MULTIAGENT TEAM FOR PLAYING SOCCERBOTS INSPIRED BY WASP LIKE TASK ALLOCATION

ÁNGEL PANIZO LLEDOT[1]

CONTENTS

LIST OF FIGURES

ABSTRACT

In this paper a multiagent system inspired on wasp like task allocation capable of playing SoccerBots[1] is presented. The several stimulus and actions that they trigger will be described. The performance of this approach will be tested against the teams made for SBTournament[2] by students of the Complutense University of Madrid.

---

1 http://www.cs.cmu.edu/ trb/TeamBots/Domains/SoccerBots/
2 http://gaia.fdi.ucm.es/research/sbtournament
[1] *Agentes inteligentes y sistemas multiagente, Universidad Politécnica de Madrid, Madrid, España*
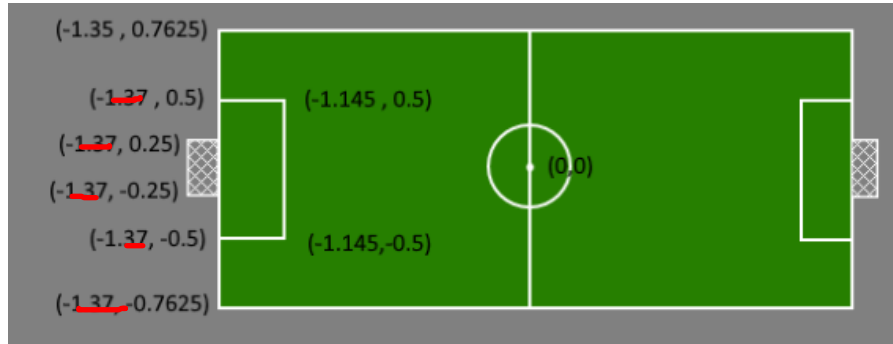
Figure 1: Dimensions of the field

1   INTRODUCTION

We proposed a team of SoccerBots based on a multiagent approach, where each agent is inspired by the wasps behavior allocating tasks and Bonabeau model[1]. Each agent has its own threshold, when a stimuli is bigger than the corresponding threshold the agent does an action in response that stimuli. For example the emptier and nearer the opponent's goal is, from certain player perspective, the bigger the stimuli for shooting. When the corresponding threshold, unique for that agent, is exceeded the agent shoots.

SoccerBots simulates the dynamics and dimensions of a regulation RoboCup[3] small size robot league game, which have 5 players in each team and the dimensions of the field is 152.5cm by 274cm. Each goal is 50cm wide. The *goalkeeper area*, centered on the goal, is 100cm wide by 22.5cm deep. In the coordinate system used in the simulation, the *center of the field* is (0,0) with +x to the right (east) and +y up (north). The team that starts on the left side (or *west field*) is called the *west team* and the team on the right (or *east field*) is called the *east team*. The following schema sketches the main points for the west field (coordinates are in meters). See figure 1.

On top of that the software SBTournament is used which enhanced the SoccerBots and has teams implemented by students of the UCM[4]. These teams will be used to evaluated the efficiency of the multiagent team proposed in this paper.

Other works in this area are: [2] which implements a team using fuzzy rules, [3] where a self-adaptive method is proposed and [4] where a case-based approach is used from which this paper is inspired.

The paper is structured as following: in section 2 the different stimulus which the agent receive are described, in section 3 the territorial division of the field is explained, in section 4 the actions and threshold available for an agent, in section 5 the method for adjusting the threshold, section 6 the result obtained compared to the teams available in SBTournament and finally in section 7 the conclusions and future work.

