3rd International Conference on Automation, Control, Engineering and Computer Science (ACECS'16)

**Proceedings of Engineering & Technology (PET)**
**pp. 503-509**


# Path Planning for mobile robots using fuzzy logic controller in the presence of static and moving obstacles

Faten CHERNI, Yassine BOUTEREAA, Chokri REKIK, Nabil DERBEL
University of Sfax, National Engineering School of Sfax, Tunisia
Control and Energy Management Laboratory (CEM Lab)
Emails: fatin.cherni@gmail.com, yassinebouteraa@gmail.com, chokri.rekik@enis.rnu.tn, n.derbel@enis.rnu.tn

***Abstract*—Designing an efficient navigation procedure of mobile robots, ensuring their securities, is an important issue in robotics. Path planning for mobile robots ensures a smoothly trajectory to the designed target without collision with static or moving obstacles. The aim of this paper is to develop an algorithm using fuzzy logic controller in order to find a feasible collision-free path with obstacles moving with varying velocities. The main idea of the present work is based on an integrated environment representation. In fact, information about the environment which contains close multiple stationary and moving obstacles are included in the representation of a sensing area of the mobile robot. Simulation results are performed to demonstrate the efficiency of the proposed approach which can be well applied in the mobile robot navigation.**

***Index Terms*—Mobile robot, path planning , integrated environment representation, fuzzy logic controller.**

## I. Introduction

Nowadays, robots represent an essential elements in society. This is due to replace humans by robots in dangerous tasks or to provide better solutions for the industry. Designing an intelligent and independent moving robot is the most modern technology in robotics. Path planning for a wheeled robot is defined as finding a free path that helps the mobile robot to reach the target without hitting obstacles. For this reason, the mobile robot must be equipped with an adequate perception system of the environment in order to give it a reactive behavior. Such a condition is provided, the robot will be able to ensure a fluid and reactive movement to the designed target without collisions with obstacles.

Indeed, the mobile robot needs to find a collision free path between any two points (from its beginning to its end). To be able to find this path, the mobile robot should run an adequate path planning algorithm. Several research works, for path planning of mobile robots, have been proposed in the literature [1], [2], [3]. Hence, there is some classical approaches dedicated to static environments are extended to dynamic ones [4], [5], [6]. However, the problem of avoiding collisions in dynamic environment is much harder. Several works have been developed for dynamic environments like velocity obstacles [7], [8], collision cones [9], the rolling window method [10], inevitable collision state [11].

In the other side, researchers have been carried out using advanced techniques such as fuzzy logic systems [12], [13], [14]. The Fuzzy Logic Controller has become a way of collecting human knowledge and experience. Now, the fuzzy logic is becoming an interesting topic in control engineering and a successful solution to a variety of industrial systems and consumer products. Moreover, the use of fuzzy logic system becomes very widespread to design a robust controller satisfying autonomous navigation problems.

This paper proposed a developed method for addressing chattering phenomenon with a simple and easy implementation. This is realized by replacing the sign function in control input used in [15] with fuzzy logic controller. This method allows chattering decrease in control input, while keeping the robustness characteristics of the robot mobile navigation. The task is to command the mobile robot in order to avoid obstacles and reach the goal while ensuring a smoothly trajectory in an static or dynamic environment.

This paper is organized as follows. In the next section, the problem formulation is presented. Section 3 contains the trajectory calculation. The fuzzy logic controller is given in Section 4. In Section 5, simulation results are presented. Finally, conclusions are given in Section 6.

## II. Problem formulation

Several navigation approaches can be founded in the literature. The main idea of our work is inspired by the approach developed in [15] and based on an integrated environment representation. In fact, this approach is efficient and very easy to implement [16], [17]. In the following, we introduce the proposed algorithm dedicated to the robot path planning in presence of static and dynamic obstacles. Hence, we assume that the positions of the robot, the obstacles and the goal are known in advance.

