

# Comparative Study between Fuzzy Logic and Interval Type-2 Fuzzy Logic Controllers for the Trajectory Planning of a Mobile Robot

Boucetta Kasmi

Department of Electronics  
Laboratory of Intelligent Systems  
University Ferhat Abbas Setif 1  
Setif, Algeria  
boucetta.kasmi@univ-setif.dz

Abdelouaheb Hassam

Department of Electronics  
Laboratory of Intelligent Systems  
University Ferhat Abbas Setif 1  
Setif, Algeria  
abdelhassam@univ-setif.dz

**Abstract**—In this study, Fuzzy Logic (FL) and Interval Type-2 FL (IT-2FL) controllers were applied to a mobile robot in order to determine which method facilitates navigation and enables the robot to overcome real-world uncertainties and track an optimal trajectory in a very short time. The robot under consideration is a non-holonomic unicycle mobile robot, represented by a kinematic model, evolving in two different environments. The first environment is barrier-free, and moving the robot from an initial to a target position requires the introduction of a single action module. Subsequently, the same problem was approached in an environment closer to reality, with objects hindering the robot's movement. This case requires another controller, called obstacle avoidance. This system allows the robot to reach autonomously a well-defined target by avoiding collision with obstacles. The robustness of the structures of the defined controllers is tested in Matlab simulations of the studied controllers. The results show that the IT-2FL controller performs better than the FL controller.

**Keywords**—Interval Type-2 Fuzzy Logic (IT2-FL); Fuzzy Logic (FL); mobile robot; non-holonomic; obstacle avoidance; trajectory planning

## I. INTRODUCTION

A mobile robot is generally equipped with perception and decision-making capabilities and actions that allow it to navigate safely and successfully in a given environment and to follow a desired trajectory. This objective is achieved without or with a reduced human intervention [1]. Today, independent mobile robots are increasingly used in many applications, such as service tasks, agriculture, handling of nuclear waste, and in space industry [2]. The developments of sensors, microprocessors, and control technology have enabled mobile robots to perform very complex tasks. Today, the main challenge regarding mobile robots is the development of intelligent navigation systems [3]. Navigation is of great importance, since virtually every task requires the robot to travel between different positions by tracking a desired trajectory while being able to localize itself and plan its future movements without human assistance [4] in order to accomplish the defined task [5]. Therefore, once the trajectory

of the robot in a complex environment is determined, it must be capable of following it [6]. The generated trajectory must take into account the environmental and kinematic constraints of the moving object [7-8]. To tackle these difficulties and to enable a large number of parameters to be managed, a new control strategy is proposed, which is broken down into weakly coupled units whose interactions are limited and perfectly controlled in a distributed manner. The purpose is to develop robots that can move safely in unstructured environments, despite any unforeseen changes.

The main problem of navigation of mobile robots can be broken down into three sub-problems: reaching the target, avoiding obstacles, and tracking an optimal trajectory in a very short time. To reach the target, we can indicate the trajectory to the robot by employing stable techniques based on the search for optimal trajectories. However, since the environments are rarely predictable, it makes little sense to provide the robot with a planned trajectory. Obstacle avoidance has often been solved by using local information, perceived by the robot via its sensors. Initially, research has focused on solving the obstacle avoidance problem by presenting it as a high-level control component of hierarchical robotic systems. Thus, the problem arose as a trajectory planning where the controller at the low level leads the robot towards its final destination while avoiding the surrounding obstacles using the trajectory found at the high level.

The main objective of this paper is the tracking of an optimal trajectory in a very short time, which is the third sub-problem. A comparative study of Fuzzy Logic (FL) and Interval Type-2 FL (IT-2FL) controllers was conducted in order to assess their performance. The robot under consideration is a non-holonomic unicycle mobile robot, represented by a kinematic model.

## II. MODELING OF A MOBILE ROBOT

The differential mobile robot is a platform with two motorized wheels (Figure 1), mounted on the same axis and controlled independently while having in addition a free front

Corresponding author: Boucetta Kasmi

wheel [7]. The simplified hypotheses considered for modeling are:

- The ground wheel contact is a contact point.
- The rolling of each wheel is done without slipping.

The generalized coordinates of the system are given by  $q = [x, y, \theta, \varphi_r, \varphi_l]^T$  where  $[x, y]$  are the Cartesian coordinates of the mobile robot,  $\theta$  is its orientation measured from the x-axis and  $\varphi_r, \varphi_l$  are the angular positions of the right and left wheel respectively [2, 3].

