Strategy Learning For A Team In Adversary Environments

Yasutake Takahashi[1], Takashi Tamura[1], and Minoru Asada[1]

Dept. of Adaptive Machine Systems,
Graduate School of Engineering Osaka University, Suita, Osaka 565-0871, Japan

Abstract. Team strategy acquisition is one of the most important issues of multiagent systems, especially in an adversary environment. RoboCup has been providing such an environment for AI and robotics researchers. A deliberative approach to the team strategy acquisition seems useless in such a dynamic and hostile environment. This paper presents a learning method to acquire team strategy from a viewpoint of coach who can change a combination of players each of which has a fixed policy. Assuming that the opponent has the same choice for the team strategy but keeps the fixed strategy during one match, the coach estimates the opponent team strategy (player's combination) based on game progress (obtained and lost goals) and notification of the opponent strategy just after each match. The trade-off between exploration and exploitation is handled by considering how correct the expectation in each mode is. A case of 2 to 2 match was simulated and the final result (a class of the strongest combinations) was applied to RoboCup-2000 competition.

1 Introduction

Team strategy acquisition is one of the most important issues of multiagent systems, especially in an adversary environment. RoboCup has been providing such an environment for AI and robotics researchers as one of the most challenging issues in the field since 1997 with increasing number of various leagues.

In the simulation league, a number of methods for team strategy acquisition have been applied. Wunstel et al.[1] proposed a method to acquire an opponent team model in order to build the own team strategy to beat the opponent. However, this method requires all kinds of information and observations on every player on the field during the game. Stone and Veloso [2] proposed a locker room agreement to decide which to take, that is, a defensive formation or an offensive one considering the score just before the match. Dynamic role exchange during the game is also attempted through the broadcasting line.

Not so many teams in the real robot league have tried to apply the method for team strategy acquisition because they cannot afford to pay any attention to such an issue but need to care about more hardware stuffs. Castelpietra et al. [3] proposed a method of cooperative behavior generation by exchanging information and roles through the wireless network. Uchibe et al. [4] proposed a method of dynamic role assignment with shared memory of task achievements.

These methods need explicit communication lines to realize cooperative behaviors. Therefore, the realized cooperation might be damaged by the perception and/or communication noises which affect the opponent model estimation and communication itself.

More deliberative approaches to the team strategy acquisition seem less useful due to its ineffectiveness in such a dynamic and hostile environment. One alternative is to use the dynamics of environment itself with behavior-based approach [5]. Depending on the environment dynamics consisting of not only the teammate actions but also the opponent ones, the cooperative behaviors are expected to happen much more often than by deliberative approaches. However, the design principle is rather conceptual and has not revealed which combination of players can emerge more cooperative behaviors.

This paper presents a learning method to acquire team strategy from a viewpoint of coach who can change a combination of players each of which has a fixed policy. Assuming that the opponent team has the same choice for the team strategy but keeps the fixed strategy during one match, the coach estimates the opponent team strategy (player's combination) by changing the own team strategy based on game progress (obtained and lost goals) and notification of the opponent strategy just after each match. The trade-off between exploration and exploitation is handled by considering how correct the prediction of the opponent team strategy is. A case of 2 to 2 match was simulated and a part of the final result (a class of the strongest combinations) was applied to RoboCup-2000 competition.

The rest of the paper is organized as follows. First, the definitions of the task and the team strategy are given along with a basic idea of the use of environmental dynamics in the context of emergence of cooperative behaviors. Next, a learning method for team strategy estimation is introduced based on the information such as game progress. Then, the results of the computer simulation are given and the real robot experiments are shown.

2 Basic Ideas and Assumptions

As the environment becomes more complex, the system usually tends to be more complicated to cope with the complexity of the environment. Especially, in a multiagent system, adding a capability of communication seems useful to realize cooperative behaviors. However, such a system becomes useless when the environment is much more dynamic and therefore the prediction of the environment changes is hard.