---

3 https://www.sonycsl.co.jp/person/kitano/RoboCup/RoboCup.html
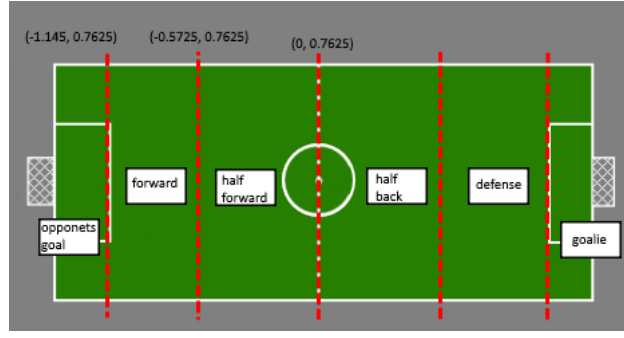4 Universidad Complutense de Madrid

Figure 2: Territories

2 TERRITORIAL TASK DIVISION

Inspired by [5] territorial task division the soccer field was divided into six territories, namely: goalie, defense, halfback, half-forward, forward and opponent's goal. These territories are shown in Figure 2. The total workspace is rectangular area that is bounded by (x_{min}, y_{min}) and (x_{max}, y_{max}). All the workspaces were assigned the same length, which was $|Y_{max} - Y_{min}|$ and the Xs as the Figure 2 shows. These divisions enables the team players to be more distributed and interfere less between each other.

3 STIMULUS

The several stimulus the agent will react to are the: score and remaining time, the defense, competition and angle factors defined in [2] and the: situation, cover and penetration(some changes were made from the original) factors defined in [3] and finally the: shooting and keep goal factors of my own invention. Therefore, their definitions will first be given in this section. All stimulus are functions between 0 and 1.

3.1 Score

This stimuli is the score of the match, 1 if our team is winning by much advantage and 0 if it is losing. The stimuli is defined as follows. For modeling the stimuli the sigmoid function[5] of the difference between our team's score and the opponent's score. The stimuli is defined as follows:

$$f_{score} = \frac{1}{1 + e^{\sigma * (S_m - S_o)}} \tag{1}$$

Where $\sigma = 0.5$, S_m our team's score and S_o the opponents score. $f_{score} = 1$ means our team is winning by far and $f_{score} = 0$ that is losing by far.

3.2 Remaining time

This stimuli is the time remaining to finish the match. The stimuli is defined as follows:

$$f_{time} = 1 - \frac{time}{total\_time} \tag{2}$$

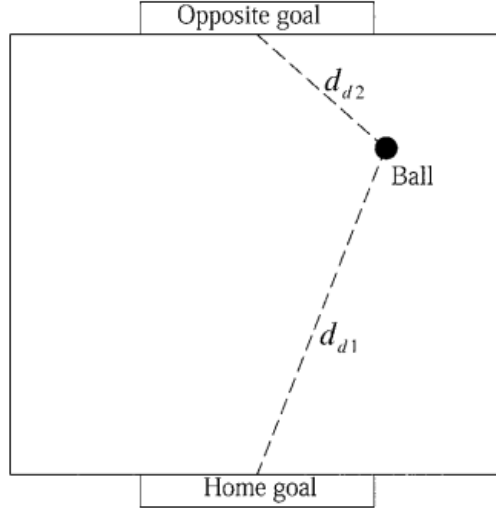5 https://en.wikipedia.org/wiki/Sigmoid_function

Figure 5. Definition of the defense factor.

Figure 3: Defense Factor, image taken from [2]

$f_{time} = 0$ means the match have finished and $f_{time} = 1$ is starting.

3.3 Defense factor

Referring to Figure 5, the defense factor is defined as follows:

$$f_{defense} = e^{-\sigma_d * (d_1/d_2)} \tag{3}$$

Where d_1 is the distance between the ball and the home goal, d_2 is the distance between the ball and the opposite goal, and $\sigma_d = 3$ is a large positive constant.

From the above definition, the value of $f_{defense}$ varies between 0 and 1. $f_{defense} = 1$ and $f_{defense} = 0$ denote that the ball is very close to the home goal and the opposite goal,respectively. If $f_{defense} = 1$, then the robot should play the role of a defender. On the other hand, if $f_{defense} = 0$, then the role of the robot is of a striker.

