

Daredevils 2D Soccer Simulation

Team Description Paper 2013

Gehad Kaseb, Marwa Ayman, Rawia Ahmed, Safaa Mohamed

Cairo University, Faculty of Engineering, Computer Department

Srmg.cmp2013@gmail.com

Abstract.

Daredevils Team is a group of four students in the last year of Computer Engineering. We participate in this competition as our graduation project 2013. This paper gives an overview on the planned techniques of the Daredevils soccer simulation team. Our short-term goal is to build an effective goalkeeper. Our long-term goal is to establish a smart team that utilizes different computational intelligence techniques such as Fuzzy Logic. In this paper, we present a brief description to our view to the team strategies and used techniques.

1. Introduction

The main proposal is to use fuzzy logic to implement a set of behaviors/strategies, based on information provided by the sensor models (aural, visual, body sensor). Fuzzy logic is then used to calculate the actions of each agent, based on its role on the team (goalkeeper, defender, midfielder or attacker), its position on the field, and the position of the ball. The importance of fuzzy logic is due to the uncertainty of the simulator data.

In our work towards building an effective team, we surveyed many of the available software libraries that help accessing the simulation server. However, we discovered many restrictions and difficulties in modifying such libraries. Therefore, we decided to start developing our library using java which it will be the first free open source library in java. We already paid extra effort in developing such library. This library will be available for other teams that are interested in java language to use it and may add to it. The design of this library considers the extensibility that might be needed.

The paper is organized as follow: section 2 describes how the different role-base agents take a decision using fuzzy logic controller; section 3 describes a terminology to avoid opponent predicting the agent's decision; section 4 describes the advantages and the architecture of the used library; The last section contains the paper conclusion.

2. Fuzzy Controllers for Robotic Soccer Agents

The goal is to develop fuzzy controllers for specialized agents in the positions: goalkeeper, center defender, wing defender, defensive midfielder, offensive midfielder and attacker.

For example, when an attacker agent kicks the ball to opponent goal, it must see if the opponent goalkeeper is on left, center or right so it can kick as farthest as possible from the goalkeeper. But there is no precise numeric definition for the linguistic terms left, center, right or far from goalkeeper. This example illustrates the imprecise nature of information that must be used in reasoning process by autonomous agents. Fuzzy Rules-Based Systems are composed of "fuzzy rules", such as: "if distance is close and the speed is medium, then the action to take is to brake a lot." Each rule relates to fuzzy linguistic variables in a simple manner, which allows the modeling of human knowledge about the problem even in the presence of imprecision. Therefore, we believe that the fuzzy controller is a suitable technique that will handle many of the team strategies. Our team will initially use the fuzzy logic type I and it will be later extended to type II.

2.1 Controller for Kick Positioning

The main objective of this controller is to find the position in the opponent goal to kick with the best chances to score. To decide where to kick, the agent will take in consideration the opponent goalkeeper position and his own position. The output variable is kickposition, it has values between -7.0 and 7.0, which represents the ycoordinates of the opponent goal. The x-coordinate is the field limit, the same as the goal line, it will always be 52.5. Figure 1 represents the linguistic terms used for this variable during fuzzy reasoning process.

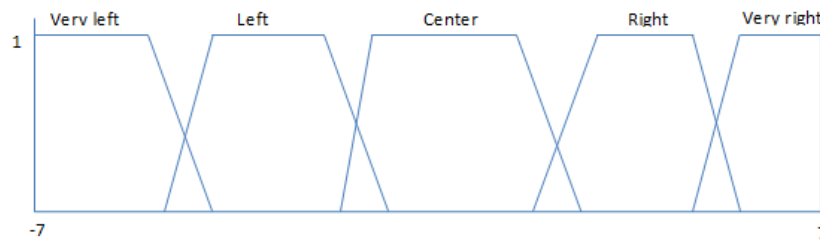


Fig.1. Linguistic terms for goal position in the y-axis

Input variables for this controller are goalkeeperPosition and kickerPosition. They also refers only to the y-coordinates, Figure 2 illustrates its linguistic terms. The min and max values ranges from -7,0 to 7,0 respectively.

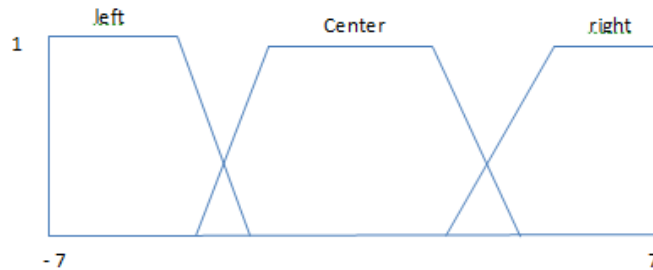


Fig.2. Linguistic terms for position of agent in the y-axis

The rule base for this controller is a combination of the input variables; it intends to find a position at the goal the farthest from the goalkeeper and the closest to the kicker as possible.

