International Journal of Sciences and Techniques of Automatic control & computer engineering
IJ-STA, Volume 5, N° 2, December 2011, pp. 1558–1575.

Design of Fuzzy Logic Controller for Autonomous parking of Mobile Robot

Ahmed Hechri[1,2], Anis Ladgham[1], Fayçal Hamdaoui[1] and Abdellatif Mtibaa[1,2]

[1] Laboratory EµE, Faculty of Sciences of Monastir, University of Monastir.
[2] National School of Engineering of Monastir, Av Ibn ElJazzar 5019 Monastir.

Ahmed.hechri@enim.rnu.tn; Ladghamanis02@gmail.com; faycel_hamdaoui@yahoo.fr and Abdellatif.mtibaa@enim.rnu.tn

Abstract. *An autonomous parking controller can provide convenience to a novice driver. However, if the controller is not designed adequately, it may endanger the car and the driver. Therefore, this paper presents a fuzzy logic controller designed for automatic parking system. The proposed system receives information about the parking environment from infrared sensors, and then generates the velocity and the steering angle of the robot to complete the parking tasks. The developed controller is simulated based on the kinematics models of three wheeled mobile robot. The simulation results under a variety of scenarios illustrate the effectiveness of the developed controllers. In order o ensure the performance of the proposed system, a prototype of three wheels mobile robot, equipped with a microcontroller control system, has been developed. An extensive experimentation has shown a good performance for different parking space.*

Keywords. *Mobile robot, fuzzy logic controller,trajectory.*

1. Introduction

Currently, an increasing amount of the robotic researches has focused on the development of new technologies that increase the autonomy of the car. For this, many researchers have developed algorithms for the autonomy of their vehicles / robots to assist drivers during all phases of driving. The parking problem of a vehicle is how we can find the parking and how it handles to complete this task adequately [1-10]. In this work, we propose an autonomously parking controller based on the fuzzy logic using measurements of infrared sensors as inputs.

To resolve this problem, [8] used the CCD camera for the overall vision of the car park which is an expensive method, [10] combined the ultrasonic sensors, encoders,

This paper was recommended for publication in revised form by the editor Staff.
Edition: CPU of Tunis, Tunisia, ISSN: 1737-7749

gyroscopes and a differential GPS system to detect and estimate the dimensions of parking which is also very expensive and imprecise for the use of the GPS and [12] used the ultrasonic sensors to discover its environment. In this paper, we will use information from infrared sensors to detect environmental parking.

This paper is organized as follows. Section 2, presents the designed three wheels mobile robot. Then, kinematic and dynamic model of the mobile robot were described. Section 3 addresses the parking lot measurement and the fuzzy logic controller of monitoring of wall. Section 4 shows the simulation curves. Experimental results are given in Section 5. Section 6 concludes this paper.

2. Robot presentation

2.1. Robot design

The robot is built with 16F877 microcontroller, L298D, IR sensors and IP camera (see figure 1). It has interfaced a ring of 6 IR sensors distributed around the perimeter of the robot. The front sensor is to explore if displacement of parking is empty or not, R1 and R2 are used to measure the dimensions of this displacement when the parking is done in the right of the street, L1 and L2 are for the left of the street and the rear sensor is placed to stop the robot when it is near wall in the end of the parking.

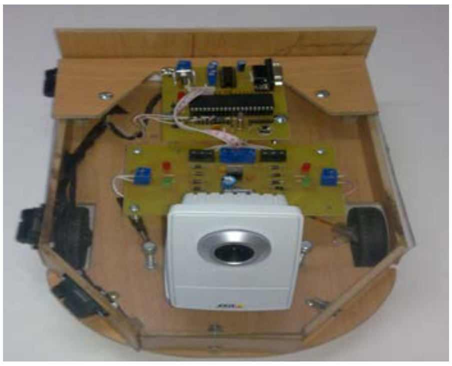


Fig. 1. (a) Mobile robot platform

The PIC16F877 microcontroller was selected for the main controller for several reasons. First of all, its size is small and equipped with sufficient output ports without having to use a decoder or multiplexer. Another significant advantage was for the presence of the PWM inside the chip itself which allow us to vary the duty cycle of DC motor drive easily.

2.2. Motor

Two DC motors are used in this robot as shown in figure 5. Control for the two motors in the system is carried out by using the L298D (figure2.a), microcontroller, enable and disable the motor excitation elements using the internal H bridges in the circuit. The selected motor (6VDC) operates at 230rpm and current up to 2 amperes (maximum). DC motors are generally bi-directional motors. That is, their direction of rotation can be changed by just reversing the polarity. But once the motors are fixed, control becomes tricky this is done using the H-Bridge type of motors. The following table resembles the movements of the robot according to the state of ENABLE, INPUT1, INPUT2, INPUT3 and INPUT4.

