

Fuzzy Control Of A Mobile Robot In A Dynamic Environment

Najmeh Asadi, Mazda Moatari, Padideh Rasouli, Elham Baneshi

Abstract: In this paper, a fuzzy control system is proposed to making intelligent the movement of a robot in a dynamic environment. We use a Kalman filter in order to determine the position of moving obstacle relative to the position of the robot at any moment. Our intelligent robot is able to change speed and direction of its movement at any moment, depending on the position of obstacle and the position of the target. Eventually, the simulation results confirm the system performance using MATLAB software.

Index Terms: Mobile Robot, Kalman Filter, Fuzzy Logic, Dynamic Environment

1. INTRODUCTION

The modern world that we are living in today, robots play a role more than attending imaginative stories or children's toys. We now see a widespread use of robots, and especially mobile robots, with a wide variety of applications in our environment. Types of intelligent robots come to help humans in difficult and dangerous environments such as space or at the bottom of the ocean. Also, the use of mobile robots is very common in order to serve humans with interesting and various applications and sometimes taking on very important and complex responsibilities such as disaster relief or help patients and people with disabilities or doctor's assistant in surgical operations. But the most basic issue with mobile robots is the performance of these robots in dynamic environments. The autonomous mobile robots must be planned in such a way that they can automatically find a path without collision with obstacles and reach the specified destination from a certain beginning point in order to properly carry out their duties. Therefore, an intelligent mobile robot should be programmed with knowledge and reasoning to have independent behavior. Our goal of this research is to making intelligent an autonomous mobile robot in a dynamic environment. The dynamic environment means an environment in which the obstacle position is changing at any moment. Using fuzzy logic is one of the methods to control the robot, in which lots of researches have been done so far. A robot arm system for rehabilitation is designed for patients with neurological and muscular disorders by force and position control using fuzzy logic [1]. Also, adaptive fuzzy controllers are designed for industrial applications [2]. In another research, a decentralized fuzzy controller for robot arms has been developed [3]. Also, a robust fuzzy controller is designed for robot arms [4]. Fuzzy logic in static environments is developed in [5] for mobile robot navigation. A sensor fusion technique is proposed to enhance the navigation rules using a modified fuzzy associative memory in [6].

And even a fuzzy-based approach to avoiding collision with obstacles for the human robot TWNHR-3 has been proposed in [7]. Another important subject that we are facing in this study is the issue of position estimation. With the introduction of Kalman filter in 1990, a great revolution in the problem of position estimation was created and many studies have been done in this regard. In the past, the extended Kalman filter and various types of it have been investigated for dealing with nonlinear estimation problem by replacing nonlinear equations with linear approximations [8]. Nam · K · H and his colleagues studied a second-order filter to access more accuracy [9]. Then an extended Kalman filter was designed to improve accuracy of approximation using the approximation of the measurement equations for the correction of last estimation instead of predicting the previous estimate [10]. Mira · R · K has checked an extended Kalman filter to transfer nonlinear equations of measurement to linear functions by selecting a suitable coordinate system [11]. It was also found in a study that a modified extended Kalman filter algorithm provides better performance rather than the estimated state vector from random equations by approximating nonlinear equations on the state vector which are calculated from algebraic equations [12]. In another study, the extended Kalman filter performance has been improved by changing the Taylor series approximation center [13]. Finally, Kalman filter has been used as a tool to estimate the obstacle position [14]. In the previous studies, the use of Kalman filter algorithm and fuzzy logic simultaneously, have not been tested. In this study by combining the powerful Kalman filter tool to estimate obstacle position and fuzzy logic to making intelligent the movement of autonomous mobile robot, an attempt has been made to design an intelligent control system and routing for an autonomous mobile robot in a dynamic environment. Also, by using input feedback linearization of the fuzzy system, we provide a new approach to control of mobile robots in dynamic environments. The proposed system is designed in such a way that our intelligent robot determines its own motion plan autonomously, by considering the position and the direction of moving obstacle, and also considering the position of target point. In this motion plan, by using fuzzy inference system and without human intervention, the robot starts to move from beginning point and estimates the position of obstacle sequentially by Kalman filter at any moment. with regard to direction of obstacle movement, the robot selects a direction to the target point in the way that it reaches target point without collision with moving obstacle. In this plan, the force applied to robot wheels is also determined according to the distance between robot and target by fuzzy inference system. So our

