

Observation Denoising in CYRUS Soccer Simulation 2D Team For RoboCup 2023

Aref Sayareh³, Nader Zare¹, Omid Amini¹, Arad Firouzkouhi⁴, Mahtab Sarvmaili¹, Stan Matwin^{1,2}, and Amilcar Soares³

¹ Institute for Big Data Analytics, Dalhousie University, Halifax, Canada

² Institute for Computer Science, Polish Academy of Sciences, Warsaw, Poland

³ Memorial University of Newfoundland, St. John's, Canada

⁴ Amirkabir University of Technology, Iran

nader@cyrus2d.com,

{omid.amini, mahtab.sarvmaili}@dal.ca,

arad.firouzkouhi@aut.ac.ir

stan@cs.dal.ca,

{asayareh, amilcarsj}@mun.ca

Abstract. The RoboCup competitions hold various leagues, and the Soccer Simulation 2D League is a major one among them. Soccer Simulation 2D (SS2D) match involves two teams, including 11 players and a coach, competing against each other. The players can only communicate with the Soccer Simulation Server during the game. This paper presents the latest research of the CYRUS soccer simulation 2D team, the champion of RoboCup 2021. We will explain our denoising idea powered by long short-term memory networks (LSTM) and deep neural networks (DNN). The CYRUS team uses the CYRUS2D base code that was developed based on the Helios and Gliders bases.

Keywords: Robotic · Denoising · Soccer Simulation.

1 Introduction

RoboCup has hosted annual robotic soccer competitions since 1997, following the proposal of robotic soccer games as a new research topic in 1992[1–3]. The primary objective of these competitions is to advance the robotics and artificial intelligence field by implementing robotic soccer games. Several leagues, including rescue, soccer simulation, and standard platform soccer, have been created to achieve this.

The Soccer Simulation 2D (SS2D) league is one of the oldest RoboCup leagues. In a soccer simulation 2D game, two teams containing 11 players and one coach compete against each other. Each game takes 6000 cycles (10 minutes). In each cycle, the players and coach receive observations from the RoboCup Soccer Simulation Server and send actions based on those observations back to the server. The server is responsible for controlling the game.

CYRUS has been a regular participant in RoboCup competitions since 2013 and emerged as the champion of the SS2D league in RoboCup 2021. Over the

years, the CYRUS team has achieved first, second, second, third, fourth, and fifth places in RoboCup in 2021, 2022, 2018, 2019, 2017, and 2014, respectively. CYRUS also achieved first place in IranOpen in 2018 and 2014, RoboCup Asia-Pacific in 2018, and second place in JapanOpen in 2020.

In this paper, we will first discuss related research by other teams, focusing on topics such as AI and robotics in soccer simulation 2D. We will then present our previous works, including our released applications and codes. Next, we will explore the issue of noise in soccer simulation 2D, presenting our ideas for addressing this challenge. Finally, we will outline our plans for future work in this area.

1.1 Related Work

In this part, we will mention some latest novel ideas presented by SS2D teams in recent years.

YuShan2022 adopted an offline training model and LambdaMART algorithm to enhance their team’s offensive ability through movement direction selection[4]. RobôCIn researched an adaptive strategy based on time, score, and the opponent team’s pressure. Also, they developed and released a log analyzer software[5]. Persepolis team improved main behaviours such as pass, shoot, dribbling, and marking; also used a ranking algorithm to improve chain-action performance[6]. Persepolis also implemented a Path Planning algorithm to improve its offensive strategy[7, 8]. Helios2022 conducted numerical experiments to investigate the effect of using Anonymous mode on game results[9]. HfutEngine2D used genetic algorithms to adjust parameters and attempted to use neural networks for rational decision-making, significantly improving the team’s strength during recent tests against other RoboCup teams.[10]. Alice2D’s work centered around optimizing defense strategy using the Monte-Carlo tree and reinforcement learning algorithms.[11]. Hades2D team improved individual players’ offensive and defensive skills, significantly decreasing the average number of conceded goals against the base.[12, 13]. FRA-UNited utilized reinforcement learning (RL) techniques to improve their ability to intercept the ball from a player in possession. They created a Markov decision process (MDP) environment with two players: the agent learning to intercept and the dribbling player. Through this environment, the agent learned and improved its intercepting skills over time.[14].