(a)

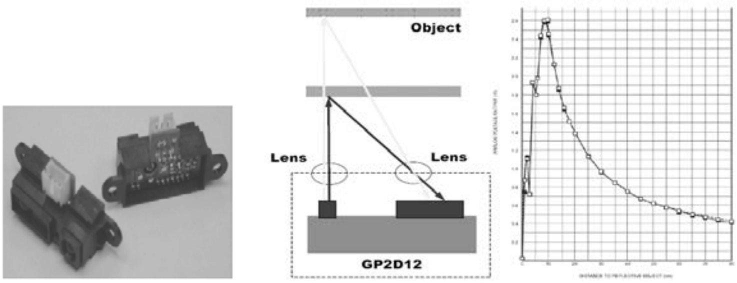
ENABLE_A	INPUT1	INPUT2	INPUT3	INPUT4	MOTEUR1	MOTEUR2	DIRECTION
0	X	X	X	X	Roue libre	Roue libre	Arrêt
1	1	0	1	0	Sens1	Sens1	Avant
1	0	1	0	1	Sens2	Sens2	Arrière
1	1	0	0	1	Sens1	Sens2	Gauche
1	0	1	1	0	Sens2	Sens1	Droite
1	0	0	0	0	Stop	Stop	Arrêt
1	1	1	1	1	Stop	Stop	Arrêt

(b)

Fig.2. (a) LM298D (b) Robot movements

2.3. IR sensor

In this paper, three GP2D12 infrared sensors are utilized for distance measurements. The infrared sensor consists of a LED emitting the infrared light and a position sensing devise (PSD) that outputs voltage based on results of the triangulation procedure. Measurement of the distance using triangulation is illustrated in Figure 3.



(a) (b)

Fig.3. (a) GP2D12 IR Sensors (b) Triangulation Measurements

The angles in a triangle connecting the IR emitter, a distant object and PSD are dependent on a position of the object with respect to LED/PSD plane and therefore, are used to calculate the distance between IR sensor and the object.

Typical Output/Distance Characteristics of GP2D12 are presented in Figure 3. Maximum range that can be detected using the GP2D12 is from 10 to 80 cm.

3. Robot model

Actually, there are a lot of types of mobile robots. Each type requires a command type and a kinematic study different from each other. For this, we choose the simplest type and least costly to achieve for testing the effectiveness of our controller.

The robot realized in this research is uni-cycle driven by two independent wheels and having a loose wheel ensuring its stability.

The parameters describing the vehicle motion are given by the following figure:

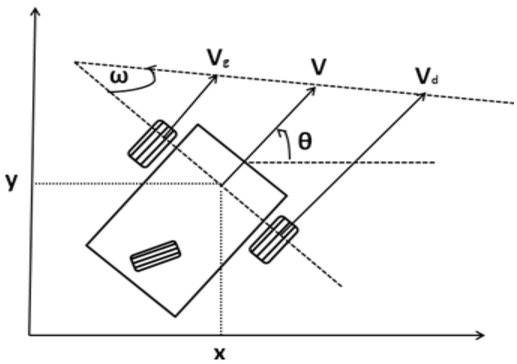


Fig. 4. Displacement parameters of the robot

Its movement is described by the following kinematic model (1):

$$\dot{r} = \begin{pmatrix} cos\theta & 0 \\ sin\theta & \mathbf{0} \\ 0 & \mathbf{1} \end{pmatrix} \begin{pmatrix} Vlin \\ Vang \end{pmatrix} \qquad (1)$$

With **r** $\dot{r} = (\mathbf{x}\,\mathbf{y}\,\theta)^{T}$ where the pair (x, y) denotes the coordinates of the center of gravity of the robot in the mark R (O, $\vec{\mathbf{x}}, \vec{\mathbf{y}}$) and θ is the angle representing its orientation relative to the axis (O,$\vec{\mathbf{x}}$) of the same mark.

Vd and Vg are respectively the velocity of the right and the left wheel. **Vlin** is the robot's linear velocity and **Vang** is its angular velocity. They are obtained as shown in (2) and (3):

$$Vlin = \frac{Vd + Vg}{2} \qquad (2)$$

$$Vang = \frac{Vd - Vg}{2} \quad (3)$$

The dynamic model is used to translate the voltages applied to each engine (U1, U2) in linear and angular velocity ($\mathbf{v}$, $\mathbf{w}$) according to the following transfer functions (4) and (5):

$$\frac{v}{Ul} = \frac{K_1}{1 + \tau_1 . p} \quad (4)$$

$$\frac{w}{Ua} = \frac{K_2}{1 + \tau_2 . p} \quad (5)$$

Where $\mathbf{Ul} = \frac{\mathbf{U1 + U2}}{\mathbf{2}}$ and $\mathbf{Ua} = \frac{\mathbf{U1 - U2}}{\mathbf{2}}$

K1, K2, τ1 and τ2 were determined by open loop tests on engines [11].

Simulations were done on the Matlab Simulink environment. Thus figure 2 shows the basic modules used which includes the kinematic model of the robot which is detailed in figure 3 and the dynamic model that includes the two blocks of the DC motors that contain their transfer functions (velocity = f(voltage)).

Angular and linear velocities which are the model kinematic's inputs are obtained from the angular velocities of the two motors Vangd and Vangg.