- *Authors affiliation:*
- *Department of Electrical Engineering, Marvdash Branch, Islamic Azad University, Marvdasht, Iran*
- *Email: n.asadi.0014@gmail.com*
- *Department of Electrical Engineering, Marvdash Branch, Islamic Azad University, Marvdasht, Iran*
- *Email: moattari@miau.ac.ir*
- *Department of Electrical Engineering, Marvdash Branch, Islamic Azad University, Marvdasht, Iran*
- *Email: p.rasouli@miau.ac.ir*

autonomous mobile robot will be able to move from beginning point and travel along the path to target without colliding with moving obstacle and ultimately stop at the target point. Therefore, the innovation of this research is in the simultaneous use of Kalman filter and fuzzy logic to making intelligent a mobile robot, and also the input feedback linearization of the fuzzy system. It should be noted that the designed system, in the simplest way, offers very good performance. In section 2, the Extended Kalman Filter algorithm and its operation are expressed. In section 3, models of the moving obstacle and the mobile robot are described. Design of the intelligent control system is explained in section 4. Simulation results are presented in section 5. And finally, the conclusion is given in section 6.

2-EXTENDED KALMAN FILTER

This filter is actually a Kalman filter, which linearizes in the current average range. In this way, we can make estimates of the current calculation, using partial derivatives of process functions and measurements, even in the face of nonlinear relationships. Assuming that the process has a state vector in the form $x \in R^n$, it is expressed by the nonlinear randomized differential equation below [15]:

$$x_k = f(x_{k-1}, u_k, w_{k-1}) \quad (1)$$

And the measurement equation is as follows [15]:

$$z_k = h(x_k, v_k) \quad (2)$$

The random variables w_k and v_k respectively represent process noise and measurement noise. The nonlinear function f connects the states in previous time step to the states in current time step and The non-linear function h links the states to the measurement. The set of time update equations, and EKF filtering measurements, are as follows [15]:

$$\hat{x}_{\bar{k}} = f(\hat{x}_{\bar{k}-1}, u_k, 0) \quad (3)$$

$$P_{\bar{k}} = A_k P_{k-1} A_k^T + w_k Q_{k-1} w_k^T \quad (4)$$

$$K_k = P_{\bar{k}} H_k^T (H_k P_{\bar{k}} H_k^T + v_k R_k v_k^T)^{-1} \quad (5)$$

$$\hat{x}_k = \hat{x}_{\bar{k}} + K_k (z_k - h(\hat{x}_{\bar{k}}, 0)) \quad (6)$$

$$P_k = (I - K_k H_k) P_{\bar{k}} \quad (7)$$

In the time update equations, states and covariance are estimated from step $k-1$ to the current step. And in the updating equations of measurement, the estimated states and covariance are corrected. H_k and v are the Jacobins of measurement in step k and R_k is the noise covariance of measurement in step k [15].

Table (1) shows an algorithm for estimating an obstacle position with a Kalman filter [15].

TABLE 1. ALGORITHM FOR ESTIMATING AN OBSTACLE POSITION WITH A KALMAN FILTER [15]

- 1-Estimate the initial target pose X_p
- 2-Augment to state variable $X_0 = \{X_p, \dot{X}_p\}$
- 3-Initialize **EKF** ; X_0, P_0, q_0, Q_0, R
- 4- **while** (**EKF** enabled)
- 5-{
- 6-Estimate next state variable and covariance matrix $X_{k|k-1} = AX_{k-1|k-1} + Bq_{k-1}$,

$$P_{k|k-1} = AP_{k-1|k-1}A^T + BQ_{k-1}B^T$$

$$7\text{-Evaluate Jacobian of measurement model } H_k = \frac{\partial h(X)}{\partial X|_{X=X_{k|k-1}}}$$

$$8\text{-Calculate Kalman gain } K_g = P_{k|k-1}H_k^T(H_kP_{k|k-1}H_k^T + R)^{-1}$$

