Team Description Paper: Rhoban Football Club

Ludovic Hofer, Hugo Gimbert, Rémi Fabre, Grégoire Passault, Olivier Ly, Steve N'guyen, and Quentin Rouxel

Laboratoire Bordelais de Recherche en Informatique, Université de Bordeaux, 351, cours de la Libération, 33405 Talence, France
{ludovic.hofer,hugo.gimbert,remi.fabre,gregoire.passault olivier.ly,steve.nguyen,quentin.rouxel}@labri.fr

Abstract. In this paper, we describe the approach and the ideas used by our team, Rhoban Football Club. Our bot is based on the WrightEagle WEBase bot, with some extra features added:

- the goalie decisions have been optimized by offline solving of a large-scale MDP,
- dribbling speed has been improved,
- shoot decisions in attack mode have been optimized.

1 Introduction

This paper describes the features of the Rhoban Football Club bot for the Robocup Simulation League 2014.

This year the Rhoban Football Club team does apply to both, humanoïd kid size league and 2D simulation league. The Rhoban Football Club team has originally been focused on the humanoid kid size league, with three participations over the last four years. Basic functions like walking, shooting, basic visual anaysis and localization are performed by our robots Sigmaban in a more and more satisfactory way, and the team is focusing some attention on the strategic aspects, hence the decision to be a candidate to participate in the simulation league.

Our bot is based on the WEBase bot, brought by the WrightEagle team, with some extra features added:

- the goalie decisions have been optimized by offline solving of a large-scale MDP,
- dribbling speed has been improved,
- shoot decisions in attack mode have been optimized.

We started quite recently the optimization of the WEBase bot and we plan to add more new features in the next few months, cf. Section 4.

2 Work done

2.1 Moving toward a goal in a given time

One of the low-level ability required for decision-making is calculating the probability of reaching a position after a specified number of turns as well as the actions required. For a given situation we have identified 5 different parameters.

- The distance from the player to the target: ρ_{target}
- The direction of the target in the player referential: θ_{target}
- The current speed of the player: ρ_{speed}
- The direction of the players speed: θ_{speed}
- The number of cycles available before reaching the target: k

Since the first four parameters are continuous, we decided to use a regular mesh to represent this problem as a MDP, allowing to find solutions through value iteration with bounded horizon. However, due to the high-dimensionality of this problem, the solutions are not computed online. Therefore, the heterogenous aspect of the players is not taken into account.

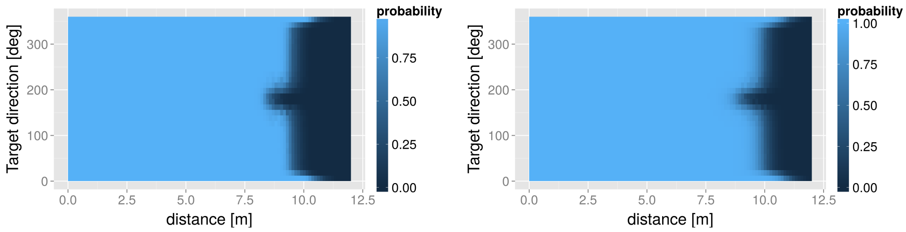
2.2 Optimizing the goalies chances of blocking the goal

Once the ball is kicked by an opponent toward the goal, our goalie has several options, catching, kicking or tackling the ball. We decided to handle these 3 situations in a very similar way. We simulate different possible trajectories based on the estimations of the ball position and on its speed. For every trajectory, we can obtain an approximation of the probability of applying successfully each action on the ball after k cycles using the values of a MDP calculated offline. Finally, we can choose the interception turn by maximizing the probability of reaching the ball over the different trajectories.

The computed probability of catching or tackling the ball depending on the distance and the direction of the target at 10 cycles in the future are shown at Fig. 1.

2.3 Dribbling faster

One of the improvement we brought to the WEBase bot concerns *Fast Dribble*. It is defined as a behavior where the player decide to throw the ball forward or laterally, planning to intercept several cycles later. Initially, this strategy had a predefined horizon and the wished speed for a kick was fixed to the maximal running speed of the player. The main issue about this policy was that the player was kicking the ball quite often, thus loosing speed and allowing its opponents to reach him. We decided to choose the horizon dynamically, and to allow higher speed for the ball, thus increasing the average speed of our forward player when applying this behavior.