2.2 Controller for Attacker Positioning Without Ball

The objective of this controller is to allow the attackers, without ball, to find a position in the opponent's field, based on his own position, on the offside line and on the ball position. The input variables are the player and the ball position and the offside position on the x-axis. The output variables are the x-axis position and y-axis position, which represent the final positions that the agents should go to. The opponent's field is divided into 9 zones. The rule for this controller is the agent could only move to adjacent quadrants. The movement occurs in shorts distances, because in each cycle the perceptions can change and so the direction that the agent should move. For example, an agents positioned at the 8th quadrant can only move to the 4, 5, 6, 7, 8 or 9 quadrant, because he can't make to the 1, 2 nor 3 quadrant within the next cycle. This was defined by creating rules that have the same values for inputs, but different outputs and weights. The rules that make the agent stay at his quadrant, when the ball is at another, have lower weights, making him always move into the ball direction

2.3 Controller for Offensive Wing Midfielders Decision Taking

The objective of the Fuzzy controller for the offensive wing midfielders is to decide what he should do when in ball possession. The attacker should make a decision analyzing his global position, the distance to the closest opponent, the distance from the closest teammate to the opponent goal, the number of opponents near the closest teammate, and his distance to this teammate. The output variable is a value, and depending on its range, a decision to kick, dribble, pass, pass through or conduct is made. The domain of the output variable varies between 0 and 5, and each range represents a decision. The rules for the controller were created from the combination of the input variables, always aiming to decrease the possibility of losing the ball and to increase the chances of scoring.

2.4 Controller for Central Defenders Positioning

The objective of this controller is to allow the central defenders without ball to position themselves searching for the ball and staying at center of the defensive field. For this controller the y-axis was divided into two zones, LeftCenter and RightCenter, making him stay at the defensive center, leaving the laterals for the wing defenders. The output variables used by the controller are the x-axis position and y-axis position, which represent the final position that the agents should go to.

2.5 Controller for Goalkeeper Positioning

The goalkeeper moves to a certain position of the goal, as he notices that a player from another team is in a kickable distance from the goal. As inputs for the rules, the following variables are calculated: extending a straight line from the ball position to the center of the goal, and intercepting this line with the virtual vertical line where the goalkeeper position itself. It calculates the distance between the actual coordinate of the goalkeeper and the coordinate of this intersection, giving an positive or negative number, depending on the positions of the ball and the goalkeeper. This value is the input for the rule-base for the goalkeeper. The rule base provides an output variable called PosCatch which is exactly the position that the goalkeeper should take to catch the ball, and with which velocity, in order to position itself to block a possible kick to its goal.

2.6 Controller for intercepting the ball

The objective of this controller is to choose the best player from a set of players who see the ball with the opponent possession in their visible range to intercept the ball. The controller takes into consideration the position, stamina and the direction of the player, the position of the ball, the closest distance from the player to the ball position and the number of opponents near the closest teammate.

2.7 Controller for obstacle avoidance

Obstacle avoidance uses a fuzzy logic system that has the angle and the distance of the agent with respect to the obstacle as its inputs and generates the speed and the position that will effectively avoid any perceived obstacle (Teammates, opponents, flags).