3.4 Competition Factor

the competition factor is defined as:

$$f_{competition} = \phi_{dist} * e^{-\sigma_1 * (d_1/d_2)} + (1 - \phi_{dist}) * e^{-\sigma_2 * (\theta_1/\theta_2)} \tag{4}$$

Where d_1 is the distance between the ball and the home robot, d_2 is the distance between the ball and the opponent robot, θ_1 is the angle between the ball and the home robot, θ_2 is the angle between the ball and the opponent robot, $\sigma_1 = 3$,$\sigma_2 = 3$ and $\phi_{dist} = 0.7$.

The definition shows that the value of $f_{competition}$ varies between 0 and 1. $f_{competition} = 1$ indicates that the ball is very close to the home robot and the angle between the ball and the home robot is quite small, such
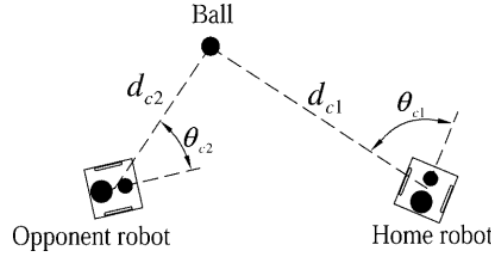
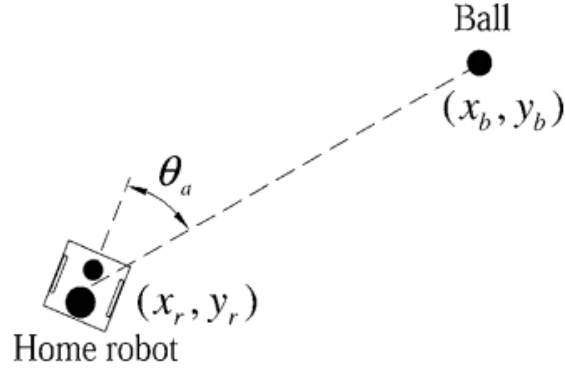**Figure 4:** Competition Factor, image taken from [2]

Figure 5: Angle Factor, image taken from [2]

that the home robot has a much better chance to control the ball than the opponent robot. On the other hand, if $f_{competition} = 0$, then the opponent robot will have an obvious advantage over the home robot.

3.5 Angle factor

The angle factor is defined as follows:

$$f_{angle} = \frac{\|y_b - y_r\|}{\sqrt{(x_b - x_r)^2 + (y_b - y_r)^2}} \tag{5}$$

where (x_b, y_b) is the position of the ball and (x_r, y_r) is the position of the home robot. When $-90 \leqslant \sigma_a \leqslant 90$, it indicates that the ball is in front of the robot such that the robot can access the ball very easily. Otherwise, value of the angle factor is set to zero, since the ball is behind the robot. Therefore, the angle factor can be considered as a measure of the easiness of the robot to access the ball.

3.6 Situation factor

Factor that evaluate how crowded is a territory, the number of team mates and opponents, in a territory, will be taken into account. The next formula describes this factor:

$$f_{situation} = \frac{1}{(1 + e^{-\frac{C_{teams} - C_{opponents}}{\sigma}})} \tag{6}$$
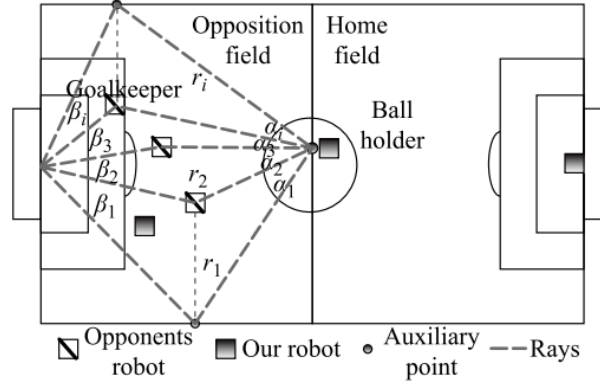
Figure 6: coverage Factor, image taken from [3]

Where $\sigma = 1.3$, C_{team} the team mates on that territory and $C_{opponent}$ the opponents on it. If $f_{situation} = 0$ means that the territory is too crowded with opponents and $f_{situation} = 1$ that the territory is not crowded enough.