In this paper, we consider a mobile robot which takes as input the angular velocity $\omega$. The kinematic model of the mobile robot is given by:

$$\begin{cases} \dot{X}_R = V \quad \cos\alpha_R \\ \dot{Y}_R = V \quad \sin\alpha_R \\ \dot{\alpha}_R = \omega \end{cases} \tag{1}$$

**ISSN: 2356-5608**

where:

- $(X_R, Y_R)$ is the robot's cartesian coordinates.
- $\alpha_R$ is the heading direction.
- $V$ and $\omega$ are the translational and rotational velocities, respectively.

Let $(X_R(0), Y_R(0), \alpha_R(0))$ the initial condition of the robot and let $\tau$ the sampling rate.

The purpose of this paper is to produce a reliable and a smooth trajectory in a static and a dynamic environment and to guide the robot towards the target direction without hitting obstacles, taking into account physical constraints of the robot.

## III. Trajectory Calculation

We assume that all obstacles are circles in order to facilitate the present work. We define the disc $C$ of the radius $R$ centred at the point $\Omega$ that is ahead of the mobile robot's position as shown in Figure 1.

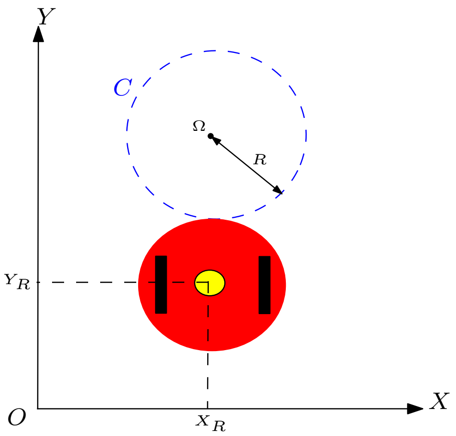
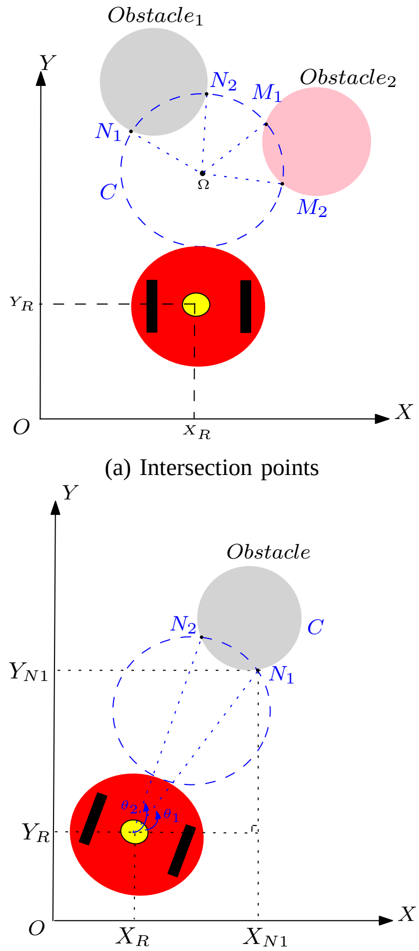



Fig. 1: Representation of the disc

The geometric sense of the disc $C$ choice is to ensure an efficient detection of obstacles. Indeed, the geometric shape covers the entire area in front of the robot. Really, the developed obstacle avoidance approach looks for having points of the intersection between virtual disc and real obstacles (see Figure 2a). In Figure 2b, we have two intersection points $N_1(X_{N1}, Y_{N1})$ and $N_2(X_{N2}, Y_{N2})$. The angle $\theta_1$ is given by :

$$\theta_1 = \arctan\left(\frac{Y_{N1} - Y_R}{X_{N1} - X_R}\right) \quad (2)$$

The idea is to compute the angle that makes the disc $C$ with two intersection points of the obstacles. Based on the computation angle, a new direction $\phi$ close to $\alpha_R$ will be provided. The objective behind the calculation of the new direction is to change the robot's heading in order to avoid obstacles detected in front. That remains now is to move towards the goal. To this end, the angular velocity, that guides the robot to the target direction without hitting obstacles, should be determined. For this, we propose a fuzzy logic controller to calculate the angular velocity of the mobile robot.