(a) Catch probability (b) Tackle probability

Fig. 1. The probability of catching and tackling the ball at 10 cycles depending on distance and direction

By increasing the average speed of dribbling players, we managed to create more situations were our forward players were able to outrun the opponent defenders. However, this improvement was not enough to score goals against all teams, because the forward player were not able to create good opportunities for passes.

2.4 Facing the goalie

Our dribble improvement allow to create more situations where one of our forward player was alone, facing the goalie. In such situations, the player had to choose between moving slightly to improve its chance of scoring or trying to kick. If choosing whether to kick or not is a difficult problem, we decided to tackle this problem by computing the best kick angle at full power and the associated goal probability. This was performed every cycle, and if the value was above a given threshold, then the player tried the given kick. In order to avoid loosing a goal opportunity, we decided to use a dynamical threshold, depending on the distance to the closest opponnent.

3 Experimental results

Since we based our code on the WEBase, we decided to use the original version as a first baseline, and we managed to obtain serious improvement. We also compared the results of our team and WEBase when facing Agent2D.

3.1 Against WEBase

We measured the improvement brought by our modifications by analyzing the scores resulting of 50 games between WEBase and our last version of the bot. The average of goal taken by game is 3.40 while the average of goal scored is 6.18, leading to an average difference of 2.78. Due to the high standard-deviation on the numbers of goals by both teams, the WEBase team won 10 games out of the 50 played. By running a t-test on the games results we obtained a significative difference ($p < 5 * 10^{-10}$). The results are shown at Figure 2.

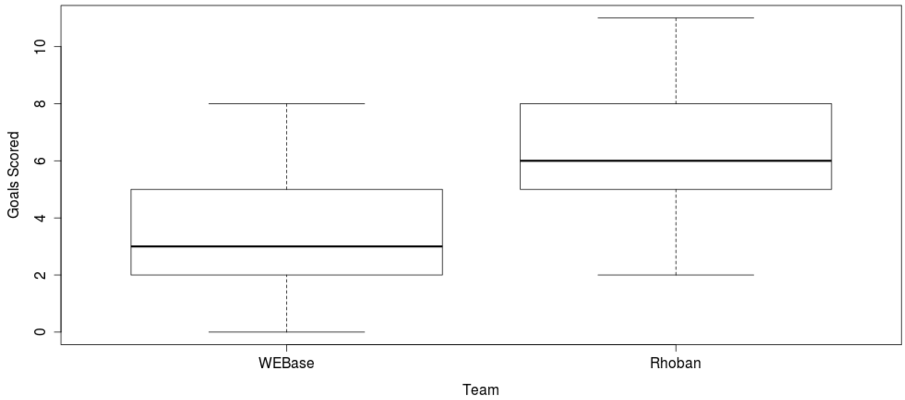


Fig. 2. The average results of 50 games opposing Rhoban to WEBase

3.2 Against Agent2D

Since the Agent2D obviously outperforms our current bot, we decided to compare the results obtained by WEBase and Rhoban when facing Agent2D. The results obtained by Agent2D over 50 games against Rhoban and 50 games against WEBase are shown at Figure 3. Even if Rhoban did not score significatively more goals than WEBase, it did take an average of 14.58 goals per game against 19.2 goals for WEBase. Running a t-test on the number of goals taken depending on the team who was facing Agent2D led to a significative difference ($p < 10^{-9}$). Therefore, we can consider that, by solving MDP offline and using them during the game, we improved the goalie's ability to stop goals.

Our low goal rate against Agent2D is mainly due to the fact that the players have a target position which depends only on the ball position and not on the position of close opponents. Therefore it is very hard for midfield players to make interesting passes.

4 Future work

4.1 Improving passing and positionning behaviors

As mentioned previously in 3.2, the weakest point of our strategy is the positionning behavior which do not allows players to be eligible for receiving passes. We plan to start by introducing a simple plan allowing our players to take into account the position of opponents in order to increase the opportunities offered to the ball owner. We also plan to extend those two behaviors by using the ideas proposed in [5], thus allowing to plan more than one pass ahead.

