RobôCIn Soccer Simulation 2D Team Description Paper for RoboCup 2024

Bonna Borsoi, Felipe Pereira, Gabriel Souza, Heitor Souza, Jeferson Araújo, Maria L. Silva, Mariana Coelho, Mateus Machado, Mateus Soares, Pedro V. Cunha, Rafael Labio, Walber M. Rodrigues, and Edna Barros

Centro de Informática, Universidade Federal de Pernambuco.
Av. Prof. Moraes Rego, 1235 - Cidade Universitária, Recife - Pernambuco, Brazil.
robocin@cin.ufpe.br
https://robocin.com.br/

Abstract. RobôCIn Soccer Simulation 2D team, based at the Universidade Federal de Pernambuco, was founded in 2018. In our debut competition at the Latin American Robotics Competition (LARC) in João Pessoa, Paraíba, Brazil, we secured fourth place against other Latin American teams. The following year, we competed in the RoboCup for the first time and achieved a ninth-place finish. In 2023, we placed in the 6th position on RoboCup and first place at LARC. In this paper, we present the work developed over the past year, especially the refinement and improvement of the agent's behaviors and changes made to the field evaluator.

Keywords: Soccer Simulation 2D · Data Analysis · Statistics · Behavior · Marking · Field Evaluator

1 Introduction

RobôCIn is a robotics research team from the Centro de Informática (CIn), Universidade Federal de Pernambuco (UFPE), created in 2015 to participate in competitions and research subjects related to robotics. We are currently working in four categories: Soccer Simulation 2D (SS2D), Very Small Size (VSS) since 2015, Small Size since 2018, and Flying Robots Trial League since 2022.

RobôCIn utilized the well-structured base of agent2d 3.1.1, as presented in [1], as the foundation for our code development in the first year. To improve performance, we subsequently integrated gliders2d-v1.6 [4] and MarlikBlock [5] to enhance the agent's movement behavior. Also, we are investigating the recently released Cyrus2D[6] and looking forward to possible future integration points. Our most recent agent release includes changes in goalie behavior, action evaluation, studies for implementing a ball prediction model, and analysis of situations for in-game marking.

2 Changes in Field Evaluator

During our matches, we noticed a consistent issue with our offensive tactics. When the ball was in our defensive territory, our players struggled to pass effectively to an open attacker, and during offensive plays, they had difficulty with central positioning. To address this, we adjusted the penalty for the opponent gaining possession from $-50 + ball_x$ to -500 to discourage errant passes and unsuccessful dribbles. It's important to note that we only considered the final state when making this change.

The FieldEvaluator class now focuses on evaluating game states based on ball positioning. It not only considers predefined terminations but also rewards the agent based on the ball's position, the chance of the last player with possession having a shooting opportunity, and the distance of the ball from both our goal and the opponent's goal. While these latter two conditions are closely related to the first criterion, they remain crucial for goal-scoring potential. However, we acknowledge that these conditions may need revision in the future. To align with our offensive strategy, we've developed a Potential Evaluation Function parameterized by the ball's position, as shown in Equation 1.

$$
\begin{aligned}
&E_x = \lambda_x^{dist_x} \\
&E_y = \begin{cases} 2*^{-\lambda_y * dist_y}, & \text{if } x > 0 \\ ^{-\lambda_y * dist_y}, & \text{otherwise} \end{cases} \\
&Evaluation = E_x + E_y,
\end{aligned} \tag{1}
$$

It was assigned weights $\lambda_{x,y}$ to functions and computed distances $dist_{x,y}$ in each axis to our goal center $(-52, 0)$. The weights, set to $\lambda_x = 1.02$ and $\lambda_y = -0.2$, were chosen arbitrarily. In future work, we will explore these parameters and their impacts. The resulting values were incorporated into the evaluation of ball position, enhancing the efficiency of action selection. Fig. 1 shows the Potential Evaluation Function.

