conf data, the line changing formations, behavior selection, player roles and so on. As a result, we could develop the more efficient and strong team than original team.



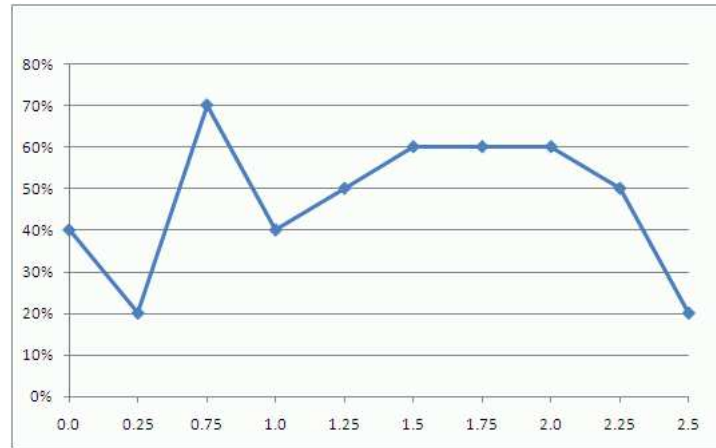
**Fig. 1.** Generation transition of ball possession



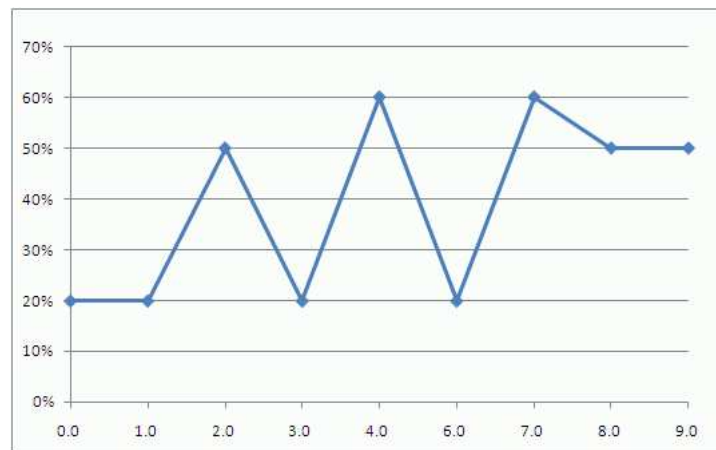
**Fig. 2.** Time transition of ball possession



**Fig. 3.** GA result of player role



**Fig. 4.** Result of parametric search (x: dribble parameter, y: winning percentage)



**Fig. 5.** Result of parametric search (x: shoot parameter, y: winning percentage)