FUZZY LOGIC CONTROLLER FOR SHOOTING ACTION OF AN INTEGRATED MULTI-AGENT ROBOT SYSTEM

Daniela COMAN, Anca PETRIŞOR
University of Craiova, Romania

Abstract. An integrated multi-agent robot system for robot soccer games consists of multiple mobile robots, a vision system, a wireless communications system and a host computer. The multiple robots can be cooperatively controlled as they play a robot soccer game in an unknown and dynamic environment. The development of the system involved mechanism design and manufacture, integration of the electromechanical system, vision system, pattern recognition, decision-making, wireless communications, motion control and centralized coordination control.

This paper proposes a fuzzy logic controller for shooting action which is one of the fundamental actions for soccer robots. The shooting action is view as a posture control problem with such a constraint that the robot should not approach the ball so that the ball moves to the home team area.

The fuzzy logic controller consists of two levels: one is the planner level that generates a path to the ball with obstacle avoidance. The other is the motion control level that outputs robot wheel velocities to follow the desired path given the robot's current posture. The effectiveness of the proposed fuzzy logic controller is verified by the simulation results.

Keywords: fuzzy logic controller, integrated multi-agent robot system, path planner, obstacle avoidance

1. Introduction

Multiple robots can cooperatively perform tasks that are difficult for a single robot. Coordination of multiple robots increases the flexibility and robustness of the system at the expense of increasing the complexity involved in performing tasks. The advantages of multi-agent robot systems have led to their extensive use in the area of manufacture automation. Although multi-agent robot systems can perform very complicated cooperative tasks in determinate and static environments, they cannot work effectively in unknown and dynamic circumstances. Therefore, an intelligent and adaptive multi-agent robot system applicable to unknown and dynamic environments is desired [1, 2].

In robot soccer different kinds of system configurations exist. The configuration focused in this work is called MiroSOT (Micro Robot Soccer Tournament) [3]. In this configuration two teams play soccer against each other on a black playground with a golf ball. The dimensions of the playground are 1.5 m of length and 1.3 m of width. A team consists of three robots and three additional team members. One or two host computers per team control the robots. The size of the robots is limited to a cube with an edge length of 75 mm. Each robot has to be marked on its top with at least one color. This is the team color and has to be according to the rules blue or yellow. In addition the robots can be marked with other colors in order to differentiate them within the team on the one hand and on the other hand to detect not only their position, but also their orientation on the playground. The color of the ball, which is a golf ball, must be orange.

The positions of the moving objects on the playground are detected by the aid of their color information. A camera is mounted at 2 m above the playground, which transmits images to the host computer. In the host computer a vision system detects the position and orientation of the moving objects. This information is the basis for decision making. With a radio communication the desired movement of the robots is sent back to the robots.

The shooting action is one of the fundamental actions for soccer robots. The shooting action is view as a posture control problem with such a constraint that the robot should not approach the ball so that the ball moves to the home team area (figure 1). The following two constraints are considered:

- the robot should approach the ball from the opposite side of the line connecting the ball and the opponent goal. In other words, it should always approach the ball so as to "bump" it towards the opponent goal;

- the robot should avoid obstacles that are not very close to the ball on the field.

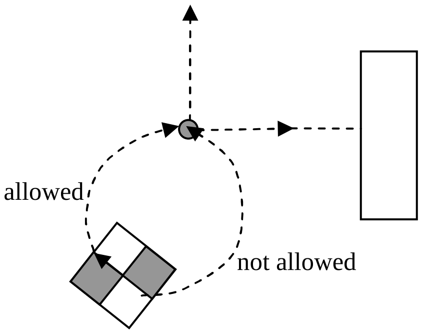


Figure 1. Shooting action

Such methods as uni-vector field method, geometric calculation method, and heuristics method have been utilized to solve these constraints so far in the robot soccer competitions [4, 5, 6].

Fuzzy logic has been used widely in controlling mobile robots, such as path following or local obstacle avoidance [7]. In most cases the desired path is given by a higher level planner.

In this paper is proposes a fuzzy logic controller for shooting action for soccer robots.

