Multi Robots Cooperative Based on Action Selection Level*

CHU Hai-tao, HONG Bing-rong

(*Department of Computer Science and Engineering, Harbin Institute of Technology, Harbin* 150001, *China*)

E-mail: chuhaitao@yahoo.com

Received September 10, 2001; accepted May 9, 2002

Abstract: In a multi robots environment, the overlap of actions selected by each robot makes the acquisition of cooperation behaviors less efficient. In this paper an approach is proposed to determine the action selection priority level based on which the cooperative behaviors can be readily controlled. First, eight levels are defined for the action selection priority, which can be correspondingly mapped to eight subspaces of actions. Second, using the local potential field method, the action selection priority level for each robot is calculated and thus its action subspace is obtained. Then, Reinforcement learning (RL) is utilized to choose a proper action for each robot in its action subspace. Finally, the proposed method has been implemented in a soccer game and the high efficiency of the proposed scheme was verified by the result of both the computer simulation and the real experiments.

Key words: multi-agent; reinforcement learning; cooperative; local potential field; action selection level

In multi-agent environment, the cooperation behaviors can usually be acquired by learning. However, conventional reinforcement learning algorithm which is well used for single agent case is not so efficient in the multi-agent case, as an environment including other learners might change randomly from the viewpoint of an individual learning agent[1]. Therefore, for multi-agent learning, it is very important for the agent to perceive the influence from both the other learners and the objectives.

Robot soccer is a good domain for researchers to study the multi-agent cooperation problem. Under the robot soccer simulation environment, Stone and Veloso[2] proposed a layered learning method that consists of two levels of learned behaviors; Uchibe[3] et al proposed a scheme in which the relationship between a learner's behaviors and those of other robots is estimated based on the method of system identification and the cooperative behaviors is acquired by reinforcement learning. These methods have developed efficient action selections of individual robot. However, all the methods mentioned above failed to consider the fact that the action selected by each robot may be unnecessarily overlapped.

In this paper, a method based on potential function is proposed to deal with the overlap of action selection. The concept named Action Selection Priority Level (ASPL) is firstly defined in this method. Then, eight ASPLs are introduced to indicate the action of different purpose. That is, the action space is divided into eight subspaces corresponding to each ASPL. The ASPL, as well as the action subspace for each robot, can be determined by the local potential field method where more factors relating to the decision of a robot are considered. Moreover, we imported the ASPL model into the conventional RL algorithm with which the proper action for each robot is chosen. In addition, it should be noted that the creation of the ASPL model enables the robot to search proper actions among

* Supported by the National Natural Science Foundation of China under Grant No.69985002 (国家自然科学基金)

CHU Hai-tao was born in 1972. He is a Ph.D. candidate at the Department of Computer Science and Technology, Harbin Institute of Technology. His research interests are multi-agent systems, and machine learning. **HONG Bing-rong** was born in 1937. He is a professor and doctoral supervisor at the Computer Application, Harbin Institute of Technology. His current research areas include intelligence robot, robot soccer, and multi-agent systems.

the action subspaces instead of the whole action space, which greatly decrease the time for learning. We have implemented the proposed method for both the computer simulation and the real experiment. By integrating this method into our robot soccer system, we've achieved the first place in the 1st Robot Soccer Competition, China, 2000.

1 Architecture of the Robot Soccer System

In multi robots environment, no learner's input can tell it everything about its environment including other changing learners[4]. In this case, there is always a conflict between the global cooperation and the local independence. In order to solve this problem, we developed a robot soccer system based on the ASPL model in which the global action space is partitioned into eight subspaces.

1.1 Learning architecture

Figure 1 shows the learning architecture for each robot in our robot soccer system.