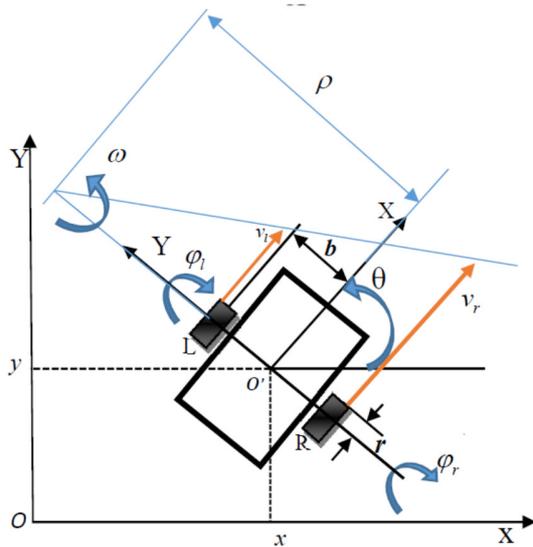


Fig. 1. The unicycle-type mobile robot under study.

### III. KINEMATIC MODEL OF THE MOBILE ROBOT

The kinematic model of the mobile robot can be written as [3]:

$$\dot{q} = \begin{bmatrix} \dot{x} \\ \dot{y} \\ \dot{\theta} \\ \dot{\varphi}_r \\ \dot{\varphi}_l \end{bmatrix} = \begin{bmatrix} \frac{r \cos(\theta)}{2} & \frac{r \cos(\theta)}{2} \\ \frac{r \sin(\theta)}{2} & \frac{r \sin(\theta)}{2} \\ \frac{r}{2b} & \frac{-r}{2b} \\ 1 & 0 \\ 0 & 1 \end{bmatrix} \begin{bmatrix} \dot{\varphi}_r \\ \dot{\varphi}_l \end{bmatrix} = \begin{bmatrix} \dot{\varphi}_r \\ \dot{\varphi}_l \end{bmatrix} \quad (1)$$

By introducing the following control inputs:

$$\begin{bmatrix} \dot{\varphi}_r \\ \dot{\varphi}_l \end{bmatrix} = \frac{b}{r} \begin{bmatrix} 1 & 1 \\ 1 & -1 \end{bmatrix} \begin{bmatrix} v \\ \omega \end{bmatrix}$$

Equation (1) may be written as:

$$\begin{bmatrix} \dot{x} \\ \dot{y} \\ \dot{\theta} \end{bmatrix} = \begin{bmatrix} \cos(\theta) \\ \sin(\theta) \\ 0 \end{bmatrix} v + \begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix} \omega \quad (2)$$

$$\begin{bmatrix} v \\ \omega \end{bmatrix} = \begin{bmatrix} \frac{r}{2} & \frac{r}{2} \\ \frac{r}{2b} & -\frac{r}{2b} \end{bmatrix} \begin{bmatrix} \dot{\varphi}_r \\ \dot{\varphi}_l \end{bmatrix} \quad (3)$$

where  $v$  and  $\omega$  are respectively the linear and angular velocities of the mobile robot and  $2b$  and  $r$  represent respectively the radius of the wheels and the distance between them.

The non-holonomic constraint is represented in a simple mathematical form [3]:

$$\dot{x} \cos(\theta) - \dot{y} \sin(\theta) = 0 \quad (4)$$

Equation (4) implies that a perfect tracking is achievable only if the reference trajectories are feasible for the physical platform.

### IV. LOCALIZATION OF A MOBILE ROBOT

One of the fundamental problems of autonomous mobile robotics is the locationing of the robot during its movement. In fact, to locate a mobile robot is to determine, in a given work reference, its position and its orientation, in order to accomplish the control structure that is based on these data.

#### A. Presentation of the Odometry

The odometry allows determining the position and the orientation of a mobile robot navigating on a plane ground, with respect to the reference mark, which is the robot's mark in its initial configuration. This technique is based on the integration of the elementary motions of the wheels measured by means of incremental encoders.

#### B. The Odometry for the Localization of a Mobile Robot

This locomotion system is very popular for indoor robots because of its maneuverability and ease of operation. In this case, the displacement  $\Delta D$  and the elementary rotation  $\Delta \theta$  of the robot model in the plane can be expressed as a function of the elementary displacements of the right and left wheels respectively  $\Delta d_r$  and  $\Delta d_l$ , by [13, 14]:

$$\Delta D = \frac{\Delta d_r + \Delta d_l}{2} \quad (5)$$

$$\Delta \theta = \frac{\Delta d_r - \Delta d_l}{2b} \quad (6)$$

where  $(x_k, y_k, \theta_k)$  is the configuration of the robot at the instant  $k$ , and  $(\Delta D_k, \Delta \theta_k)$  the components of the elementary displacement measured between instants  $k$  and  $k+1$ . The elementary rotation at time  $k+1$  is:

$$\theta_{k+1} = \theta_k + \Delta \theta_k \quad (7)$$

These very simple formulas are obtained by considering that the robot moves in a straight line  $\Delta D_k$  in the direction defined by  $\theta_k$ , and then makes a rotation on site of  $\Delta \theta_k$ :

$$x_{k+1} = x_k + \Delta D_k \cos(\theta_k + \frac{\Delta \theta_k}{2}) \quad (8)$$

$$y_{k+1} = y_k + \Delta D_k \sin(\theta_k + \frac{\Delta\theta_k}{2}) \quad (9)$$

This robot control was applied in both FL and IT-2FL strategies.