Simon pointed out the use of the environment itself [6]. The walking of ants on the sands is not so complicated but just simple to adapt it to the environment without any predictions or plans. A behavior-based approach is one of such ways to utilize the dynamic environments. However, it seems difficult to design each robot in a multiagent hostile environment because the environment dynamics seriously depends on the actions of other agents who might have their own policies to take actions.

Then, we propose a learning method of team strategy acquisition assuming the following settings.

- Each team has various kinds of players that are designed by behavior based approach without communication lines to each other.
- A team strategy is defined as a combination of players.
- A coach may change its strategy by exchanging players.
- The opponent team has the same choice of the player's combination, that is, it can take one of the team strategies.

The problem here is to find the strategy to beat the opponent from a viewpoint of coach based on the game progress and notification of the opponent strategy after the match. Since the opponent has the same choice, the task here is to identify the unknown opponent strategy and to know the relationship (stronger or weaker) between the opponent's and its own. Then, we add one more assumption.

- The opponent team strategy changes randomly every match but is fixed during one match while the learning team's coach may change its team strategy to estimate the opponent's team strategy during one match. Hereafter, the coach means the learning team's coach.

3 Team Strategy Learning

First of all, we prepare two modes of the coach policy.

Exploration Mode Sampling data in order to fill the game scores in the obtained and lost goals table.

Exploitation Mode Estimating the better team policy against the predicted opponent team.

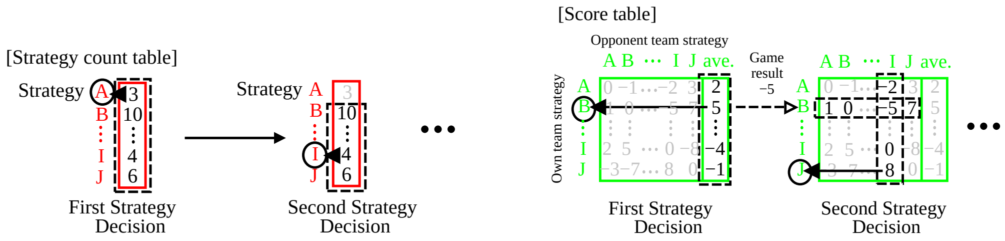
The coach decides the team strategy based on a weighted sum of the two modes. The weight w_{er} is the ratio between the exploration mode and the exploitation one, and updated gradually from exploitation to exploration as the coach becomes able to estimate the own team strategy to beat the opponent. We define an probability to select the j-th strategy $P(j)$ as following:

$$P(j) = (1 - w_{er})P_r(j) + w_{er}P_e(j) \tag{1}$$

where $P_r(j)$ and $P_e(j)$ are the occurrence probabilities to select the strategy j defined by exploration mode and exploitation one, respectively.

3.1 Exploration Mode

The exploration mode is simple and straightforward; select the most inexperienced strategy to obtain more data. One match consists of several periods between which the coach may change its team strategy while the opponent one



(a) Exploration Mode

(b) Exploitation Mode

Fig. 1. Strategy selection

is fixed during one match. The coach has a table $T_r(j)$ which counts number of periods in which the j-th one among n strategies is applied. We define the probability $P_r(j)$ to select the j-th strategy in the next period as following:

$$P_r(j) = \frac{p_r(j)}{\sum_{l=1}^{n} p_r(l)}, \tag{2}$$

where

$$p_r(j) = \begin{cases} 0 & \text{(if coach has selected the } j \text{ th strategy in the same match already)} \\ 1 & \text{(if } T_r(j) = 0 \text{)} \\ \dfrac{1}{T_r(j)} & \text{(else)} \end{cases} . \tag{3}$$

Fig.1 (a) shows the idea of the exploration mode. The coach selects the most inexperienced strategy according to the strategy count table at the first period. At the second and the following periods, the coach skips the previously taken strategies in the current match and selects the most inexperienced strategy among the rest.

