NCL07

Yasushi Nakabayashi, Yuji Ogawa, Tomoyuki Ito,
Masaharu Shishiyama, and Kazunori Hara

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

Abstract. NCL07 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 Dr. Hidehisa Akiyama. The main research topics of our team are the optimization of formation, dynamic optimization and so on. 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. Finally, we could develop the more efficient and strong team than original team.

Key words: 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 UvA trilearn 2004 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 viewpoint of developing multi-agents system.

This year we adopt a new approach to attending RoboCup competition. We use the source code of HELIOS team developed by Dr. Hidehisa 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.

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

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 Dr. Hidehisa Akiyma and is support formation.conf data.

We investigated which agents should use formation.conf data. After using some heuristic method, we decided using type B of Fig.1. UvA-Trilearn mark means using formation.conf data. Type B formation shows high goal ability.

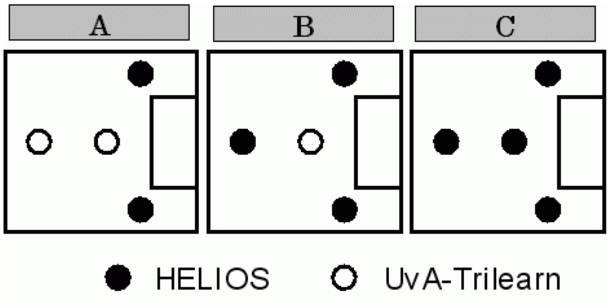

Fig. 1. Selection of Agents using formation.conf

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. Fig.2 shows initial lines of changing formation. 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. The 5-2-3 and 3-3-4 formations are shown in Fig.3. 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 extreamly defensive and optimized for not giving goals.

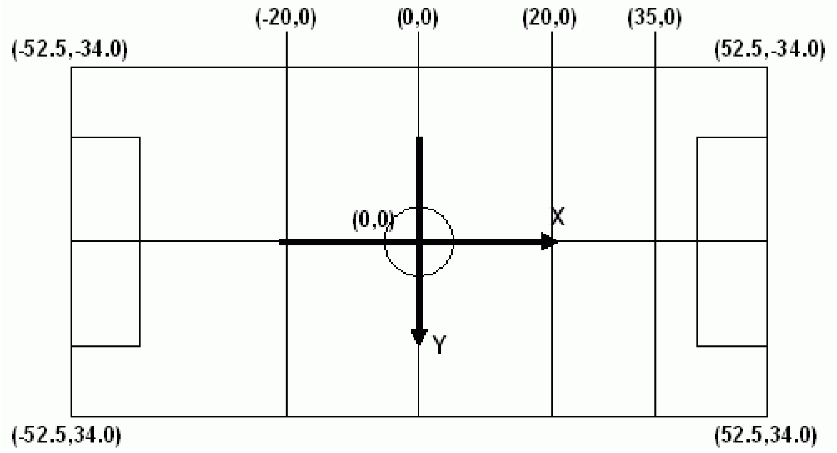

Fig. 2. Line of Changing formation

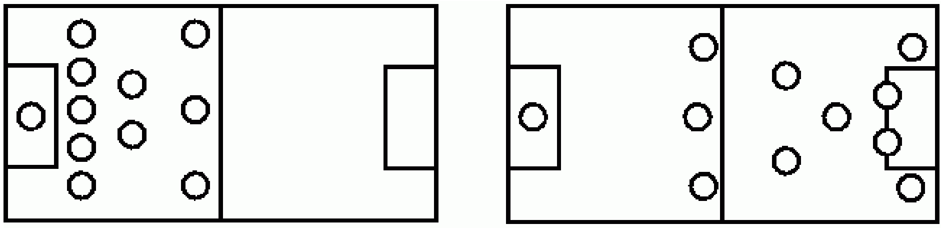

Fig. 3. 5-2-3 and 3-3-4 formations

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 oppornetn is used for diciding next behavior is pass or dribble. This parameter was set as constant. 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 optimization scheme using the GA is shown in Fig.4. 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 decresed the mutation parameter for local search. Total 1,000 generation is one set of the GA optimization. We optimized formations.conf data using many set of this method.