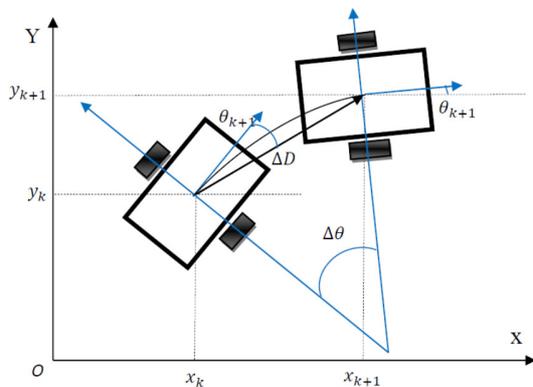


Fig. 2. The odometry applied to the mobile robot.

V. STRUCTURE OF THE FL CONTROLLER

A classic fuzzy controller consists of a fuzzification interface, a rule base, an inference system, and a defuzzification interface [15-21]. The structure of an FL system is illustrated in Figure 3.

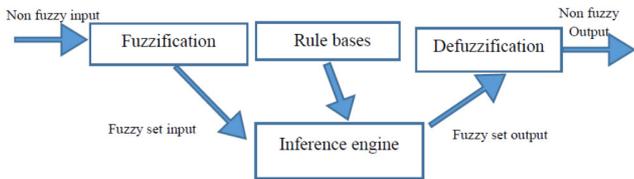


Fig. 3. Structure of an FL system.

In our work, the fuzzy controller (FL and IT-2FL) has 4 triangular-shaped membership functions for the robot-target distance, 7 membership functions for the variation of the robot-target angle and an interval-type of fuzzy sets for the linear velocity and angular velocity output for IT-2FLC. The output variables for FLC are 4 triangular-shaped membership functions for linear velocity and 7 triangular-shaped membership functions for the angular velocity.

VI. THE PROPOSED NAVIGATION SYSTEM

The movement of the unicycle robot is carried out on a flat ground and the position of the robot can be expressed at every moment according to its kinematic model  $(x, y, \theta)$ . When meeting obstacles or walls, the relevant decision is made by two controllers. Figure 4 shows the structure of the system, consisting of a free navigation controller and an obstacle avoidance controller [8-10].

A. Implementation of the Free Navigation Controller

If we take a mobile robot operating in a non-binding environment, then the optimal path from an initial configuration to a final situation would naturally be a straight line joining the two situations (Figure 5) [8, 11, 12].

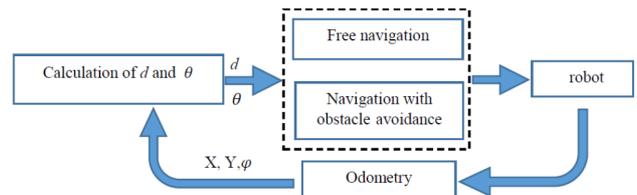


Fig. 4. Structure of the proposed navigation system.

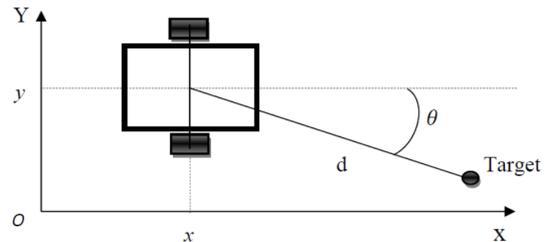


Fig. 5. The mobile robot in a free environment.

1) Input Variables

For the robot-target distance  $d$  (m), we have chosen 4 membership functions (Figure 6): very near (VN), near (NR), far (F), and very far (VF) distributed over the discourse universe  $[0, 30]$ . For the entry  $\theta$  (rad), the robot-target angle has 7 membership functions that are associated (Figure 7): negative big (NB), negative medium (NM), negative small (NS), zero (Z), positive small (PS), positive medium (PM), and positive big (PB) distributed over the discourse universe  $[-3, 3]$ .

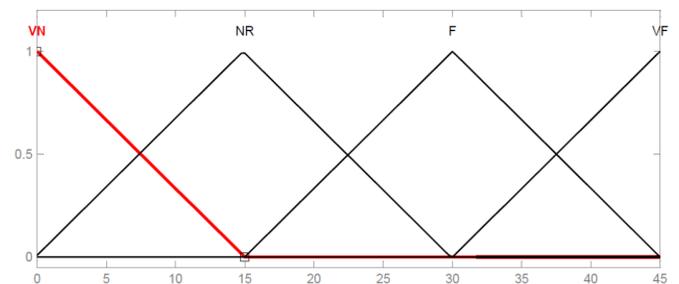


Fig. 6. The membership functions of the input variable distance  $d$ .

