Wright Eagle 2-D Simulation Team 2005

BAI Peng, FAN Changjie, WU Feng, CHEN Xiaoping

Multi-agent Lab, Department of Computer Science and Technology,
University of Science and Technology of China,
230027,Hefei,P.R.China
baipeng@ustc.edu, cjfan@mail.ustc.edu.cn
WWW home page: http://mas.ai.ustc.edu.cn

Abstract. This paper introduces the implementation of Wright Eagle 2-D Simulation team 2005. The team is implemented on the basis of WrightEagle 2004 and thus inherits the decision-making framework of its antecedent, which is a combination of a simplified model of MDP and PRS system[1–3]. In WrightEagle2005, we re-build several key basic-action modules of the team in order to enhance the performance of the agent, such as the Unknown-player Matching algorithm, the new Fast-Dribble skill and the fixed Pass-Ball algorithm, and we also use a new decision-making structure which is based on POMDP[4–6] method to implement the tactic behavior.

1 Overview

In this paper we briefly present the new features of WrightEagle 2005. WrightEagle Simulation League team development has been started since 2000. To the year 2003, we had not changed much of the basic codes of the team. In contrast, compared with the WrightEagle2004, the new team has four-fifth completely new-written codes, including the seventy-percent of the fundamental modules and almost all the decision codes. As a result, the new team gets far better performance than before, that rely on the strong foundation and the well-debugged decision algorithms.

In the following sections, we focus on the pass module and the dribble module.

2 Pass Analysis

To begin with, we note some difficulties in the development of the pass module. Generally speaking, the common implementation of the pass-ball behavior is based on the decision module which contains the estimation of the possibility and the effort of the actions. The difficulty of the application of the method is that the estimation can not be as objective as it should be in the dynamic and indeterminable environment as RoboCup Simulation. The application also has a shortcoming that the evaluation lacks the foresight when the complex behavior is expected.

Instead of the traditional decision method[7, 8] based mediator used in the WrightEagle2004 and before, we build a simplified MDP/POMDP[4, 6] decision module to implement the multi-step evaluation of the continues states. We use a approximate estimation of the states and by which the algorithm searches for the best pass behavior in a certain depth. Despite of the complexity of the program, it acts more rationally like a human team. To solve the too-much CPU time problem, we rebuild the fundamental codes of the world model, which based on the relative-coordinate updating system.

3 Fast Dribble Technique

In this part, we employ several functions to evaluate the state of the agent and to estimate the style of the dribble action in order to make the dribble proceeding continuing. Our goal is to dribble as fast as possibility while keeping the ball near around the body. Fig1. shows one simple situation of the dribble proceeding that the ball goes paralleled to the player, we deals with more practical situations in the program such as kick-away dribble and the break-through dribble when considering the opponents faced.

One principle of dribble is to ensure the ball is in kickable-area of self in the following cycles and not in the opponents'. The dribble behavior consist of one kick followed by several dashes.

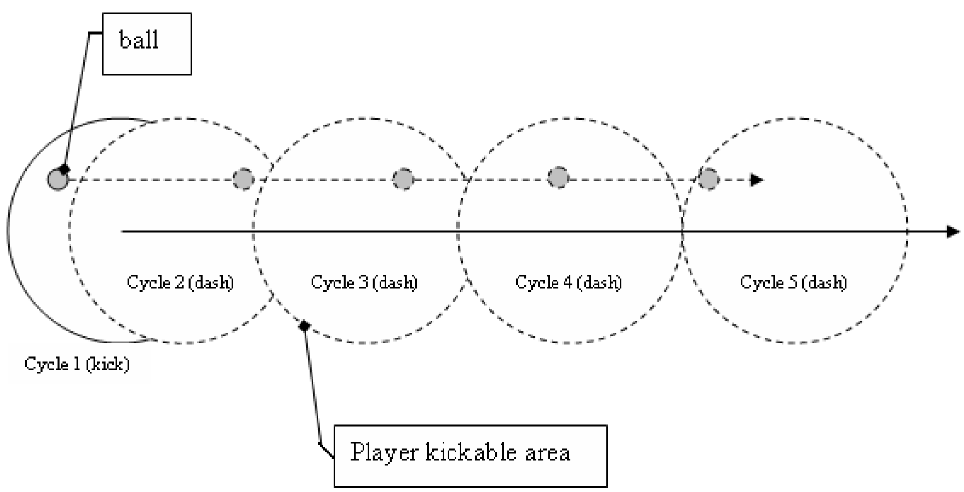

Fig. 1. dribble proceeding

As a result, our dribble model achieves a good outcome, the experiments and the details will be on the further documents.

4 Future Work

We only refer to some rough ideas here, and there is too much work to do in the future. The agent is expected to act more rationally in the collaborative and competitive environment, and we also have work to do for the basic level of the agent. The dribble modules will also be one of the most important parts in the future work, and the complexity of the calculation which takes more and more CPU-time have became a serious problem. The program needs to be optimized to a great extent. Above all, we need to continue with the problems mentioned above, whether the 2-D competition will be still exist, because in all probability the problems are going to show up in the new 3-D environment.

References

1. J. Lee, M. J. Huber, E. H. Durfee, and P. G. Kenny. *UM-PRS: an implementation of the procedural reasoning system for multirobot applications.* AIAA/NASA Conf. on Int. Robots in Field, Factory, Service and Space. American Institute of Aeronautics and Astronautics, 1994.
2. F.F. Ingrand, R. Chatila, R. Alami, and F. Robert, *PRS: A High Level Supervision and Control Language for Autonomous Mobile Robots* IEEE International Conference on Robotics and Automation, Mineapolis, USA, 1996.
3. T. F. Bersano-Begey, P. G. Kenny, and E. H. Durfee. *Multi-agent teamwork, adaptive learning, and adversarial planning in robocup using PRS architecture.* submitted to the Robocup 97 Workshop, 1997.
4. Sutton, R.S., Precup, D., Singh, S. *Between MDPs and semi-MDPs: A Framework for Temporal Abstraction in Reinforcement Learning*, Arificial Intelligence 112:181-211. 1999
5. Craig Boutilier, Thomas Dean, Steve Hanks, *Decision Theoretic Planning: Structural Assumptions and Computational Leverage.* In: Journal of Artificial Intelligence Research, 11:1–94 (1999).
6. L.R. Rabiner, *A Tutorial on Hidden Markov Model and Selected Applications in Speech Recognition*
7. Li Zhenyu, Chen Xiaopin, Fan Changjie, Bai Peng, *Wright Eagle 2004 Simulation Team Description*, 2004.
8. Yao Jinyi, Lao Ni, Yang Fan, Cai Yunpeng, and Sun Zengqi. *Technical solutions of TsinghuAeolus for Robotic Soccer.*In: Proceedings of the RoboCup Symposium 2003, Padova, Italy, 2003.