Evaluating the Minority Game Strategy in Agent Role Assignments

Tingting Wang
Department of Computer Science
Hong Kong Baptist University
Kowloon Tong, Hong Kong
ttwang@comp.hkbu.edu.hk

Jiming Liu
Department of Computer Science
Hong Kong Baptist University
Kowloon Tong, Hong Kong
jiming@comp.hkbu.edu.hk

ABSTRACT

A team-based competitive environment is a complex multi-agent environment, in which agents are required to coordinate with each other not only to enhance their collective behavior, but also to compete with the other team. An interesting research problem in such an environment is the role assignment problem (RAP). The problem requires agents to decide the roles they should take based on real-time feedback from a dynamically changing environment. In this paper, we aim to provide a new strategy that is based on the Minority Game (MG) model, i.e., the MG strategy, for assisting a team to perform effective role assignments in such an environment. Through experiments in our previous work, we have demonstrated that the MG strategy is helpful for RAP in RoboCup Simulation League. In this paper, with a more *generic* competitive environment such as `DynaGrid`, we find that the MG strategy is not always effective. It can help agents do effective roles assignments in the case that the targets move in a nonlinear motion.

Categories and Subject Descriptors

I.2.11 [**Computing Methodologies**]: Artificial Intelligence—*Distributed Artificial Intelligence*; I.2.6 [**Computing Methodologies**]: Artificial Intelligence—*Learning*

General Terms

Algorithms, Experimentation

Keywords

multi-agent, competition, teamwork, role assignment

1. INTRODUCTION

In distributed multi-agent team-based systems, an attractive problem is the *role assignment* problem (RAP). Agents in such systems have limited sensing range and uncertainty about their surroundings. They have to work together in a team with an appropriate distribution of roles to optimally fulfill their tasks, as well as compete with other teams.

The Minority Game (MG), proposed by Challet and Zhang [1], is a game theory for studying cooperation and competition of agents given limited resources. Agents playing MG do not interact directly with each other. They achieve their implicit collective behavior through an information sharing mechanism.

MG has been demonstrated to be efficient in a competitive game environment with limited resources. By relating the *alternatives* of players in MG to the *roles* of agents in a competitive environment, we can simulate a MG inspired method for solving RAP in a competitive multi-agent system. We intend to examine whether or not MG works in RAP and how we can validate it.

To address the above issues, first, we select RoboCup Simulation League (RSL) [2] as a specific platform where agents cooperate through taking different roles over time; second, we provide a more common competitive environment, i.e., `DynaGrid` in our research, in which coordination among agents and targets are fully described.

2. THE MG STRATEGY IN RSL

In MG, each player is equipped with a set of *rules* [1]. During a game, a player learns to use its rules best. Specifically, at each cycle, a player uses the rule with the highest virtual value. In [3], we have developed a RSL team, named BUTT, which is equipped with the MG strategy. The meaning of the *MG strategy* is that by relating the alternatives of MG (0 and 1) to the roles in RSL (attacker and defender), the agents in a team learn a set of useful rules by utilizing the *ball-controlling time* of a team in RSL.

Figure 1 shows that as time goes by, BUTT performs better in controlling the ball than TsinghuAeolus. By increasing the memory size M, the ball-controlling time is increasing, and the improvement of BUTT becomes more and more obvious in a nonlinear fashion because of more effective role assignments. Thus in RSL, the role assignments among agents can be dynamically improved by using the MG strategy.

3. THE MG STRATEGY IN DYNAGRID

[1]Note that here we use the term of *rule* instead of a commonly used term *strategy* in a minority game, in order to differentiate it from the *the MG strategy* that we proposed.

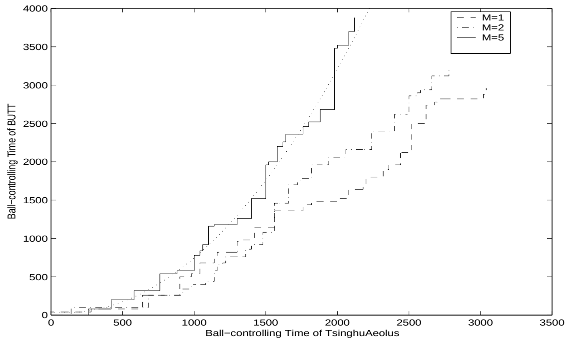


Figure 1: The ball-controlling time of BUTT vs. TsinghuAeolus in the case of $M = 1, 2, 5$.