1.2 CYRUS Previous Works

In this section, we briefly explain the history of our research. CYRUS was established in 2013 and has participated in many competitions since then. CYRUS Team researched various aspects of the game, starting with passing behavior, unmarking strategy, and formation[15, 16]. After that, we shifted our attention to defensive behaviour[17], shooting algorithms, and communication modules[18]. In 2017, the CYRUS team developed the opponent behavior prediction module[19, 20]. In recent years, CYRUS Team has mainly been developing the pass prediction and unmarking module, and Data Generator framework [21–25]. Some

of our research work, including Cyrus2D Base code, CYRUS 2014 Source Code, PYRUS - Python SS2D Base Code, and SS2D Log Analyzer, has been released in recent years.

Cyrus 2014 Cyrus team released its source code used in the 2014 RoboCup competition, where they achieved fifth place. The source code can be an excellent resource for other teams and new teams interested in participating in the RoboCup competition. The code provides valuable insights into the team’s approach, strategies, and implementations that can be helpful for others to improve their performance. The code is available on our GitHub⁵.

Cyrus2D Base Cyrus2D Base[26] code is a soccer simulation base code created by combining Helios Base[27], Gliders base[28], and the best features of the Cyrus team, which won first place in the RoboCup Soccer Simulation 2D league in 2021. This base code is an excellent starting point for new teams just starting to work on their soccer simulation projects. The code provides a solid foundation for building the functionalities required for the RoboCup competition. The code is available on our GitHub⁶.

PYRUS Base Pyrus Base code is a soccer simulation base code developed in Python, providing an excellent platform for researchers to develop and test their Machine Learning algorithms. This base code benefits teams interested in applying artificial intelligence techniques to enhance performance. The code is available on our GitHub⁷.

Cyrus AutoTest2D This software was initially developed by the WrightEagle team[29], and Cyrus team members upgraded and simplified it. The software helps teams run many parallel games to test their performance. Cyrus AutoTest provides a systematic approach to evaluating teams’ performance in the soccer simulation league, and the software can help teams identify their strengths and weaknesses to improve their performance. The code is available on our GitHub⁸.

AutoTune2D Cyrus team developed this tool that allows researchers to fine-tune their parameters alongside AutoTest. AutoTune can be a valuable tool for researchers who want to optimize their team’s performance by adjusting various parameters to find the best combination. The code is available on our GitHub⁹.

⁵ Cyrus 2014 Source <https://github.com/naderzare/cyrus2014>.

⁶ Cyrus Base Source <https://github.com/Cyrus2D/Cyrus2DBase>

⁷ Pyrus Base <https://github.com/Cyrus2D/Pyrus-SS2D-Base>

⁸ AutoTest <https://github.com/Cyrus2D/AutoTest2D>

⁹ AutoTune <https://github.com/Cyrus2D/AutoTune2D>

Cyrus Log analyzer This software can analyze soccer simulation 2D game logs to find statistical values, including the number of passes, dribbles, and other relevant statistics. This software can provide valuable insights into a team’s performance, helping teams identify areas of improvement and fine-tune their strategies. The Cyrus Log analyzer can help teams optimize their performance by providing detailed information about their gameplay. The code is available on our GitHub¹⁰.

2 Denoising

2.1 Observation Noise In SS2D

In the Soccer Simulation 2D league, accurately calculating the position of each object, including players and the ball, presents a significant challenge due to agents’ visual sensor limitations.

In this multi-agent environment, the agent can use three different vision angles: 180(Wide), 120(Normal), and 60(Narrow) degrees. However, using each angle comes at a cost. For instance, if the agent chooses to use the Wide angle, it must wait three cycles before using the sensor again. Similarly, for the Normal angle and Narrow angle, it should wait for 2 and 1 cycle, respectively. Choosing the right vision angle is important to balance performance and reduce the delay caused by limited sensor usage.

To address this issue, a parameter called ‘pos_count’ is defined for each object, which indicates the number of cycles that have elapsed since the last time the sensor received the object’s position.

In addition, the relative distance and angle information that the agent receives from the visual sensor are quantized, resulting in more noise in the distance measurement when the object is farther away. Furthermore, the floating-point part of the angle measurement is rounded.

2.2 Helios Base Denoising Method

HeliosBase applies a technique to tackle the quantized noise by considering the object’s distance. Specifically, it calculates the average range of distances at which the object can be detected. This method is also applied to the relative angle between the player and the object to refine the estimation further. After these initial estimates, the method can determine the object’s relative position before calculating its global position. This approach allows for improved accuracy in object localization, which is crucial in many applications such as robotics and computer vision.