(a) Intersection points

(b) Definition of angle $\theta$

Fig. 2: Obstacles detection

The flowchart algorithm presented in Figure 3, contains four steps which are explained in the following:

**Step 1:** Let $\theta_T$ the set which contains all values of $\theta$ calculated by equation (2).

$\theta_T = \{\theta_1, \theta_2 ... \theta_{j-1}, \theta_j\}$ where $j \in \{1, 2, ..., 2n\}$ and $n$ is the number of obstacles. Figure 4 illustrates the distribution of different intervals.

**Step 2:** We note $I_{nd}$ the index of the angle that is closest to the robot current heading $\alpha_R$.

$$I_{nd} = \arg \ \min(|\theta_T - \alpha_R|) \quad (3)$$

Moreover we note:

$$S_1 = \alpha_R - \tfrac{\pi}{2} \quad ; \quad S_2 = \theta_T(I_{nd} - 1)$$
$$S_3 = \alpha_R + \tfrac{\pi}{2} \quad ; \quad S_4 = \theta_T(I_{nd} + 1)$$
$$S_5 = \theta_T(I_{nd})$$

**Step 3:** If the robot is in front of obstacles, there are four cases depending on the value of the index $I_{nd}$:

- If $I_{nd}$ is odd:
  - Case 1: If $I_{nd} = 1$ then $\varphi = S_1$
  - Case 2: If $I_{nd} \neq 1$ then $\varphi = S_2$

- If $I_{nd}$ is even:
  - Case 3: If $I_{nd} = 2n$ then $\varphi = S_3$
  - Case 4: If $I_{nd} \neq 2n$ then $\varphi = S_4$

**Step 4:** Based on the computation angle $\varphi$, we compute the new direction:

$$\gamma = \begin{cases} \frac{S_5+\varphi}{2} & \text{if length}(\theta_T) \neq 0 \\ \alpha_g = \arctan\left(\frac{Y_g - Y_R}{X_g - X_R}\right) & \text{elsewhere} \end{cases} \tag{4}$$

Where: $\alpha_g \in [-\frac{\pi}{2}, \frac{\pi}{2}]$ the desired goal direction and assumed to be known to the robot and $(X_g, Y_g)$ is the goal's cartesian coordinates. If length $(\theta_T) \neq 0$, the direction $\gamma$ represents the middle of the interval closest to $\alpha_R$ . If there isn't any obstacles in front of the mobile robot, then $\gamma$ represents the desired goal direction $\alpha_g$.

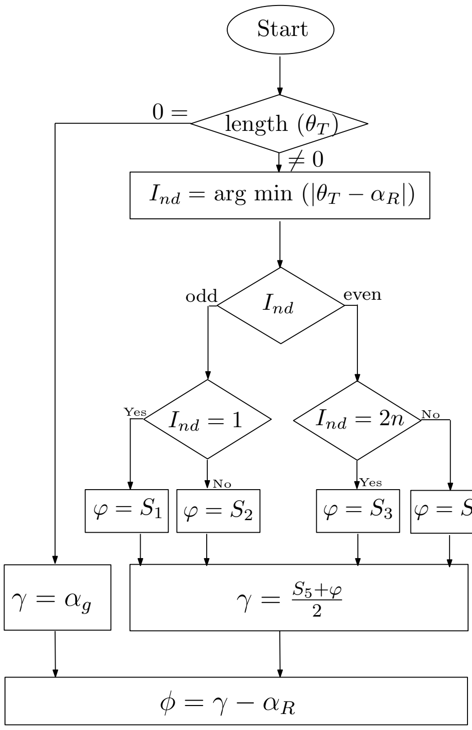



Fig. 3: The proposed algorithm

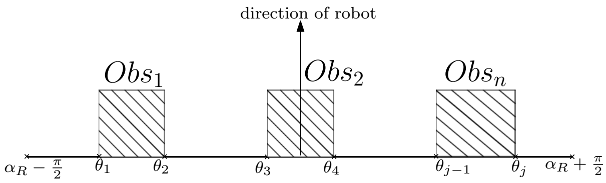



Fig. 4: Illustration of the intervals

## IV. FUZZY LOGIC CONTROLLER

The Fuzzy logic controller can be used to control the navigation of the mobile robot. In fact, the fuzzy system allows the robot to find the path from the starting point to the target.