In Section 2 the overall fuzzy logic control structure is described which consists of a fuzzy planner and a fuzzy motion controller. They are explained in Sections 3 and 4, respectively. In Section 5 simulation results are reported.

2. Overall fuzzy controller structure

2.1. Integrated multi-agent robot system

The integrated multi-agent robot system for robot soccer games consists of four major sub-systems: multiple mobile robots, a vision system, a wireless communication system and a host computer [8, 9]. The overall integrated system is depicted in figure 2.

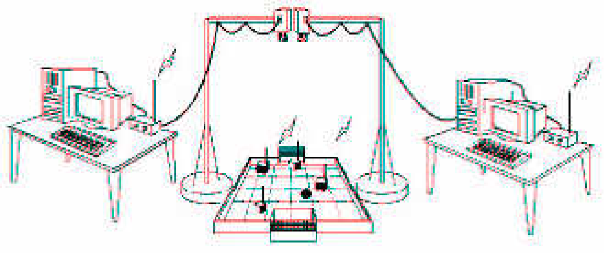


Figure 2. Overall integrated system

The robot for soccer games is designed as a mobile platform (Figure 3), which can be used for any kind of multi agent system and consists of the following parts: power supply, two DC-motors with digital encoder, single stage gear, two wheels, micro controller for controlling the rotation of the DC-motors, power electronic and radio module to send tasks to the robot.

Vision system locates the objects (the robots and the ball) on the field by the global camera fixed above the field and the host computer calculates strategies and decides actions for each robot.

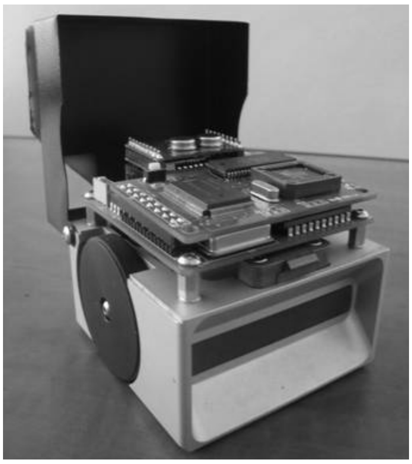


Figure 3. A robot soccer

2.2. Overall fuzzy controller

The relative posture of a robot is characterized by three variables (ρ, φ, θ) where the polar coordinate (ρ, φ) is the robot's position relative to the ball's, as depicted in figure 4. These variables are needed to implement the shooting action in view of the above mentioned constraints.

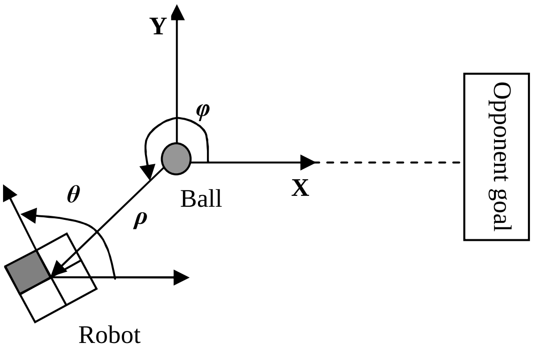


Figure 4. Variables for relative posture characterization

Therefore, the fuzzy logic controller, as shown in figure 5 consists of two levels: one is the planner level that generates a path to the ball with obstacle avoidance. The other is the motion control level that outputs robot wheel velocities to follow the desired path given the current robot posture.