¹⁰ CyrusLog Analyzer <https://github.com/Cyrus2D/CyrusLogAnalyzer>

2.3 Denoising by Machine Learning.

Our proposed approach aims to reduce the noise in the data when the object’s pos_count is more than a cycle. To achieve this, we plan to investigate various machine learning techniques, including Deep Neural Networks (DNNs)[30] and Long Short-Term Memory (LSTM)[31] networks. These models’ predicted positions are closer to the true objects’ than the last seen positions. The time series models will be trained on input sequences that include the position and sensor data from the last five cycles. We aim to find the best machine learning model to predict the opponents’ position.

2.4 Generating Data

In order to gather the necessary data for our study, we ran a total of 400 games between Cyrus2DBase and HeliosBase. The collected data includes the positional and velocity information of the ball and 22 players within the game and the direction of the players’ bodies. The data also includes the pos_count parameter for each player. The position data of each object consists of both noisy and original values.

3 Results

In addition to training the DNN and LSTM models, we also experimented with different sizes and configurations of these models to find the optimal architecture for denoising the positional data. After extensive testing, we found the LSTM model with 512 and 256 neurons and all ReLU activations in the time series model and the DNN model with 512, 256, 128, 64, and 32 neurons all ReLU activations achieved the best performance. Furthermore, we discovered that the LSTM model was more effective in predicting the accurate position of the players than the DNN model. Therefore, we focused on improving our solution’s DNN and LSTM models.

Figures 1, 2 and 3 presented in the heatmap compare errors among three methods: DNN, LSTM, and Agent Vision data. The heatmaps are based on the number of cycles an agent has not seen the player and the distance between the agent and the player. The agent represented in the figures is number 9 of the left team, and the object being observed is player 5 of the left team. LSTM has a higher level of accuracy than DNN, while DNN is more precise than player vision. Therefore, LSTM performs more accurately than the other two methods for this specific task.

4 Future work

There are several challenges associated with the proposed models. Firstly, the accuracy of the models may decrease if a team’s strategy and formation change, as the models primarily learn from the team’s historical data. Secondly, there

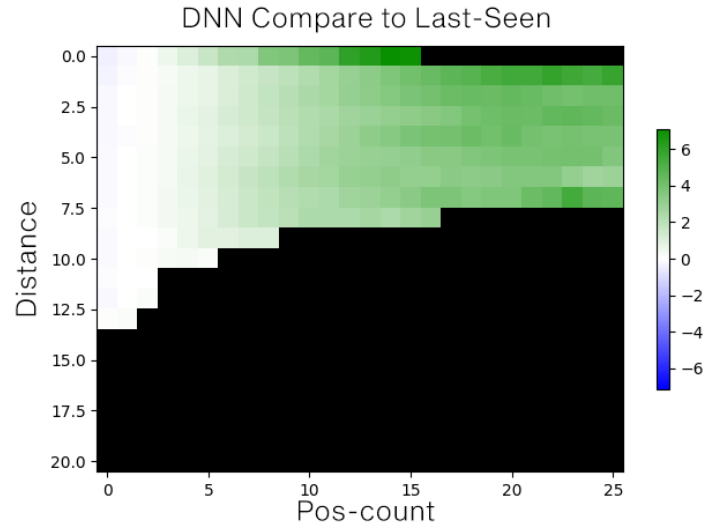


Fig. 1. Comparison of DNN accuracy and last-seen position accuracy. The green color shows a better performance with DNN, whereas the blue color indicates better performance in the last seen position. Insufficient data to compare these two methods are shown in black color.

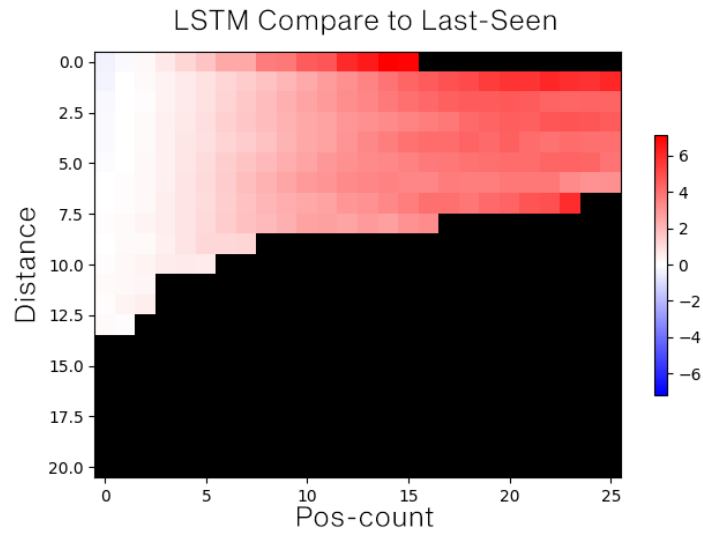


Fig. 2. Comparison of LSTM accuracy and last-seen position accuracy. The red color shows a better performance with LSTM, whereas the blue color indicates better performance in the last seen position. Insufficient data to compare these two methods are shown in black color.