9-Update state variable and covariance matrix

$$X_{k|k} = X_{k|k-1} + K_g (Z_k - h(X_{k|k-1})),$$

$$P_{k|k} = P_{k|k-1} - K_g H_k P_{k|k-1}$$

10-Output $X_{k|k}$

11-}

3-MODELING

3-1 Model of the moving obstacle

Considering a moving arm as an obstacle, which moves in two-dimensions, the state vector of obstacle is defined as follows [15]:

$$X = \{x_{TO}, \dot{x}_{TO}, y_{TO}, \dot{y}_{TO}\}^T \quad (8)$$

Where $\{x_{TO}, y_{TO}\}^T$ is coordinates of the fixed body of obstacle and $\{\dot{x}_{TO}, \dot{y}_{TO}\}^T$ represents derivation with respect to time. By assuming that the obstacle motion is uniform within a sample time interval t and if t is sufficiently small, then the higher order motion can be regarded as process noise in the form of $w = \{\dot{x}_{TO}, \dot{y}_{TO}\}^T$. So the system model is defined as [15]:

$$X_k = AX_{k-1} + Bw_{k-1} \quad (9)$$

Notes that the subscripts k and $k-1$ indicate current and previous sampling time. The coefficient matrix of system model A , and the coefficient matrix of process noise B , are defined as follows [15]:

$$A = \begin{bmatrix} a & 0 \\ 0 & a \end{bmatrix}, \quad a = \begin{bmatrix} 1 & t \\ 0 & 1 \end{bmatrix}$$

$$B = \begin{bmatrix} b & 0 \\ 0 & b \end{bmatrix}, \quad b = \begin{bmatrix} \frac{t^2}{2} \\ t \end{bmatrix}$$

(10)

The measurement model from a pin-hole camera model, is expressed in the following vector equation [15]:

$$Z_k = C(X_k) + \mu_k \quad (11)$$

Where Z is the real measurement vector and $C(X_k)$ is the estimated measurement vector based on the camera model and μ stands for the measurement noise of the camera, which may be caused by different natural conditions. This kind of noise can be considered as Gaussian noise and obeys the normal distribution [15].

3-2 Model of the mobile robot

The robot, which studied in this research is a holonomic omnidirectional mobile robot with three orthogonal wheels (Fig. 1). In this image the global coordinate system $x-o-y$ and the robot's body coordinate system $x_b-o_b-y_b$ are shown. θ denotes the orientation of the robot, D_1, D_2 and D_3 are the driving forces for the assemblies [16].

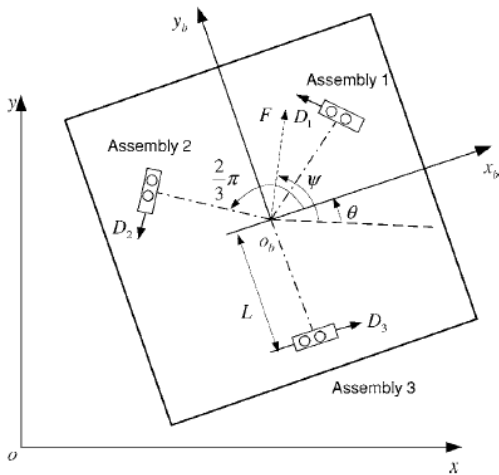


Fig 1. omnidirectional mobile robot model [16]

The applied force which determines the translational motion of the robot is given by the following equation [16]:

$$ma = F \tag{12}$$

To convert the applied force F into the driving input torque to individual wheel, we need to define the dynamic model of the robot. The driving system for each assembly can be modeled by [16]:

$$I_{\omega} \dot{\omega}_i + c\omega_i = ku_i - rD_i \quad (i = 1,2,3) \tag{13}$$

Where I_{ω} is the moment of inertia of the wheel around the driving shaft, ω_i is the rotational rate of the wheel, c is the viscous friction factor for the wheel, r is the radius of the wheel, k is the driving gain factor and u_i is the driving input torque. Define the state variable for the robot as $x = [\dot{x}, \dot{y}, \dot{\theta}]^T$ and the manipulated variable as $u = [u_1, u_2, u_3]^T$. Denote I_r as the moment of inertia of the robot and L as the distance between any assembly and the center of gravity of the robot, the dynamic model of the robot is as follows [16]:

$$\dot{x} = A(x)x + B(x)u \tag{14}$$

Where the matrixes A and B are equal to [16]:

$$A(x) = \begin{bmatrix} \frac{-3c}{3I_{\omega} + 2m_{rob}r^2} & \frac{-3I_{\omega}\dot{\theta}}{3I_{\omega} + 2m_{rob}r^2} & 0 \\ \frac{3I_{\omega}\dot{\theta}}{3I_{\omega} + 2m_{rob}r^2} & \frac{-3c}{3I_{\omega} + 2m_{rob}r^2} & 0 \\ 0 & 0 & \frac{-3cL^2}{3I_{\omega}L^2 + I_vr^2} \end{bmatrix},$$