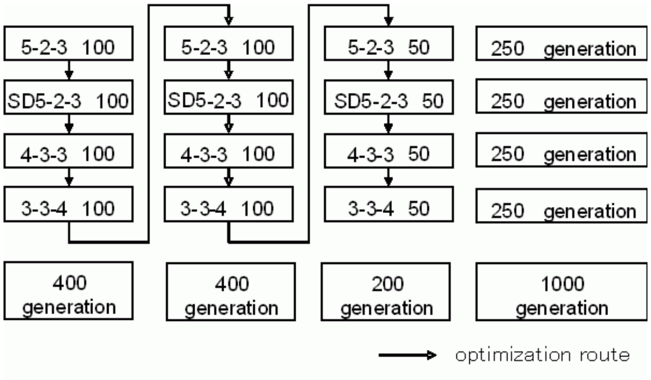

Fig. 4. Optimizaton scheme for formation.conf data

3 Result

Fig.5 shows the generation transition of ball possession. This result shows the efficiency of our approach. Fig.6 shows the time transition of ball possession against HELIOS and WE2006. This result shows that our developed team is stronger in the second half.

4 Conclusions

We developed RoboCup soccer simulation 2D team based on HELIOS source code. Our approach is not developing agents, but optimizing many parameters

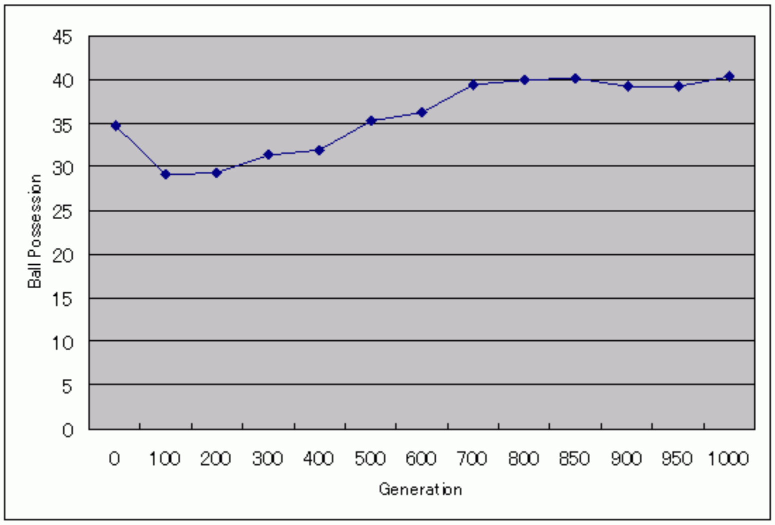

Fig. 5. Generation Transititon of Ball Possesion

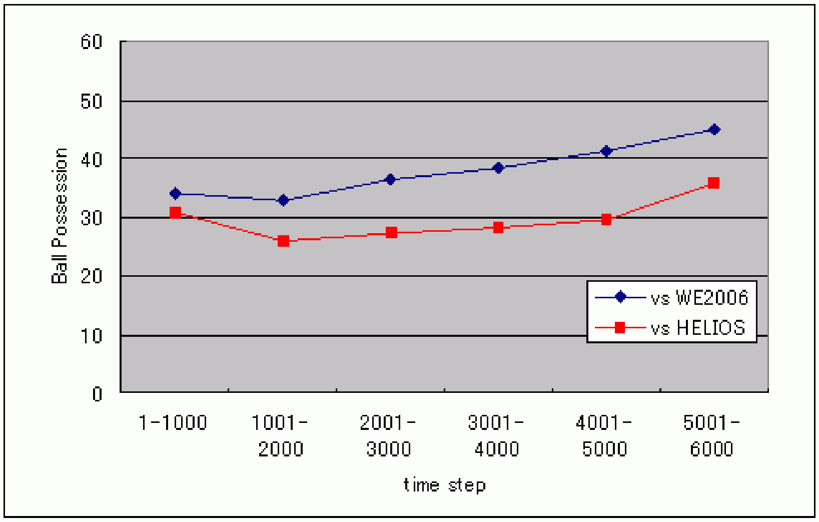

Fig. 6. Time Transition of Ball Possesion

including the formations.conf data, the line changing formations, behavior selection and so on. As a result, we could develop the more efficient and strong team than original team.

References

1. Hidehisa Akiyama: RoboCup Soccer Simulation 2D League Victory Guide. ISBN-13: 978-4798013312 (2006) (in Japanese)