WrightOcean Team Description Paper

Yifan Liu, Shiyu Ren, Qian Liu, Kaixuan Sun, Hao Wang, Kai Liu, Pengtao Yu, Yansong Gao, Gaohuan Lv, and Fei Liu✉

Lab of Robotics,
School of Information and Electrical Engineering,
Ludong University, China
wrightoceanoffical@126.com, liufei@ldu.edu.cn
http://www.wrightocean.org

Abstract. In this paper, we describe the research focus, ideas implemented and future research directions in the WrightOcean team. Our code is based on the UT Austin Villa base code. We've optimized the basic skills such as walking and kicking using CMA-ES algorithm, and added a lot of high level strategies.

1 Introduction

As a new team of RoboCup Soccer Simulation 3D League, WrightOcean team was established in July 2017 and it belongs to Lab of Robotics, School of Information and Electrical Engineering, Ludong University in China. It still is a young and passionate team, and all of the members are undergraduates that are keen on robotics.

In order to focus on the field we are interested in, we use the base code release of the UT Austin Villa RoboCup 3D simulation team[1]. We have improved and added many of our own ideas based on the open source code. To test our progress, we took part in the RoboCup China open in Shaoxing, China in April 2018 and received the third rank finally.

2 Overview

In order to quickly focus on areas of interest, such as robot motion control, intelligent decision-making and multi-agent cooperative strategy, we use the UT Austin Villa Base Code Release[2]. The code release provides a fully functioning agent. On this basis, we designed different kinds of kicks and goalie dives, and we used CMA-ES (Covariance Matrix Adaptation Evolution Strategy) algorithm[3] to optimize these basic skills, such as walking, kicking, and getting up (Section 3). In addition, we implemented many of our own ideas in the high-level strategy, which enabled the agent to make appropriate decisions according to different environments and make the whole team more consistent (Section 4). And the cooperative control of multi-agent will be our focus in the future.

3 Agent Skills

3.1 Walk

Walking like a human is very important for robots. Especially in the 3D competition, stable and fast walking is a key factor to improve the competitiveness of a 3D soccer team. The UT Austin Villa code release has an omnidirectional walk engine based on a double inverted pendulum model[4], and includes slow and stable walk engine parameters. According to Patrick MacAlpine's overlapping layered learning method[5], we optimize the parameters using CMA-ES algorithm.

3.2 Kick

Kicking as a basic skill is as important as walking. A powerful kick can help us score. And we can also use kicking to pass ball. It is the basis of the team cooperation system.

UT-Austin-Villa base code supports a stable but short kick skill. But it can't threat the opponents' goal. So we create a new kick skill. Then, we put the parameters of the skill into the CMA-ES algorithm[3] to optimize. These are as follows.

Create Kick Skill UT-Austin-Villa base code has a kicking skill execution mechanism based on key frame. One kicking skill is generated by adding a keyframe composed of joint angles. To do it more efficiently, we use the AIUT3D Motion Editor[6]. It is a open source software, developed by the AIUT Team. The action we designed is shown in Figure 1.

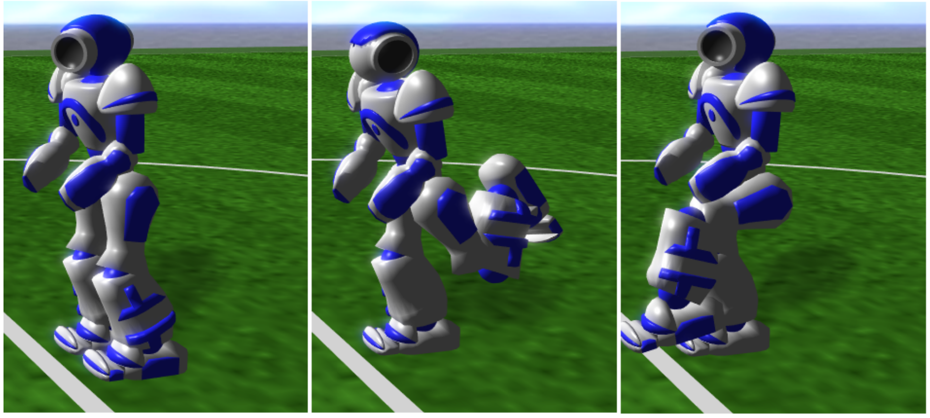

Fig. 1. kick skill

Optimize Because the kicking skill made by hand is not powful enough, we use the CMA-ES algorithm[3] to optimize it. Here we draw on the Patrick MacAlpine's overlapping layered learning method[5].

3.3 Get Up

Due to collision and other reasons in the game, the robot will inevitably fall down. So a fast and stable get up skill can help team to grab the ball. UT-Austin-Villa base code supports a stable but slow get up skill. We use the CMA-ES algorithm[3] to optimize the get up skill so that the robot can get up more fastly. Meanwhile, in order for the robot to walk steadily after getting up, we have trained both the getting up skill and walking together.

