Journal of Communications Technology, Electronics and Computer Science, Issue 13, 2016
ISSN 2457-905X

Passing in Robot Football Players Using Fuzzy Logic Systems

Alireza Rezaee
Assistant Professor at Department of System and Mechatronics Engineering, Faculty of New Sciences and Technologies,University of Tehran,Tehran,Iran
arrezaee@ut.ac.ir

***Abstract*— Robot football systems are an important part of smart systems these days. The goal in developing these robots is to create a machine that can imitate human actions in a football game. Fields such as Robotics, smart controls and Machine vision are involved in this process. Success of a robot football team will depend on both the individual players (technique) as well as the team play ability of each player (tactic). Having a controllable and appropriate speed, sense of direction, dribbling ability and others are examples of individual player skills (techniques). The ability to comprehend the overall game direction, whether to pass the ball or not best player to pass the ball to, and etc., are examples of team skills (tactic) of a robot football team. In this article, implementation of smart passing system that includes identification of best player and most suitable location for passing and other related issues are examined.**

Keywords—Robot football; fuzzy logic; smart system; pass, rule base; membership function.

I. INTRODUCTION

Soccer Robot Systems (SRS) are a complex smart system that are regularly discussed in smart system panels. Figure 1 demonstrates a general overview of the soccer game played by the robots. Progress in robotic soccer systems, their impact on other areas of science, and more importantly the field of intelligent systems is such that Robo-cup tournaments are held every year for researchers active in the field. In fact, due to the great advancements made by participating teams each year, the rules are regularly changed to reflect the level of progress. As such, interest in Robo-cup tournaments has increased resulting in more participation in groups [1]. As an example, in the past few years the number of participants in small size soccer robots has increased such that the playing field is almost doubled in size. Despite the great advancements in individual player abilities, the greatest progress in soccer robot systems has been in team tactics and increased intelligence of these systems. One of the reasons for the doubling of the playing field is to be able to better implement team tactics. In football, similar to other sports, there is the possibility of players falling over, making fouls [2], and passing the ball to other players. Moreover, players should be able to adapt to new environments [3].

Decision makings are based on a set of components such as tactics and robot skills [4]. One of the most important characteristics of a team with high tactical ability is their passing skill. Passing between players is common method to create opportunities [5].

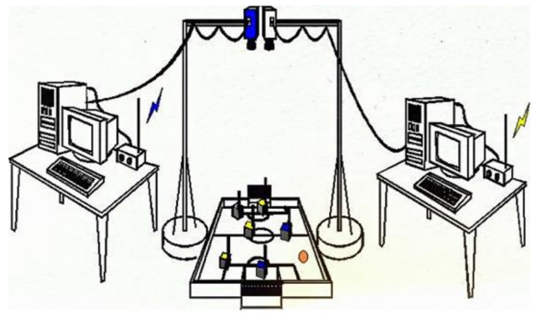

Fig. 1. General overview of robots playing _eld.

In this regard, each player must have complete knowledge of the playing field and position of every player to be able to identify the best target for passing and execute the pass. In this paper, the implementation of the passing action for Mono brain robot systems is investigated. The specifications of good passing is defined first and after extracting proper rule bases, fuzzy logic system for decision making during passing is deployed. A friendly simulation environment is also provided by MATLAB graphical user interface and the fuzzy system prepared earlier is used.

II. DEFINITION SPECIFICATIONS OF A GOOD PASSING

2.1 Differences between human and robots while playing football

Before defining a good pass, the differences between human and robot while playing football must be pointed out. Robots have limited accuracy when kicking the ball to reach a specific target. This limitation is due to positioning errors and orientation which is by itself due to uncalibrated images resulted from sensor errors. Although methods such as gyroscope and periodic camera sensor calibration are used for each robot to reduce such errors, positioning accuracy is still limited in these robots. Automation systems and robots are dependent on data codes received from a network [6]. To study the effect of positioning error, let us concentrate on figure 2. In this figure, the distance between the robot and the goal is 2 meters and positioning error caused the robot not to face the goal exactly which resulted in 4 degrees of positioning error.

If kicking the ball, it will land on:

$$2 * \tan(40) * 100 = 14cm \qquad (1)$$

Unlike human playing football, it is not applicable for a robot to pass the ball to a distant robot player.