3.7 Coverage factor

First, auxiliary points are introduced to facilitate the modeling. Two auxiliary points are located in the upper and lower boundaries of the stadium and their x-coordinates are equal to the x-coordinates of the opponent robot that is closest to the upper and lower boundary.

Second, rays can be made from the ball center to each opponent robot center and auxiliary points, denote the angle between adjacent rays by α_i.

Third, rays are be made from the midpoint of opponent goal line to each opponent robot centers and the auxiliary points, denote the angle between adjacent rays by β_i.

Denote the region between angle α_i and β_i by r_i, $r_i = (\alpha_i, \beta_i)$. The larger the angle $\alpha_i + \beta_i$, the greater possibility for our ball holder to break through the opponent defense and threaten the goal in region r_i. Let the set S comprised the angles $\alpha_i + \beta_j$. $S = \alpha_i + \beta_i, i = 0, 1, ..., n$, and the difficulty to break through the opponents defense in region r_i is inversely proportional to S_i, $S_i \in S$.

The coverage factor is defined as follows:

$$f_{coverage} = 1 - e^{\frac{-r_i}{\sigma}} \tag{7}$$

Where $\sigma = 30$. If $f_{coverage} = 0$ means that is impossible to approach the area, if $f_{coverage} = 1$ means the area is free.

3.8 Penetration factor

The weakest opponent region r_i can be calculated according to set S, $r_i \leftrightarrow max(S) = max(\alpha_i + \beta_i)$. The bisector of angle α_i is $l_{\alpha i}$, and the bisector of
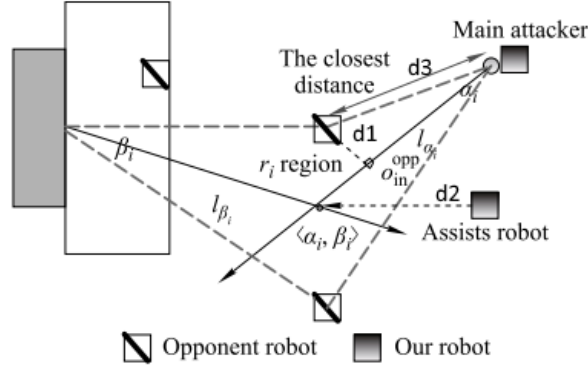
Figure 7: Penetration Factor, image taken from [3]

angle β_i is $l_{\beta i}$. Thus the line $l_{\alpha i}$ is considered to be the optimum offensive route of the main attacker in region r_i.

The penetration factor depends of the distant of the closest opponent robot to the line $l_{\alpha i}$, d1 in the figure, and to the main attacker. If these two distances are high the *penetration factor* will also be high and the agent will dribble through the line $l_{\alpha i}$.

The distant of other assists robots to the line $l_{\alpha i}$, d2 in the figure, will be also taken into account with the distance d3 to form the *Penetration assist factor*. If d3 is small and d2 is also small the main attacker will kick the ball in the $l_{\alpha i}$ direction.

The *Penetration factor* and *Penetration assist factor* functions are defined as follows:

$$f_{penetration} = (1 - e^{\frac{-d_3}{\sigma_1}}) + 0.5 * e^{\frac{-d_3}{\sigma_1}} * (1 - e^{\frac{d_1}{\sigma_2}}) \tag{8}$$

$$f_{penetration} = 1 - \frac{1}{2} * e^{\frac{-d_3}{\sigma_1}} - \frac{1}{2} * e^{\frac{-d_3 * \sigma_2 - d_1 * \sigma_1}{\sigma_1 * \sigma_2}} \tag{9}$$

$$f_{penetrationAssist} = e^{\frac{-\sigma_3 * d_3 - \sigma_1 * d_2}{\sigma_1 * \sigma_3}} \tag{10}$$

Where:
$\sigma_1 = 0.28$
$\sigma_3 = 0.16 = \sigma_2$

If $f_{penetration} = 1$ the player should dribble along l_α, if 0 should not. Also if $f_{penetrationAssist} = 1$ the player should kick the ball along l_α, if 0 should not.