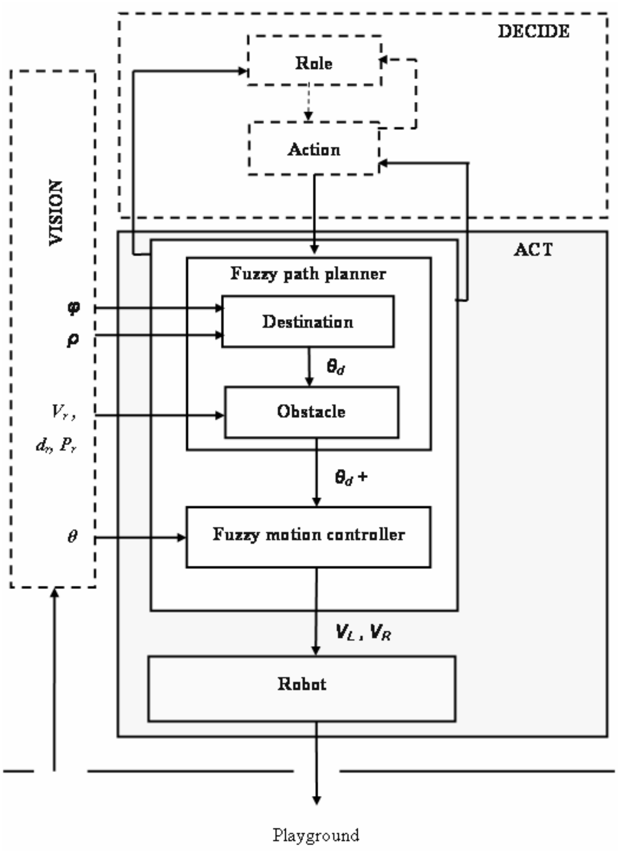


Figure 5. Overall fuzzy control structure

3. Fuzzy planner

The fuzzy planner generates a desired global path by calculating the robot's desired heading angle θ_d at each relative position (ρ, φ), without violating the two constraints. It comprises two blocks: the destination block that generates a path which leads the destination (the ball) and the obstacle block for obstacle avoidance.

3.1. Destination block

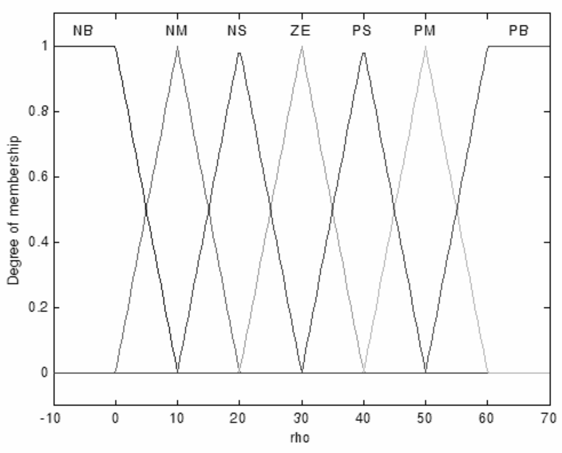
The destination block determines the desired heading angle θ_d at each relative position (ρ, φ).

The fuzzy input is the relative position of the robot to the ball (ρ, φ) and the output for the destination block is the desired heading angle at that position θ_d.

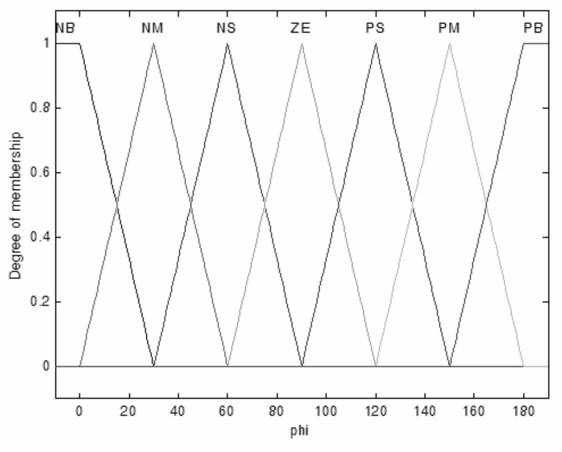
We decompose the input linguistic variables ρ and φ into 7 fuzzy term sets, which are denoted by NB (Negative Big), NM (Negative Medium), NS (Negative Small), ZE (Zero), PS (Positive Small), PM (Positive Medium) and PB (Positive Big).

The input variables membership functions are depicted in figure 6.

