Team Description of Kshitij3D

Shadab S. , Karteek Babu J. , Sireesh B. , Ravi Sankar P.
Kamalakar Karlapalem

Center for Data Engineering,
International Institute of Information Technology, Hyderabad
{s_shadab,karteek_bj,sireesh_b,ravi_s}@students.iiit.net , kamal@iiit.net

Abstract. Kshitij3D is a new robosoccer simulation team, developed based on the agent-test given in the package. In this paper, we brief our team strategy, some of the low-level and high-level skills developed (or still in development) by us.

1 Introduction

Transition from 2D to 3D introduced many open challenges in Multi-agent simulations like searching of huge state space, spatial reasoning etc.. The 3D-server [3, 2] mimics the real world soccer environment by considering the physics involved in it. Unlike in 2D simulation, 3D simulation is practically continuous and the 3D-server uses SPADES [5] middle-ware agent to remove some of the drawbacks in 2D-server like fluctuation of the team performance due to the machine and the network load. For agent evaluation, it also tracks the thinking time of the agent.

2 Team Strategy

Zone based approach is used, where the field is divided into 3 zones (shown in Figure 1(A)) namely defense zone, mid-field zone and attack zone. We used 3x4x3 offensive formation for scoring many goals. Our agent architecture is same as in the agent-test given in the package but has advanced behavior than the normal kick-and-run behavior. Currently passive agents follow SBSP [4]. One-to-one opponent marking is used in the defensive zone for minimizing opponent-team chance of scoring goals. Currently, our goalie behavior is derived from the goalie in UvA base code [1].

2.1 Localization

Positions of all the objects in the environment are obtained relative to the agent (similar to 2D). Global position of the agent is calculated by using the nearest flag relative to the agent in the simulation environment as shown in Figure 1(A). Let O, F, P be the positions of the origin, flag and agent $A3$ in the environment respectively. The global position of $A3$ is given by

$$OP = OF - PF$$

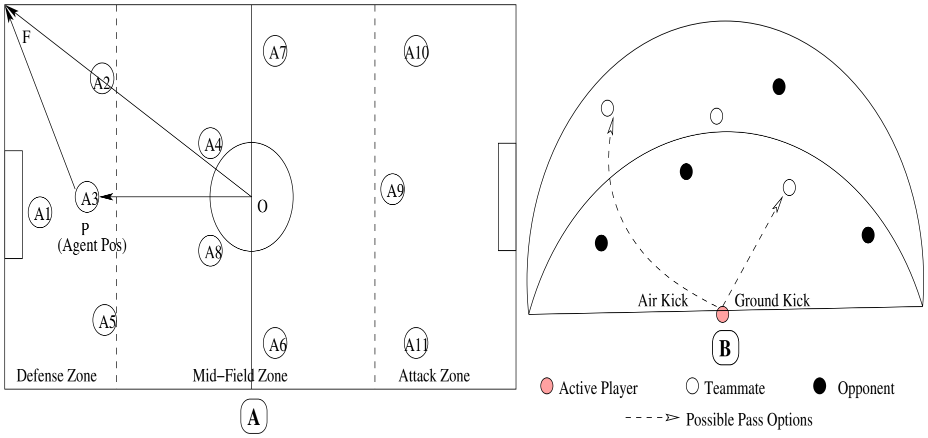

Fig. 1. (A) Formations, Zones and Agent Localization (B) Pass

3 Low Level Skills

Most of the developers in our team have little experience in 2D, so it took some time to understand the behavior and functionality provided by the 3D-server. In the initial phase, we tested drive, kick provided by the server and explored various issues like

- Working of kick and drive effectors
- Relationship between the kick power, velocity imparted to the ball and its variation in the following time cycles
- The maximum distance an agent can kick
- Drive behavior with different parameters

Drive to a point Using rotational and translational dynamics inherent in the drive effector, we obtained a differential equation that guide the motion of an agent. By solving the obtained differential equation, we get a relation between the velocity of an agent and drive power at a given time cycle. Using this relation, we can have controlled motion of the agent to any point on the field.

Kick to a point Agent needs to kick the ball with some calculated power so as to reach the defined point with a desired end-speed. For air-kicks, we found that power is linearly proportional to the distance traveled by the ball and for ground kicks, speed of the ball decreases following some decay series. So, for a given end-speed of the ball, initial speed can be found by using geometric progression. Force and torque equations used in kick effector are applied to obtain desired power for kicking such that the desired speed is imparted to the ball.

4 Predictions

Several prediction methods like $FastestToBall$, $NearestToBall$, $NearestToOpponent$, $GetNrCyclesToPoint$, $GetNrPlayersInRegion$, $PredictStateAfterCommand$

etc. were developed. The high-level skills and our basic decision module are based on these prediction methods.

5 High Level Skills

Pass Straight passes between the agents, forces the receiving agent to turn around the ball for correct alignment. Since aligning takes considerable amount of time, we prefer lead passes between the agents where the receiving agent can easily collide with the ball. Air kicks are preferred when the selected teammate to pass is at a longer distance relative to the active agent. Otherwise, ground kicks are used as shown in the Figure 1(B).

Dribble When there is no teammate available to receive the ball, we use this skill to maintain ball possession. We use a 2-phase algorithm for finding the optimal dribble direction. In the first phase, we consider all the opponents within a certain area and enumerate all the possible directions in which our agent can dribble. In the second phase, we select the best direction by considering our teammates within a limited region. Dribble length is based on the two nearest opponents in the dribble direction.

Interception Using the predictions for the ball and the agent, we determine an optimal interception point in order to reach the ball before an opponent does.

Align Position This skill helps the agent to drive to a particular point within the kickable region so that the agent can kick the ball in the desired direction. Currently our agents are taking more time for aligning compared to other teams. We need to improve this skill in our team.

Clear Ball Align position skill takes considerable time to kick the ball in the desired direction. So, in our defense zone, we use air-kicks in the direction of the widest angle between the opponents to move the ball away from our goal post.

6 Conclusions

As new features are going to be included in the 3d-server, we mainly concentrated on a basic team having strong low-level skills. In the future, we will concentrate on improving our high-level skills, prediction methods and a concrete decision module.

7 Acknowledgments

We are extremely thankful to our guide kamal karlapalem for giving us valuable suggestions through out the project development. We thank UvA-Trilearn team members for providing UvA Base code with comprehensive documentation that helped us in gaining new ideas. We finally thank developers of 3D-server for providing a wonderful server and for answering to our doubts patiently.

References

1. Remco de Boer and Jelle R. Kok. The incremental development of a synthetic multi-agent system: The uva trilearn 2001 robotic soccer simulation team. Master's thesis, University of Amsterdam, The Netherlands, feb 2002.
2. 3D-Server development team. Text instead of manual.
3. Marco Kogler and Oliver Obst. Simulation league: The next generation. In *Robocup symposium, Padova, Italy*, 2003.
4. Nuno Lau Luis Paulo Reis and Eugenio Oliveria. Situation based strategic positioning for coordinating a simulated robosoccer team. In *MAS Springer, LNAI, Berlin*, volume 2103, page 175.
5. Patrick Riley and George Riley. SPADES — a distributed agent simulation environment with software-in-the-loop execution. In S. Chick, P. J. Sánchez, D. Ferrin, and D. J. Morrice, editors, *Winter Simulation Conference Proceedings*, volume 1, pages 817–825, 2003.