3.9 Shooting factor

The shooting factor depends of the distant d_1, d_2 and d_3. the smaller d_1 and d_3 are and the bigger d_2, the bigger this factor will be. The function $f_{shooting}$ is defined as follows:

$$f_{shooting} = (1 - e^{\frac{-d_2}{\sigma_2}}) * (e^{\frac{-d_1}{\sigma_1}} - \phi_{penalizacion} * (1 - e^{\frac{-d_3}{\sigma_3}})) \tag{11}$$
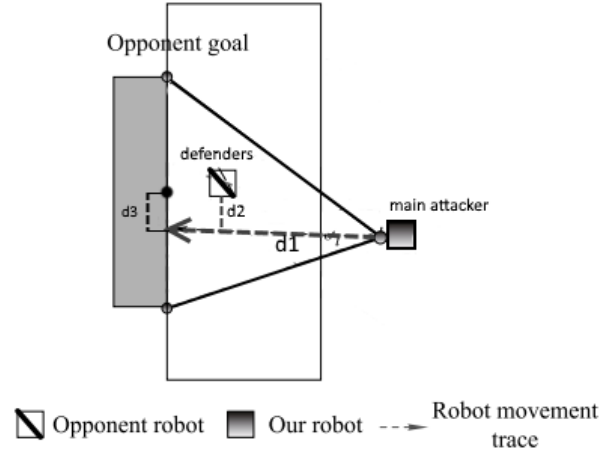
Figure 8: Shooting Factor

$$f_{shooting} = e^{\frac{-d_1}{\sigma_1}} - e^{\frac{-d_1*\sigma_2-d_2*\sigma_1}{\sigma_1*\sigma_2}} + \phi_{penalizacion}*(-1-e^{\frac{-d_2}{\sigma_2}}+e^{\frac{-d_3}{\sigma_3}}-e^{\frac{-d_3*\sigma_3-d_2*\sigma_2}{\sigma_3*\sigma_2}})$$

(12)

Where:
$\sigma_1 = 0.33$
$\sigma_2 = 0.16$
$\sigma_3 = 0.5$
$\phi_{penalizacion} = 0.4$

$$f_{intercept} = e^{\frac{-d_1}{\sigma_1}} * e^{\frac{-d_2}{\sigma_2}} * e^{\frac{-d_3}{\sigma_3}}$$

(13)

$$f_{intercept} = e^{\frac{-d_1*\sigma_2*\sigma_3-d_2*\sigma_1*\sigma_3-d_3*\sigma_1*\sigma_2}{\sigma_1*\sigma_2*\sigma_3}}$$

(14)

Where $\sigma_1 = 0.13$, $\sigma_2 = 0.05$ and $\sigma_3 = 0.04$

If $f_{shooting} = 1$ the player should shoot thought d_1, if 0 should not.
If $f_{intercept} = 1$ the player should intercept the ball through d_1, if 0 should not.

3.10   Keep Goal factor

Use the distance of each player inside the goal area. The function decrements with the number of players in the area. The function is defined as follows:

$$f_{keepGoal} = \frac{5 - \Sigma_i^5 e^{\frac{-d_i}{\sigma}}}{5}$$

(15)

Where $\sigma = 0.05$ and d_i the distance of the player i to the center of the goal. If $f_{keepGoal} = 0$ the goal is keep right , if $f_{keepGoal} = 1$ the goal need to be keep.
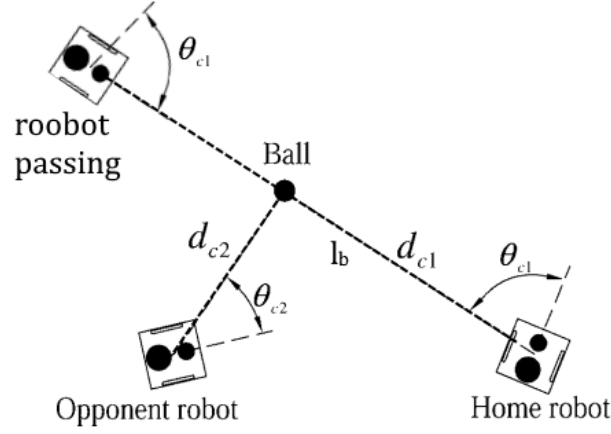
Figure 9: Diagram for the pass action