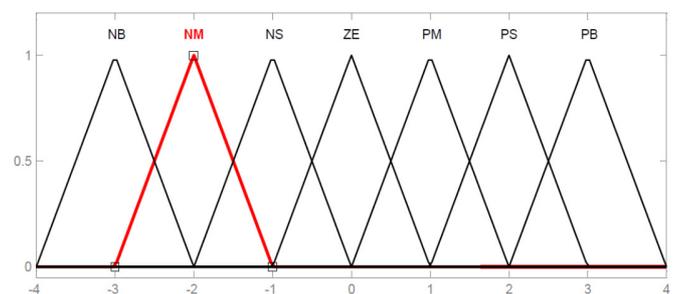


Fig. 7. The membership functions of the input variable  $\theta$ .

2) Output Variables

For the linear velocity  $v$  (m/s), 4 intervals were chosen: very slow (VS), slow (S), fast (F), and very fast (VF) distributed over the discourse universe  $[0, 0.2]$ .

The angular velocity  $\omega$  (rad) has seven (7) intervals: negative big (NB), negative medium (NM), negative small (NS), zero (Z), positive small (PS), positive medium (PM) and positive big (PB), distributed over the discourse universe  $[-0.8, 0.8]$ .

3) Representation of Input Variables

The membership functions of the triangular input variables are shown in Figures 8, 9.

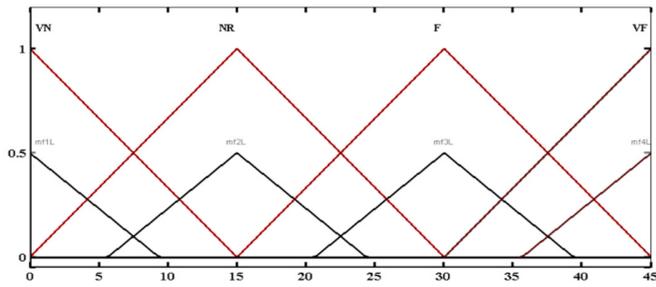


Fig. 8. The membership functions of the input variable  $d$  in Matlab Toolbox.

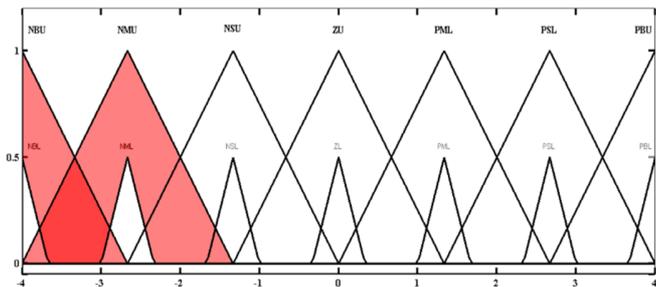


Fig. 9. The membership functions of the input variable  $\theta$  in Matlab Toolbox.

4) The Rule of the Free Navigation Controller

TABLE I. THE RULE BASE OF THE FREE NAVIGATION CONTROLLER

Orientation $\theta$	Distance $d$			
	VN	N	F	VF
NB	VS, PB	VS, PB	S, PB	S, PM
NM	VS, PM	VS, PS	S, PS	F, PS
NS	VS, S	S, PS	F, PS	F, PS
Z	VS, Z	S, Z	F, Z	V, FZ
PS	VS, NS	S, NS	F, NS	VF, NS
PM	VS, NM	VS, NM	F, NM	F, NS
PB	VS, NB	VS, NB	S, NB	S, NB

The rule base of the free navigation controller is the following:

- Rule 1: IF  $d$  is VN AND  $\theta$  is NB  
THEN  $v$  is VS AND  $\omega$  is PB
- Rule 2: IF  $d$  is VN AND  $\theta$  is NM  
THEN  $v$  is VS AND  $\omega$  is PM
- .....

Rule 28: IF  $d$  is VF AND  $\theta$  is PB

THEN  $v$  is S AND  $\omega$  is NM

We applied these rules of the free navigation controller in both FL and IT-2FL strategies.

B. Implementation of the Obstacle Avoidance Controller

In the case where the robot moves close to an obstacle, another fuzzy controller is used to avoid the obstacle and steer the robot away from  $d_{ob}$  using the navigation controller. Figure 10 shows the configuration of the robot in the obstacle area [12, 14, 20].