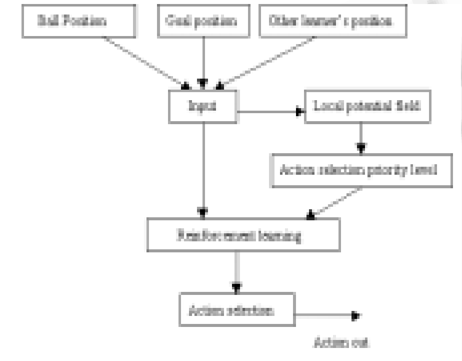

Fig.1 Learning architecture of each robot

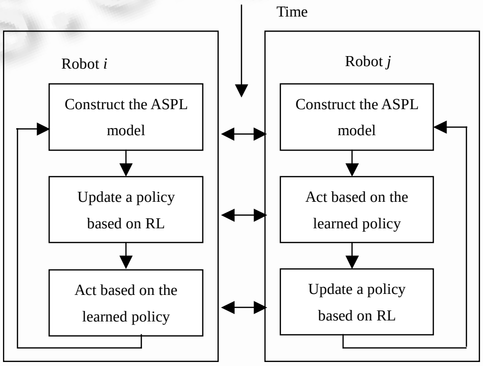
Time
Robot *i*
Construct the ASPL model
Update a policy based on RL
Act based on the learned policy
Robot *j*
Construct the ASPL model
Act based on the learned policy
Update a policy based on RL

Fig.2 Learning process for the robot soccer

In most multi-agent system, the input to the agent consists of the final goal and the goals of subtasks. For the robot soccer competition, the final goal of the agent team is to kick the ball into the opponent team's goal. So the position of the ball and the goal of the opponent team are two key input vectors of each learning robot. In addition, the state of other learners within each learner's view range should also be considered. Therefore, in our learning architecture, the learning robot firstly analyzes its surrounding environment based on the information gathered by the sensors. Secondly, based on those partial observations, each learning robot constructs its local potential field. Thirdly, according to the attractive potential and repulsive potential calculated, the ASPL for each robot is determined. Then, the action subspace determined corresponding to the ASPL model together with the state vector of each robot are passed to the RL algorithm. Finally, the action chosen by the RL algorithm combined with ASPL model is performed which can show good coordinated relations among the learning robots.

1.2 Learning sequence

Under the soccer game environment, the action space achieved by each robot as certain time may be different, so it is reasonable to expect every robot to act with a suitable strategy. However, if multiply robots learn together, the feedbacks of the environment would be confusing, even be conflicting. In traditional cooperative leaning system of multi robots, the policy-sharing method is always utilized to achieving the multi robots parallel learning. Due to the same strategy used by multi robots, the feedback of the environment at certain time can be easily added together to the corresponding pair of station/action. However, the randomly chosen of the action for each robot in the early stage of the learning process will decrease the efficiency of the cooperative leaning system[5]. Therefore, in our robot soccer system, the behaviors of the robots are obtained by learning instead of random generation. In detail, each robot constructs its local potential field first. Then one robot is selected to learn while keeping the actions of other robots as same as before. So in the first learning cycle, only the selected robot is acting while others are fixed. In other words, only the selected robot executes the action calculated by the RL algorithm and is in the state of learning phase. All the other robots take actions acquired previously and are in the state of non-learning phase. After the selected robot finishes the learning phase, another robot is chosen to learning. The process is repeated until all robots finished learning. Figure 2 shows the learning process.

2 RL algorithm Combined with the ASPL Model

Using conventional reinforcement learning method, a learner has to try every action in the action space during the learning phase before an optimal action is obtained. If the action space is large enough and there are many learners in the system, it will take tediously long time for the robots to learn. In addition, in a multi-agent system, the optimal action found by RL method for each robot doesn't mean the optimal cooperative behaviors among robots, i.e. the cooperative behaviors can not be acquired efficiently by the RL algorithm alone[6]. In the soccer robot field, one of the most typical cooperative learning method is the TPOT-RL learning algorithm proposed by Peter Stone[2]. Basically saying, the TPOT-RL is an improvement reinforcement learning method in which the Q-space is decreased by both the partition of the state space of the robots and the temporal reward. However, with the increase of the action space, the Q-space in TPOT-RL will be increased/boom in exponent levels[6]. Due to the problems mentioned above, we propose the ASPL model.

2.1 ASPL model

In this part, the ASPL model is proposed based on the local potential field method.

First, we introduce some parameters. Suppose the mass of each robot is 1 and the mass of the ball and goal is 2 each. The position of the robots and ball is w.r.t the world coordinate.

· P: the position of the learner.

· n: the number of opponents within the view of the learner.

· Q_i: the position of each opponent, i=1…n.

· m: the number of teammates within the view of the learner.

· T_j: the position of each teammate, j=1…m.

· B: the position of the ball.

· G: the position of the opponent's goal.

· $RP_{Q_i} = 1/|P-Q_i|$: repulsion from opponents.

