Robo2D Soccer Simulation Team Description Paper

Matin Nouri[1], Alireza Hasankhani[2]
[1]RSQ LC Institute, Tehran, Iran
{matin.no.al , alirezahasankhani86}@gmail.com

Abstract. This article contains descriptions of the activities of the Robo2D soccer 2D simulation team. This year, our team tried to increase the accuracy of both defense and offense, decision, managing and etc. strategies. So, we developed our ideas by designing and developing algorithms. Here are some of these algorithms and solutions.

Keywords: RoboCup. Soccer 2D simulation. Conditional decision. Machine Learning

1 Introduction

Robo2D soccer simulation team was founded 1 year ago. We started coding on Cyrus Base[1] and developed the base code. We tried to use machine learning in our team with using Keras but it doesn't helpful for us. So, we used Cyrus Base defaults.

1.1 Related Work

Our team used some wonderful and good ideas from the other teams in recent years. For example, we used an algorithm that like Cyrus 2014 Marking Algorithm that they published on their GitHub [1] and Hades Marking [2]. Also, we used Hades2D Shoot Action [3]. We used Formation Detector of Cyrus Team [4], Glider2D[5] and Fractals[6] Stamina managing and HillStone 2019 [7] Formation Type.

2 Strategies

In our experiences we know that players cannot find their position with thinking and we have to choose a rule for each player so we use Helios Base [8] Defaults and formation editor(fedit2) [9] source for make formation files with a bit change like HillStone 2019 [7]. Helios Base Goalie wasn't good for catching balls. So, we write an algorithm for goalie to change his position by configures when ball is close to our Goal instead thinking about if ball is shot find intersection of ball and Goalie Y (Figure 1). And now for stamina managing we code like Glider v.1.1 in Glider2D[5] and Fractals 2019[6]

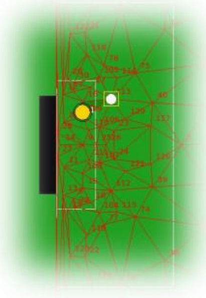

Fig. 1. Goalie Formation using Helios Base formation Editor [9]

[1] Cyrus Base , https://github.com/cyrus2d/cyrus2d_base.git

3 Decisions

For Decision we used Helios Base Defaults [3]. But in Jiling Base [4] we use configure files for decisions and it could be change in game cycles, so that is really good algorithm for machine learning and team experience. Also for extra Decisions we have written a library and use it in our teams.

4 Pass Strategy

As we mentioned above, we had a library for extra decisions and some important Decisions. Players think Sensitivity of Situation with many parameters and choose best pass type for Pass and set variables with low or high sensitivity, because Helios Base Defaults for Pass was focus on side attack and that was helpless.

5 Through Pass

We used a Voronoi Diagram algorithm such as Glider2D[10] for Through Pass Decision (Figure 2). It imagine lots of Area of Ground, If in sides of ground opponent player detected, player ignoring to pass to player who are in side of ground.

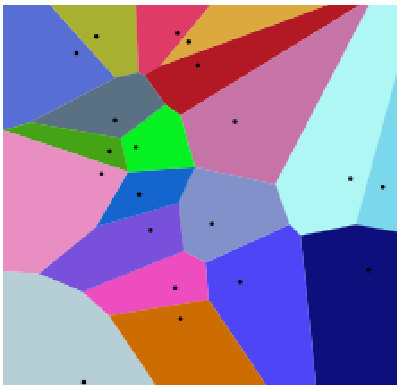

Fig. 2. Simple Voronoi Diagram

6 Dribble

Our team considered three types for Dribble. Every Player who can kicking will decide and choose one of these types. If opponent player blockade kicker it chooses small and dodge and if kicker has enough space for dribble, he will dribble without dodge in a longer distance than small Dribble type. We called this type Self-Pass. But if kicker cannot decide in these situations, he going to doing a normal Dribble.

Table 1. Dribble Types

Name	Dodge	Blockade Situation
Small	Yes	Yes
Self-Pass	No	No
Normal	-	-

[3] Helios Base Decisions, https://github.com/helios-base/helios-base/blob/master/src/bhv_basic_move.cpp,https://github.com/helios-base/helios-base/blob/master/src/bhv_basic_offensive_kick.cpp,https://github.com/helios-base/helios-base/blob/master/src/bhv_goalie_basic_move.cpp

[4] Jiling Base, https://github.com/RSQLCGROUP/Jiling-Base.git

7 AI Decision

For AI Decision we tried a lot and reach a good result. But it has many bugs with their weights so we used Cyrus Base Defaults for Unmark algorithm [1] using CppDNN [11] (Figure 3). Also, we tried to use Jiling Base DeciThink [5] for players experiences. That means player can gain experiences with each Game play. But it doesn't practical for competitions so we ignore that idea. Also, we tried to use Formation Detection that Cyrus2D 2017 [4] used, but regarding to the undesirable result we didn't use it in our team.