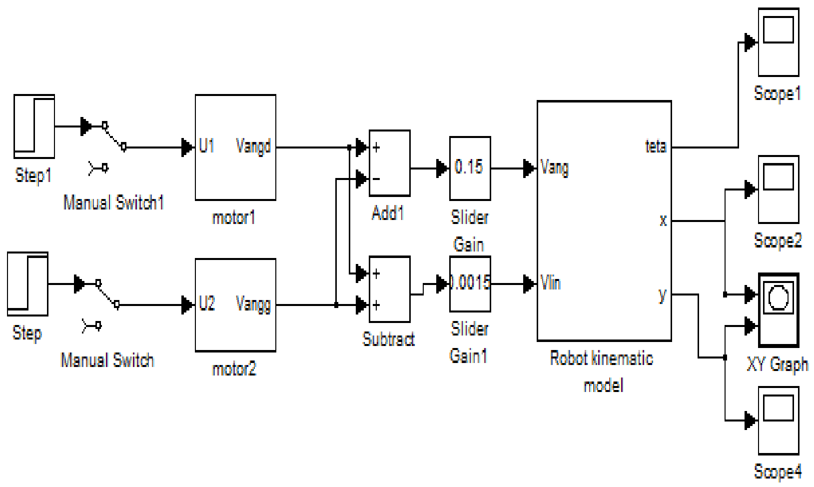


Fig. 5. Model of the robot under "Simulink"

This module shows the kinematic model of the robot, we use the trigonometric functions of matlab to elaborate this block.

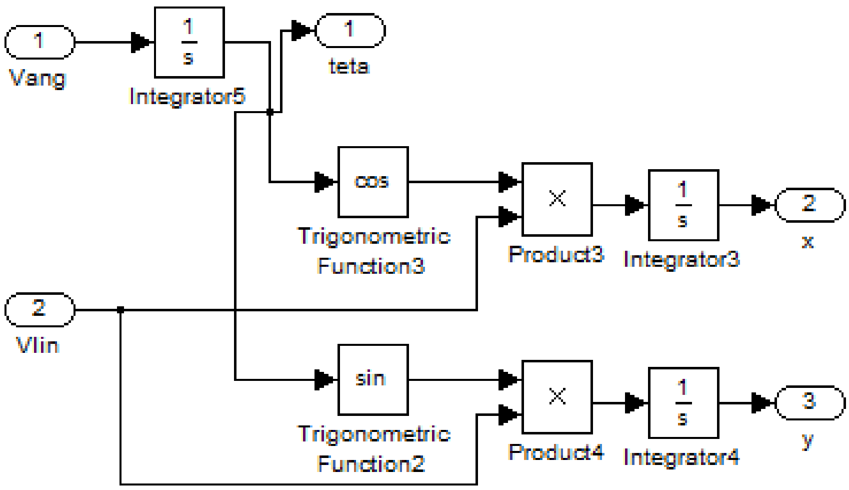


Fig.6. Internal architecture of the block "robot kinematic model"

4. Robot control system

4.1. Parking steps

In this paragraph we will explain the parking steps. The robot begins by searching the wall, following it, keeping a security distance noted D (for this study D = 10cm), to seek a location where it can do the autonomous parking.

<u>First step</u>

To perform the parking task of our project, the robot must, first, look for the parking which is modeled by the wall in our case. To perform this step, the procedure is as follows:

- If the wall is very far from the robot so that the distance is greater than the range of distance meters, the robot will move forward while deviating to the right for a small angle. Once the wall is within the reach of rangefinders, the robot will move closer to the latter until it reaches the distance to keep from him during the monitoring phase will be initiated immediately.

- If the range placed behind, finder detects that the robot approaches orthogonally to the wall during the research phase of the latter, a right turn command will be generated to prevent the robot hits the wall.

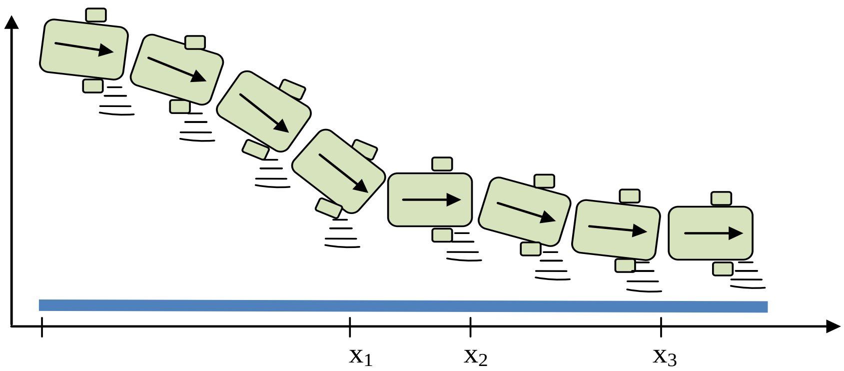


Fig. 7. The wall searching of the robot

The algorithm of this step can be clearly shown in the flow chart below.