Another major difference between human and robot football players is in kicking speed. Human player can send the ball slowly to a nearby player and faster to a distant one however, because robots use compressed/stretched spring mechanism, they can only send the ball with fixed or multivariable speed. This happens because to kick the ball, robot must compress the springs first and release them at the moment of kicking the ball. So another major limitation is the fixed speed of kicking the ball while sending it for other robot players. While keeping these limitation is mind let us de_ne a good pass for a robot player.

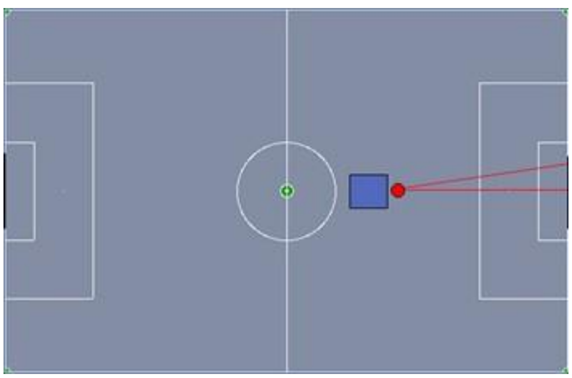

Fig. 2. The effect of positioning error.

The recipient robot is not too close nor too far Opposing players are as far as possible from the source and recipient robots pass line Opposing robots are as far as possible from the recipient robot The passing action gets you closer to the opponents goal It is possible that a player is not in an ideal position to receive the ball, but If it moves, it can create a better position. Therefore, analyzing players position should be dynamic. Certainly, the above parameters are not the only ones for a good pass and other parameters such as game score, opposing teams tactic and others can be input in the fuzzy logic system. However, for a simple fuzzy logic system the above parameters should be adequate.

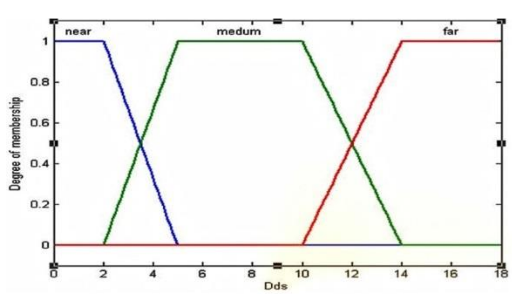

Fig. 3 The effect of positioning error.

III. Analyzing parameters of a good pass

3.1 Distance between the source and recipient robot

This distance is the Euclidean vector norm between the source and destination robots. As per the dimensions considered in the simulation, for the football field, this distance is on the [0 18] vector. For this distance, three sets of near, medium and far can be considered on the membership function shown in Figure 3.

Given the stated limitations, the distance between the source and destination robots should be in the medium set as much as possible. The effect of being in each of the near, medium or far sets is evident in extraction of rule bases.

3.2 Distance of opposing player from the line of departure and destination

Opposing players closeness to the passing line can be dangerous. This is illustrated in Figure 4. In the figure 5, player number 5 is not a suitable target for receiving the ball as players 2 and 4 from the opposite team are in the balls path.

As per Figure 5 and given that the robots maximum speed is considered $\frac{1}{3}$ of the ball speed, (assuming for simplification that the opponent can intercept the ball at point M) in such a situation if the opponents distance to point M is less than a specific level, the opponent can intercept the ball. Otherwise, interception is not possible.

The distance needed to catch the ball is calculated as below:

Vp : Player Speed

Vb : Ball Speed

Vb = 3Vp

The time needed for the opposition to reach point M is equal to:

$$trM = \frac{h}{v_p} \quad (1)$$

The distance that the ball travels at this time frame is:

$$D_b = V_b : trM = V_b : h/V_P \quad (2)$$

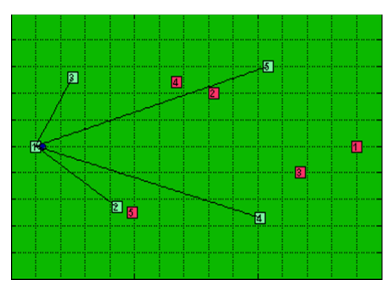

Fig. 4 Risk of opposite player being close to the passing line.

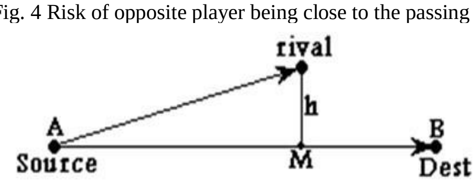

Fig. 5 Calculation of Risk for players along the pass line.