3.4 Dive

In order to defend the opponents' shot, we design some dive actions for the goalie. Just like kicking skill, we used AIUT3D Motion Editor[6] to accomplish it. We designed three dives to defend shots from any angles. They are shown in Figure 2.

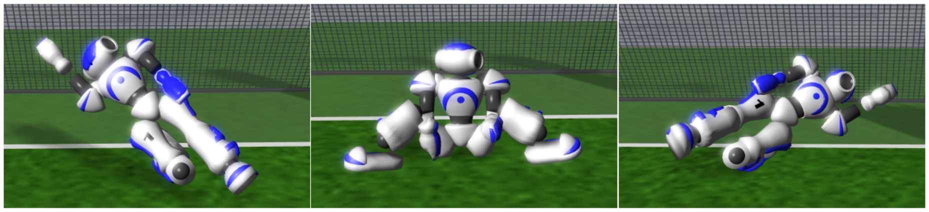

Fig. 2. dive skill

4 Stretagy

4.1 Role Assign

As is known to all, soccer is a team sport, the tacit cooperation between players is the key to the team's winning. A team has 11 players, and how to control them to make the whole team show consistency is an interesting and challenging work. There are many roles in the soccer game like stricker and defender, each with his own mission. In the 3D competition, each robot is isomorphic, so we can dynamically change their roles to improve the team's offensive and defensive efficiency.

First of all, we determine the current team formation according to the position of the ball. Of course, our formation is not static. If an opponent enters an area that is threatening to us, we consider changing the current formation and adopting a man-to-man defensive strategy.

After the formation is determined, we consider how to assign agents to these positions. We first select a player holding the ball based on factors such as position, orientation, whether it falls, and the opponents' position. We consider the match between the remaining players and positions as a bipartite graph matching problem. Our goal is to make the convergence time of the formation as short as possible. Therefore, we use the distance between the player and the position as the basis weight, and adjust according to the player's orientation, whether it falls, and whether there are obstacles between the player and the target. Finally, the Hungarian algorithm is used to solve the problem[7].

4.2 Pass

To make the shot more efficient, robots should pass ball to create a better position to shoot. So we set up some points based on the kick distance from the ball, and calculated scores for these points according to the situation on the field. The robot would kick the ball to the point with the highest score, and the role assignment system assigned a player to the point to receive the ball.

4.3 Kick Type Selection

We have three types of kick: long range kick, fast kick and dribble. When the kicker gets the ball, how to choose a reasonable kick type is important. We build up an evaluation system for kick type selection. The robot will evaluate each type of kick according to the situation on the field, and the result is whether to choose this type of kick. If all three kick types are available, we prefer long range kick, then fast kick, and finally dribble.

5 Future Work

At present, we still have some gaps in basic skills (like kicking and walking) compared with the top teams. So our priority is to continue to improve these skills. With reliable skills, our focus will be on multi-agent cooperative control. We will explore different ways to implement a variety of team strategies. At the same time, we are studying the use of reinforcement learning to train agents, and we are doing some work in this area.

References

1. Patrick MacAlpine and Peter Stone. UT Austin Villa RoboCup 3D Simulation Base Code Release. In Sven Behnke, Daniel D. Lee, Sanem Sariel, and Raymond Sheh, editors, RoboCup 2016: Robot Soccer World Cup XX, Lecture Notes in Artificial Intelligence, pp. 135–43, Springer Verlag, Berlin, 2017.
2. UT Austin Villa Base Code Release. https://github.com/LARG/utaustinvilla3d

3. N. Hansen, S. Müller, P. Koumoutsakos, Reducing the time complexity of the derandomized evolution strategy with covariance matrix adaptation (cma-es), Evolutionary Computation 11 (1) (2003) 1–18.
4. Patrick MacAlpine, Samuel Barrett, Daniel Urieli, Victor Vu, and Peter Stone. Design and Optimization of an Omnidirectional Humanoid Walk:A Winning Approach at the RoboCup 2011 3D Simulation Competition. In Proceedings of the Twenty-Sixth AAAI Conference on Artificial Intelligence (AAAI), July 2012.
5. Patrick MacAlpine and Peter Stone. Overlapping Layered Learning. Artificial Intelligence, 254:21–43, Elsevier, January 2018.
6. A motion editor for designing arbitrary key frame based motions for Nao to be used in RoboCup 3D soccer simulation league. Available: https://github.com/AIUT3D/aiut3d-motion-editor
7. Munkres, James. Algorithms for the Assignment and Transportation Problems[J]. Journal of the Society for Industrial and Applied Mathematics, 1957, 5(1):32-38.