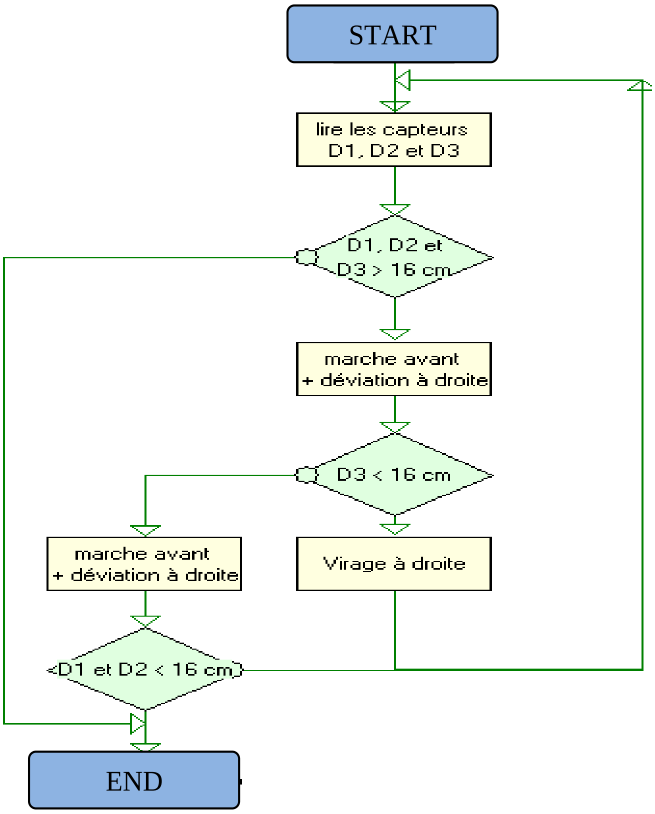


Fig.8. Wall searching step flow chart

Second step

First, we fix a distance which the robot must keep to the wall when they are parallel. If it exceeds this distance, he tries to come back immediately. The wall tracking algorithm works as follows:

- If the two infrared sensors front and rear give the same voltage (the margin is chosen according to the selected speed), so it has the same distance between the sensors and the wall. In this case, the microcontroller provides the same pulse width for both motors, and the robot will displace forward with a speed proportional to the pulse width data.

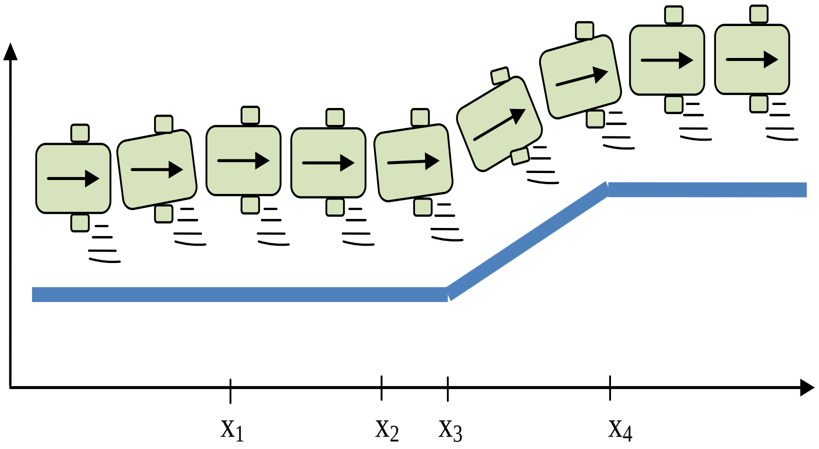


Fig. 9. The wall searching of the robot

- If both front and rear sensors provide two different voltages then we have the following two cases:

• If the front sensor provides a voltage higher than the rear sensor (the inequality is not strict, There are a certain amount equal to the first case) then the robot will turn left to avoid striking the wall. In this way, the robot corrects its position until both sensors give the same indication as the first case (the robot returns parallel to the wall).

• If the front sensor provides a voltage lower than the other then the robot will turn right.

The algorithm of this step can be clearly shown in the flow chart below.

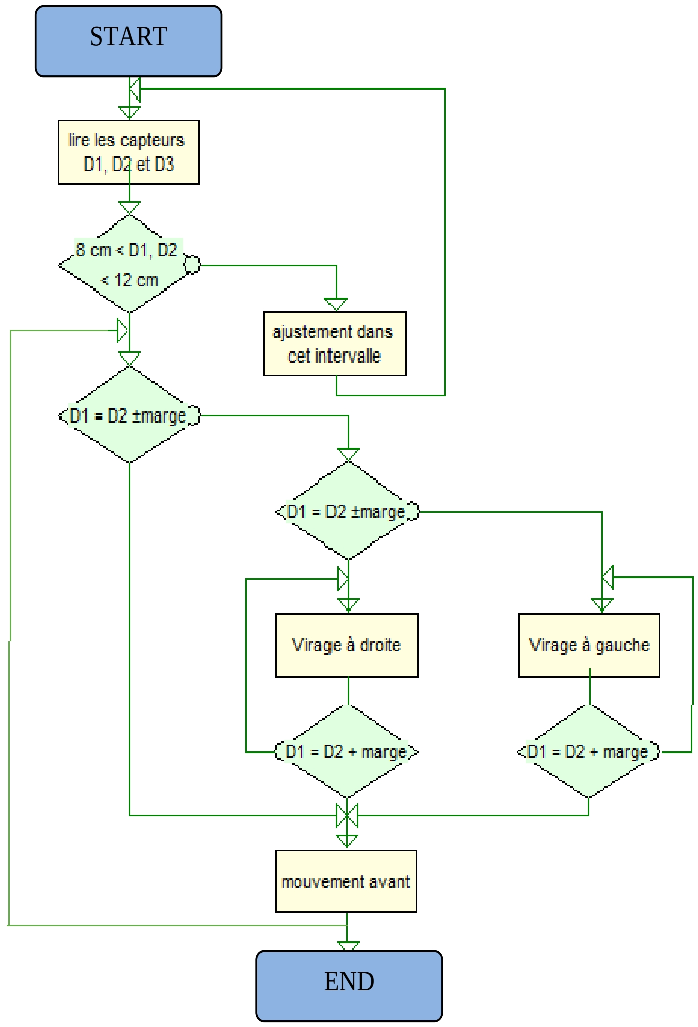


Fig.10. Wall following step flow chart

Third step

First state of the parking maneuvers, when the robot moves in a straight forward direction, searching for a free parking space. If the space is too small, the state returns to Searching for parking space. If the space is large enough, the robot move backward and the steering wheel is turned to the right, so in this way the vehicle starts to enter in the parking space. After the alignment with the parking space is done, the state changes to Stopped and the maneuver is terminated with success.