· $RP_{T_j} = 0.5/|P-T_j|$: repulsion from teammates.

· $AR_B = 1\times 2/|P-B|$: attraction from the ball.

· $AR_G = 1\times 2/|P-G|$: attraction from the goal.

with the parameters introduced and potentials calculated, we define rule ε to determine the action priority level. The rule ε can be determined according to the environment.

The ASPL for each robot at time t can be calculated as follows:

$$ASPL(t) = \varepsilon(RP_{Q_i}(t), RP_{T_j}(t), AR_B(t), AR_G(t)) . \quad (1)$$

In accordance with ASPL model, we partition the action space into eight subspaces, as shown in Table 1.

Table 1 Definition of the action space

Level	Definition	Level	Definition	Level	Definition
Level 1	Pass-ball-to-Teammate	Level 2	Kick-Ball-to-Goal.	Level 3	Dribble-Ball
Level 4	Kick-Away	Level 5	Dash-to-Ball	Level 6	Dash-to-Goal
Level 7	Avoid-Opponent	Level 8	Search-Ball and Search-Goal		

For each action subspace, there are two parameter spaces, i.e. the direction parameter and force parameter. We represent the two parameter spaces by discretization. *DP* and *FP* denotes the direction and force set respectively.

$$DP=\{-\pi/2, -\pi/4, 0, \pi/2, \pi/4\} \quad FP=\{0, F_{\max}/4, F_{\max}/2, 3*F_{\max}/4, F_{\max}\}$$

where $F_{\max}$ denotes maximum force used by the robot when executing the action. Each action chosen by the ASPL model needs proper *DF* and *FP* parameters chosen by the RL algorithm described in the next section.

2.2 *Q*-Learning algorithm combined with the ASPL model

Denote the input vector at time t for robot p by $\vec{i}(p,t)$, the action subspace chosen by ASPL model by u_{aspl}, the start time of a game by t^s and the end time by t^e. Then the *Q*-value of the selected action u_{aspl} for the given input $\vec{i}(p,t)$ can approximate as follows:

$$Q(\vec{i}(p,t),u_{\text{aspl}}(DP(d),FP(f))\cong\varepsilon(\gamma^{t^e-t}R(t^e)), \tag{2}$$

where ε is the expectation operator. $R(t^e)$ denotes the reinforcement value obtained at the end of each trial and it is equal –1 if the opponent team scores and 1 if the agent team scores and 0 otherwise. The discount factor γ is chose between 0 and 1. *DP* and *FP*, the direction and force sets, has been defined in Section 3.1, each of which consists of 5 elements. *d* and *f* denote the position of each element in the two sets, so $d\in[0,4]$ and $f\in[0,4]$. In other words, there will be 25 possible combine action of the elements in the two sets. The *Q*-learning algorithm is to find a proper combination in the subspace determined for each robot.

To calculate the *Q*-value at each step for a robot, we establish a record of history list with maximum size $H_{\max}$ for each player. We set $H_{\max}$=100 in the simulation. At the trial end, the history list *H*(*p*) of player *p* is

$$H(p)=\{\{\vec{i}(p,t^s),u_{\text{aspl}}(p,t^s),v(\vec{i}(p,t^s))\},...,\{\vec{i}(p,t^e),u_{\text{aspl}}(p,t^e),v(\vec{i}(p,t^e))\}\}, \tag{3}$$

here t^s denote the start of the history list. $v(\vec{i}(p,t))=\max_k(Q(\vec{i}(p,t),u_k(DP(d),FP(f)))$. If $t^e>H_{\max}$, $t^s=t^e-H_{\max}+1$. At the end of the each trial, we calculate the *Q*-value according to the history lists of each player.

$$Q^{new}=R(t^e), \tag{4}$$

$$Q^{\text{new}}(p,t)=\gamma(\lambda Q^{\text{new}}(p,t+1)+(1-\lambda)v(\vec{i}(p,t+1))), \tag{5}$$

here $t^s\le t\le t^e$, λ represents the influence of the learner's future experience to the present *Q*-value and we set it 0.9. Figure 3 shows the frame of *Q*-learning algorithm combined with the ASPL model.