We propose a navigation scheme which is divided into three general functions: the trajectory calculation , the fuzzy logic controller and the mobile robot as shown in Figure 5.

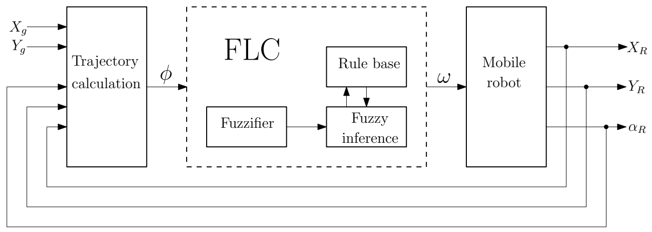



Fig. 5: Block diagram of the navigation algorithm

- $(X_g, Y_g)$ represents the goal's cartesian coordinates.
- $(X_R, Y_R)$ is the robot's cartesian coordinates and $\alpha_R$ is the heading direction.
- $\phi$ represents the new angle that changes the robot's direction.
- $\omega$ is the angular velocity of the mobile robot.

During the actual move, the mobile robot acquires information about its environment containing static or dynamic obstacles. To achieve the goal, the robot uses the new direction angle $\phi$ as the main parameter. Then, the angular velocity $\omega$ is determined and given to the robot in order to be able to bypass between obstacles without being collided.

The controller input is the angle $\phi$ calculated by the trajectory calculation and represents the new direction that makes the robot changes its direction when it senses obstacles in front. The controller output is the angular velocity $\omega$ that will be given to the mobile robot in order to guide the robot smoothly to the target without hitting obstacles.

### A. *Fuzzy partition of input variables*

The angle $\phi$ is defined in $[-\frac{\pi}{2}, \frac{\pi}{2}]$. Membership functions of the parameter $\phi$ is Gaussian. From several experiments and from different desired precision, we have associated seven linguistic values for the angle $\phi$ (NL: Negative Large; NM: Negative Medium; NS: Negative Small; Z: Zero; PS:Positive Small; PM: Positive Medium; PL: Positive Large).

### B. *Rules basis*

In this part, we determine relations between the fuzzy input variable $\phi$ and fuzzy output variable $\omega$. These rules are given by:

If ($\phi$ is $A_i$ ) then ($\omega$ = $y_i$)

with $i = 1, 2..., n$, where $n$ is the rule number.

Following several simulations and experiment tests, we have manually constructed the fuzzy inference table (situation/action). Tables I represents the suggested fuzzy rules.

### C. *Fuzzy controller outputs*

Fuzzy controller output is the angular velocity $\omega$. It is defined by:

$$\omega = \frac{\sum_{i=1}^{n} \alpha_i y_i}{\sum_{j=1}^{n} \alpha_i} \tag{5}$$

TABLE I: Inference table for the angle $\omega$

| $\phi$ | NL | NM | NS | Z | PS | PM | PL |
|---|---|---|---|---|---|---|---|
| $\omega$ | NB | NM | NS | Z | PS | PM | PB |

with $\alpha_i$ is the level activation of rule $i$.

The elaborated fuzzy controller is a Sugeno fuzzy logic system of order zero. Thus, fuzzy rule consequences are constants. We have chosen seven values as linguistic variables of the fuzzy rule consequences as following: NL: Negative Large; NM: Negative Medium; NB: Negative Big; Z: Zero; PS: Positive Small; PM: Positive Medium; PB: Positive Big. Based on simulations and experimentation tests, we have attributed to each linguistic variable a numerical value as shown in Figure 6.