3.2 Exploitation Mode

The exploitation mode can be decomposed into two parts. One is the estimation process of the opponent team's strategy, and the other is the decision of an own team's strategy to beat the opponent. The coach has a score table $T_s(k, j)$ which stores the difference between obtained and lost goals in a period in which the opponent team takes the k-th strategy and own team takes the j-th strategy. When the coach had score (difference between the obtained and lost goals) s_j^i using the j-th strategy at the i-th period, the coach estimates the opponent team's strategy at the i-th period by:

$$P_s^i(k) = \frac{p_s^i(k)}{\sum_{l=0}^{n} p_s^i(l)}, \tag{4}$$

where

$$p_s^i(k) = \max_l |s_j^i - T_s(l,j)| - |s_j^i - T_s(k,j)|. \tag{5}$$

The first term at right-hand side of equation (5) means the difference between the obtained s_j^i to the largest different score. The second term means an error of the expected value of the score (the difference between the obtained and lost goals) assuming that the opponent takes the k-th strategy compared to the current score in the period. This term is small if the estimation is correct. Then, $p_s^i(k)$ becomes large when the opponent strategy is the k-th one. Since we assume that the opponent keeps the fixed strategy during one match, the coach can estimate the opponent team's strategy after m periods in the match as following:

$$P_s(k) = \frac{1}{m}\sum_{i=0}^{m} P_s^i(k) \tag{6}$$

At the second step, the coach predicts an expected score (difference between the obtained and lost goals) $x_s(j)$ in the next period when the coach takes the j-th strategy by:

$$x_s(j) = \sum_{k=1}^{n} P_s(k)T_s(k,j). \tag{7}$$

We define a occurrence probability $P_e(j)$ to select the the j-th strategy in the next period using $x_s(j)$ by:

$$P_e(j) = \frac{p_e(j)}{\sum_{l=0}^{n} p_e(l)}, \tag{8}$$

where

$$p_e(j) = \begin{cases} 0 & (\text{if } x_s(j) \le 0) \\ x_s(j) & (\text{else}) \end{cases}. \tag{9}$$

Fig.1 (b) shows the idea of the exploitation mode. The coach selects the strongest strategy according to the average in score table $T_s(k,j)$ at the first period (taking the strategy B which has the largest average score 5). At the second and the following periods, the coach estimates the opponent team's strategy using the score at the previous periods (according the match result, the opponent strategy is predicted as I), and estimates the best strategy against the estimated opponent team's strategy (taking the best strategy J against the regarded opponent as I).

3.3 Update of w_{er}

The coach updates the weight w_{er} which is the ratio between the exploration mode and the exploitation one after a match consisting of n_p periods, and the coach took strategy a^i at the i-th period as following:

$$w_{er} \leftarrow w_{er} + \Delta w_{er}, \tag{10}$$

where

$$\varDelta w_{er} = \alpha \sum_{i=2}^{n_p} s^i \frac{i}{n_p} x_s(a^i). \tag{11}$$

α is an update rate. If the weight w_{er} becomes more than 1 or less than 0, it is set to 1 and 0, respectively. These equations intend to update the weight w_{er} so that the coach takes the exploitation mode if the estimated score is correct, or it takes the exploration mode. $\frac{i}{n_p}$ means that the coach makes the weight of estimation at early periods small because it doesn't have data enough to estimate the opponent team's strategy at early periods.

4 Experiments

4.1 Match Setting

We assume the following settings.

- Players and field obey the regulations of the RoboCup2000 middle size league.
- Each team has two players.
- Players have its own fixed policy.
- A player has no communication devices with other teammate.
- One match consists of 500 trials.
- One period consists of 100 trials. That means the coach has five opportunities to select the team strategy during one match.
- The opponent strategy changes randomly every match.
- Players' initial positions are random in their own areas and ball is set at the center of the field.

4.2 Team Strategy

Team strategy is defined from a viewpoint of coach who can change a combination of players each of which has a fixed policy. We prepare 4 types of players' policies from view points of ball handling and social behaviors.