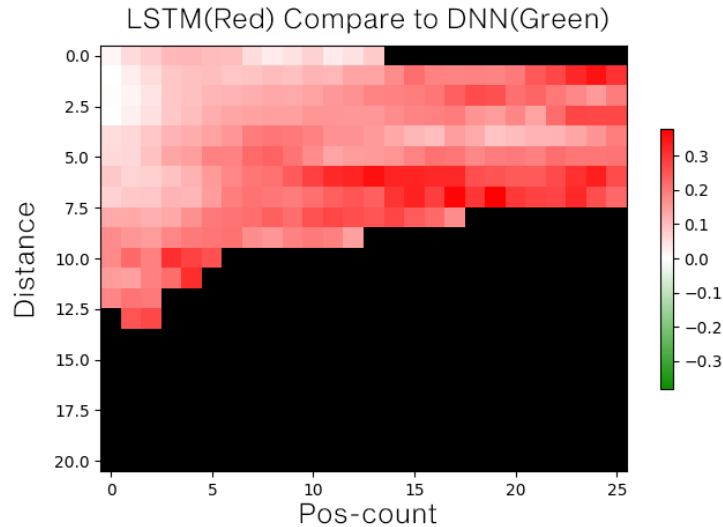


Fig. 3. Comparison of LSTM accuracy and DNN accuracy. The RED color shows a better performance with LSTM, whereas the green color indicates better performance in DNN. Insufficient data to compare these two methods are shown in black color.

may not be existing models for new teams in the league due to the lack of data. In such cases, one approach could be choosing a pre-trained model for other teams similar to the new ones to predict their performance. Another approach could be implementing online learning techniques that learn the opponent's strategy during a game, which can be challenging and require further investigation.

References

1. Kitano, H., Asada, M., Kuniyoshi, Y., Noda, I. and Osawa, E., 1997, February. Robocup: The robot world cup initiative. In Proceedings of the first international conference on Autonomous agents (pp. 340-347).
2. Noda, I. and Matsubara, H., 1996, November. Soccer server and researches on multi-agent systems. In Proceedings of the IROS-96 Workshop on RoboCup (pp. 1-7).
3. Kitano, H., Asada, M., Kuniyoshi, Y., Noda, I., Osawa, E. and Matsubara, H., 1997. RoboCup: A challenge problem for AI. AI magazine, 18(1), pp.73-73.
4. Cheng, Z., Guo, J., Ren, Y., Tang, Y., Rong, L.: YuShan2022 Team Description Paper for RoboCup2022. In: RoboCup 2022 Symposium and Competitions. Thailand (2022).
5. de Oliveira, C. S., Machado, M. G., Rodrigues, W. M., da Silva Araújo, T., Vitor Cunha, P., dos Reis de Labio, R., de Almeida Pereira, F. N. Borges Soares, M. F., da Silva Barros, E. N., de Mattos Neto, P. S. G, Ing Ren, T.: RobôCIn Team Description Paper 2022. In: RoboCup 2022 Symposium and Competitions. Thailand (2022).

