Persepolis Soccer 2D Simulation Team Description Paper 2022

Morteza Noohpisheh[1], Maryam Shekarriz[1], and Fatemeh Zaremehrjardi[2]
Monireh Karimi[3]

Dept. of Computer Engineering, Shiraz University, Iran
Dept. of Computer Science, St. Francis Xavier University
Dept. of Electerical Engineering, Kashan University
Morteza.Noohpisheh@gmail.com
m.shekarriz31@gmail.com
X2020fpvv@stfx.ca
Monir.Karimi.1373@gmail.com

Abstract. This paper includes some explanations about algorithms implemented by Persepolis team members. We will introduce algorithms that are used for pass, shoot, dribbling and marking, and in particular with the ranking algorithm for the states of chain action. The base code used by Persepolis is agent-2d 3.1.1[15].

Keywords: RoboCup · Soccer Simulation · AI.

1 Introduction

The goal is to implement intelligent agents, that have the ability to decide on using soccer individual skills as well as the ability to work with other agents in the soccer field. In addition, programming has been done by agents to improve individual skills (shoot, pass, dribble, positions of players on the field). Razi team is formed from students, has started its activities from November 2012 and has made plan in order to achieve its goal, participating in World Cup competitions. This team has participated in prestigious competitions and has won the third place in IranOpen2017 and IranOpen2018. This team has begun to use artificial intelligence in its decision making. It continues its work with the aim of using artificial intelligence algorithms and online analysis.

Note : The name of this team was changed from Razi to Persepolis after the Japan Open 2020.

2 Activities Done in Other Iranian Teams

In the recent years, the following tasks have been done in other Iranian teams: A formation 2 detection system as well as a software called Tournament Planning and Analyzing Software(TPAS) have been developed by Namira[1][2][3]. Dynamic formation changing, field evaluation system for offense, field evaluation system optimization, Optimizing defensive decision making by using message passing between agents, Shooting behavior optimization using Deep Neural Network, Predicting opponent's behavior using Rough Neural Network, Predicting the behavior of the agents with ball possession, Defensive Decision Making using Deep Reinforcement Learning ,Pass prediction using Deep Neural Network,Pass Prediction have been developed by Cyrus[5-12]. Nexus has implemented Reinforcement Learning for decision making in penalty area[13].

3 Previous Works in Team

2018: Razi has organized its own system for rating states in chain action[16].
2019: Use the Gravity Strategy[17].

4 Pass Decision Making

This DM gets the ball position and the pass receiver as input parameters. to know where to use which type we implemented a fuzzy algorithm[19]. This fuzzy algorithm according to the ball position and the player situation tells us which type is more proper.
Fuzzy rule base for pass is as follows:
1. If (position is Danger) then (pass type is Secure).
2. If (position is Safe) and (situation is Bad) then (pass type is Secure).
3. If (position is Safe) and (situation is Normal) then (pass type is Normal).
4. If (position is Safe) and (situation is Good) then (pass type is Risky).
5. If (position is Risk) and (situation is Bad) then (pass type is Secure).
6. If (position is Risk) and (situation is Normal) then (pass type is Normal).
7. If (position is Risk) and (situation is Good) then (pass type is Risky).

After the pass type and Max Pass risk became determined, agent checks his ability for performing this type of pass. First the agent checks if he can shoot to the pass position or not then the amount of risk will be calculated. The following factors have to be checked in order to calculate the risk of pass:
1. Opponents around the ball
2. The relative agent position with ball.
3. The max risk for getting position behind the ball
4. The position of opponents around the player we want to pass to, and their distances with him.
5. The ball path
Let t1 be the time takes player to get position behind ball. It depends on factors 1,2, and 3 which vary with pass type, and t2 be the time takes for opponents to reach the ball, the player can pass when t1 is less than t2. Let t(i) be the time takes for opponent i to intercept the ball at position p(i) with error E1, and t1(i) is the time takes for ball to reach p(i) with error E2, where E1 and E2 vary

with pass type. If t1(i) is less than t(i) for all i opponents then player can pass. And the last factor is when the target player, gets the ball. He must have Secure Time to control the ball, which varies with pass type. Now with the amount of risk, the player can determine whether he can pass with requested type or not.