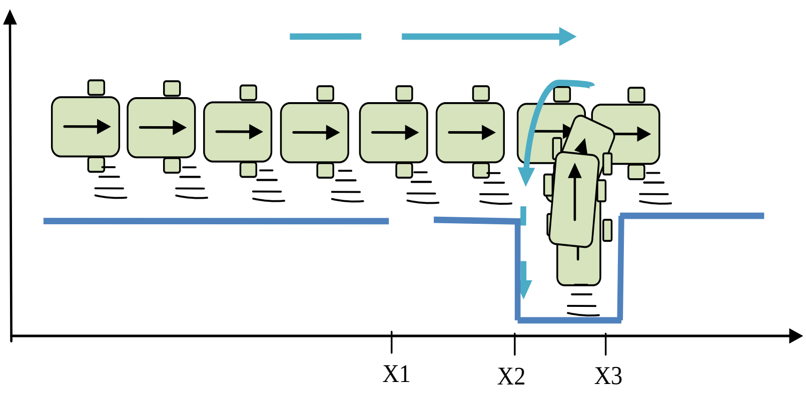


Fig. 11. Parking step of the robot

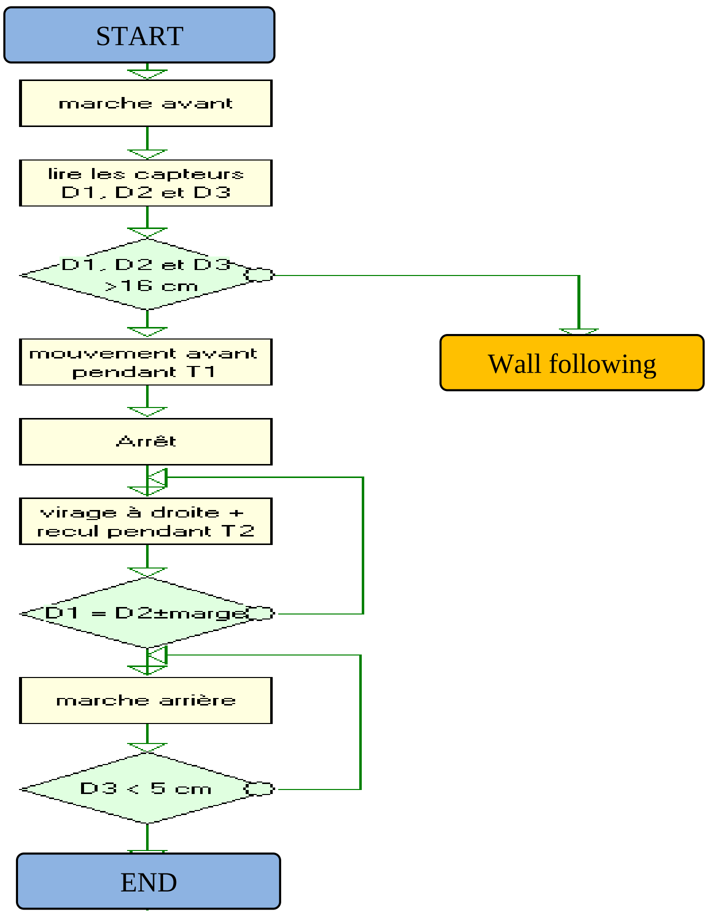


Fig.12. Parking step flow chart

4.2. System description

Firstly, the robot should search the wall in an autonomous way, whatever its position it needs to find the wall. Secondly, we address the fuzzy wall following logic controller. For this step, we use only the two sensors R1 and R2 as follow:

- If dR1 = dR2 then the robot is parallel to the wall.
- If dR1 < dR2 then the robot approaches the wall.
- If dR1 > dR2 then the robot is away from the wall.

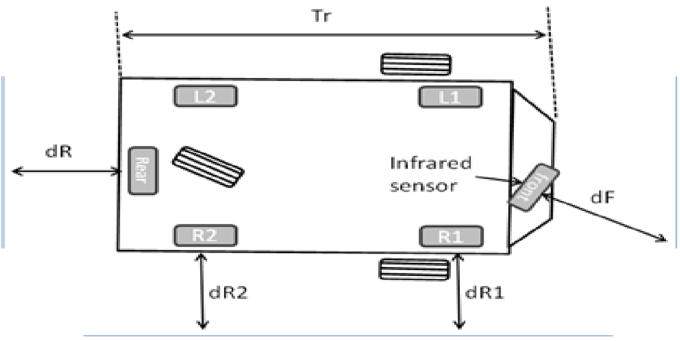


Fig.13. Sensors arrangement of the robot

XR1 and XR2 are the inputs and Φ is the output to our controller. XR1 and XR2 are respectively the differences between dR1 and dR2 and the safety distance D that are defined in equations (6) and (7) and Φ is the steering angle of the robot vehicle.

$$XR1 = dR1 - D \quad (6)$$

$$XR2 = dR2 - D \quad (7)$$

We choose the triangular form for the fuzzy membership functions for the inputs XR1 and XR2 and the output Φ . These functions are shown in figures 14 and 15 follows.