The output θ_d has singleton values obtained at sampled positions (at each input region around the ball at (0, 0)), as shown in figure 7.



a) For ρ



b) For φ

Figure 6. Membership functions

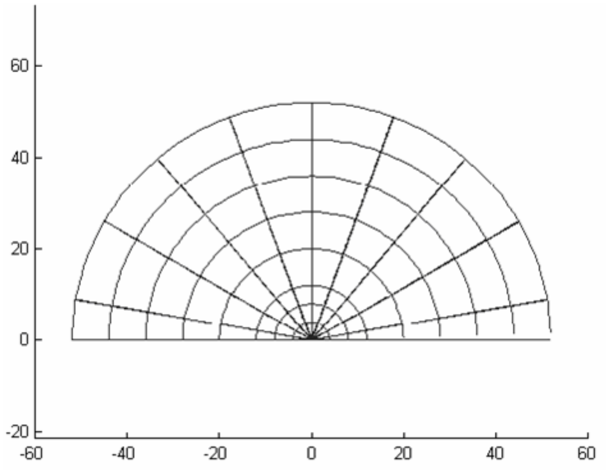


Figure 7. Sampled output

According to the designed membership functions, 7×7 fuzzy rules should be determined. These resultant rules for the destination block are represented in table 1.

Table 1. Rule for the destination block

θ_d	*P*						
φ	*NB*	*NM*	*NS*	*ZE*	*PS*	*PM*	*PB*
NB	-270.0	-216.9	-202.6	-196.3	-192.7	-190.4	-188.8
NM	-270.0	-201.5	-180.0	-171.0	-166.1	-163.0	-161.0
NS	-200.0	-187.9	-155.5	-143.6	-137.6	-134.0	-131.6
ZE	-170.0	-180.0	-126.9	-120.6	-114.3	-110.7	-108.4
PS	-140.0	-135.0	-80.0	-76.9	-71.8	-69.0	-67.4
PM	-20.0	-30.0	-34.2	-35.9	-35.5	-40.2	-45.4
PB	0.0	0.0	0.0	0.0	0.0	0.0	0.0

3.1. Obstacle block

The obstacle block determines the desired heading if there are any obstacles nearby. Four variables are utilized to obtain the angle θ_f in the presence of obstacles. These variables are:

- velocity V_r;
- direction D_r;
- distance d_r;
- position P_r (positive if the obstacle is in front and negative otherwise).

The velocity V_r and direction D_r, are necessary to obtain the escape radius R_s so as to avoid any obstacle that is either stationary or moving, as shown in figure 8.

The distance d_r and position P_r are used to obtain the proportional gain, w_0, which is multiplied with θ_s to produce θ_f .

The angle θ_s is calculated with relation:

$$\theta_S = \tan^{-1}\left(\frac{R_S}{d_0}\right). \tag{1}$$

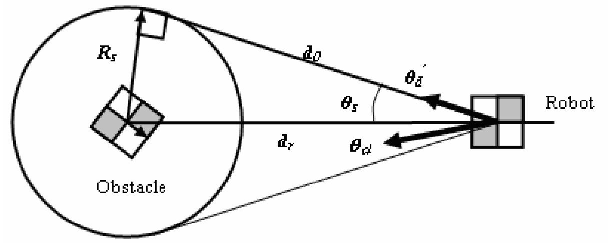


Figure 8. Obstacle avoidance

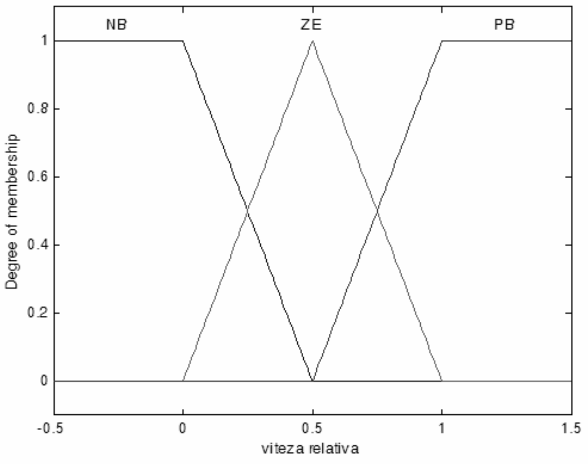
The input space (V_r, D_r, d_r, P_r), the velocity, direction, distance and position of an obstacle relative to the robot are defined as follows:

$$\begin{aligned} &V_r \in [0.5,\ 1.5] \\ &D_r \in [0\ \text{deg.},\ 180\ \text{deg.}] \\ &d_r \in [0\ cm,\ 90\,cm] \\ &P \in [-0.5,\ 1.5] \end{aligned} \tag{2}$$