4 ACTIONS AND THRESHOLDS

In this section all the actions available to a player and their respective thresholds are described. Not all actions are available to a certain player at any moment, for example for some actions like *shoot* or *pass* the player need to be in control of the ball. Only one action can be taken so when more than one thresholds are exceeded the one exceeded by most will be the one selected. The probability of selecting a undefined action i for certain player j, is defined as follows:

$$P_{action_i} = \frac{StimulusFunction(action_i)^{\sigma_1}}{StimulusFunction(action_i)^{\sigma_1} + \varphi_j * \theta_j^{\sigma_2}} \quad (16)$$

Where σ_1 and σ_2 are positive integer numbers, φ_j is a number between [0,1] and θ_j is the threshold of the player j for action i.

4.1 With the ball

These actions can only be made by the player who has the ball

4.1.1 *Advance*

Move a player forward, depends of: the situation factor of the next territory, the defense factor and the coverage factor. The stimuli function is defines as follows.

$$S_{advance} = (\varphi_{sn} * f_s(next) + (1 - \varphi_{sn}) * f_c(player)) * (f_d) \quad (17)$$

Where:
$\varphi_{sn} = 0.30$
$f_s(next)$ is the situation factor next territory
$f_c(player)$ is the coverage factor of the player
f_d is the defense factor.

If this action is taken the player will: dribble along the *coverage factor* line.

4.1.2   *Pass to player*

The player i pass the ball to other player j, depends on the: coverage factor of player i and player j, the distance between them, situation factor of player's j area and the competition factor between the: player j , he's nearest opponent and the middle of the line l_a between player j and player i. The stimuli function is defined as follows:

$$S_{pass}(i) = \varphi_c * \frac{1}{(1 + e^{-\mu * \frac{f_c(j)}{f_c(i)} + \beta})} + \varphi_d * e^{-\sigma * l_b} + \varphi_{cp} * f_{cp}(j, opp) + \varphi_s * f_s(j)$$

(18)

Where:
$\varphi_c = 0.40$
$f_c(j)$ is the coverage factor of player j.
$f_c(i)$ is the coverage factor of player i.
$\mu = 10$
$\beta = 15$
$\varphi_d = 0.25$
$\sigma = 1.5$
l_b is the line in the figure.
$\varphi_{cp} = 0.25$
$f_{cp}(j, opp)$ is the competition factor of j and the opponent robot
$\varphi_s = 0.1$
$f_s(j)$ is the situation factor of j's territory.

If this action is taken, the player i pass the ball along l_b to player j.

4.1.3   *Shoot*

The players shoot to the opponent's goal, depends on the: shooting, defense and penetration assist factors. The stimuli function is defined as follows:

$$S_{shoot} = (1 - f_d) * (\varphi_s * f_s + (1 - \varphi_s) * (1 - f_{pa}))$$

(19)

Where:
f_d is the *defense factor*
f_s is the *shooting factor*
$\varphi_s = 0.5$
f_{pa} is the *penetration assist* factor

If this action is taken the player shoots along the *shoot factor* d_1 line.

4.1.4   *Pass assist*

The players pass the ball to a team mate that is assisting , depends on the: shooting, defense and penetration assist factors. The stimuli function is defined as follows:

$$S_{passAssist} = (1 - f_d) * ((1 - \varphi_s) * f_{pa} + \varphi_s) * (1 - f_s))$$

(20)

Where:
f_d is the *defense factor*
f_s is the *shooting factor*
$\varphi_s = 0.5$

f_{pa} is the *penetration assist* factor

If this action is taken the player pass the ball along the *penetration factor* l_α line.