5 Offensive Strategy

In the 2D soccer simulation league, the field state has many properties describing offence and defense, such as the relative positions, the distance from the ball, the members in a certain region and so on. It is easy to see that the field state is a continuous space. To deal with the state data quickly and effectively, we need to divide them into many equivalent states by a process of discretization. In the course of a soccer game, the attacking strategy varies with the state. That is to say, attacking strategy is a piecewise function: the best attack action may be different if the field state changes. Furthermore, we must identify the best action for each state. In this strategy we set a path for each action in each state, and found the best action in a given state via the preferences of the paths. We should regard each attack action as a path in the solution space, consider each agent's training in the same way we would foraging behavior of an ant, and treat an effective attack as success foraging. In order to solve the problem of "a solution per state", we set an ant group for each equivalent state. Within this setting, each ant lays down a trail pheromones on a path (corresponding to the preference value of an attack action), and each ant group may have a path that has the highest pheromone level (corresponding to the best preference value and the best attack action).
We can simulate different attacking environment by setting agents at different points on the field. Regarding three attack actions (shooting, passing and dribbling) as three different paths for foraging, we can obtain the best attack action by using the preference values of ants' foraging behaviors. The following two steps should be made in order to find the best attack action: 1) train the preference values of actions in each environment, and 2) select the best attack action based on preference value. The first step runs offline in the background for a long time to obtain the preference value; the second step is designed using the decision module of the attacker and runs online to select the best action. The state of attacking environment is a continuous space. In order to deal with the state data quickly and effectively, we should transfer them into many discrete equivalent states. For every attacking environment (state), we generated a group of ants and used pheromone concentration to record the dynamic success-rate of each attack action.

6 Experimental evaluation

After being created, the 2D soccer simulation team with the MACO[14] was run in the RoboCup simulation platform to evaluate our new algorithm. Firstly, every agent took random attack actions. The results of these actions have been recorded and used for the next training session. The underlying code and underlying database are Agent2d-3.1.1 and Librcsc, respectively. The positions of all players (two attackers, two defenders and one goalkeeper) were set randomly by an offline coach. After five cycles of competition, the results of two teams are recorded as the format of "αshoot, dshoot, αpass, dintp, ddrib, dintd, aa, suc". The training will stop after 150,000 times. The training scene is shown in Fig. 1. In order to show the situations of the attack and defense sides more clearly, only the penalty area is depicted.

Fig. 1. Screenshot of training scene arranged randomly.

In the following experiments, the maximum value for each state data (αshoot, dshoot, αpass, dintp, ddrib, dintd) was defined as ($\pi/2$, 15, π /2, 15, 3, 3), respectively, with each state discretized into 30 elements. If state data are larger than the defined maximum, the discretized data will be the maximum (e.g., if the value of αshoot is equal to $2\pi/3$, then the discretized value of αshoot will be 30). The number of states was 2,700 (3*302), the number of groups was 2,700 and the number of ants was 8,100 in total. The adjustment factors for pheromones and heuristics were set as follows: α=2, β=4. In order to keep the previous experiences obtained during the pre-event training, we set the coefficient of pheromone evaporation at a low level (ρ=0.1).

Application effect in competition is the sole criterion for evaluating an offensive strategy. This section concerns our creation of a 2D soccer team named Persepolis, which employs the MACO algorithm as the offensive strategy. Meanwhile, using the underlying code Agent2d-3.1.1, which is an open source by Helios (former world champion in the RoboCup 2D simulation league), we created another team, which adopted the searching-space offensive strategy. The only difference between the Persepolis team and the BASE team is the attacking selection strategy. There were 10 games in total in the test, with each game running 6,000 standard clock cycles. The results of the games are presented below: As listed in Table 1, the Persepolis team dominated in the games with 53.4% possession and 82.61% passing-success on average. As a result, the Persepolis team enjoyed a 100% winning advantage with 2.9 goal differences per game, as well as scoring three times as many goals as its opponent on average.

Games	Possession	Passing Success	Goal Scored	Goal Against	Result	Goal Difference
1	53.2%	80.1%	5	1	Win	4
2	54.1%	82.8%	4	1	Win	3
3	52.8%	81.7%	6	2	Win	4
4	53.9%	83.1%	4	2	Win	2
5	54.2%	82.6%	3	1	Win	2
6	51.7%	82.1%	5	1	Win	4
7	53.4%	83.2%	4	2	Win	2
8	52.4%	84.2%	4	2	Win	2
9	54.6%	83.6%	6	3	Win	3
10	53.7%	82.7%	5	2	Win	3

Table 1. Result of Persepolis vs Base Team games

7 Conclusions

The offensive strategy is an important factor in the attacking capability of a soccer team. Given the unpredictability of the environment and the influence amongst agents, it is very difficult to select action for the ball-handler. Attacking strategy has becomes a hotly discussed issue in the domain of multi-agent cooperation in a real-time, asynchronous and noisy environment. We analyzed the principal factors of the main attack actions for the ball-handler in the 2D simulation league and presented a novel offensive strategy algorithm based on multi-group ant colony optimization (MACO-OS)[18] in order to improve attacking efficiency. Using the pheromone evaporation mechanism, the algorithm creates an AIT, which is the basis of decision-making action. Our future work will focus on defensive strategy and defense conversion strategy to enhance the team's overall performance.