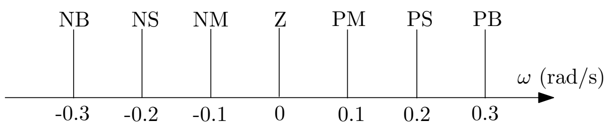



Fig. 6: Numerical values of fuzzy rules output

## V. Simulation Results

To conclude the performances of the developed method using the principle of the fuzzy logic controller, we will present simulations of an arbitrarily environment including static obstacles. In all simulations, the linear velocity and the maximum angular velocity have been chosen respectively as $V = 0.2 \quad ms^{-1}$, $\alpha_{max} = 0.3 \quad rad.s^{-1}$. The sampling rate is set to be $\tau = 0.1s$.

### *A. Navigation with static obstacles*

In order to prove that the developed approach is efficient in partially known environment, we have constructed an environment containing static obstacles. We assume that the robot start its motion from the initial position $[X_R(0), Y_R(0), \alpha_R(0)] = [0, 0, 0]$. Figure 7 and Figure 8 illustrate the scene supplied to the mobile robot and show the mobile robot trajectory depicted with small circles. In this scene, five static obstacles, with different shapes, are placed with an arbitrary way. The robot should begin at point $(0, 0)$ and finish at point $(0, 12)$. In such a crowd environment, we are compared the algorithm of the original method and the developed algorithm with fuzzy logic controller. In both methods, the mobile robot fulfill successfully the task and reach the final destination without being collide with obstacles.

### *B. Navigation with moving obstacles*

In this case, the robot is navigating with 3 dynamic obstacles starting its motion as mentioned in table II. The desired goal $(x_g, y_g) = (8, 9)$. The mobile robot start its motion from the initial position $[X_R(0), Y_R(0), \alpha_R(0)] = [0, 0, \alpha_g]$. In the following simulations, we show different scenarios illustrating

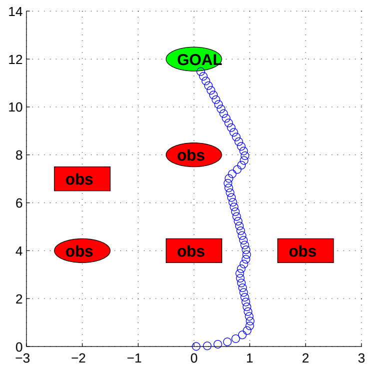



Fig. 7: Path planning with original method [15] in static environment starting at (0,0) and ending at (0,12).

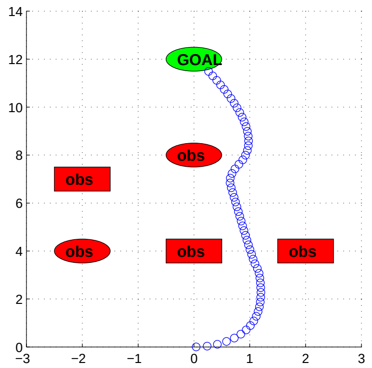



Fig. 8: Path planning with developed method using fuzzy logic controller in static environment starting at (0,0) and ending at (0,12).

TABLE II: Initial conditions of mobiles obstacles

| | $x_{obs}$ | $y_{obs}$ | $\theta_{obs}$ | $V_{obs}$ |
|---|---|---|---|---|
| $obstacle_1$ | 2 | 4 | $-\frac{\pi}{4}$ | 0.05 |
| $obstacle_2$ | 11 | 6 | 0 | -0.05 |
| $obstacle_3$ | 3 | 7 | $-\frac{\pi}{4}$ | 0.1 |

the mobile robot moving towards the goal in a dynamic environment. As it can be seen in Figure 9 and Figure 10, the robot tries to detour the moving obstacles from its front and changes its direction when it detects the obstacle. Finally, the mobile robot accomplishes successfully the navigation mission and reaches the stationary goal. Moreover, we define $\triangle S = s_2 - s_1$ the covered distance between the starting and the ending points and the period $\triangle t = t_2 - t_1$.