3. Avoid predicting decisions

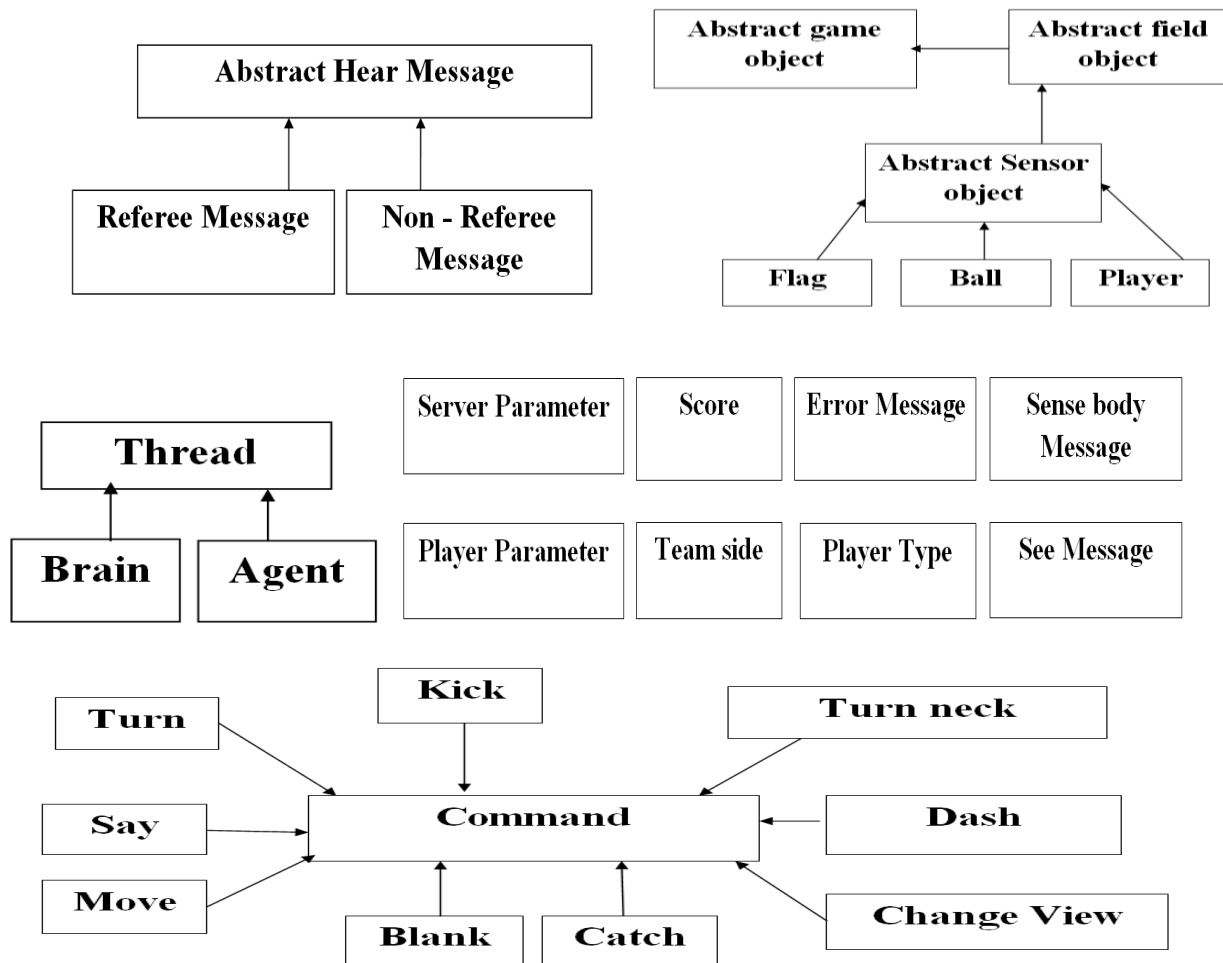
Taking the same decision every time a situation happens may lead that the opponent will be able to predict our actions. So, we are trying to make our decisions dynamic using more than method to take a decision such that each method give a valid different decision.

4. Interface with server

There are a few java clients/libraries available online. Among them, there is a simple client example- [Krislet](#). It is fairly old and very basic but it has errors in parsing the information of server version 15.0.1. It only handles non full-state messages from an outdated protocol; it isn't usable for us, but it showed us how to communicate with the server. [Atan](#) is a more complex client in java but it still has bugs. [Sebbot](#), it implements some strategies but still has problems in parsing the message. So we decide to build our own interface with server, and we took care of many things:

- Full parsing of every string coming from server including warnings, errors, see, hear, sense_body, init, yellow\red card messages, server_param, player_param, and player type messages.
- Providing mapping for every message
- Managing timing constraints, allowing only one command per cycle via a return statement
- Providing complete shielding of every technical detail of the server

And this is architecture of our library



5. Conclusion

In this team description paper we have outlined the characteristics of the Daredevils team participating in RoboCup's 2D Soccer Simulation League. Our team will utilize the fuzzy logic in different strategies and we believe that fuzzy logic will be effective in different roles. Also, our team will develop a new library based java. We believe that such library will be the first in its type that will posted as open source for java developers. We expect that such library will add a great value to RoboCup in later years.

6. References

1. Marco A. C. Simões, Helder Guimarães Aragão, Victor Souza, Simon Viegas, Hugo da Luz Silva, Jessica Meyer, "Using Fuzzy Logic to build a Heterogeneous Multiagent System for the Robotics Soccer Problem", Computer Architecture and Operating Systems Group (ACSO), State University of Bahia (UNEB), Intelligent Computing Research Group (GPCI), University Center of Bahia (FIB).
2. <http://www.larc10.fei.edu.br/TDPs/F180/78344.pdf>
3. http://cdn.intechopen.com/pdfs/9353/InTech-Fira_mirosot_robot_soccer_system_using_fuzzy_logic_algorithms.pdf
4. Vahid Khodabakhshi, Mojtaba Mesri, Navid KeyhaniRad, Hosein Zolanvar, "Paranoid 2D soccer simulation, Team paper description 2011", Department of Information Technology Engineering, Department of Information Technology Engineering, 2011.
5. Helena Åberg, Kungliga Tekniska Högskolan, "Agent Roles in RoboCup Teams", School of Computer Science and Techn Åsa Åhman. Decision Control, Department of Computer and Systems Sciences, The Royal Institute of Technology, Stockholm ology, February 1998.