μ_{XR1}, μ_{XR2}

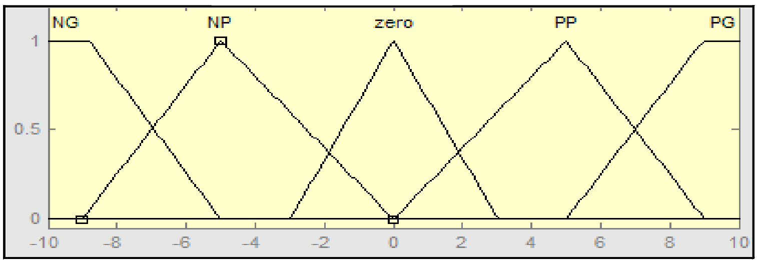


Fig. 14. Fuzzy membership function for the inputs XR1 and XR2

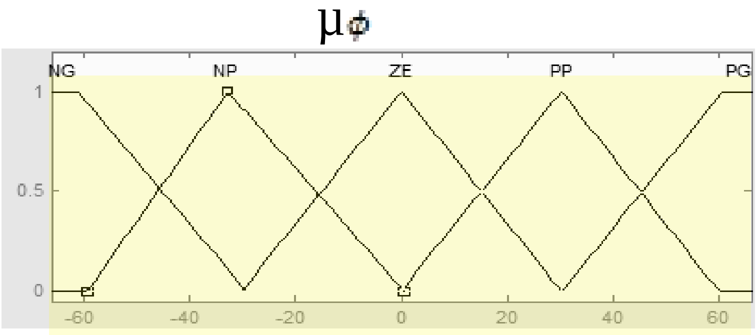


Fig.15. Fuzzy membership function for output

The fuzzy rules are giving below; it contains 11 rules that give the steering angle of the mobile robot. This angle each angle is achieved by providing a suitable control for each wheel.

If	(XR1 is PP)	and	(XR2 is PP)	then	(ϕ is NP)
If	(XR1 is NP)	and	(XR2 is NP)	then	(ϕ is PP)
If	(XR1 is ZE)	and	(XR2 is ZE)	then	(ϕ is ZE)
If	(XR1 is NP)	and	(XR2 is ZE)	then	(ϕ is ZE)
If	(XR1 is ZE)	and	(XR2 is NP)	then	(ϕ is ZE)
If	(XR1 is PP)	and	(XR2 is NP)	then	(ϕ is NP)
If	(XR1 is NP)	and	(XR2 is PP)	then	(ϕ is PP)
If	(XR1 is PG)	and	(XR2 is NG)	then	(ϕ is NG)
If	(XR1 is NG)	and	(XR2 is PG)	then	(ϕ is PG)
If	(XR1 is PG)	and	(XR2 is ZE)	then	(ϕ is NP)
If	(XR1 is NG)	and	(XR2 is ZE)	then	(ϕ is PP)

Finally we will talk about parking step, in time when the robot is following the wall, if the sensor R1 detects a distance greater than Tr and the front sensor detects a gap, the vehicle-robot continues to walk until dR1and dR2 become greater than Tr then the location of parking is found. After this, the vehicle continues to walk and it only stops when dR2 ≈ dis to finally make the parking maneuvers.
If the front sensor detects an obstacle in the location of parking, the vehicle continues to navigate even if dR1> Tr.

4.3. Simulation results

Following curves show the simulated movement of the robot when tracking wall. The figure 16 shows the distances measured by R1 and R2 sensors when monitoring the wall.

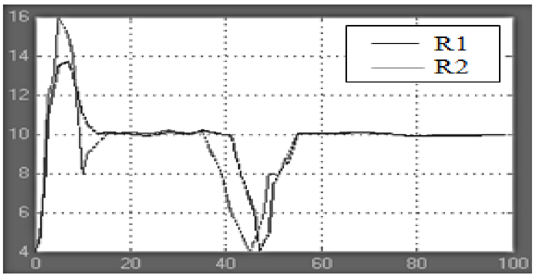


Fig. 16. Distances calculated by the sensors R1 and R2 when monitoring Wall

Between 0 and 15s the shapes of the curves exceed the 10 cm and reach the 16 cm which implies that the robot is in a right turn, between 15 cm and 35 cm there is no turn and the robot is parallel to the wall. Between 35 and 55 cm, the curves show a turn in the left. Finally, for the rest the robot is parallel to the wall.

Figure 17 bellow shows the trajectory of the robot in the plane (x,y) corresponding to the distances measured by R1 and R2 sensors in figure 13.