$$B(x) = \begin{bmatrix} \frac{-kr(\sqrt{3}\sin\theta - \cos\theta)}{3I_{\omega} + 2m_{rob}r^2} & \frac{kr(\sqrt{3}\sin\theta - \cos\theta)}{3I_{\omega} + 2m_{rob}r^2} & \frac{2kr\cos\theta}{3I_{\omega} + 2m_{rob}r^2} \\ \frac{kr(-\sqrt{3}\cos\theta - \sin\theta)}{3I_{\omega} + 2m_{rob}r^2} & \frac{kr(\sqrt{3}\cos\theta - \sin\theta)}{3I_{\omega} + 2m_{rob}r^2} & \frac{2kr\sin\theta}{3I_{\omega} + 2m_{rob}r^2} \\ \frac{krL}{3I_{\omega}L^2 + I_vr^2} & \frac{krL}{3I_{\omega}L^2 + I_vr^2} & \frac{krL}{3I_{\omega}L^2 + I_vr^2} \end{bmatrix} \tag{15}$$

Apparently, the system is nonlinear and coupled. Nevertheless, using feedback linearization, we know that control $u = B(x)^{-1}(u^* - A(x)x)$, can transform Equation (14) into the following form [16]:

$$\dot{x} = u^* \tag{16}$$

Where $u^* = [\alpha^{*T} \ddot{\theta}^*]^T \in R^3$ is the new control signal to be determined. α^* is the desired acceleration of robot and θ^* is the desired orientation of the robot. For this holonomic omnidirectional robot, the orientation and the position can be controlled independently. In the following discussions, the orientation of the robot is assumed to be kept unchanged, i.e., $\ddot{\theta}^* = 0$ [16].

4-THE INTELLIGENT CONTROL OF ROBOT

In this research, intelligent control of a robot in a dynamic environment means the choice of a path, which the robot chooses completely independently and without human intervention and this path can bring the robot to the target without collision with the obstacle. The proposed fuzzy control system, which is designed using Fuzzy Mamdani systems, consists of two separate Fuzzy control systems, including a Force Fuzzy Control System, and a Rotation Angle Fuzzy Control System. Force Control means that, by determining the distance between itself and the target, the robot determines its own motion force. So that, at long distances with more force and at near distances with less force, moves toward the target. until it finally stops at the target. Rotation Angle Control, allows the robot to instantly detect the position of obstacle and the position of target relative to its own location, and to change its direction, in order to avoid obstacle and approaching the target.

4-1 Design of the Force Fuzzy Control System

In order to control the applied force to robot wheels, a fuzzy system is designed with 1 input and 1 output, which at any moment determines amount of the applied force to robot wheels by considering the distance between robot and target. The input of this system; R_2 , is the distance between robot and target. for this variable 4 fuzzy sets are defined as "Zero", "Very Close", "Close" and "Far", which are briefly named as Z , VC , C , and FA . The output of this system; F , is the applied force to robot wheels. This variable also belongs to four fuzzy sets, which include "Zero", "Very Slow", "Slow" and "Fast", and are named as Z , VSL , SL , and FS respectively.

The task of this controller system, which is designed using Mamdani fuzzy model, is to determine the amount of output force and is completely dependent on the distance between robot and target. The scenario of this system is: the more the distance between robot and target is, the greater the force applied to robot, and as the distance decreases, the system output will also decrease. Table (2) is the rule base, which consists of 4 fuzzy rules in order to fulfill system's task.

TABLE 2. RULE BASE OF THE FORCE CONTROL SYSTEM

R2			
Z	VC	C	FA
Z	VS	S	FS
	L	L	

4-2 Designing of the Rotation Angle Fuzzy Control System

The task of this fuzzy control system is to track target and obstacle, and helps robot to avoid collision with obstacle and rotation at the right angle in a way that is in the direction of reaching the target point. It can be said that the task of the Angle Controller System is very important, and in the event of

a poor performance of this system, the possibility of collision with obstacle is very probable in our dynamic environment. On the other hand, the system must also monitor the target, and adjust the robot's path so that robot eventually reaches target. This controller has 3 input variables called θ_1 , θ_2 and R_{12} , which are respectively "The obstacle position relative to robot", "The target position relative to robot" and "The subtraction of the distance between robot and obstacle and the distance between robot and target". The variable α , which is "The rotation angle of robot", is defined as the single output of this system. Fuzzy sets of this variables are as follows:
 The variable θ_1 belongs to 4 fuzzy sets "Left Down", "Left Up", "Right Up" and "Right Down", which are briefly called as LD, LU, RU and RD respectively. The variable θ_2 belongs to 5 fuzzy sets called as "Zero", "Left Down", "Left Up", "Right Up" and "Right Down", which are briefly called as Z, LD, LU, RU and RD. And the last input of this system; R_{12} belongs to 2 fuzzy sets "Near" and "Far", called N and FA respectively. α ; The output of this system also belongs to 5 fuzzy sets called as "2 Step To Right", "1 Step To Right", "Go Straight", "1 Step To Left" and "2 Step To Left" Which are briefly named as 2SR, 1SR, GS, 1SL, and 2SL. In order to perform a desirable task, the rule base of this system consist of 40 fuzzy rules, which are shown in Table (3).