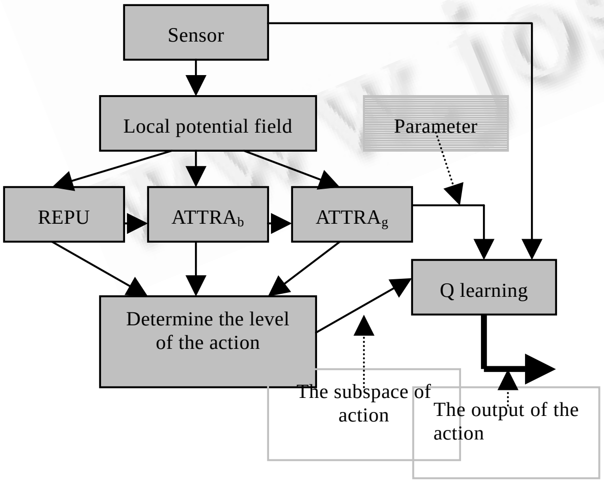

Fig.3 Architecture of combine ASPL and *Q*-learning

Using the algorithm above, the *Q*-value for each action can be obtained. We use Boltzman Gibbs distribution at temperature *T* to select action and T increase linearly from 0-60.

$$\frac{e^{Q(x,u)}/T}{\sum_{u\in U} e^{Q(x,u)}/T}. \qquad (6)$$

3 Result of Simulation and Experiment

3.1 Computer simulation

We played 1000 games, each of which is of the time period 3000. Every 20 games we sum the score. Each game consists of separate trials. The trials stop once one of the teams score or the game runs out of time. We utilize the Robocup Soccer Server[7] to conduct the simulation. The team size is 1, 3 and 11. One team adopted the Q-learning algorithm and the other adopted Q-learning algorithm combined with the ASPL model. Figures 4~6 show the performance of the two algorithms when the team size is 1, 3 and 11. As shown in the figure 4, when the team size is 1, the scores of the two algorithms are nearly same. So for the single robot game, there is no cooperation involved. However, when the scores reach the maximum, the average amount of the game number of the first method and the second method is 600 and 550 separately. That is, the learning speed is faster with second method. When the team size increase to 11, our algorithm shows obvious advantages over the pure Q-learning method with the great increase of the scores.