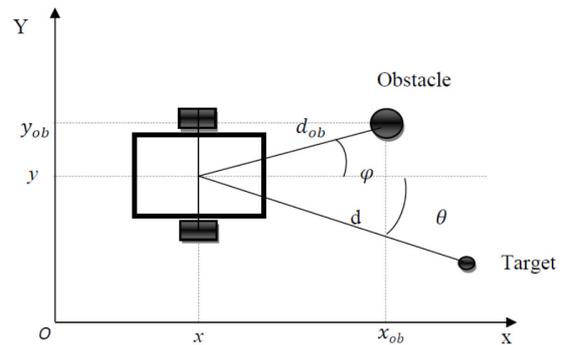


Fig. 10. The mobile robot in an environment with an obstacle.

The obstacle avoidance controller that we used has two input variables: robot-obstacle distance and robot-obstacle angle ( $d_{ob}$  and  $\phi$ ) respectively and two output variables: robot linear velocity  $v$  and angular velocity  $\omega$ .

1) The Membership Functions of the Input Variables for FLC

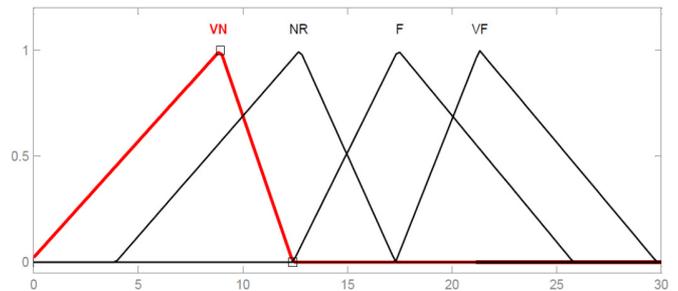


Fig. 11. The membership functions of the input variable  $d_{ob}$ .

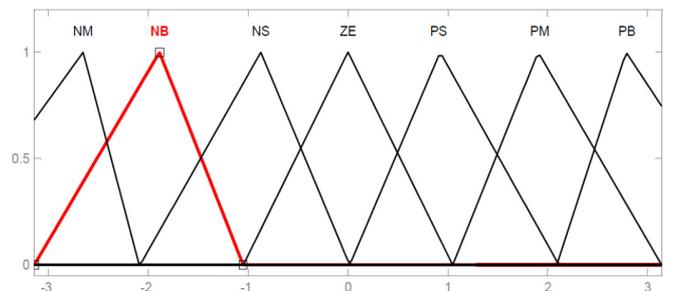


Fig. 12. The membership functions of the input variable  $\phi$ .

2) The Membership Functions of the Input Variables for IT-2FLC

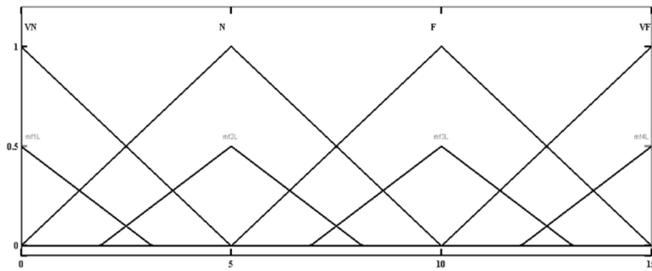


Fig. 13. The membership functions of the input variable  $d_{ob}$  in Matlab Toolbox.

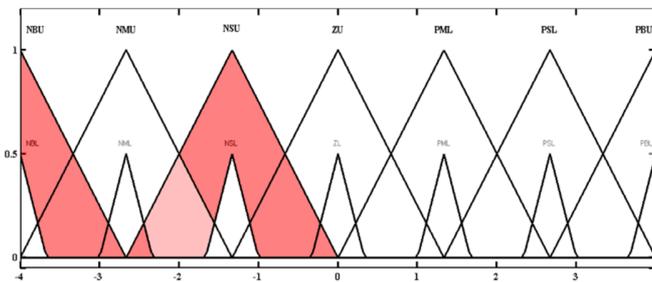


Fig. 14. The membership functions of the input variable  $\phi$  in Matlab Toolbox.

3) The Rule Base of the Obstacle avoidance controller

TABLE II. THE RULE BASE OF THE OBSTACLE AVOIDANCE CONTROLLER

Orientation $\phi$	The obstacle distance $d_{ob}$			
	VN	N	F	VF
NB	VS, PM	VS, PM	S, PM	S, PM
NM	VS, PM	VS, PS	S, PS	F, PS
NS	VS, PS	S, PS	F, PS	F, PS
Z	VS, PM	S, NS	F, PS	VF, PS
PS	VS, NM	S, NS	F, NS	VF, NS
PM	VS, NM	VS, NM	F, NS	F, NS
PB	VS, NB	VS, NB	S, NB	S, NB

The rule base of the obstacle avoidance controller is the following:

Rule 1: IF  $d_{ob}$  is VN AND  $\phi$  is NB  
THEN  $v$  is VS AND  $\omega$  is PM

Rule 2: IF  $d_{ob}$  is VN AND  $\phi$  is NM  
THEN  $v$  is VS AND  $\omega$  is PM

.....  
Rule 28: IF  $d_{ob}$  is VF AND  $\phi$  is PB  
THEN  $v$  is S AND  $\omega$  is NB

We applied these rules of the obstacle avoidance controller in both the FL and IT-2FL strategies.