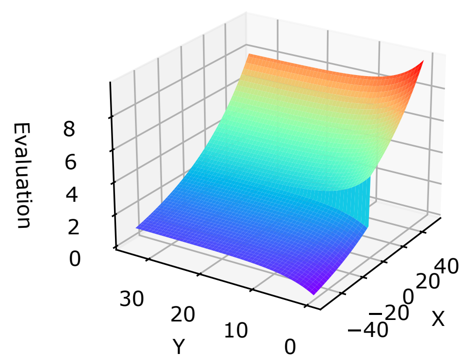

Fig. 1: The plot of the Potential Evaluation Function of a state according to the ball X and Y positions within the game field.

3 Set Play marking

After analyzing several games, it became evident that our positioning during set-play situations was not ideal for marking opponents, which allowed the opposing team to position themselves freely on the field. To tackle this issue, the players were positioned in a way that they could mark the opposing players and prevent potential moves from set-play situations.

To accomplish this, each player will try to mark the nearest opposing player outside a certain radius originating from the ball. This radius varies depending on the current set-play situation and is big enough to avoid player's positioning too close to any opponent near the ball, since there is a minimum distance to be respected, thus preventing a loss of stamina during the set play.

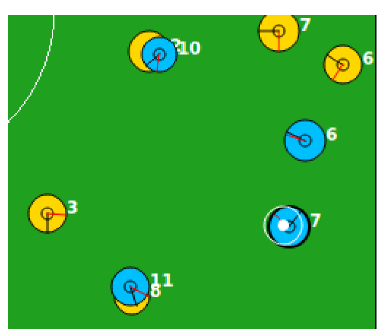

Fig. 2: Example of a setplay marking situation

4 Goalie Improvements

During the past year of activities and tests, we noticed some problems in our goalie behavior, such as trying to catch the ball at times that did not match its position and positioning close to the goal posts. Additionally, we observed that the goalie remained passive in situations where it should be more aggressive.

In the catching situations, better guidelines have been established regarding when the goalkeeper should or should not execute a catch, reducing the occurrence of unnecessary catches. Additionally, our goalkeeper utilizes orthogonal projection to the ball's trajectory to determine the point at which they should defend, thereby avoiding delayed catches.

Regarding positioning near the goalposts, it was defined specific scenarios in which the goalkeeper should act to narrow the shooting angle of the opponent closest to the goalposts.

Lastly, certain situations have been noted where the goalie should be more aggressive and move toward the ball instead of simply waiting. For instance, in scenarios involving a lead pass from the opposing team, our goalie actively advances toward the ball when possible to prevent the opposing player from gaining control of the ball unmarked near the goal.

5 Mark Analyzer

In order to implement new defensive behaviors and enhance our opponent marking system, we noticed the need to create a structure capable of analyzing the dynamics of the game and identifying marking situations. This structure was then implemented to identify possible teammate markers and their respective opponents, establishing a mapping between markers and their targets for all game modes. The implementation of the Mark Analyzer provided players with a more accurate perception of which opponents pose a threat because they are unmarked, allowing a more effective defensive response.

To address the target matching issue, the Hungarian algorithm was employed. It has been integrated into our SIM2D agent, inspired by the open source implementation developed by RobôCIn's Small Size League team[2]. The Hungarian algorithm[3], known for its efficiency in solving assignment problems in polynomial time, was a suitable choice to address the target matching issue in this context. It was used to consider only valid targets, using their distances and other positioning characteristics in relation to our player in the metrics for choosing the best assignment, as can be seen on Fig. 3.

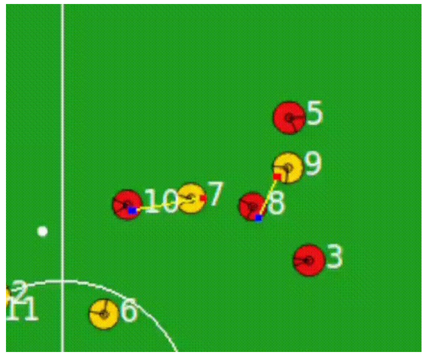

Fig. 3: Marking situations identified by Mark Analyzer

6 Ball Predictor

We explored methods to improve player's environmental perception to improve agent decision-making. The ball is the most critical element regarding player positioning in pre-defined formations, opponent marking behavior, and player effort to avoid or score a goal. However, it is intrinsic to the Simulation Soccer 2D environment that a Quantization error is added to the ball position, adding a layer of difficulty to the player's decision-making. That being said, we experimented with machine learning strategies to predict the vision residual error of the ball, using the noiseless coach vision as target.