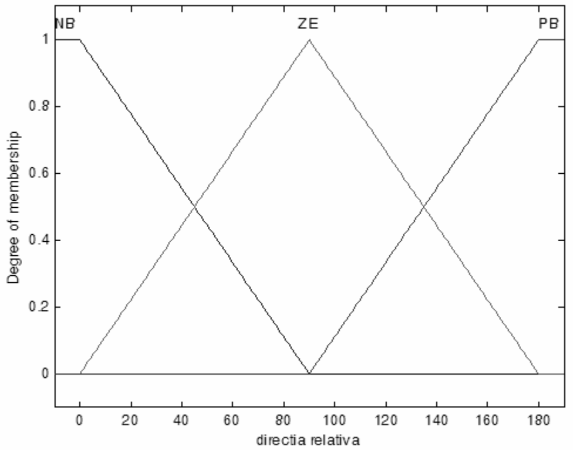
The output space θ_f, the offset angle to be added to θ_d is defined as follows:

$\theta_f, \in$ [0 deg., 180 deg.].

The input variables membership functions are depicted in figures 9 and 10.

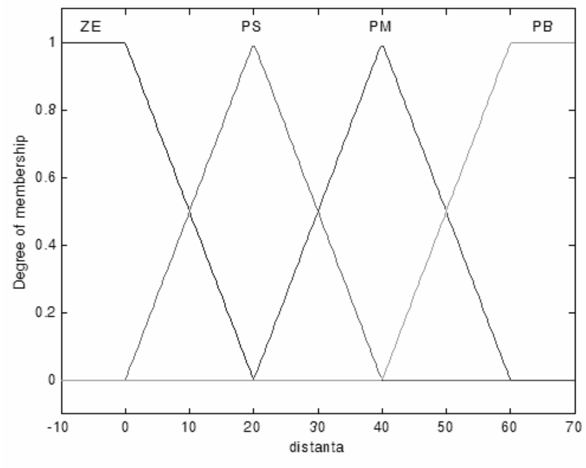


a) For V_r

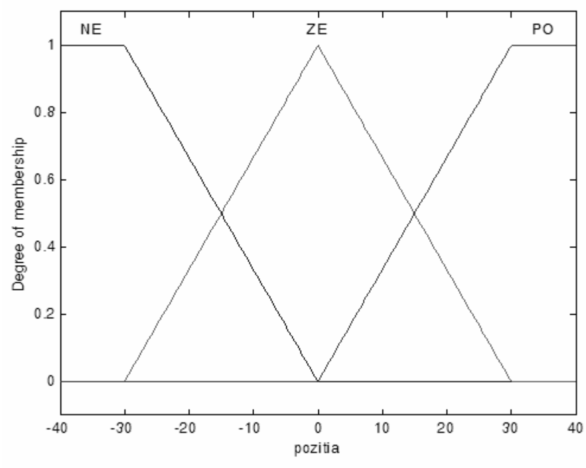


b) For D_r

Figure 9. Membership functions



a) For d_r



b) For P_r

Figure 10. Membership functions

According to the designed membership functions, 9 fuzzy rules are obtained for R_s, with V_r and D_r each characterized by three linguistic variables (*NB*, *ZE*, *PB*). 12 fuzzy rules are obtained for w_0, with P_r characterized by three linguistic variables (*NE*, *ZE*, *PO*) and d_r characterized by four linguistic variables (*ZE*, *PS*, *PM*, *PB*). These resultant rules for the obstacle block are represented in table 2.