VII. SIMULATION RESULTS

To compare the control and planning performances of FL and IT2-FL controllers, simulations were conducted and analyzed with MATLAB Fuzzy Logic Toolbox R2014a.

A. Trajectory without Obstacles

From Figure 15, we notice that from its initial position, ( $x_i=2.5$  and  $y_i=4.5$ ,  $\theta=45^\circ$ ) the mobile robot could reach the target whose coordinates are ( $x_f=7$ ,  $y_f=-1$ ). In Figures 16 and 17 we can see the velocity and the angular velocity of the mobile robot. From the obtained results, we notice that the mobile robot adopts the following behavior: when the robot-target angle is large, the angular velocity is high, whereas the linear velocity is small. Once the robot-target angle becomes zero, the linear velocity reaches its maximum. The latter gradually decreases by canceling once the target is reached.

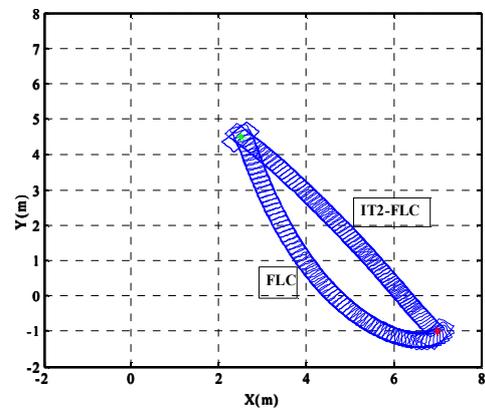


Fig. 15. Attraction to the target: barrier-free environment.

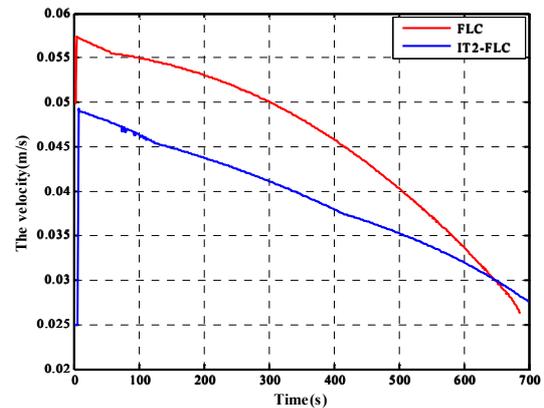


Fig. 16. The velocity of the mobile robot.

B. Obstacle Avoidance

From Figure 18 we can see how the mobile robot reached the target ( $x_f=6$ ,  $y_f=7$ ) from its initial position ( $x_i=3$  and  $y_i=8$ ,  $\theta=-45^\circ$ ).

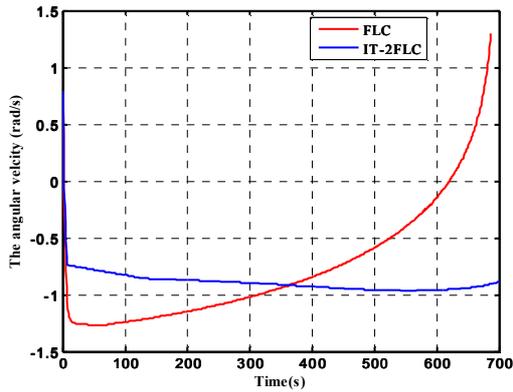


Fig. 17. The angular velocity of the mobile robot.

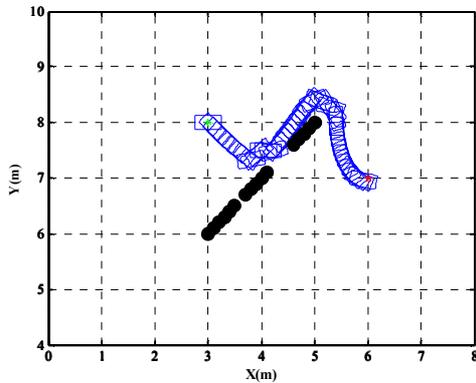


Fig. 18. Convergence towards the target by FLC in the presence of obstacles.