Our strategy modifies the player's vision pipeline to add a predicted residual error for the ball vision. The error prediction is a two-layer Multi-Layer Perceptron f_θ with 100 neurons in the hidden layer, trained using 5 features: the ball position, player position, and noisy distance between the player and the ball. These 5 features are used for the error between the agent's vision and the noiseless coach's vision, following Equation 2.

$$\mathcal{L}(\theta) \leftarrow E[(coach\_vision - player\_noisy\_vision)^2] \tag{2}$$

To fix the ball position and avoid unnecessary computation, we modified the player's vision algorithm, adding the ball error prediction as in Algorithm 1.

Algorithm 1 Player ball vision chain with error prediction

1: p: previous ball position
2: $p \leftarrow$ updateBallByHear(p) ▷ Update ball data from other players
3: $p \leftarrow$ updateByGameMode(p) ▷ Update ball data using game mode information
4: $p \leftarrow$ updateSelfRelated(p) ▷ Update ball data using inertia and velocity
5: $p \leftarrow p + f_\theta(p, player\_pos, p - player\_pos)$ ▷ Predicts ball residual error

7 Experiments and Results

To validate the modifications developed, the RoboCIn Testing Module is used to evaluate statically. It is designed as a series of games and runs around five hundred times to generate a sufficient amount of data. This process allows us to analyze the performance of the changes made to the code and ensure they meet our standards.

7.1 Changes in Field Evaluator

In response to observed deficiencies in offensive tactics, it was implemented strategic adjustments in the Field Evaluator module. These alterations aimed to optimize player positioning and enhance playmaking capabilities. Here, we present the outcomes of these modifications and their impact on overall team performance.

	RCPotentialField	**RCMaster**
Victories	50.91	22.94
Defeats	22.94	50.91
Draws	26.16	26.16
Score	777	512
G.B	265	-265
S.P.G	1.56	1.03

Table 1: Statistical analysis results for Potential Field on Field Evaluator. G.B stands for Goal Balance and S.P.G for Score per Game.

7.2 Goalie Improvements

Implemented enhancements have markedly refined our goalie's performance, with improved catch execution guidelines and strategic positioning adjustments. Below are the results that reflect our observations on the adjusted goalie behaviors.

	RCGoalie	RCMaster
Victories	57.17	16.97
Defeats	16.97	57.17
Draws	25.86	25.86
Score	756	368
G.B	388	-388
S.P.G	1.53	0.74

Table 2: Statistical analysis results for Goalie improvements. G.B stands for Goal Balance and S.P.G for score per game.

7.3 Mark Analyzer

To evaluate the effectiveness of the Mark Analyzer, we conducted a series of 500 games against our master branch. By comparing the results of these games, as can be seen in Table 3, we were able to quantify the improvement provided by the new structure.

	RCMarkAnalyzer	RCMaster
Victories	57.23	14.86
Defeats	14.86	57.23
Draws	27.91	27.91
Score	763	361
G.B	402	-402
S.P.G	1.53	0.72

Table 3: Statistical analysis results for Mark Analyzer. G.B stands for Goal Balance and S.P.G for score per game.

7.4 Ball Predictor

For our experiments with the ball prediction algorithm, we collected data in every cycle of 10 games and paired the player's observation with the coach's observation. The coach's observation is regarded as the ground truth while the player's vision is the noisy data. The distance metric between the player position and the ball position is computed on the noisy data, decoupling any input data from the ground truth. We performed non-stratified bootstrapping, performing 100 dividing the dataset into 20% for test and 80% for training.