If the distance traveled by the ball is less than the distance between the source and point M, then the opposing robot can (but not certainly) intercept the ball. Otherwise the rival robot cannot intercept the ball. Therefore:

$$T: C := \frac{AM}{D_b} = AM \Big/ h.\frac{V_P}{V_p} = \frac{1}{3}.\ \frac{AM}{h} \qquad (3)$$

Can be considered as a risk factor for the players situated along the passing

line between the source and destination. In a crisp situation,

If (T:C: > 1 ↔(AM/h > 3), then the opposing player is considered dangerous.

If (T:C: < 1↔ (AM/ h < 3), then the opposing player is not considered dangerous.

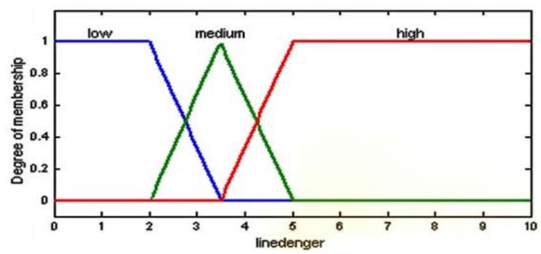

Fig. 6 Membership Functions related to risk of players distance to the balls path..

However, not every player with a T.C. greater than one can intercept than ball, because there are other parameters such as early detection of ball trajectory, rotating in time and etc. that affect interception success rate. Moreover, given that robots have different maximum speeds, using a crisp model would not

be correct and we would have to use fuzzy logic functions for these parameters. Considering AM/h instead of T.C., as risk factor indicating parameter, we can consider three membership functions of Low, Medium, and High for this parameter. These functions are shown in Figure 6.

The more the AM/h parameter is in the Low function the better. As the parameter moves towards the Medium and High functions, the situation gets worse.

3.3 Distance of opponents from target player

Proximity of opponents to the target player can be dangerous. If an opponent gets too close to the target player, he can hunt the ball before it reaches its destination. Figure 7 illustrates the proximity of an opponent to the target player. In this figure, player 5 is not a suitable player to pass the ball to because player opponent 2 is near this player. In a crisp model, if the distance between the opponent and target player is less than a specific value, opponent can hunt the ball otherwise he can't. This distance is calculated according to figure 8 as follows: Vp : PlayerSpee

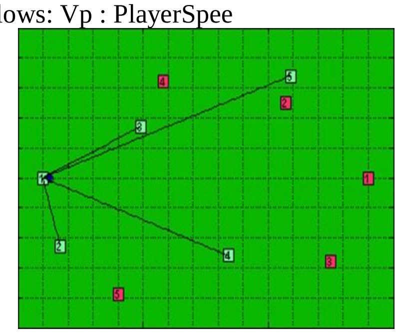

Fig. 7 Danger of proximity of opponent to the target player.

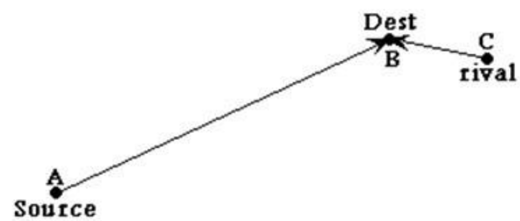

Fig. 8 Calculating the danger factor of proximity of opponent to target player.

$V_b = 3V_p$

Time required for the ball to reach to destination from source is equal to:

$$t_b = dAB \big/ V_b \qquad (4)$$

The distance that rival can cruise during this period is equal to:

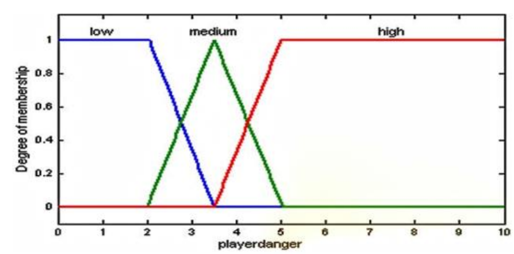

Fig. 9 Functions of the danger stages related to proximity of opponent to the target player.

$$d_r = V_p t_b = \frac{v_p}{v_b} d_{AB} \qquad (5)$$

In crisp model, if the distance of dr gets more than the distance of rival and Dest, opponent player can kick the ball or else he cant. So

T:C: = dr/ dBC

can be considered a danger factor. In crisp model:

If (T:C: > 1(dAB/ dBC> 3) then opponent is dangerous

If (T:C: < 1(dAB<dBC< 3) then opponent is not dangerous

However, for the mention reasons in the previous paragraph, defining the above border in crisp model is troublesome. Hence fuzzy stages must be considered for this parameter. Considering T:C: instead of AB/dBC as the danger parameter, 3 stages of low, medium, and high can be defined for this parameter. Functions of these stage parameters is shown in figure 9.