TABLE 3. RULE BASE OF THE ANGLE CONTROL SYSTEM

		Theta2					
		LD	LU	RU	RD	Z	
R12	LD	2SL	1SL	1SR	2SR	GS	N
	LU	2SL	1SL	1SR	2SR	GS	
	RU	2SL	1SL	1SR	2SR	GS	
	RD	2SL	1SL	1SR	2SR	GS	
	LD	GS	1SL	1SR	2SR	GS	
T	LU	2SL	GS	1SR	2SR	GS	FA
	RU	2SL	1SL	GS	2SR	GS	
			1SL		GS	GS	
	RD	2SL		1SR		GS	



It is worth noting that the forces applied to the robot wheels (control forces) will ultimately be a combination of force and angle (the outputs of the fuzzy system), F and α .

$$F_y = F \sin \alpha \quad , \quad F_x = F \cos \alpha \quad (17)$$

5. SIMULATION

As we know, the robot's purpose in this system is to reach to the coordinates of the target point without collision with the moving obstacle. In order to achieve this goal, robot must decide at any moment that with how much force and at what angle to move. The obstacle position is estimated at any moment by Kalman filter. Then, our intelligent robot, considering the position of obstacle and target, measures the distances between itself and obstacle and target. Also, robot considering the Current and previous coordinates of obstacle, will be able to detect the direction of obstacle's motion. Therefore, robot, with the available information, assesses force and angle of its motion to reach to the target without collision with obstacle. After starting to move in a collision-free path to target, it stops at the target. In order to test the performance of proposed system, simulations are carried out in two steps. In the first step, the estimation of moving obstacle position and, in the second step, the results of proposed system's performance are displayed in different situations.

5-1 Step 1

At this step, with the knowledge of the system model and the measurement model, the obstacle position is estimated at any given moment. Figures 2 to 4 represent results of this simulation.

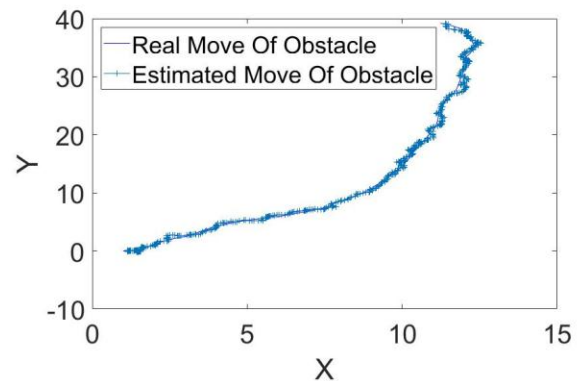


Fig 2. Estimated position of obstacle along the axes x and y

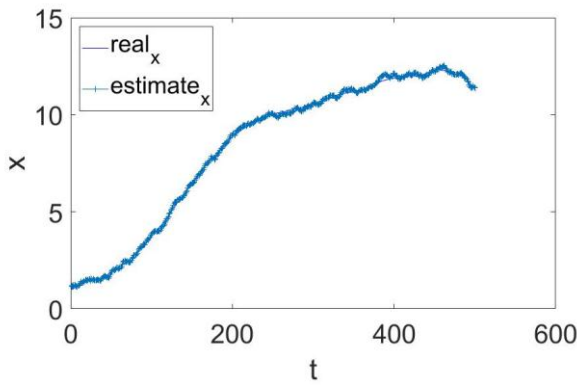


Fig 3. Estimated position of obstacle relative to time along the axis x

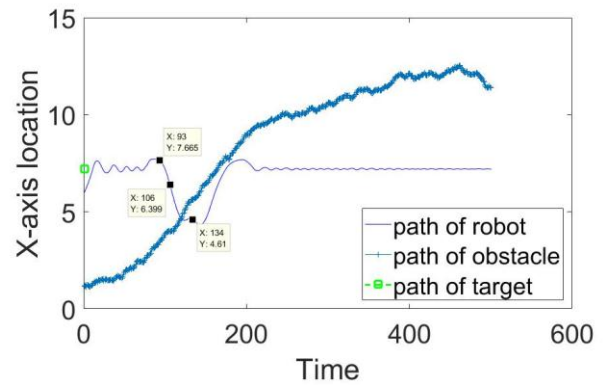


Fig 6. The path of robot, obstacle and target, relative to time along axis x (Case A)