4.1.5 *Clear*

The players clear the ball towards the opponent's field, depends on the: defense factor and situation factor. The stimuli function is defined as follows:

$$f_{clear} = f_d * ((1 - f_s(actual)) \tag{21}$$

Where:
f_d is the *defense factor*
$f_s(actual)$ is the situation factor of the actual territory

If this action is taken, the player kick the ball along the middle of the bigger coverage area.

4.2 Without the ball

These actions can only be made by players who does not has the ball.

4.2.1 *Advance*

Move a player forward, depends of the: situation factor of the next territory, defense and situation of the actual territory factors. The stimuli function is defines as follows.

$$S_{advance} = \varphi_s * \frac{1}{(1 + e^{-\mu * \frac{f_s(next)}{f_s(actual)} + \beta})} + (1 - \varphi_s) * (1 - f_d) \tag{22}$$

Where:
$\varphi_s = 0.40$
$\mu = 7$
$\beta = 7$
$f_s(next)$ is the situation factor of the next territory
$f_s(actual)$ is the situation factor of the actual territory
f_d is the defense factor.

If this action is taken the player will advance to the less crowded area. To calculate this first a vector in opposite direction of every opponent, in the field of view of the player, and with module proportional to the distance of the opponent to the player are calculated on the players position. Second the same is done for the team mates but in the same direction. Third all the vector are summed, if the direction on x axis is not pointing to the opponent's goal, the direction in that axis is inverted.
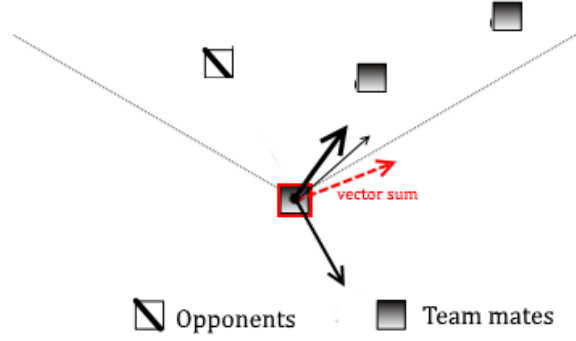
Figure 10: Diagram for the calculation of the less crowded area

4.2.2 *Retreat*

Move a player back to it's field, depends of the: situation factor of the previous territory, defense and situation of the actual territory factors. The stimuli function is defines as follows.

$$S_{retreat} = \varphi_s * \frac{1}{(1 + e^{-\mu * \frac{f_s(actual)}{f_s(prev)} + \beta})} + (1 - \varphi_s) * (f_d) \qquad (23)$$

Where:
$\varphi_s = 0.40$
$\mu = 7$
$\beta = 7$
$f_s(prev)$ is the situation factor of the previous territory
$f_s(actual)$ is the situation factor of the actual territory
f_d is the defense factor.

If this action is taken the player will retreat to the area with more opponents, this can be calculated the same way of the *advance action* but swapping the opponents and the team mates.

4.2.3 *Block player*

Move the player i to block the opponent j, depends of the: *j's* assist factor and penetration factor and competition factor between i and j. The stimuli function is defined as follows:

$$\varphi_c * (1 - f_c(i, j)) * (\varphi_a * f_a(j) + (1 - \varphi_a) * f_p(j)) \qquad (24)$$

Where $\varphi_c = 1$, $f_c(i, j)$ is the competition factor between players i and j, $\varphi_a = 0.4$, $f_a(j)$ is the assist factor of j and $f_p(j)$ is the penetration factor of j. If this action is taken the player i will go to block the player j.

4.2.4 *Cover goal*

Move the player to cover the goal, depends of the: defense and keep goal factors. The stimuli function is defined as follows:

$$S_{coverGoal} = (1 - \varphi_d) * (f_d) + \varphi_d * f_k \qquad (25)$$

Where:
$\varphi_d = 0.4$

f_d is the defense factor
f_k is the keep goal factor

If this action is taken the player moves to less crowded area of the goal.