If dAB/dBC has more members in low, condition is better and if it has more members in medium and high categories, condition gets worse.

3.4 Advancing towards opponents goal

In order to evaluate the player with a better position, we need to introduce another parameter into the fuzzy logic process. Although the distance between the player and the opposing teams goal is not the only parameter, in order to simply the calculations, only the distance was considered. Figure 10 illustrates

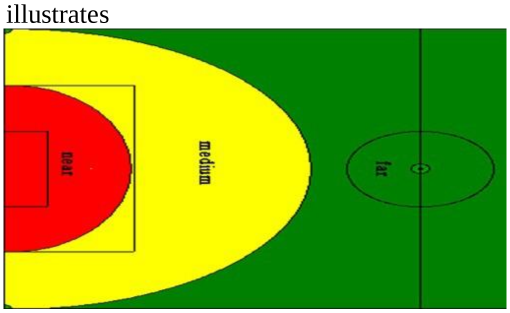

Fig. 10 Division of playing _eld in a crisp state according to the distance to. Opposing.

the three areas of Near, Medium and Far in the crisp state. In a fuzzy state, the membership functions in Figure 11 can also be considered.

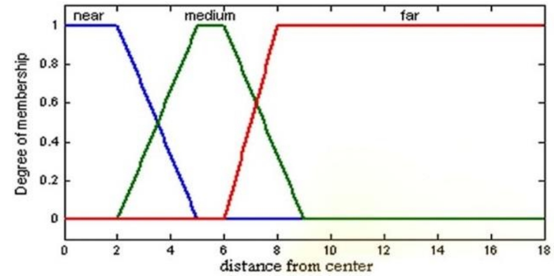

Fig. 11 Membership functions related to distance to opponents goal parameter.

3.5 Dynamic analysis of destination player position

It is possible that a player is not in a suitable position to receive the ball, but by moving to a specific location, it can get to a position to receive the ball with minimum risk. The importance of this movement is shown in Figure 12.

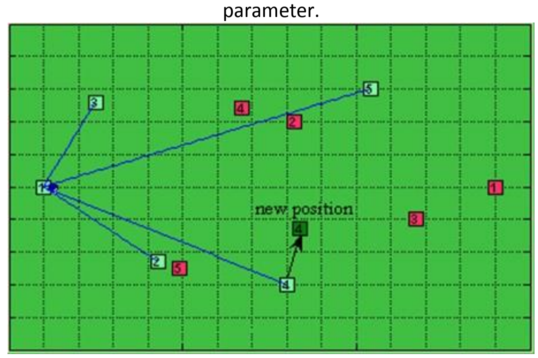

Fig. 12 Dynamic analysis of destination player position.

In this Figure, player number 4, due to the presence of player number 5 from the opposing team, is not in a suitable position to receive the ball. However, by moving to a new position as the ball is being delivered can get the ball with minimum risk.

The maximum distance the destination player can move until the ball reaches to him, creates a certain area in the _eld. In order to _nd this area, we consider Figure 13.

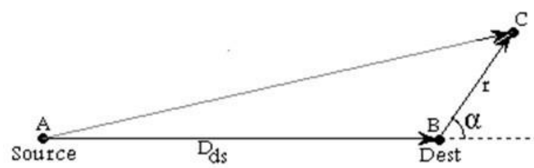

Fig. 13 Calculation of destination player displacement.