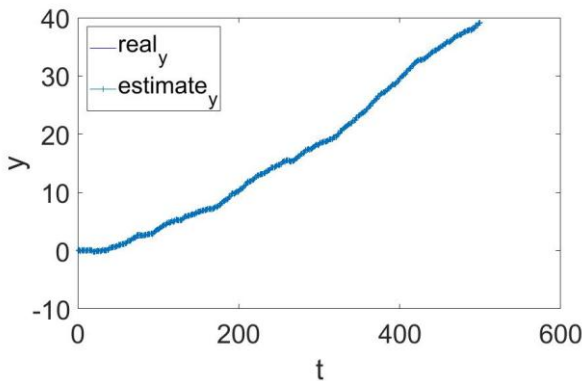


Fig 4. Estimated position of obstacle relative to time along the axis y

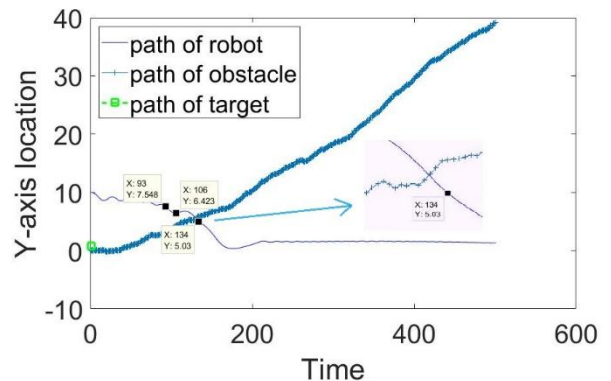


Fig 7. The path of robot, obstacle and target, relative to time along axis y (Case A)

As it can be seen, the estimated values are very close to the real values.

5-2 Step 2

At this step, simulations in different scenarios are taken in terms of coordinates of the beginning point of robot's motion and target point. The results of these simulations are shown below.

Case A:

The target's coordinates: $P_{tar} = [7.2, 0.8]$

The robot's initial coordinates: $P_{rob} = [6, 10]$

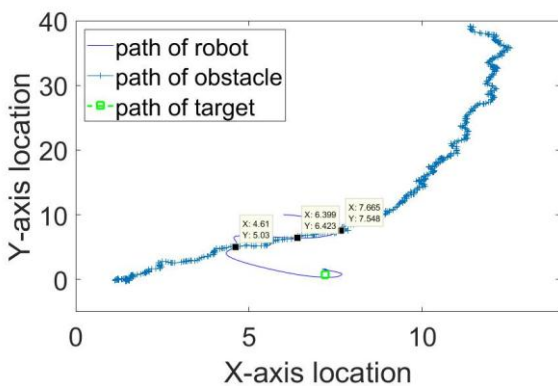


Fig 5. The path of robot, obstacle and target along axes x and y (Case A)

Case B:

The target's coordinates: $P_{tar} = [10, 9.5]$

The robot's initial coordinates: $P_{rob} = [6, 12]$

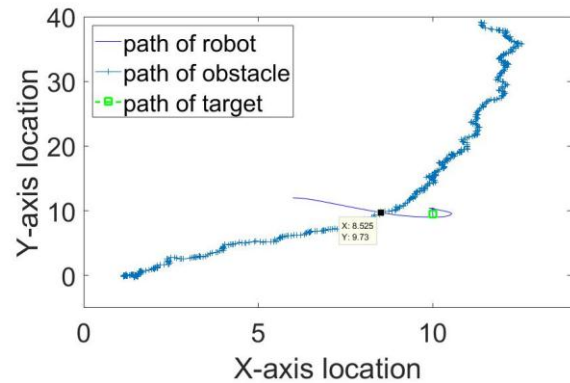


Fig 8. The path of robot, obstacle and target along axes x and y (Case B)