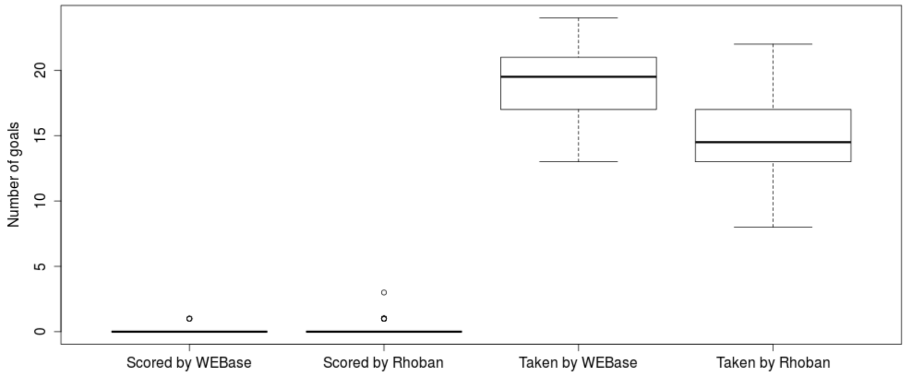


Fig. 3. The average results of 50 games opposing Rhoban to Agent2D and 50 games opposing WEBase to Agent2D

4.2 Solving MDPs online

We plan to use the delayed Q-learning algorithm associated with adaptative mesh refinement in order to solve the approach problems online, thus allowing us to take in consideration the heterogeneity of players. One of the interest of this method is that it allows to bound the values of the states without knowing the model of the MDP [1].

4.3 Visual informations

For WEBase, the orientation of the neck is updated every turn in order to maximize a given criteria. The body direction is modified in order to increase the set of possible directions only if no other interesting actions where selected. While this algorithm leads to satisfying result in many situations, it does not allow to choose between acquiring more information and using existing informations.

The evaluation function is a crucial point of the game [3]. Therefore, we plan to include the uncertainty about teammates and players as a part of the evaluation function.

When using the current version of the trainer, one of the issue is to initialize the knowledge of the robot. A solution proposed in [6] is to use an initial time in order to let each robot obtain information about the current state. However, this solution cannot be used when the aim of the trainer is to determine whether information should be used or applied. Therefore, we plan to implement a compact way of representing the belief-space of each robot in order to load it when using the trainer.

4.4 Strategies with low descriptive complexity

We plan to keep the descriptive complexity of the strategies low and be able to combine the strategies in a modular way. For that we will focus on strategy improvement algorithms [2] rather than value iterations [4].

5 Conclusion

We presented our Robocup soccer simulation 2D team, Rhoban Football Club, based on the bot WEBase, published by the WrightEagle team. We explained the improvements we brought to the existing bot and showed that our version was performing better. We finally described the main modifications that we want to implement before the robocup, including dynamic positionning, multi-step passes, online computation of MDP and low descriptive complexity of strategies.

References

1. Tomás Brázdil, Krishnendu Chatterjee, Martin Chmelik, Vojtech Forejt, Jan Kretínský, Marta Z. Kwiatkowska, David Parker, and Mateusz Ujma. Verification of markov decision processes using learning algorithms. In *Automated Technology for Verification and Analysis - 12th International Symposium, ATVA 2014, Sydney, NSW, Australia, November 3-7, 2014, Proceedings*, pages 98–114, 2014.
2. Cyrus Derman. On sequential decisions and markov chains. *Management Science*, 9:16–24, 1962.
3. Edward Moore, Oliver Obst, Mikhail Prokopenko, Peter Wang, and Jason Held. Gliders2012 : Development and Competition Results. 2012.
4. L. S. Shapley. Stochastic games. *Proceedings of the National Academy of Science USA*, 39:1095–1100, 1953.
5. Haochong Zhang, Miao Jiang, Haibo Dai, Aijun Bai, and Xiaoping Chen. WrightEagle 2D Soccer Simulation Team Description 2012. *Wrighteagle.Org*, 2012.
6. Haochong Zhang, Miao Jiang, Haibo Dai, Aijun Bai, and Xiaoping Chen. WrightEagle 2D Soccer Simulation Team Description 2013. *Wrighteagle.Org*, 2013.