Destination player must be able to reach position C before the ball reaches there. Therefore considering

Vp : PlayerSpeed

Vb : Ball Speed

Vb = 3Vp

The time needed for the ball to reach point C from the source is:

$$t_b = \frac{\sqrt{(D_{ds}+r\cos\alpha)^2+(r\sin\alpha)^2}}{V_b} \quad (6)$$

The maximum distance the destination player can move in this time period is:

$$r = V_p t_b = V_p.\frac{\sqrt{(D_{ds}+r\cos\alpha)^2+(r\sin\alpha)^2}}{V_b} \quad (7)$$

Considering the equation Vb= 3Vp and by simplifying, we get:

$$9r2 = D2 + r2 + 2rDdscos\alpha = \Rightarrow 8r2 - 2rDdscos\alpha - D2 = 0 \quad (8)$$

And by solving the equation we get:

$r = \frac{\cos\alpha + \sqrt{8+\cos\alpha^2}}{8} D_{ds}$ (9)

This equation is a circle in the polar coordinate system with a radius of $\frac{3}{8D_{ds}}$

and a center $\frac{1}{8D_{ds}}$ points away from the destination player. It is also along the passing line of the ball. Destination players movement border is

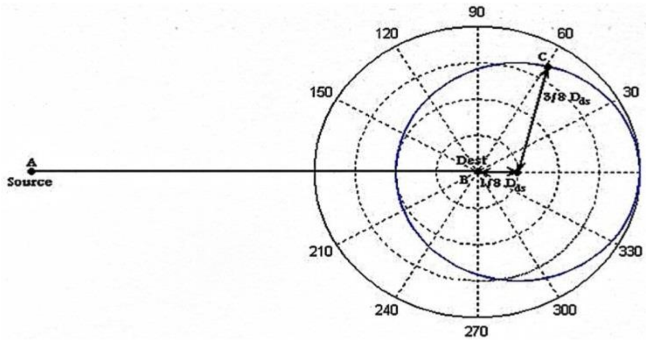

Fig. 14 Destination player movement border.

shown in Figure 14. For a dynamic analysis, we can assume that the destination player is at the new position (circle center) and can move a maximum distance equal to the circle radius. As such, for different angles and distances from the circle center, we can study the destination robots displacement and identify

the best location for it to receive the ball. The source robot can then send the ball to that location.

IV. Implementation of the fuzzy logic system

In order the implement the fuzzy logic system, we used the Fuzzy Logic Tool- box embedded in Matlab. Initially using FIS Editor, the system is created and implemented. Then by loading the system in the GUI, the modeled playing field is used. Figure 15 shows the graphic user interface in the Fuzzy Logic Toolbox that were used to implement the fuzzy logic system.

4.1 Rule base extraction

During the process of completing this project, the Rule Bases were modified often. At the end, with more assessments and various testing the following were used as final rules.

Fig. 15 Graphic user interface in the Fuzzy Logic Toolbox.

4.2 Defining the Fuzzy Logic System in FIS editor

FIS Editor is a graphic user interface that enables the configuration of a fuzzy logic system. For the fuzzy logic system to implement the passing action, it needs four inputs, Dds line danger player danger and distance from center and one output, pass grade. Figure 16 shows the system in FIS Editor along with all inputs and output.

4.3 Defining membership functions in membership function editor

After configuring the system in FIS Editor, the membership functions for each input and output need to be defined in Membership Function Editor. For each of the inputs and outputs the necessary functions are defined and introduced

TABLE I. Rule base extraction.

Distance to opponents goal Risk of opponent close to target	Distance between source and destination pass point	Risk of opponent close to line	Risk of opponent close to destination	Points of pass
N	*	H	*	g4
N	*	*	H	g4
N	F	M	L	g15
N	F	L	M	g15
N	N	M	L	g13
N	N	L	M	g13
N	M	L	M	g16
N	M	M	L	g16
N	F	L	L	g17
N	N	L	L	g15
N	M	L	L	g20
N	F	M	M	g14
N	N	M	M	g13
N	M	M	M	g15
M	*	H	*	g2
M	*	*	H	g2
M	F	M	L	g12
M	F	L	M	g12

M	N	M	L	g9
M	N	L	M	g9
M	M	L	M	g14
M	M	M	L	g14
M	F	L	L	g14
M	N	L	L	g13
M	M	L	L	g18
M	F	M	M	g10
M	N	M	M	g9
M	M	M	M	g12
F	*	H	*	g0
F	*	*	H	g0
F	F	M	L	g9
F	F	L	M	g9
F	N	M	L	g6
F	N	L	M	g6
F	M	L	M	g11
F	M	M	L	g11
F	F	L	L	g11
F	N	L	L	g10
F	M	L	L	g15
F	F	M	M	g6
F	N	M	M	g5
F	M	M	M	g9