Using the developed approach, we remark that the mobile robot selects the shortest smooth path in the shortest possible time until reaches the target. In fact, the robot accomplishes the navigation mission with $\triangle S = 14m$ path length and spends $\triangle t = 70s$ to achieve the target. However, in the original method, the robot spends $\triangle t = 82.1s$ with $\triangle S = 16.42m$ path length.

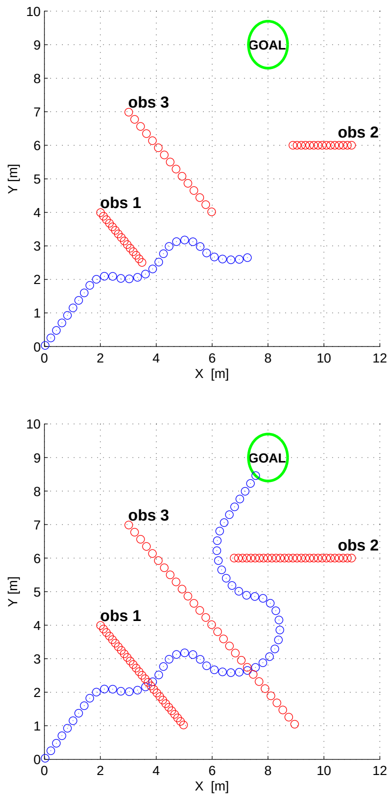



Fig. 9: Mobile robot navigation with original method [15] in dynamic environment

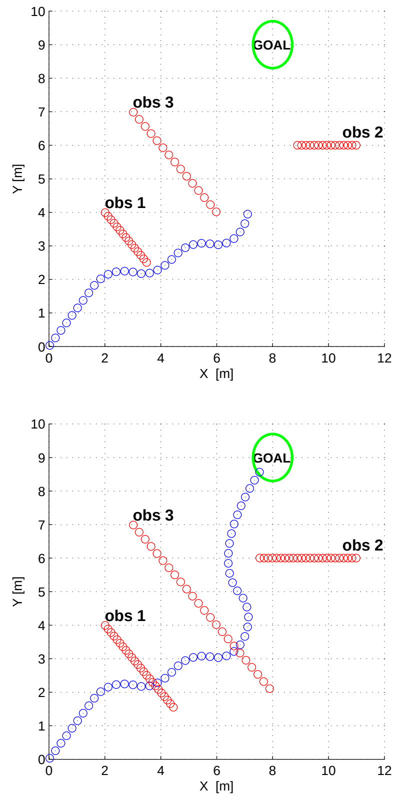



Fig. 10: Mobile robot navigation using fuzzy logic controller in dynamic environment

Indeed, the path is optimized and the robot spends a short time.

In order to prove that there are no collision between the robot and obstacles, we illustrate in Figure 11 the curves of the robot's cartesian coordinates $(X_R, Y_R)$ depicted with continuous line and the curves of cartesian coordinates of different obstacles represented by dotted lines. If there is a collision between the robot and obstacles, they will have the same cartesian coordinates at the same time. Observing Figure 11, it is easy to conclude that there is no collision between the robot curves and obstacles. This proves that the robot moves away from mobile obstacles and doesn't collide them.

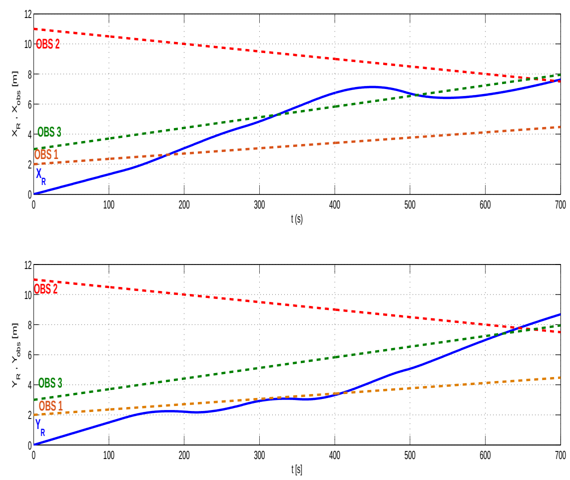



Fig. 11: Cartesian coordinates curves of the robot and obstacles

### C. Chattering problem

The chattering phenomenon is an undesirable phenomenon that generates oscillations in the control input which can result the deterioration anticipated of the control system.