8 Upcoming Work

What we are trying to do, and to some extent succeed, is Defensive Decision Making using Deep Reinforcement Learning, which Cyrus Team's 2019 article has been a great help to us in further understanding and finding this path.[4]

References

1. E. Asali, F. Negahbani, S. Tafazzol, M. S. Maghareh, S. Bahmeie, S. Barazandeh, S. Mirian and M. Moshkelgosha, "Namira Soccer 2D Simulation Team Description Paper 2018," in RoboCup 2018, Monreal, Canada, 2018.
2. E. Asali, A. Moravej, S. Akbarpoor, O. Asali, M. Katebzadeh, S. Tafazol, F. Negahbani, M. Valipour, S. Mirian, H. Mehran, A. Bitaraf Haghighi and Y. Khoshghadam Borazjani, "Persian Gulf Soccer 2D Simulation Team Description Paper 2017," in RoboCup 2017, Nagoya, Japan, 2017.
3. E. Asali, M. Valipour, N. Zare, A. Afshar, M. Katebzadeh and G. Dastghaibyfard, "Using Machine Learning approaches to detect opponent formation," Artificial Intelligence and Robotics (IRANOPEN), pp. 140-144, 2016.

4. Zare, N., Sarvmaili, M., Mehrabian, O., Nikanjam, A., Khasteh, S.-H., Sayareh, A., Amini, O., Barahimi, B., Majidi, A., Mostajeran, A.: Cyrus 2D Simulation 2019. In:RoboCup(2019).
5. Zare, N., Sayareh, A., Sarvmaili, M., Amini, O., Soares, A., Matwin, S.: CYRUS 2D Soccer Simulation Team Description Paper 2021. In: RoboCup 2021 Symposium and Competitions, Worldwide (2021)
6. Zare, N., Sayareh, A., Sarvmaili, M., Amini, O., Matwin, S., Soares, A.: Engineering Features to Improve Pass Prediction in 2D Soccer Simulation Games. In: RoboCup 2021: Robot World Cup XXIV, Springer (2021)
7. Zare, N., Amini, O., Sayareh, A., Sarvmaili, M., Firouzkouhi, A., Soares, A., Matwin, S.: Improving Dribbling, Passing, and Marking Actions in Soccer Simulation 2D Games Using Machine Learning. In: RoboCup 2021: Robot World Cup XXIV, Springer (2021)
8. N. Zare, M. Sadeghipour, A. Keshavarzi, M. Sarvmaili, A. Nikanjam, R. Aghayari, A. Firouzkoohi, M. Abolnejad, S. Elahimanesh and A. Akhgari, "Cyrus 2D Simulation Team Description Paper 2018," in RoboCup 2018, Montreal, Canada, 2018.
9. N. Zare, A. Najimi, M. Sarvmaili, A. Akbarpour, M. NaghipourFar, B. Barahimi and A. Nikanjam, "CYRUS 2D Simulation Team Description Paper 2017," in 21th annual RoboCup International Symposium, Nagoya, Japan, 2017.
10. R. Khayami, N. Zare, M. Karimi, P. Mahor, A. Afshar, M. S. Najafi, M. Asadi, F. Tekrar, E. Asali and A. Keshavarzi, "CYRUS 2D simulation team description paper 2014," in RoboCup 2014 Symposium and Competitions: Team description papers, Joao Pessoa, 2014.
11. R. Khayami, N. Zare, H. Zolanvar, M. Karimi, P. Mahoor, F. Tekrar, E. Asali and M. Fatehi, "Cyrus Soccer 2D Sumilation Team Description Paper," in 17th annual RoboCup International Symposium, Eindhoven, Netherlands, 2013.
12. M. Hausknecht and P. Stone, "Deep Reinforcement Learning in Parameterized Action Space," in International Conference on Learning Representations (ICLR), San Juan, Puerto Rico, 2016.
13. M. Abolfazli Esfahani, M. Ghafouri, M. Jamili, S. Askari, R. Etemadi, H. Mohebbi, M. Dara, M. Erafani and M. Malekabadi, "Nexus2D Team Description Paper," in RoboCup 2017, Nagoya, Japan, 2017.
14. Chen, S., Lv, G., Wang, X. (2016). Offensive Strategy in the 2D Soccer Simulation League Using Multi-Group Ant Colony Optimization. International Journal of Advanced Robotic Systems, 13.
15. Akiyama H, Nakashima T. Helios base: An open source package for the robocup soccer 2d simulation. InRobot Soccer World Cup 2013 Jun 24 (pp. 528-535). Springer, Berlin, Heidelberg.
16. M. Noohpisheh, M. Shekarriz, S. Barzegar, D. Borzoo, A. Kariminia, "Razi Soccer 2D Simulation Team Description Paper 2018," in RoboCup 2018, Montreal, Canada, 2018.
17. Noohpisheh M, Shekarriz M, Bordbar A, Liaghat M, Salimi A, Borzoo D, Zarei A. M. Noohpisheh, M. Shekarriz, A. Bordbar, D. Borzoo, M. Liaghat, A. Salimi, A. Zarei, "Razi Soccer 2D Simulation Team Description Paper 2019," in RoboCup 2019, Sydney, Australia, 2019.
18. Wei-guo Zhang, Tian-yu Lu, The Research of Genetic Ant Colony Algorithm and Its Application, Procedia Engineering, Volume 37, 2012
19. Y. Dote, "Introduction to fuzzy logic," Proceedings of IECON '95 - 21st Annual Conference on IEEE Industrial Electronics, 1995, pp. 50-56 vol.1