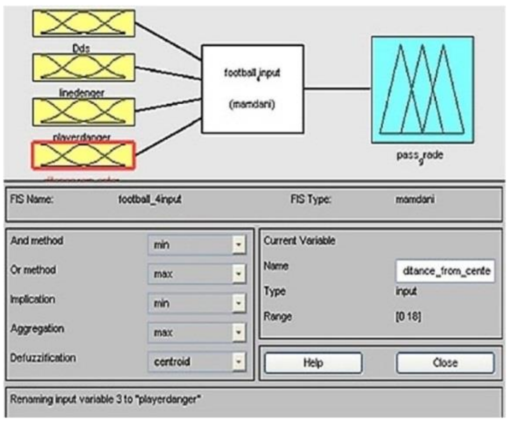

Fig. 16 System in FIS editor.

to the fuzzy logic system. Figure 17 illustrates the membership functions for
passgrade:

4.4 Implementation of rules using Rule Editor

Rule Editor is a graphic user interface used to define the relationships between inputs and outputs in fuzzy logic system. After preparing the rules, we can implement them using Rule Editor. Figure 18 shows the implementation of rules in the environment.

4.5 Testing the Fuzzy Logic system using rule viewer

After performing the above steps, now is the time to test the system. For this, the Rule Viewer application embedded in the Fuzzy Logic Toolbox is used. Rule Viewer enables us to observe, in a graphical interface, which by applying various inputs into the system and seeing how they affect the rules and the output, the efficiency of the system can be studied. Figure 19 shows Implementation of inputs [11 3 2 3] and their impact on the simulation of the rules.

The designed system can be stored in a graphic user interface (GUI) after testing and implementation of necessary changes.

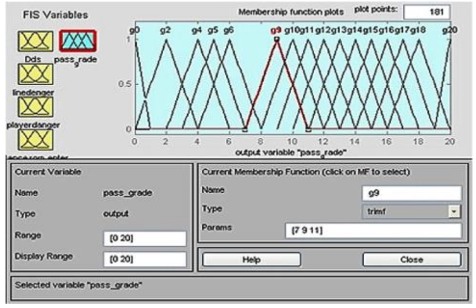

Fig. 17 Membership functions for "pass grade".

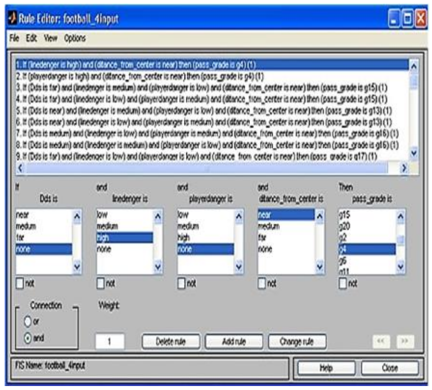

Fig. 18 Implementation of rules using Rule Editor.

Fig. 19 Implementation of inputs [11 3 2 3] and their impact on the simulation of the rules.

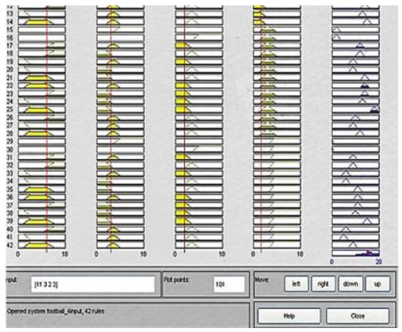

Fig. 19 Implementation of inputs [11 3 2 3] and their impact on the simulation of the rules.

V. USE OF SIMULATION ENVIRONMENT

Figure 20 illustrates the overall view of the simulation environment for the implementation of the passing action.

5.1 Random positioning of players

In order to achieve this, we only need to press the "Random Positioning" button for the players to be positioned in a new randomly selected positions.

5.2 Changing position of a player

This can be done in two ways.

5.2.1 Changing a players position using position setting panel

A players position can be changed using the text edits of "me" and "rival" located in the Position Setting Panel. For this, after clicking on the Edit text button for the desired player, the changes can be made and then applied by using the "SET" button. Figure 21 shows the Position Setting panel.

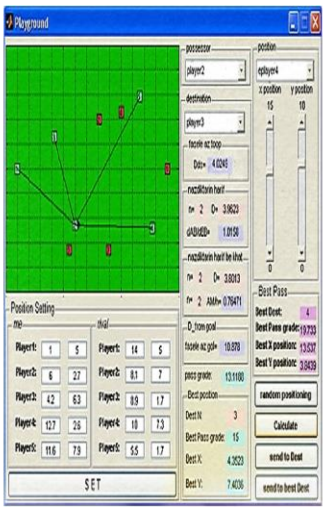

Fig. 20 Overall view of the simulation environment for the implementation of the passing action.