The above analysis of the MG strategy is carried out on RSL, a special competitive environment. Though our MG strategy has shown to be effective in RSL, we still cannot fully generalize our results to other competitive environments due to the special assumptions made in RSL. We need to work with a more *generic* competitive environment, i.e., `DynaGrid` in this paper. This environment has the following characteristics:

1. It has different parts with different varying utility values to represent the importance of positions.

2. The agents in this environment can be classified according to their roles. Therefore the roles of agents (e.g., reactive agents, coordinate agents) can be related and guided with the alternatives of MG (0 and 1).

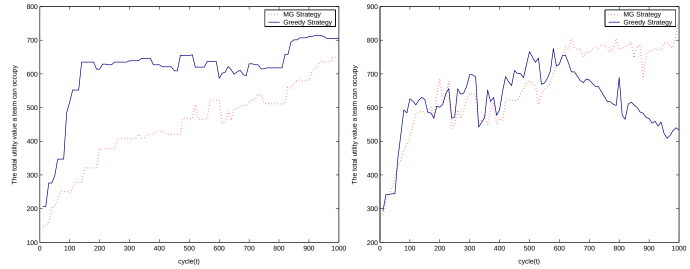
Then we can design the MG strategy in `DynaGrid`. First, Let $M_k(t)$ be the values that a team of agents occupy, so $\triangle M_k(t)$ indicates the variation of the important positions that a team can occupy at different cycles.

Then, suppose $V_{is}(t)$ is the virtual value of agent i with *rule* s. The virtual value can be updated as follows:

$$V_{is}(t+1) = V_{is}(t) + \omega \cdot \triangle M_k(t) \qquad (1)$$

where ω is a positive constant. The predicting result of a MG player is determined by its current rule. And the role of an agent is related to its MG result. Eq. 1 indicates that the total values that a team of agents can occupy influence the ranking of their roles, which can finally help them to learn an effective combination of role assignments.

Figures 2 and 3 show the competition results between a team with the *MG strategy* and a team with a *greedy strategy*. For a linear motion, the complexity is relatively low. Agents do not need much external information in selecting their roles. On the contrary, the complexity of a high-order, nonlinear motion is high. When a target is in a curved motion, agents that use the MG strategy can learn to choose their roles by means of predicting the motion of the target more accurately. Thus, the MG strategy is found to have better performance in a more dynamically changing environment, that is, the environment that involves a target in a nonlinear motion. Also, agents with a larger memory size perform better. This is consistent with the results that we obtained in our previous work.



(a) **(b)**

Figure 2: The total utility value of the cells occupied by each team. (a) Target is in a linear motion. (b) Target is in a curved motion.

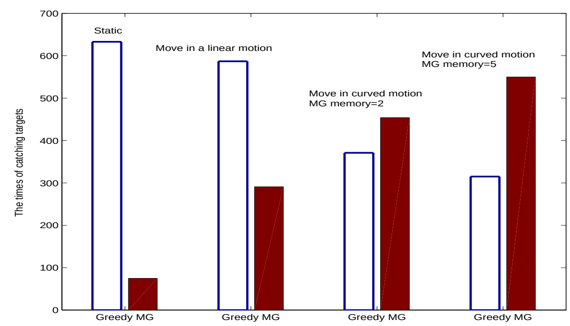


Figure 3: The times that each team catches the targets.

4. CONCLUSIONS

In this paper, we formulated the MG strategy for solving the role assignment problem (RAP) in RoboCup Simulation League (RSL) and examined the effectiveness of the MG strategy in real matches. However, RSL is just a case study of RAP. We design a grid environment `DynaGrid` to simulate the coordinate behaviors among agents. We observed that, MG strategy is not always effective. Its performance depends on the motion of the targets. When targets are static or move in a linear motion, MG strategy is not quite effective than the greedy strategy. However, if the targets move in a nonlinear motion, i.e., the complexity of the motion is higher than a linear motion, MG strategy can help agents do effective roles assignments to occupy good positions and catch more targets.

5. REFERENCES

[1] D. Challet and Y.-C. Zhang. Emergence of cooperation and organization in an evolutionary game. *Physica A*, 246(3-4):407–418, 1997.

[2] H. Kitano. Robocup: The robot world cup initiative. In *Proceedings of the 1st International Conference on Autonomous Agent (Agents-97), Marina del Ray, The ACM Press*, 1997.

[3] T. Wang, J. Liu, and X. Jin. Minority game strategies for dynamic multi-agent role assignments. In *Proceedings of IEEE/WIC/ACM Intelligent Agent Technology (IAT'04)*, Beijing, 2004.