Table 2. Rule for the obstacle block

R_s	D_r		
V_r	*NB*	*ZE*	*PB*
NB	20	20	20
ZE	20	25	30
PB	20	35	40

w_0	d_r			
P_r	*ZE*	*PS*	*PM*	*PB*
NE	0.8	0.7	0.6	0.0
ZE	1.0	1.0	0.9	0.0
PO	1.0	1.0	1.0	0.0

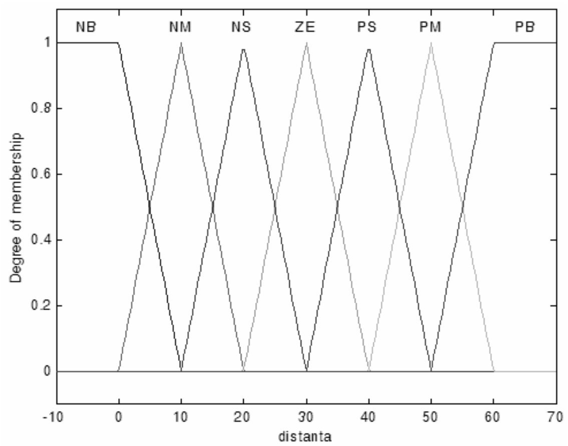
4. Fuzzy motion controller

In the overall structure of figure 5, the fuzzy motion controller block receives θ_d from the fuzzy planner block and part of robot posture information (ρ, θ) from the vision processing system. Then the motion controller block generates appropriate left and right wheel velocities to make θ follow θ_d at non-zero linear speed before ρ diminishes.

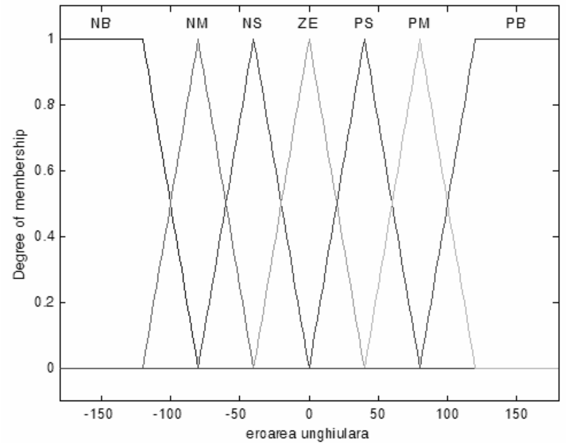
For this conventional problem of mobile robotics, the following heuristics are incorporated:

- if ρ big → V_L, V_R big.
- if $|\theta_e| = |\theta'_d - \theta|$ big → $|V_L - V_R|$ big.

The input variables membership functions are depicted in figure 11.



a) For ρ



b) For θ_e

Figure 11. Membership functions

According to the designed membership functions, 49 fuzzy rules are acquired each for left and right wheel velocities. Table 3 is the rule table for the right-wheel speed V_R, with ρ and θ_e each characterized by seven linguistic variables (*NB*, *NM*, *NS*, *ZE*, *PS*, *PM*, *PB*). The left-wheel speed is symmetrical about the X-axis (i.e., $\varphi = 0$).

Table 3. Rule for the right wheel

V_R	ρ						
φ	*NB*	*NM*	*NS*	*ZE*	*PS*	*PM*	*PB*
NB	-35	-27	-27	-3	-3	-3	-3
NM	-25	8	8	18	31	31	42
NS	-15	15	22	35	57	67	67
ZE	30	30	50	60	90	100	100
PS	15	40	44	65	82	92	92
PM	25	51	51	61	68	68	77
PB	35	63	63	67	67	67	67

5. Simulation

The simulation is performed based on the following robot kinematics:

$$\theta(k+1) = \theta(k) + \frac{v_R(k) - v_L(k)}{D} \cdot T_s \qquad (3)$$

$$x(k+1) = x(k) + \frac{v_R(k) - v_L(k)}{2} \cdot \cos\frac{\theta(k)+\theta(k+1)}{2} \cdot T_s \qquad (4)$$

$$y(k+1) = y(k) + \frac{v_R(k) - v_L(k)}{2} \cdot \sin\frac{\theta(k)+\theta(k+1)}{2} \cdot T_s \qquad (5)$$

Robot’s physical characteristics are:

- wheelbase (D) = 6.7 cm
- wheel radius (r) = 3.15 cm
- sampling interval (T_S) = 10 ms.

Figure 12 shows the simulation results with and without the obstacle avoidance block. The initial heading angle of all the trials is 0 deg. The obstacles are marked with ’o’ and they are well avoided as seen from figure 12.