4.2.5 *Intercepts*

Move the player to the intercept the ball, depends only of the *intercept factor*. If this action is selected the player move to the nearest point of d_1.

4.2.6 *Assist*

Move the player trough the line d_2 in the *penetration factor* figure. ~~Depends of the defense and assist factor. The Stimuli function is defined as follow:~~

$$S_{assist} = (1 - f_d) * (f_{pa}) \qquad (26)$$

~~If this action is taken, the player moves along d_2 line of the *penetration factor* figure.~~

4.2.7 *Go to ball*

Move the player to the ball. ~~depends only of the competition factor. If this action is taken the player moves along d_1 of the *competition factor* figure.~~

4.2.8 Avoid coilision: if blocked avoid the colission.

5 ADJUSTING THRESHOLDS: POSITIVE NEGATIVE FEEDBACK

In Bonabeu model [1] the agent adjust their threshold when a task is performed, this approach need to be changed. Threshold will only be decreased for the action the players were doing if a beneficial action is done, for example retrieving the ball from the opponents, scoring , passing correctly ...etc. As Bonabeu proposed the reinforcement will be:

$$\theta_j = \theta_j - \epsilon(f_s, b_j) \qquad (27)$$

$$\epsilon(f_s, b_j) = valor(b_j) + \left(\frac{1}{1 + e^{-\sigma * f_s + \alpha}} - 0.5\right) * valor(b_j) \qquad (28)$$

Where θ_j is the threshold of a player to action j ,which was the one that the player did when the beneficial action happened, $\epsilon(f_s, j)$ is a function that depends of the: *score factor* and the beneficial action that triggers the update b_j.$\sigma = 8$, $\alpha = 4$ and $valor(b_j)$ is the ones in the table below.

action	value
score	0.2
retrieve ball	0.2
~~intercepts ball~~	0.2
correct pass	0.~~1~~2
~~player with ball~~ enter opponents area	0.1
ball kick off our area	0.1
~~player with ball exit our field~~	0.1
ball exit our field	0.05

All the threshold will be restored to normal as time goes by, until a maximum is reach, so if not good action are happening the player forgets his bias. This forget function is as follows:

$$\theta_j = \theta_j + \phi(f_s, f_t) \tag{29}$$

$$\phi(f_s, f_t) = \mu - (0.5 * f_s * \mu + f_t * (\frac{1}{1 + e^{-\sigma * f_s + \alpha}} - 0.5) * \mu) \tag{30}$$

Where θ_j is the threshold of a player to action j and $\phi(f_s, f_t)$ is a function that depends of the: *score factor*, *time factor*. $\sigma = 8$, $\alpha = 4$.

6   RESULTS

7   CONCLUSIONS AND FUTURE WORK

REFERENCES

[1] Eric Bonabeau, Andrej Sobkowski, Guy Theraulaz, and Jean-Louis Deneubourg. Adaptive task allocation inspired by a model of division of labor in social insects. In *BCEC*, pages 36–45, 1997.

[2] Chia-Ju Wu and Tsong-Li Lee. A fuzzy mechanism for action selection of soccer robots. *Journal of Intelligent and Robotic Systems*, 39(1):57–70, 2004.

[3] Haobin Shi, Lincheng Xu, Lin Zhang, Wei Pan, and Genjiu Xu. Research on self-adaptive decision-making mechanism for competition strategies in robot soccer. *Frontiers of Computer Science*, 9(3):485–494, 2015.

[4] Raquel Ros, Josep Lluís Arcos, Ramon Lopez De Mantaras, and Manuela Veloso. A case-based approach for coordinated action selection in robot soccer. *Artificial Intelligence*, 173(9):1014–1039, 2009.

[5] Khashayar R Baghaei and Arvin Agah. Multi-agent task allocation for robot soccer. *Journal of Intelligent Systems*, 16(3):207–240, 2007.