Ball handling

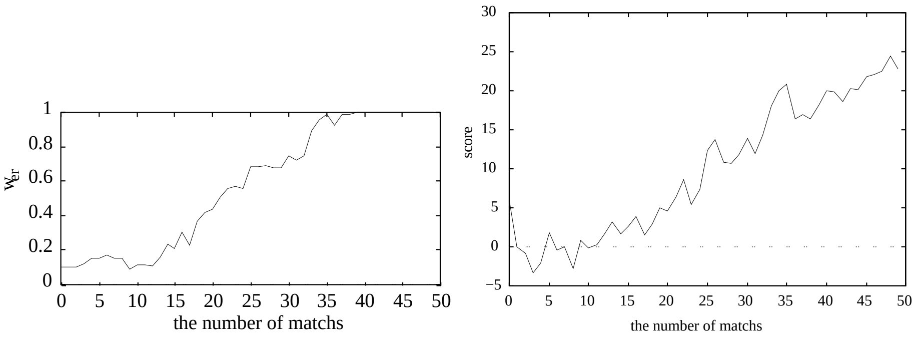
ROUGH a quick approach and rough ball handling
CAREFUL a slow approach and careful ball handling

Social behaviors

SELFISH regarding other all players as obstacle
SOCIAL yielding a way to the teammate if the teammate is on its own way

Totally there are 4 types of players' policies because of the combination of the ball handling controllers and the social behaviors. Then, there are 10 kinds of team strategies because one team has two individual players: ${}_4C_2$ strategies of heterogeneous controllers and ${}_4C_1$ strategies of homogeneous controllers.

4.3 Results



(a) w_{er} (b) score

Fig. 2. The change of w_{er} and score (difference of obtained and lost goals)

Fig.2 (a) shows the change of w_{er} and that the coach switches mode from exploration to exploitation over the matches. Fig.2 (b) shows the sequence of average score (difference of obtained and lost goals) in the last 10 matches. The team becomes able to beat the opponent team while the coach switch mode to exploitation one. The team obtained more points against the opponent when the coach takes exploitation mode. These figures show that the coach appropriately controls the switch of two modes.

Table 1 shows the score table after the learning (after 50 matches). The values in the table are the the differences of obtained and lost points in one period. The table has data of stronger strategies and no data of weaker strategies. This means the coach eliminates the experience of week strategies and the learning process is efficient. The learning time needs 1/3 compared to the exhaustive search.

Fig.3 shows the learned relationship among strategies. The strategy pointed by a head of arrow is stronger than strategy pointed by a tail of arrow. The system has a cycle at the top 3 strategies. It seems impossible to select the best team among the all teams, and the coach has to learn the relationship between strategies to beat the opponents.

We applied the part of the final result to the match in the RoboCup2000 middle size league, that is two forward players take a strategy of CAREFUL-SELFISH(Type A) and ROUGH-SOCIAL(Type B) and obtained many goals. Fig.4 shows an example applying the strategy. In this situation, the two robots recover each others' failures quickly. ① indicates that the two different robots follow a ball. The type B robot tries to shoot a ball to the opponent goal at ②. But it failed at ③ because the ball handling skill of type B is not so good, and type A robot recovers the failure soon. The type A robot tries to shoot the ball, but the opponent goalie defends it at ④. The type A robot tries to shoot the

Table 1. The result of score table by learning

own	opponent										average
	1	2	3	4	5	6	7	8	9	10	
1	-	2.3	-4.2	5.5	5.7	6.3	3.0	-	3.2	11.6	4.2
2	-2.3	-	1.2	1.1	6.4	4.1	11.3	-	-	8.3	4.4
3	4.2	-1.2	-	-	-	-3.4	-	-	-3.8	-	-0.6
4	-5.5	-1.1	-	-	-	-	-	-	-	-	-2.8
5	-5.7	-6.4	-	-	-	-	-	-	-	-	-4.1
6	-6.3	-4.1	3.4	-	-	-	-	-	-	-	-2.9
7	-3.0	-11.3	-	-	-	-	-	-	-	-	-4.3
8	-	-	-	-	-	-	-	-	1.1	-	-1.3
9	-3.2	-	3.8	-	-	-	-	-1.1	-	-	-1.0
10	-11.6	-8.3	-	-	-	-	-	-	-	-	-4.3
average	-4.2	-4.4	0.6	2.8	4.1	2.9	4.3	1.3	1.0	4.3	-