To limit this phenomenon, the fuzzy logic controller is introduced to replace the sign function and to solve the chattering problem in order to provide the stability and the robustness of the system.

In fact, the switching caused by the sign function presented in equation (6)[15] involve the appearance of the chattering phenomenon which is characterized by large oscillations (see Figure 12 and Figure 14).

$$\begin{array}{l} \text{if} \quad m(k\delta)=0 \quad \text{then} \quad u(t)=U_{max} \quad \text{sign} \quad [\theta_0(t)-\theta(t)] \\ \text{if} \quad m(k\delta)=1 \quad \text{then} \quad u(t)=U_{max} \quad \text{sign} \quad [C(t)-\theta(t)] \\ \forall t \in [k\delta, (k+1)\delta) \end{array} \tag{6}$$

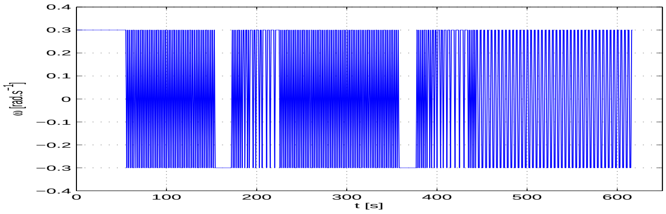



Fig. 12: Control input of the original method in static environment

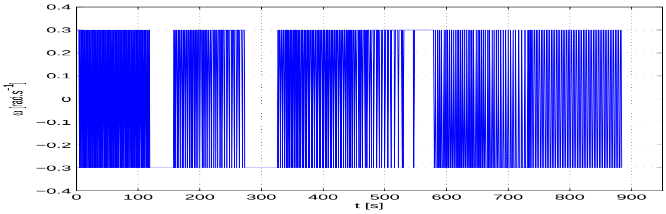



Fig. 14: Control input of the original method in dynamic environment

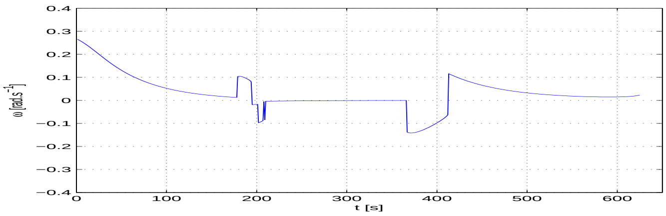



Fig. 13: Control input with fuzzy logic controller in static environment

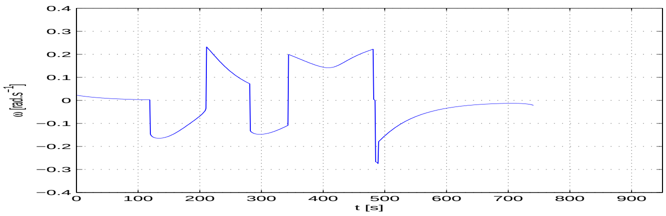



Fig. 15: Control input with fuzzy logic controller in dynamic environment

Where:

$m(k\delta)$ is a function taking 0 when the mobile robot does not sense the environment in front itself or 1 when the mobile robot senses the environment. $u(t)$ the angular velocity, $U_{max}$ the maximum angular velocity. $\theta_0$ represents the desired goal direction and $C(t)$ the new direction. $\delta$ is the sampling period.

Simulations results, given by Figure 13 and Figure 15, prove that the chattering phenomenon has been eliminated from the signal of the angular velocity and show clearly that the use of the fuzzy logic controller gives good performances and reduces the chattering phenomenon.