6. Noohpisheh, M., Shekarriz, M., Zaremehrijardi, F., Karimi, M.: Persepolis Soccer 2D Simulation Team Description Paper 2022. In: RoboCup 2022 Symposium and Competitions. Thailand (2022).
7. Noohpisheh, M., Shekarriz, M., Zaremehrijardi, F., Khademi Ardekani, F., Khor-sand, S. A.: Persepolis Soccer 2D Simulation Team Description Paper 2021. In: RoboCup 2021 Symposium and Competitions: Team Description Papers. Worldwide (2021).
8. Noohpisheh, M., Shekarriz, M., Bordbar, A., Liaghat, M., Salimi, A., Borzoo, D., Zarei, A.: Razi Soccer 2D Simulation Team Description Paper 2019. In: RoboCup 2019 Symposium and Competitions: Team Description Papers. Sydney, Australia (2019).
9. Akiyama, H., Nakashima, T., Hatakeyama, K.: HELIOS2022: Team Description Paper. In: RoboCup 2022 Symposium and Competitions. Thailand (2022).
10. Wang, F., Lin, X., Xiao, Y.: HfutEngine2022 Simulation2D Team Description Paper. In: RoboCup 2022 Symposium and Competitions. Thailand (2022).
11. Yu, Y., Zhao, G., Wang, H., Fang, B.: Alice2022: Team Description Paper. In: RoboCup 2022 Symposium and Competitions. Thailand (2022).
12. 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).
13. Esmaelifar, S., Rokni, S. R., Esmaelifar, S., Akhondi, F., Rajabi, A., Pourmoghaddam, R.: Hades2D soccer2D simulation Team Description paper. In: RoboCup 2022 Symposium and Competitions. Thailand (2022).
14. Gabel, T., Eren, Y., Sommer, F., Vieth, A., Godehardt, E.: FRA-UNited — Team Description 2022. In: RoboCup 2022 Symposium and Competitions. Thailand (2022).
15. Khayami, R., Zare, N., Zolanvar, H. M. Karimi, P., Mahor, F., Tekara, E., Asali, Fatehi, M. : Cyrus Soccer 2D Simulation Team Description Paper 2013. In The 17th annual RoboCup International Symposium, Eindhovenm, The Netherlands. (2013)
16. Khayami, R., Zare, N., Karimi, M., Mahor, P., Afshar, A., Najafi, M. S., Asadi, M., Tekrar, F., Asali, E., Keshavarzi, A.: CYRUS 2D simulation team description paper 2014. In: RoboCup 2014. Joao Pessoa, Brazil, (2014).
17. Zare, N., Karimi, M., Keshavarzi, A., Asali, E., Ali Poor, H., Aminian, A., Beheshtian, E., Mola, H., Jafari, H. , Khademian, M. J.: Cyrus Soccer 2D Simulation Team Description Paper 2015. In: RoboCup 2015. Hefei, China, (2015).
18. Zare, N., Keshavarzi, A., Beheshtian, S. E., Mowla, H., Akbarpour, A., Jafari, H., Arab Baraghi, K., Zarifi, M. A., Javidan, R.: Cyrus 2D Simulation Team Description Paper 2016. In: RoboCup 2016. Leipzig, Germany, (2016).
19. Zare, N., Najimi, A., Sarvmali, M., Akbarpour, A., NaghipourFar, M., Barahimi, B., Nikanjam, A.: Cyrus 2D Simulation Team Description Paper 2017, In: Robocup(2017), Hefei, China (2017).
20. Zare, N., Sadeghipour, M., Keshavarzi, A., Sarvmali, M., Nikanjam, A., Aghayari, R., Firouzkoochi, A., Abolnejad, M., Elahimanesh, S., Akhgari, A.: Cyrus 2D Simulation Team Description Paper 2018. In: RoboCup(2018), Montreal, Canada (2018).
21. Zare, N., Sarvmali, 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).
22. Zare, N., Sayareh, A., Sarvmali, M., Amini, O., Soares, A., Matwin, S.: CYRUS 2D Soccer Simulation Team Description Paper 2021. In: RoboCup 2021 Symposium and Competitions, Worldwide (2021)

23. Zare, N., Firouzkouhi, A., Amini, O., Sarvmali, M., Sayareh, A., Soares, A., Matwin, S.: CYRUS 2D Soccer Simulation Team Description Paper 2022. In: RoboCup 2022 Symposium and Competitions, Worldwide (2022)
24. Zare, N., Sayareh, A., Sarvmali, 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)
25. Zare, N., Amini, O., Sayareh, A., Sarvmali, M., Firouzkouhi, A., Matwin, S., Soares, A.: Improving Dribbling, Passing, and Marking Actions in Soccer Simulation 2D Games Using Machine Learning. In: RoboCup 2021: Robot World Cup XXIV, Springer (2021)
26. Zare, N., Amini, O., Sayareh, A., Sarvmali, M., Firouzkouhi, A., Ramezani Rad, S., Matwin, S., Soares, A.: Cyrus2D base: Source Code Base for RoboCup 2D Soccer Simulation League. In: RoboCup 2022: Robot World Cup XXV, Springer (2022)
27. 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.
28. Prokopenko, M., Wang, P.: Gliders2d: Source Code Base for RoboCup 2D Soccer Simulation League. CoRR abs/1812.10202 (2018)
29. Bai, A., Lu, G., Zhang, H., Chen, X. (2011, July). WrightEagle 2D soccer simulation team description 2011. In RoboCup 2011 Symposium and Competitions: Team Description Papers, Istanbul, Turkey.
30. Simon Haykin. 1998. Neural Networks: A Comprehensive Foundation (2nd. ed.). Prentice Hall PTR, USA.
31. Hochreiter, S. and Schmidhuber, J.: Long short-term memory. 9(8):1735–1780, 1997. MIT Press.