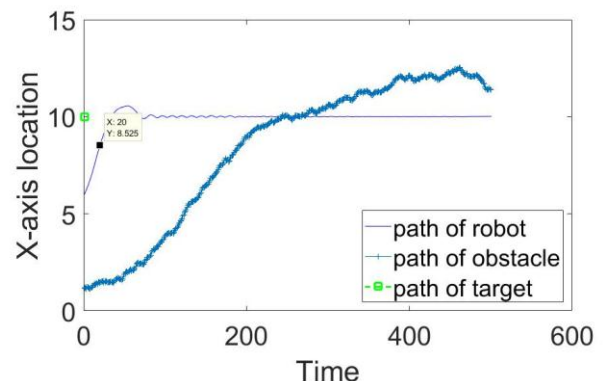
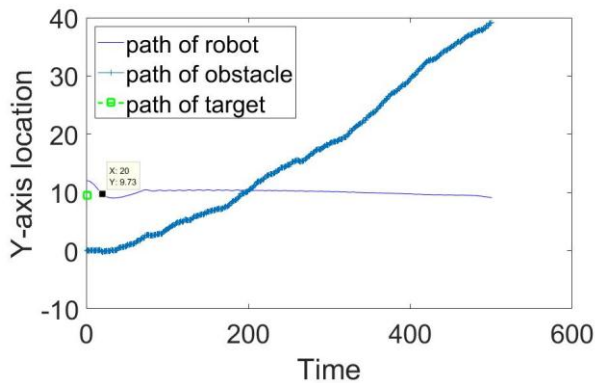
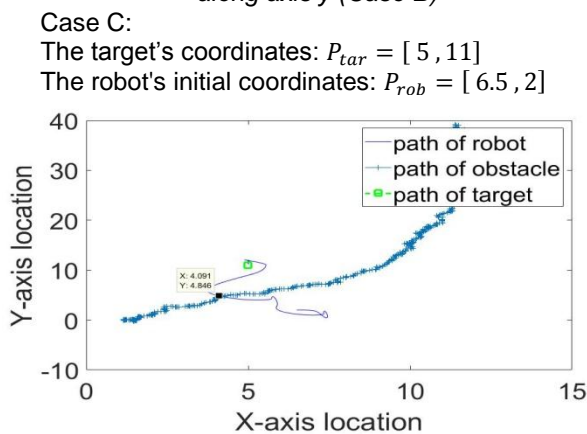
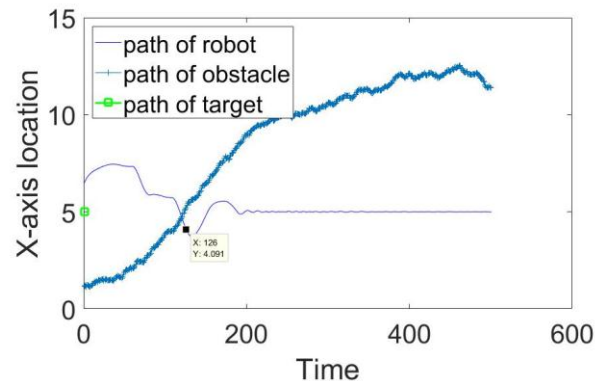
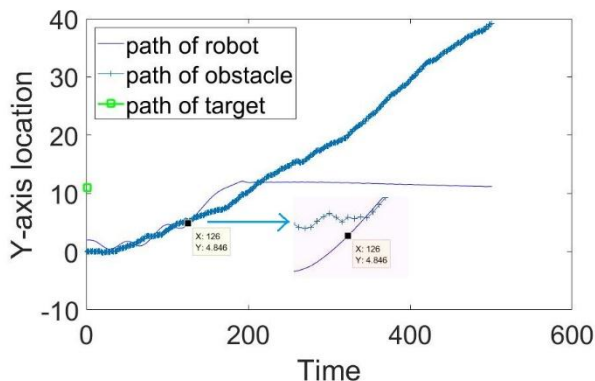


Fig 9. The path of robot, obstacle and target, relative to time along axis x (Case B)**Fig 13.** The path of robot, obstacle and target, relative to time along axis y (Case C)

As the images are illustrative, in all cases, robot has reached the goal without collision with the obstacle. For each case, three images are displayed. In all simulation results, the first image shows the path of robot, obstacle and the coordinates of target point along axes x and y, the second image shows the path of robot, obstacle and the coordinates of target point relative to time along x-axis and the third image shows the path of robot, obstacle and the coordinates of target point relative to time along y-axis. At some points in the images of axes x and y, it seems that robot has hit obstacle but by referring to these points in the images that are relative to time, it has been shown that at the given time there was no collision.