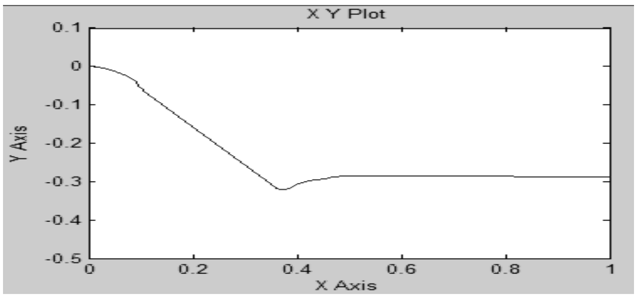
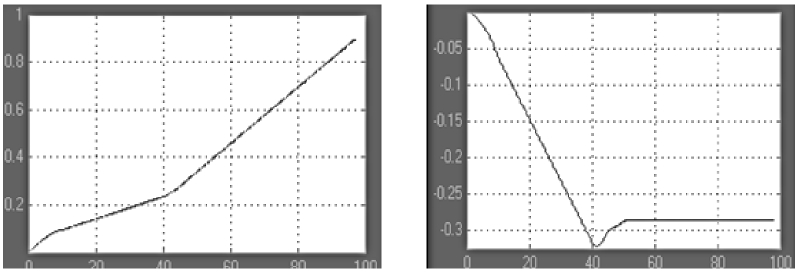


Fig. 17. Circuit of the robot when tracking the wall

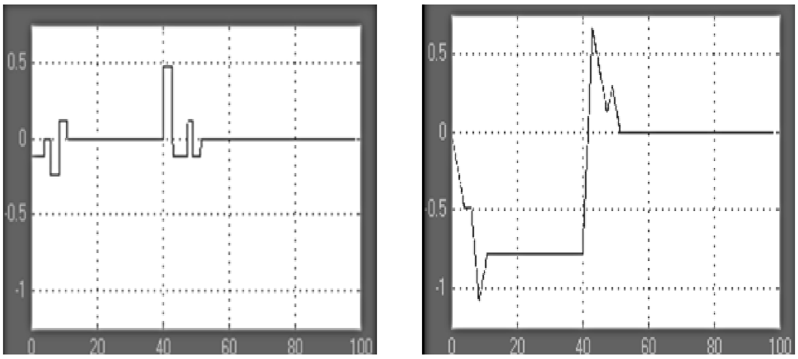
Displacement along x only and along y only as functions of time are shown in the figure bellow.



(a) (b)

Fig. 18. (a) Displacement along the x-axis (b) displacement along the y-axis

Steering angle and angular vitesse of the robot when performing the last trajectory are shown in figure 19 bellow.

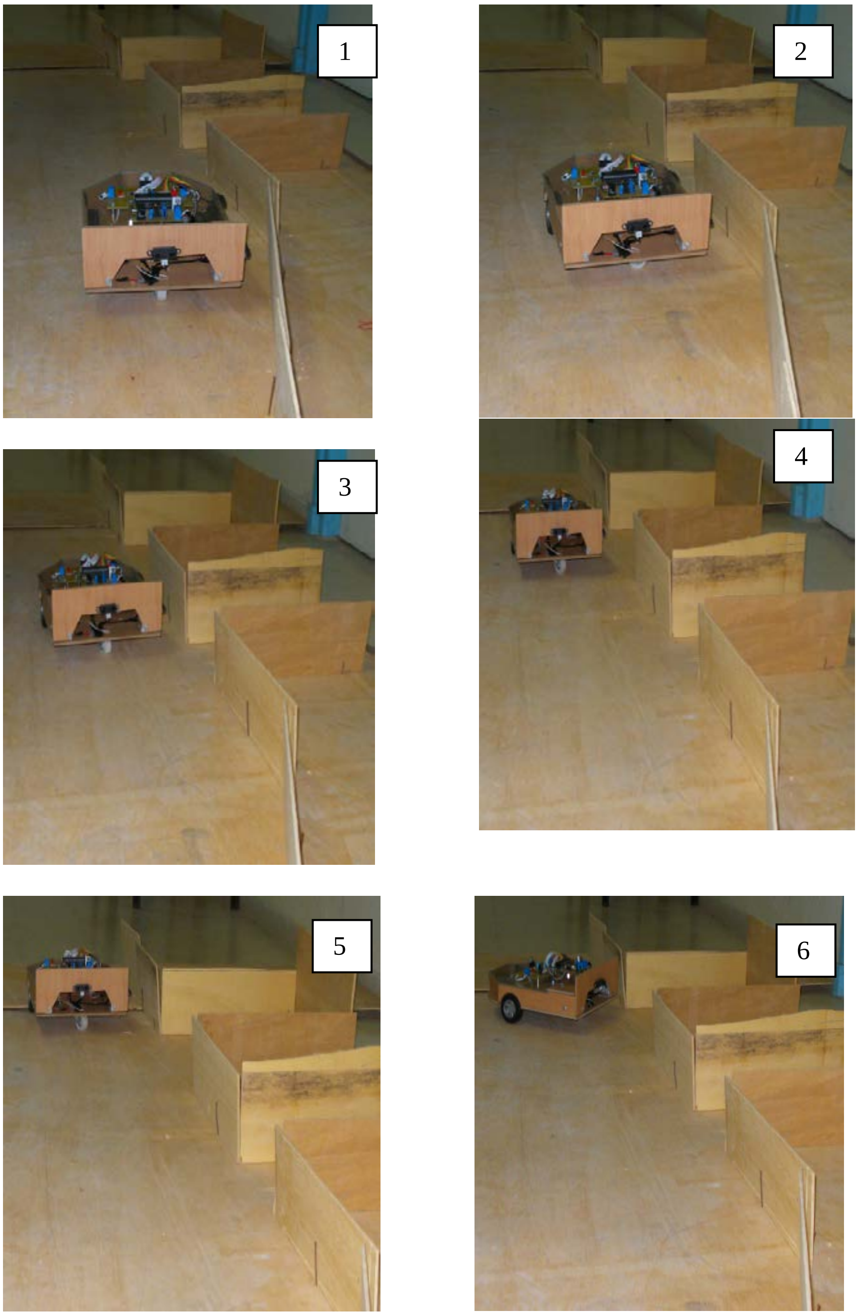


(a) (b)

Fig. 19. (a) Steering angle (b) angular velocity

5. Experimental results

Figure 20 shows that our mobile robot is stationary in the proper location. Images 1 and 2 show how the robot follows the adequately wall, image 3 shows that it continues the wall following when it discovers that dimensions of the first displacement are smalls then its size. The other figures 4,5,6,7 and 8 show that the robot founds a displacement for parking that supports its size, the robot will park in this place.