1:(RO-SE,CA-SO) , 2:(RO-SE,CA-SE) , 3:(RO-SE,RO-SO) , 4:(CA-SE,CA-SO) ,
5:(RO-SO,CA-SE) , 6(CA-SE,CA-SE): , 7:(RO-SO,CA-SO) , 8:(RO-SE,RO-SE) ,
9:(RO-SO,RO-SO) , 10:(CA-SO,CA-SO)
RO:ROUGH , CA:CAREFUL , SE:SELFISH, SO:SOCIAL

ball from left side of the goal at ⑤ and ⑥, but unfortunately fails again while the type B robot moves its position behind the type A robot. The type B robot tries to recover the failure of type A robot's shooting at ⑦, and it shoots the ball successfully after all at ⑧.

5 Conclusion

This paper presented a learning method to acquire team strategy from a viewpoint of coach who can change a combination of players each of which has a fixed policy. Through the learning process the coach gradually obtained the knowledge about the opponent strategy and which strategy to take in order to beat the opponent. The part of the results is applied to the RoboCup2000 middle size league matches and obtained many goals (Top of the preliminary games). As future works, we will make the match settings for the experiments be a real RoboCup middle size scenario, and investigate the theoretical formulation of our approach.

References

[1] Michael Wunstel, Daniel Polani, Thomas Uthmann, and Jurgen Perl. Behavior classification with self-organization maps. In *The Fourth International Workshop on RoboCup*, pages 179–188, 2000.

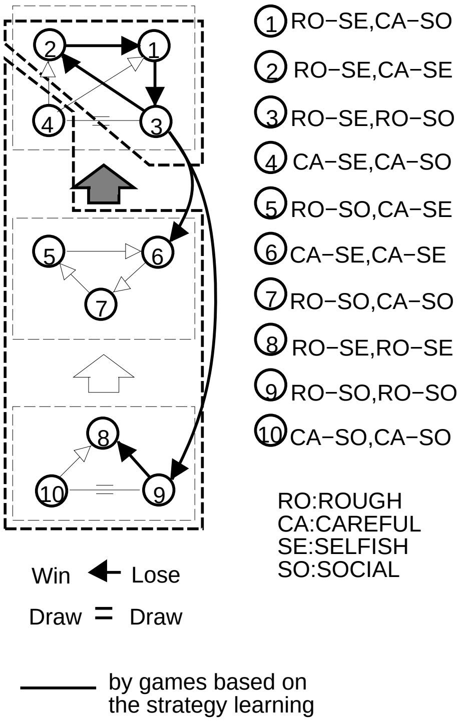


Fig. 3. The learned relationship among strategies

[2] Peter Stone and Manuela Veloso. Layerd learning and flexible teamworks in robocup simulation agents. In *Robocup-99: Robot Soccer World Cup III*, pages 495–508, 1999.

[3] Claudio Castelpietra, Luca Iocchi, Daniele Nardi, Maurizio Piaggio, Alessandro Scalso, and Antonio Sgorbissa. Communication and coordination among heterogeneous mid-size players: Art99. In *The Fourth International Workshop on RoboCup*, pages 149–158, 2000.

[4] Eiji Uchibe, Tatsunori Kato, Minoru Asada, and Koh Hosoda. Dynamic Task Assignment in a Multiagent/Multitask Environment based on Module Conflict Resolution. In *Proc. of IEEE International Conference on Robotics and Automation*, 2001 (to appear).