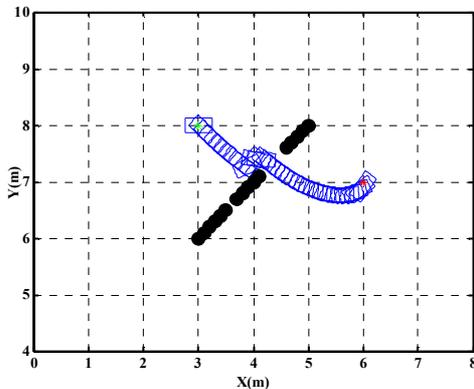


Fig. 19. Convergence towards the target by IT-2FLC in the presence of obstacles.

We can highlight from the above results that the IT-2FL sets can be quite useful when considering the control of a mobile robot. It was shown in depth that the proposed IT2-FL controller is more efficient in terms of saving time, smooth trajectory, and optimal distance than its FL counterpart.

VIII. CONCLUSION

In this paper, movement control methods of a wheeled mobile robot were studied. To achieve this control target, we

have designed the FL and IT-2FL controllers and simulated the mobile robot movement from an initial to a desired position in different environment configurations, with and without obstacles. The proposed fuzzy control exploits the interactive variables between the mobile robot and the unknown environment to generate the robot's velocity and steering, which makes it possible to bring the mobile robot towards the target while avoiding any obstacles present in this environment.

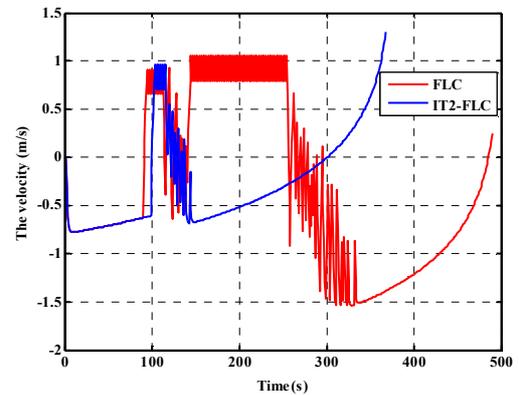


Fig. 20. The velocity of the mobile robot in the presence of obstacles.

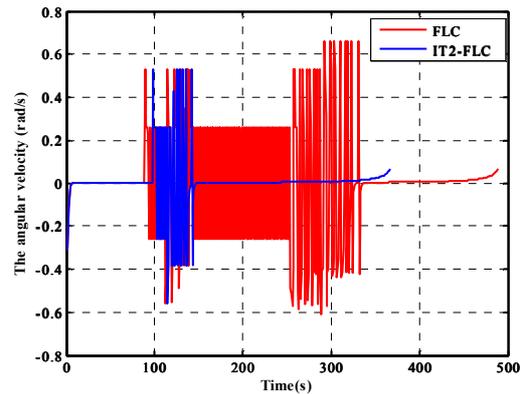


Fig. 21. The angular velocity of the mobile robot in the presence of obstacles.

In order to test the applicability of the IT-2FLC system, we compared its performance with that of an FL controller. The IT-2FLC offers better results than its FL counterpart in environments with obstacles. The main characteristic of the IT-2FLC sets is its ability to handle uncertainties more efficiently than FLC. This is made possible because a larger number of parameters and more freedom degrees are available in the IT-2FLC sets.

REFERENCES

[1] P. Gil, Y. Mezouar, M. Vincze, and J. A. Corrales, "Robotic Perception of the Sight and Touch to Interact with Environments," *Journal of Sensors*, vol. 2016, Dec. 2016, Art. no. e1751205, <https://doi.org/10.1155/2016/1751205>.  
 [2] U. Libal and J. Plaskonka, "Noise sensitivity of selected kinematic path following controllers for a unicycle," *Bulletin of the Polish Academy of Sciences, Technical Sciences*, vol. 62, no. 1, pp. 3-13, Mar. 2014, <https://doi.org/10.2478/bpasts-2014-0001>.