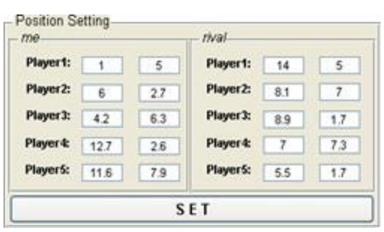

Fig. 21 Position Setting Panel.

5.2.2 Changing a players positions using the x position and y position sliders

It is possible to change a players position using the x and y position sliders. For this, after selecting the desired player from the popup menu, the sliders can be used to change its position. Figure 22 shows the decrease in the y position of number 4 player from the rival team.

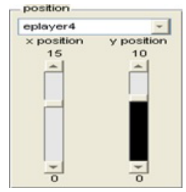

Fig. 22 Decrease in y positon of player number 4 in opposite team using the x and y position sliders.

Fig. 23 Selecting the Player in Possession of Ball.

5.3 Selecting the player in possession of ball

The player who currently holds the ball can be selected using the popup menu in the Possessor Panel. Figure 23 shows the Possessor Panel.

5.4 Selecting the Target Player for performing calculations and Passing

This action can be done using the popup menu in the Destination Panel. Figure 24 shows the Destination Panel.

5.5 Calculations and selecting best player for passing

For this, a player other than the source player is selected and the button is clicked. After the calculations are done, the best player for passing, the pass point, and the best position the destination player can move to are displayed in the Best Pass Panel. This Panel is shown in Figure 25.

5.6 Sending the ball to Destination Player

For this action, we just need to click on " send to DEST ".

Fig. 24 Selecting the Target Player.

Fig. 25 Best Pass Panel.

5.7 Sending the ball to Best Destination Player

In order to send the ball to the best destination player, we just need to click on " send to best DEST ". It should be noted that before clicking on this button, the "Calculate" button is pressed to have the necessary calculations done beforehand.

VI. CONCLUSION

In addition to individual skills and technics of players, the success of a robot soccer team depends team tactics as well. One of the most important high tactic specifications is the ability to give passes. Each and every player must have complete knowledge about the game field and position of each player so that one can identify the best candidate for giving passes to. In this paper, the application of giving passes within mono robot soccer players was studied.

The specifications of a good pass were defined and after extracting suitable rule bases, fuzzy logic system was deployed to use when giving passes. Then a user friendly simulation environment was provided via MATLAB graphical user interface and the prepared fuzzy system was used in it.

REFERENCES

[1] R. Gerndt., D. Seifert, J.Baltes J, Sadeghnejad S. & S. Behnke (2015). Humanoid Robots in Soccer - Robots Versus Humans in RoboCup 2050. IEEE-RAS Robotics Automation Magazine , 22 (3), 147{154.

[2] J. Ruiz-del-Solar, R. Palma-Amestoya, R. Marchanta, I. Parra-Tsunekawa,& P. Zegers (2009). Learning to fall: Designing low damage fall sequences for humanoid soccer robots. Robotics and Autonomous Systems, 57 (x), 796{807.

[3] A.Rezaee, (2017) Designing a system for ordering the opening bidder football players on the field, Majlesi Journal of Mechatronic Systems, 5(4) pp 1-5.

[4] J. Bruce, S. Zickler, M.Licitra & M.Veloso (2008). CMDragons: Dynamic Passing and Strategy on a Champion Robot Soccer Team IEEE International Conference on Robotics and Automation , 4074{4079.

[5] J. Biswas, J.P Mendoza, D. Zhu,B. Choi,S. Klee & M. Velos (2014). CMDragons: Opponent-Driven Planning and Execution for Pass, Attack, and Defense in a Multi-Robot Soccer Team International Conference on Autonomous Agents and Multi-Agent Systems (AAMAS), 493{500.

[6] B. Keho, S. Patil.,P. Abbeel &K. Goldberg (2015). A Survey of Research on Cloud Robotics and Automation IEEE Transactions on Automation Science and Engineering (T-ASE): Special Issue on Cloud Robotics and Automation , 12 (2), 1{11.