Fig 10. The path of robot, obstacle and target, relative to time along axis y (Case B)**Fig 11.** The path of robot, obstacle and target along axes x and y (Case C)**Fig 12.** The path of robot, obstacle and target, relative to time along axis x (Case C)

6- CONCLUSION

In this research, we tried to use the Kalman filter algorithm and fuzzy logic simultaneously, and also the linearization of input feedback of the fuzzy system, to making intelligent and controlling mobile robots in dynamic environments. Using the Kalman filter to estimate the obstacle position, provides detailed information about the position of the moving obstacle, to the robot. Then, the fuzzy control system, consisting of force control and angle control, has succeeded in achieving the target in addition to a desirable performance in order to avoid the moving obstacle. Finally, it can be concluded that the application of the proposed methodology and design, despite the simplicity, has been successful.

REFERENCES

- [1] Ming-Shaung Ju, Chou-Ching K. Lin, Dong-Huang Lin, Ing-Shiou Hwang, and Shu-Min Chen, A Rehabilitation Robot With Force-Position Hybrid Fuzzy Controller: Hybrid Fuzzy Control of Rehabilitation Robot, IEEE Transactions on neural systems and rehabilitation engineering, Vol. 13, No. 3, September 2005.
- [2] Marius-Constantin Popescu, Ilie Borcosi, Onisifor Olaru, Luminita Popescu, Florin Grofu, The simulation hybrid fuzzy control of Scara robot, Wseas transactions on systems and control, 2008.
- [3] Yaochu Jin, Decentralized Adaptive Fuzzy Control of Robot Manipulators, IEEE Transactions on systems, man, and cybernetics-Part B: cybernetics, Vol. 28, No. 1, February 1998.
- [4] C Ham, Z Qu, R Johnson, Robust fuzzy control for robot manipulators, IEE Proceedings-Control Theory and Applications, 2000.
- [5] Ngangbam Herojit Singh and Khelchandra Thongam, Mobile robot navigation using fuzzy logic in static environment, 6th International Conference on Smart Computing and Communications, ICSCC 2017, 7-8 December 2017, Kurukshetra, India.
- [6] S.Parasuraman, sensor Fusion for Mobile Robot navigation: Fuzzy Associative memory, International Symposium on Robotics and Intelligent Sensors 2012 (IRIS 2012)
- [7] Ching-Chang Wong, Chi-Tai Cheng, Kai-Hsiang Huang, and Yu-Ting Yang, Fuzzy Control of Humanoid Robot for Obstacle Avoidance, International Journal of Fuzzy Systems, Vol. 10, No. 1, March 2008.

- [8] Kalman, R.E. and R.S. Bucy, New results in linear filtering and prediction theory. *Journal of basic engineering*, 1961. 83 :3(p. 95-108.
- [9] Nam, K. and M.-J. Tahik, A second-order stochastic filter involving coordinate transformation. *IEEE transactions on automatic control*, 1999. 44(3): p. 603-608.
- [10] Bellaire, R.L., E.W. Kamen, and S.M. Zabin. New nonlinear iterated filter with applications to target tracking. in *SPIE's 1995 International Symposium on Optical Science, Engineering, and Instrumentation*. 1995. International Society for Optics and Photonics.
- [11] Mehra, R., A comparison of several nonlinear filters for reentry vehicle tracking. *IEEE Transactions on Automatic Control*, 1971. 16(4): p. 307-319.
- [12] Wall, D. and F. Gaston. Modified extended Kalman filtering. in *Digital Signal Processing Proceedings, 1997. DSP 97., 1997 13th International Conference on*. 1997. IEEE.
- [13] Chui, C., G. Chen, and H. Chui, Modified extended Kalman filtering and a real-time parallel algorithm for system parameter identification. *IEEE Transactions on Automatic Control*, 1990. 35(1): p. 100-104.
- [14] Zezhong, X., et al. Position estimation for a mobile robot with augmented system state. in *Neural Networks and Brain, 2005. ICNN&B'05. International Conference on*. 2005. IEEE.
- [15] Dong, G. and Z.H. Zhu, Autonomous robotic capture of non-cooperative target by adaptive extended Kalman filter based visual servo. *Acta Astronautica*, 2016. 122: p. 209-218.
- [16] Ge, S.S. and Y.J. Cui, Dynamic motion planning for mobile robots using potential field method. *Autonomous robots*, 2002. 13(3): p. 207-222.