- [3] G. Abdelhakim and H. Abdelouahab, "A New Approach for Controlling a Trajectory Tracking Using Intelligent Methods," *Journal of Electrical Engineering & Technology*, vol. 14, no. 3, pp. 1347–1356, May 2019, <https://doi.org/10.1007/s42835-019-00112-1>.
- [4] B. Hua, E. Rama, G. Capi, and M. Jindai, "A human-like robot intelligent navigation in narrow indoor environments," *International Journal of Information and Electronics Engineering*, vol. 6, no. 5, pp. 308–312, Sep. 2016, <https://doi.org/10.18178/IJIEE.2016.6.5.644>.
- [5] K. Chen, F. Yang, and X. Chen, "Planning with task-oriented knowledge acquisition for a service robot," in *Proceedings of the Twenty-Fifth International Joint Conference on Artificial Intelligence*, New York, NY, USA, Jul. 2016, pp. 812–818.
- [6] Y. Gigras and K. Gupta, "Artificial Intelligence in Robot Path Planning," *International Journal of Soft Computing and Engineering*, vol. 2, no. 2, pp. 471–474, May 2012.
- [7] B. Damas and J. Santos-Victor, "Avoiding moving obstacles: the forbidden velocity map," in *2009 IEEE/RSJ International Conference on Intelligent Robots and Systems*, St. Louis, MO, USA, Oct. 2009, pp. 4393–4398, <https://doi.org/10.1109/IROS.2009.5354210>.
- [8] W. Yu, J. Peng, X. Zhang, and K.-C. Lin, "A Cooperative Path Planning Algorithm for a Multiple Mobile Robot System in a Dynamic Environment," *International Journal of Advanced Robotic Systems*, vol. 11, no. 8, Aug. 2014, Art. no. 136, <https://doi.org/10.5772/58832>.
- [9] Xiaoyu Yang, M. Moallem, and R. V. Patel, "A layered goal-oriented fuzzy motion planning strategy for mobile robot navigation," *IEEE Transactions on Systems, Man, and Cybernetics, Part B (Cybernetics)*, vol. 35, no. 6, pp. 1214–1224, Dec. 2005, <https://doi.org/10.1109/TSMCB.2005.850177>.
- [10] D. J. Huh, J. H. Park, U. Y. Huh, and H. I. Kim, "Path planning and navigation for autonomous mobile robot," in *IEEE 2002 28th Annual Conference of the Industrial Electronics Society. IECON 02*, Seville, Spain, Nov. 2002, vol. 2, pp. 1538–1542, <https://doi.org/10.1109/IECON.2002.1185508>.
- [11] J. Courbon, Y. Mezouar, L. Eck, and P. Martinet, "A generic framework for topological navigation of urban vehicle," presented at the ICRA09 - Workshop on Safe navigation in open and dynamic environments Application to autonomous vehicles, May 2009.
- [12] T. Belker and D. Schulz, "Local action planning for mobile robot collision avoidance," in *IEEE/RSJ International Conference on Intelligent Robots and Systems*, Lausanne, Switzerland, Sep. 2002, vol. 1, pp. 601–606, <https://doi.org/10.1109/IRDS.2002.1041457>.
- [13] M. Defoort, J. Palos, A. Kokosy, T. Floquet, W. Perruquetti, and D. Boulinguez, "Experimental Motion Planning and Control for an Autonomous Nonholonomic Mobile Robot," in *Proceedings 2007 IEEE International Conference on Robotics and Automation*, Rome, Italy, Apr. 2007, pp. 2221–2226, <https://doi.org/10.1109/ROBOT.2007.363650>.
- [14] E.-H. Guechi, J. Lauber, and M. Dambrine, "Suivi de trajectoire d'un robot mobile non holonome en présence de retards sur les mesures," in *2010 IEEE CIFA*, Nancy, France, 2010.
- [15] N. N. Karnik, J. M. Mendel, and Qilian Liang, "Type-2 fuzzy logic systems," *IEEE Transactions on Fuzzy Systems*, vol. 7, no. 6, pp. 643–658, Dec. 1999, <https://doi.org/10.1109/91.811231>.
- [16] O. Kahouli, B. Ashammari, K. Sebaa, M. Djebali, and H. H. Abdallah, "Type-2 Fuzzy Logic Controller Based PSS for Large Scale Power Systems Stability," *Engineering, Technology & Applied Science Research*, vol. 8, no. 5, pp. 3380–3386, Oct. 2018, <https://doi.org/10.48084/etasr.2234>.
- [17] O. Castillo and P. Melin, *Type-2 Fuzzy Logic: Theory and Applications*. Berlin Heidelberg, Germany: Springer-Verlag, 2008.
- [18] B. Bouchon-Meunier and C. Marsala, *Logique floue. Principes, aide à la décision*. Paris, France: Hermes Science Publications, 2002.
- [19] Qilian Liang and J. M. Mendel, "Interval type-2 fuzzy logic systems: theory and design," *IEEE Transactions on Fuzzy Systems*, vol. 8, no. 5, pp. 535–550, Oct. 2000, <https://doi.org/10.1109/91.873577>.
- [20] N. N. Karnik and J. M. Mendel, "Type-2 fuzzy logic systems: type-reduction," in *SMC'98 Conference Proceedings. 1998 IEEE International Conference on Systems, Man, and Cybernetics*, San Diego, CA, USA, Oct. 1998, vol. 2, pp. 2046–2051, <https://doi.org/10.1109/ICSMC.1998.728199>.
- [21] H. A. Hagra, "A hierarchical type-2 fuzzy logic control architecture for autonomous mobile robots," *IEEE Transactions on Fuzzy Systems*, vol. 12, no. 4, pp. 524–539, Aug. 2004, <https://doi.org/10.1109/TFUZZ.2004.832538>.