1
2
3
4
5
6

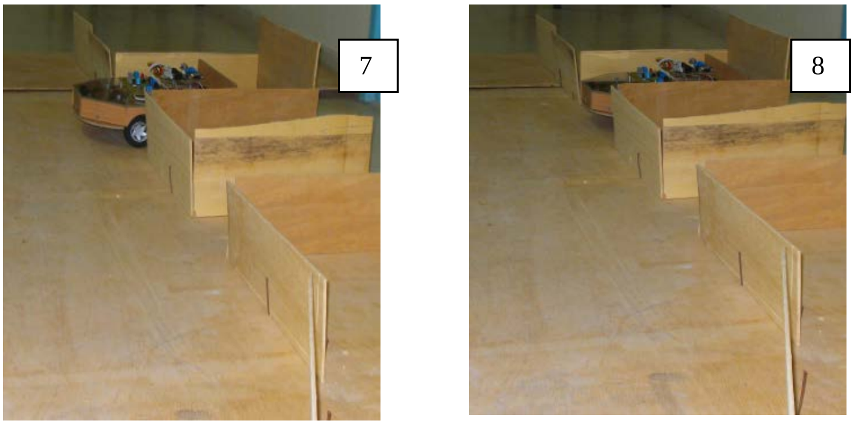


Fig. 20. The robot parking

6. Conclusion and future work

This paper addressed the problem of parking control in unkown environement taking into account the dynamics of the three wheeled mobile robot. The proposed solution, based on fuzzy logic controller, has been implemented and evaluated on real mobile robot placed in differents situations. The experimental results demonstrate effectiveness and applicability of our proposed approach. For future study, we want to use the camera sensor on our mobile robot. Despite its cost, this sensor does not need reflective object and can get more information about the environment.

References

1. M. Sugeno, T. Murofushi, T. Mori, T. Tatematsu, and J. Tanaka, “Fuzzy algorithmic control of a model car by oral instructions,” Fuzzy Sets Syst., vol. 32, pp. 207–219, 1989.
2. T.-H. S. Li, S.-J. Chang, and Y.X. Chen, “Implementation of human-like driving skills by autonomous fuzzy behavior control on an FPGA-based car-like mobile robot”, IEEE Trans on Industrial Electronics, vol. 50, NO.5, pp. 867-880, 2003.
3. A. Ohata and M. Mio, “Parking control based on nonlinear trajectory control for low speed vehicles,” in Proc. IEEE Int. Conf. Industrial Electronics, 1991, pp. 107–112.

4. S. Yasunobu and Y. Murai, “Parking control based on predictive fuzzy control,” in Proc. IEEE Int. Conf. Fuzzy Systems, vol. 2, 1994, pp. 1338–1341.

5. W. A. Daxwanger and G. K. Schmidt, “Skill-based visual parking control using neural and fuzzy networks,” in Proc. IEEE Int. Conf. System, Man, Cybernetics, vol. 2, 1995, pp. 1659–1664.

6. K. Jiang, “A sensor guided parallel parking system for nonholonomic vehicles,” in Proc. IEEE Conf. Intelligent Transportation Systems, 2000, pp. 270–275.

7. J. Xiu, G. Chen, and M. Xie, “Vision-guided automatic parking for smart car,” in Proc. IEEE Intelligent Vehicles Symp., 2000, pp. 725–730.

8. T.-H. S. Li and S.-J. Chang, “Autonomous fuzzy parking control of a car-like mobile robot,” IEEE Transactions on Systems, Man, and Cybernetics, vol. 33, pp. 451–465, 2003.

9. T.-H. S. Li, S.-J. Chang, and Y.X. Chen, “Implementation of human-like driving skills by autonomous fuzzy behavior control on an FPGA-based car-like mobile robot”, IEEE Trans on Industrial Electronics, vol. 50, NO.5, pp. 867-880, 2003.

10. I. Baturone, F. J. Moreno-Velo, S. Sánchez-Solano, and A. Ollero, “Automatic design of fuzzy controllers for car-like autonomous robots” IEEE Transactions on Fuzzy Systems, vol. 12, NO. 4, pp 447-465, 2004.

11. Rambaud, D., Exécution assisté par ordinateur de manœuvres de stationnement en espace restreint, Rapport de projet de fin d’études, ESISAR, Ecole Polytechnique de Montréal, Juillet 2004.

12. Y. Han, M. Han, H. Cha, M. Hong, and H. Hahn, “Tracking of a moving object using ultrasonic sensors based on a virtual ultrasonic image” Robotics and Autonomous Systems, vol.36 pp. 11–19, 2001.