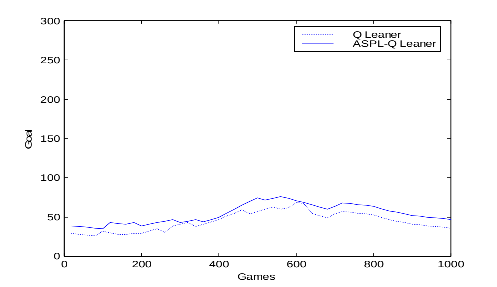

Fig.4 Comparison of the two algorithms with team size 1

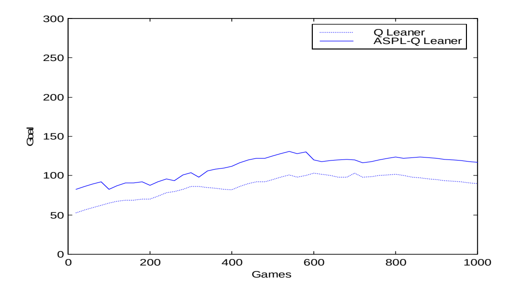

Fig.5 Comparison of the two algorithms with team size 3

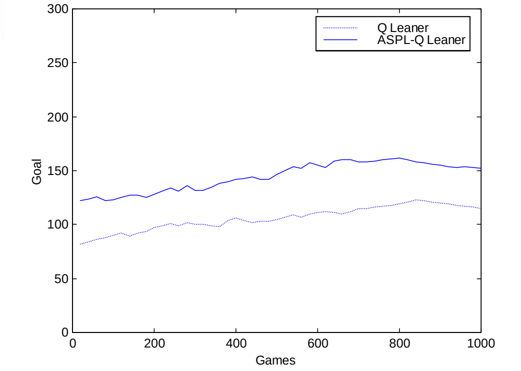

Fig.6 Comparison of the two algorithms with team size is 11

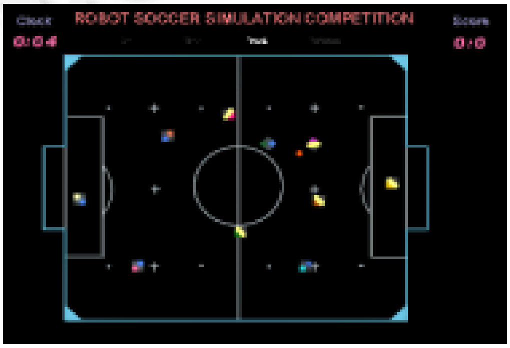

Fig.7 Microbsoccer simulation environment

3.2 Real robot competition

Besides the computer simulation, we also apply the proposed method to the real robot soccer competition with team size 5. To simulate the real competition, we first developed a simulation environment according to the size of the real robot and ball as well as the ground. In addition, the dynamic property of the robot was also considered. The Microbsoccer simulation environment[8](Fig.7) was established using Visual C++ and DirectX. Even though there is some difference between the simulation and real competition, it is reasonable to use the training result obtained from the simulation as the initial date of the real competition. As a result, the learning speed of the real robots is greatly improved. The statistical data of the rate of success of shooting, the rate of success of passing and the number of collisions were shown in the Table 2.

Table 2 Statistical data of the robot soccer competition

Test item	*Q*-Learni*ng*	*Q*-Learning + ASPL
Success of shooting	63/100	82/100
Success of passing	26/100	53/100
Number of collisions	33/100	16/100
Number of trials	100	

4 Conclusion

This paper proposed an approach to acquire cooperative behaviors in a multi-robot environment. Based on the local potential field method, the ASPL model corresponding to eight action subspaces was developed and the rules for determining the ASPL as well as subspace for each robot was explained. The RL algorithm combined with the ASPL model is employed to choose the proper direction and the force parameters for the action are selected. Finally, the performance of both the computer simulation and the real competition shows that the cooperation behavior can be obtained efficiently and the speed of learning can also be improved.

References:

[1] Kitano, H., Asada, M., Kuniyoshi, Y., *et al*. A challenge problem. AI Magazine, 1997,18(1):73~85.

[2] Stone, P., Veloso, M. Using machine learning in the soccer server. In: Proceedings of the IROS Workshop on Robocup. Osaka, Japan, 1996. 105~203.

[3] Uchibe, E., Asada, M., Hosoda, K. State space construction for behavior acquisition in multi-agent environments with vision and action. In: Proceedings of the International Conference on Computer Vision. 1998. 870~875.

[4] Salustowicz, R.P., Wiering, M.A., Schmidhuber, J. Learning team strategies: soccer case studies. Machine Learning, 1998,33: 263~282.

[5] Sandholm, T.W., Crites, R.H. On multiagent *Q*-learning in a semi-competition domain. In: Weib, G., Sen, S., eds. Adaptation and Learning in Multiagent Systems. Berlin: Springer-Verlag, 1996. 188~190.

[6] Stone, P., Veloso, M. Team-Partitioned, opaque-transition reinforcement learning. In: Asada, M., Kitano, H., eds, Robocup-98: Robot Soccer World CupII. Berlin: Springer Verlag, 1999.

[7] Kim, Do-Yoon, Chung, Myung Jin. Path planning for multi-mobil robots in the dynamic environment. In: Proceedings of the Micro-Robot World Cup Soccer Tournament. 1996. 127~132.

[8] Noda, I., Matsubara, H., Hiraki, K., *et al*. Soccer server: a tool for research on multiagent systems. Applied Artificial Intelligence, 1998,12:25~27.

基于动作选择级的多机器人协作

褚海涛, 洪炳熔

(哈尔滨工业大学 计算机科学与工程系,黑龙江 哈尔滨 150001)

摘要: 在多机器人环境中,由于每个机器人动作选择的重叠现象,让机器人之间的协作变得很差.提出了一个方法用于确定动作选择级别.在此基础上,可以很好地控制多机器人的协作行为的获取.首先,定义了用于动作选择级优先级的 8 个级别,这 8 个级别相应的映射到 8 个动作子空间.然后,利用局部势场法,每个机器人的动作选择优先级被计算出来,并且因此,每个机器人获得了各自需要搜索的动作子空间.在动作子空间中,每个机器人利用加强学习方法来选择一个适当的动作.最终,把该方法用于机器人足球比赛的机器人局部协作训练中.试验的效果在仿真和实际比赛中得到了证实.

关键词: 多智能体;加强学习;协作;局部势场;动作选择级

中图法分类号: TP242 文献标识码: A