[5] Barry Werger. Cooperation without deliberation: Minimalism, stateless, and tolerance in multi-robot teams. *Artificial Intelligence*, 77:293–320, 1999.

[6] H. A. Simon. *The sciences of the artificial.* MIT Press, 1969.

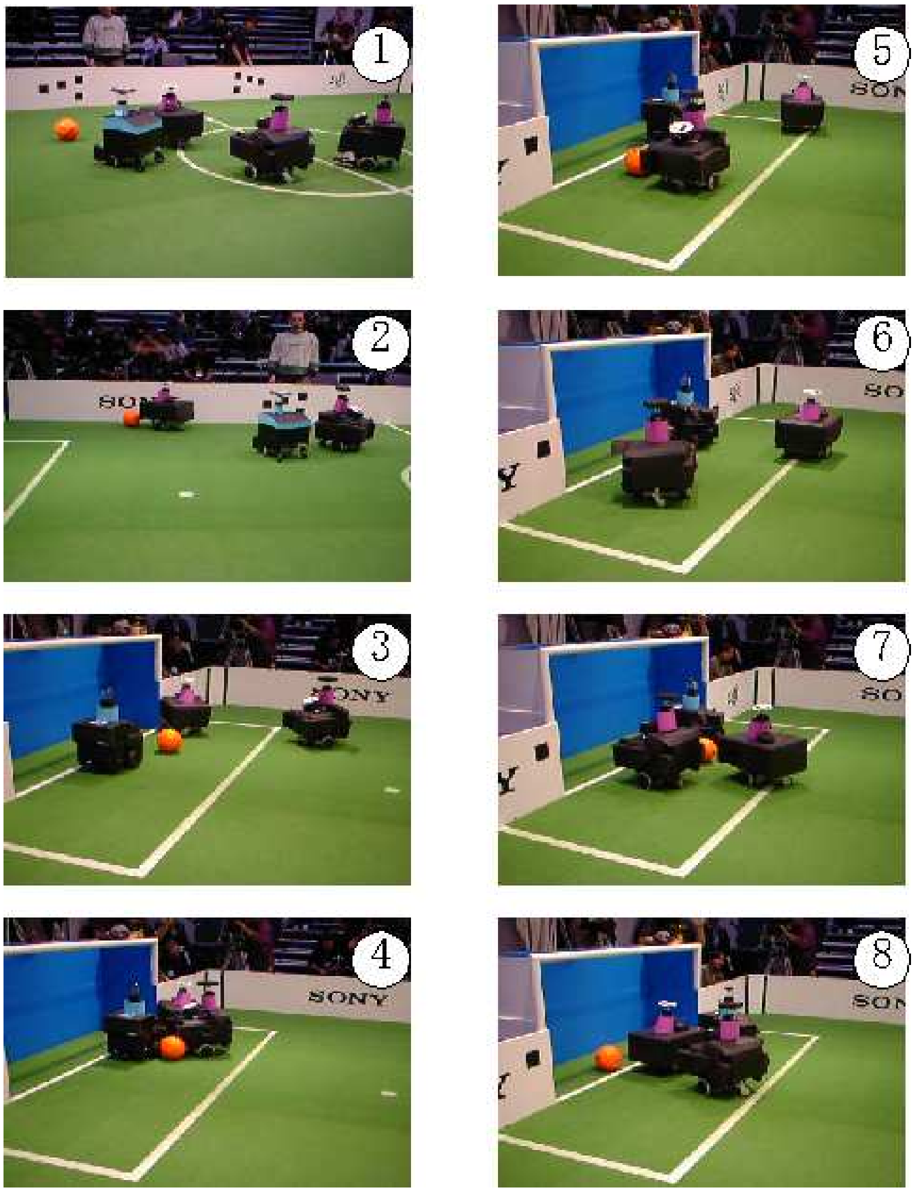


Fig. 4. A sequence of a failure recovery behavior among two robots