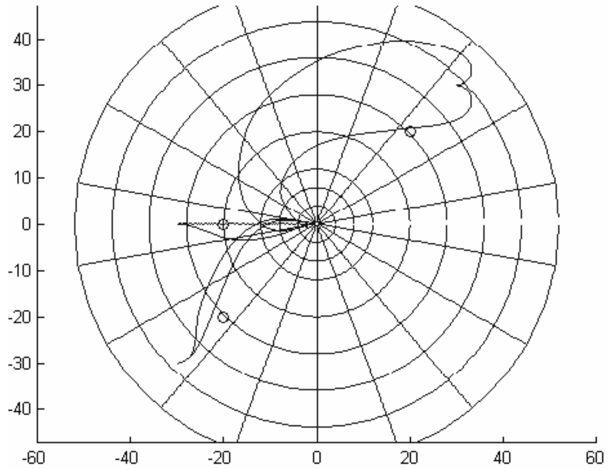


Figure 12. Simulation results with and without obstacles

In the simulation it was observed that smooth velocity profiles were generated as shown in figure 13.

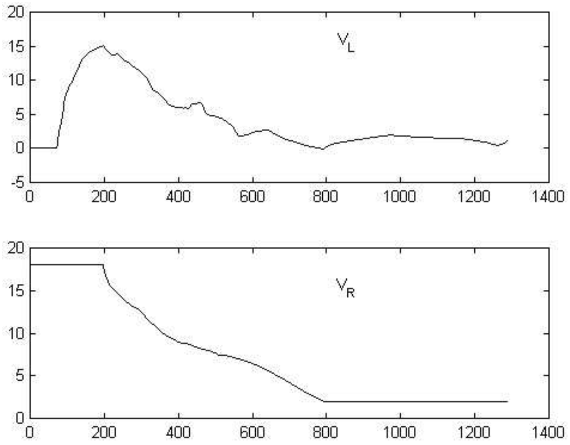


Figure 13. Velocity profile

References

1. Shen, W.M., Adibi, J., Adobbati, R., Cho, B., Erdem, A., Moradi, H., Salemi, B., Tejada, S.: *Building integrated mobile robots for soccer competition*. IEEE Int'l. Conf. on Robotics and Automation, 1998, p. 2613-2618, Leuven, Belgium
2. Park, S.W., Kim, J.H., Kim, E.H., Oh, J.H.: *Development of a multi-agent system for robot soccer game*. IEEE Int'l. Conf. on Robotics and Automation, 1997, p. 626-631, Albuquerque, NM, U.S.A
3. ***: http://www.fira.net. Accessed: 2006-11-05
4. Springer, G., Hannah, P., Stonier, R.J., Smith, S., Wolfs, P.: *Simple strategies for collision-avoidance in robot soccer*. Robotics and Automous Systems, Vol. 21, No. 2, 1997, p. 191-196
5. Shim, H.S., Kim, H.S., Jung, M.J., Choi, I.H., Kim, J.H., Kim, J.O.: *Designing distributed control architecture for cooperative multi-agent system and its realtime application to soccer robot*. Robotics and Automous Systems, Vol. 21, No. 2, 1997 p. 149-166
6. Dong-Han, K., Yong-Jae, K., Kwang-Choon, K., Jong-Hwan, K., .Prahlad, V.: *Vector Field based Path Planning and Petri-net based Role Selection Mechanism with Q-learning for the Soccer Robot System*. Int. J. Intelligent Automation and Soft Computing, Vol. 6, No. 1, 2000, p. 75-87
7. Ishikawa, S.: *A method of autonomous mobile robot navigation by using fuzzy control*. Advanced Robotics, Vol. 9, No. 1, 1995, p. 29-52
8. Mizuno, H., Kourogi, M., Kuroda, Y., Muraoka, Y.: *An architecture for soccer shooting robot*. ALIFE V, p. 7-12, Kyoto, Japan, 1996
9. Kim, J.H., Kim, D.H., Kim Y.J., Seow, K.T.: *Soccer Robotics*. Springler, 2004