

UI-AI3D 2004 Team Description

Reza Zafarani¹, Mohammad Amin Jashki¹, and Mohsen Moosavi²
<http://eng.ui.ac.ir/uisim>

¹ Computer Engineering Department , University of Isfahan ,
81745 Isfahan, Iran

rzafarani@eng.ui.ac.ir, uisim@comp.ui.ac.ir

² Faculty of Engineering, University of Isfahan,
81745 Isfahan, Iran

mohsen@mossavi.com

Abstract. For the first time ever, the 3D soccer simulation environment[1] is introducing a new test bed for creating and simulating 3D soccer teams. Teams are based on spades[2] and the new 3D simulator born. Despite the little time we had in developing the soccer server 3d binary, the new UIAI3D team had some achievements in developing a new team.

1 Introduction

The development of UIAI3D team started right after the server release time, the development process therefore was quickened by designing a simple binary which had some simple but effective advancements.

2 Team Architecture

Each agent's main loop is based on several simple action which he can execute in each sensation and acting cycles. In each cycle the agent checks if the ball is kickable or not. The agent kicks the ball to the opponents goal if he is standing in the right position for scoring toward the opponents goal, otherwise he will adjust himself in a way so that he can score in near future. The field is divided into several parts which each part is controlled by an agent. The agents will move to the ball if the ball is in their controlled area. Movement of the agent is defined in a way so that the agent accelerates for a while and breaks after a number of cycles so that he stops right behind the position he is moving to. The movement process is described through out the paper. The agents would move to their strategic positions when they should not move to the ball or kick the ball toward the opponent's goal.

3 Formation

The players will move to their formation based on the ball position they receive. The agents play offensive 3-4-3 as their default formation system which

is changed based on the goals they score or concede. The formation system is currently read from a file which tells the agents where they should be positioned.

4 Movement

The small time we had in developing the soccer3D team was enough to show us how important movement was in a soccer3d simulation game. Therefore deep calculations were needed so that our player would accelerate and break in the right directions to move to a certain point on the field. If the movement is not considered correctly the agents would simply pass the point or stop behind it, thus it is a key factor in a soccer3D match. We simplified the problem by assuming that the starting speed is the same as the breaking speed, in fact the breaking speed is an opposite vector of the starting speed. We assumed that the player speeds up to the maximum speed possible with the starting drive vector before breaking, our experiments showed that around 35 sensations of acceleration with the starting speed would result in that. We found out that if the player speeds up for 35 cycles with a starting speed 10 cycles of breaking with the same speed (speeding in the opposite angle) would stop him. By testing different speeds and breaking we found out that a linear relation could be found between the distance player travels by a starting speed for 35 sensation and breaking for 10 cycles with different speeds. Therefore we calculated the distance player travelled for drive vector(1,0,0) which was around 0.451 and each time the players used this constant to determine their first speed. E.g. if the player wants to travel 45 meters then he should accelerate for 35 sensations with the drive vector length of $45/0.45$ (Drive(100,0,0)) and he should break after that for ten cycles with the drive vector in opposite angle (Drive(-100,0,0)).

5 Kicking the Ball

Since kicking the ball in 3d Server is not the same as the traditional 2d server, which kicking was possible in different directions. Thus it is obvious that the players should place themselves in a strategic point behind the ball so that they can kick the ball toward the direction they want. Our players check that by the angle their position makes with the line created from the center of the opponents goal and the ball (currently, our agents can only kick to opponent's goal). If it is higher than a certain threshold or the player is not behind the ball, the player would adjust and replace himself behind the ball. The player would do this until he is placed on a good position to kick. Since it is possible to kick higher than the ground the player would first calculate the angle which ball hits the upper post from the current location, subtracts a fixed amount of error from the angle derived and then kicks to the max angle between the angle derived and zero angle.

6 Conclusion and Future Work

The main aspect for the current UIAI3D-2004 was to create a working team for the new born generation soccer simulation server. A simplified version of important actions such as moving to a position and kicking the ball was implemented. Several problems were ignored since there was not enough time to implement them in systematic way. For example in our current movement model a player cannot change the direction of move until he is near the point he was ordered to go before. Therefore the players will always travel any distance in a fixed amount of time which makes it totally inflexible. Our future work is based on fixing the low levels of our code in a better way so that better high levels and decision layers could be based on that.

References

1. Kogler, M.: Simulation and Visualization of Agents in 3D Environments.
2. Riley, P.: SPADES System for Parallel Agent Discrete Event Simulation, User's Guide and Reference Manual For Version 0.91