## VI. Conclusion

In this paper, one of major tasks of path planning of mobile robot has been presented. Based on the developed algorithm, we have found a collision-free path in cluttered environment containing static and moving obstacles. In the other side, we have used a fuzzy logic controller to reduce the chattering phenomenon in the control law of the original method which uses the sign function. Simulation results prove that the developed algorithm shows a high effectiveness in obstacle avoidance with the shortest path to the destination and the lowest elapsed time. Indeed, the high frequency in the control input has been successfully reduced.

## References


[1] Jun-Hao Liang, Ching-Hung Lee,*"Efficient collision-free path-planning of multiple mobile robots system using efficient artificial bee colony algorithm"*, Advances in Engineering Software, 79:47–56, 2015

[2] Ippei Nishitani, Tetsuya Matsumura,*"Human-centered X-Y-T space path planning for mobile robot in dynamic environments"*, Robotics and Autonomous Systems, 66:18–26, 2015

[3] Xunyu Zhong, Xungao Zhong, Xiafu Peng, *"Velocity-Change-Space-based dynamic motion planning for mobile robots navigation"*, Neurocomputing, 143:153–163, 2014.

[4] S. Ge, Y. Cui, *" Dynamic motion planning for mobile robots using potential field method"*, Auton. Robots, 13(3):207–233, 2002.

[5] D. Hsu, R. Kindel, J.C. Latombe, S. Rock, *"Randomized kinodynamic motion planning with moving obstacles"*, Int. J. Robot. Res, 21(3):233–255, 2012.

[6] K. Fujimura, H. Samet, *"Roadmap-based motion planning in dynamic environments"*, IEEE Trans. Robot. 21(5):885–897, 2005.

[7] P. Fiorini, Z. Shiller, *"Motion planning in dynamic environments using velocity obstacles"*, Int. J. Robot. Res, 17:760–772, 1998.

[8] F. Large, C. Lauger, Z. Shiller, *"Navigation among moving obstacles using the NLVO: principles and applications to intelligent vehicles"*, Auton. Robots, 19:159–171, 2005.

[9] A. Chakravarthy, D. Ghose, *"Obstacle avoidance in a dynamic environment: a collision cone approach"*, IEEE Trans. Syst. Man Cybern, 28:562–574, 1998.

[10] Chun-gang Zhang, Yu-geng Xi, *" Rolling path planning and safety analysis of mobile robot in dynamic uncertain environment"*, Control Theory Appl, 20(1):37–44, 2003.

[11] T. Fraichard, H. Asama, *"Inevitable collision states. A step towards safer robots"* in: IEEE International Conferenceon Intelligent Robots and Systems, 388–393, 2004.

[12] A. Fahmi, L. Khriji, A. Al-Yahmadi and N. Masmoudi, *"Contribution of fuzzy logic based autonomous robot navigation in unknown environment"*, Proc. of IEEE conf. on Signals, System and Decision, SSD'07, 2007.

[13] H. Maaref and C. Barret, *"Sensors-based fuzzy navigation of an autonomous mobile robot in an indoor environment"*, Robotics and Autonomous System, 38:1–18, 2002.

[14] Foudil Abdessemed, Khier Benmahammed, Eric Monacelli,*"A fuzzy-based reactive controller for a non-holonomic mobile robot"*, Robotics and Autonomous Systems 47:31–46, 2004.

[15] Andrey V. Savkin, Chao Wang, *"Seeking a path through the crowd: Robot navigation in unknown dynamic environments with moving obstacles based on an integrated environment representation"*, Robotics and Autonomous Systems, 62:1568–1580, 2014.

[16] Faten Cherni, Yassine Boutereaa, Chokri Rekik, Nabil Derbel, *"Autonomous mobile robot navigation algorithm for planning collision-free path designed in dynamic environments"*, $9^{th}$ Jordian Int. Electrical and Electronics Engineering Conference (JIEEEC).

[17] Faten Cherni, Maisaa Boujlben, Lotfi Jaiem, Yassine Boutereaa, Chokri Rekik, Nabil Derbel, *"Autonomous mobile robot navigation based on an integrated environment representation designed in dynamic environments"*, Int. Journal of Automation and Control (under review).