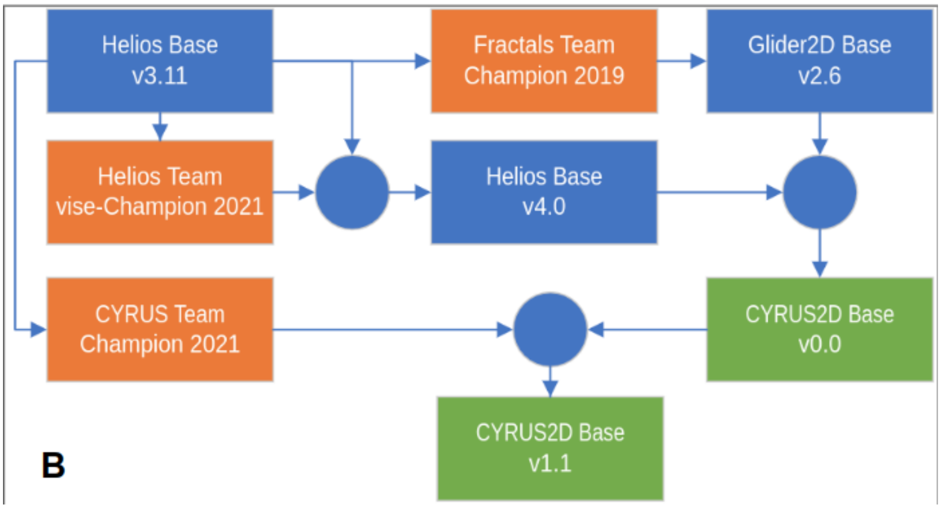

Fig. 3. Cyrus2D Base AI

8 Mark

Players Mark Opponent Player when it is in Error Handling situation. But sometimes for keeping Situation better than Marking Situation, they won't execute Mark Command. Also, we Mark our Goal that means when Opponent are close to our Goal Players will go to a good point by Strategy Positioning or sometimes in some data handling, they find a good position for Marking our Goal.

9 Block

In some situations, when we haven't ball owner, players will find opponents near to ball kicker and check them with their own algorithm and block pass path to them and try to give ball from ball kicker.

[5] DeciThink, https://github.com/RSQLCGROUP/Jiling-Base/tree/main/Agent/src/DeciThink

References

1. Nader Zare, Aref Sayareh, Mahtab Sarvmaili, Omid Amini, Amılcar Soares, and Stan Matwin," CYRUS Soccer Simulation 2D Team Description Paper 2022," In RoboCup 2022 Symposium and Competitions, WorldWide (2022).
2. Akhondi, F., Esmaelifar, S., Esmaelifar, S., Rokni, S. R., Rajabi, A, Hasanpour, G.: Hades2D soccer2D simulation Team Description paper. In: RoboCup 2022 Symposium and Competitions: Team Description Papers. Worldwide (2022).
3. Akhondi, F., Esmaelifar, S., Esmaelifar, S., Rokni, S. R., Rajabi, A, Hasanpour, G.: Hades2D soccer2D simulation Team Description paper. In: RoboCup 2021 Symposium and Competitions: Team Description Papers. Worldwide (2021).
4. Nader Zare , Ali Najimi , Mahtab Sarvmaili , Aryan Akbarpour , Mohsen NaghipourFar , Borna Barahimi , Amin Nikanjam," CYRUS Soccer Simulation 2D Team Description Paper 2017," In RoboCup 2017 Competitions, WorldWide (2017).
5. Prokopenko, M., Wang, P.: Gliders2d: Source Code Base for RoboCup 2D Soccer Simulation League. CoRR abs/1812.10202 (2018)
6. Mikhail Prokopenko and Peter Wang "Fractals2019: Guiding Self-Organisation of Intelligent Agents," In Robocup 2019
7. Taiga Ito, Junichiro Iseki, Norikazu Sato,Yuki Arimura, Norifumi Watanabe, Takashi Omori "HillStone 2019 Team Description Paper".: "We are using the fedit version 2-0.0.0 for the defensive formation development, and are creating an allocation of players for the fedit2. A sample of created allocation is shown in Fig.2 [1,2]." In Robocup 2019
8. Akiyama, H., Nakashima, T.: Helios base: An open source package for the robocup soccer 2d simulation. In Robot Soccer World Cup 2013 Jun 24 (pp. 528-535). Springer, Berlin, Heidelberg
9. fedit2-0.0.1, online, available at: https://osdn.net/projects/rctools/downloads/6853 1/fedit2-0.0.1.tar.gz/. Consulted on April 2021.
10. M. Prokopenko, P. Wang, and O. Obst," Gliders2015 Soccer Simulation 2D Team Description Paper 2015," In RoboCup 2015 Competitions, WorldWide (2015).
11. Nader Zare, Aref Sayareh, Mahtab Sarvmaili, Omid Amini, Amılcar Soares, and Stan Matwin," CYRUS Soccer Simulation 2D Team Description Paper 2021," In RoboCup 2021 Symposium and Competitions, WorldWide (2021).