Figure 4 shows the Mean Squared Error (MSE) between the player's noisy observation and the ground truth (Vision Error), and the MSE between our denoise algorithm and the ground truth (Our Approach). We can note that despite the error being a Quantization error, for some positions the vision can be slightly improved. We note that the highest error is concentrated in the goalie position, as it is further away from the ball, we achieve a reduction of 12% in the error.

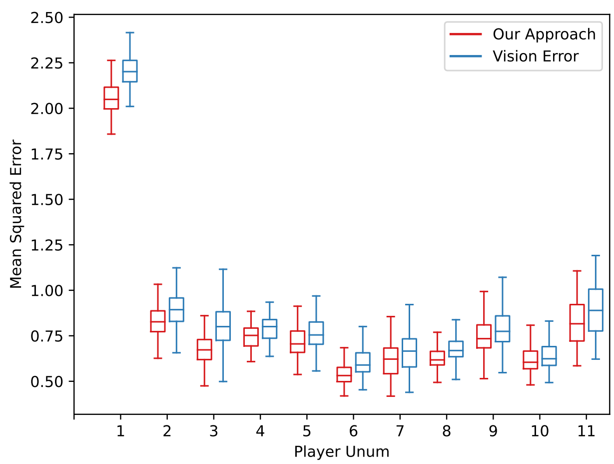

Fig. 4: Mean Squared Error between the player noisy vision (Vision Error) and our denoise algorithm.

8 Conclusions and Future Works

In conclusion, the changes made in the field evaluator enhanced our heuristic definitions for action evaluation, leading to better offensive behavior, which is a critical aspect of any successful team's performance. Other implemented changes make references to our defensive field, and getting all together returns a series of behaviors, both from markers and the goalie, that improve the way the team acts in front of an opponent opportunity in some situations. Overall, these changes have allowed our team to perform better in a variety of different scenarios.

For future work, we are planning to research and apply more machine-learning approaches for our agents while continually enhancing the models built since then. In the next steps of development, we aim to optimize code function parameters by exploring bioinspired algorithms, which offer more optimized and efficient performance. Additionally, we intend to build a pass model generator to enhance our pass accuracy and potentially generate better moves.

9 Acknowledgement

First, we would like to thank our advisors and the Centro de Informática (CIn) - UFPE for all the support and knowledge during these years of project and development. We also would like to thank all our sponsors: Moura, ITEMM, Microsoft, CESAR, Neurotech, Incognia, HSBS, Embraer, Instituto Nacional de Engenharia de Software (INES) and Mathworks.

References

1. Akiyama, H., Nakashima, T.: Helios base: An open source package for the robocup soccer 2d simulation. In: Behnke, S., Veloso, M., Visser, A., Xiong, R. (eds.) RoboCup 2013: Robot World Cup XVII. pp. 528–535. Springer Berlin Heidelberg, Berlin, Heidelberg (2014)
2. Cruz, J.: Soccer common (2022), `https://github.com/robocin/soccer-common`
3. Kuhn, H.W.: The hungarian method for the assignment problem (1995), `https://web.eecs.umich.edu/~pettie/matching/Kuhn-hungarian-assignment.pdf`
4. Prokopenko, M., Wang, P.: Gliders2d: Source code base for robocup 2d soccer simulation league (2018)
5. Tavafi, A., Nozari, N., Vatani, R., Yousefi, M.R., Rahmatinia, S., Pirdeyr, P.: MarliK 2011 Soccer 2D Simulation team description paper (2011), `http://archive.robocup.info/Soccer/Simulation/2D/TDPs/RoboCup/2011/MarliK_SS2D_RC2011_TDP.pdf`
6. Zare, N., Amini, O., Sayareh, A., Sarvmaili, M., Firouzkouhi, A., Rad, S.R., Matwin, S., Soares, A.: Cyrus2d base: Source code base for robocup 2d soccer simulation league. arXiv preprint arXiv:2211.08585 (2022)