Flexible Path Planning of Mobile Robot for Avoiding the Dynamic Obstacles Using Fuzzy Controllers

Pham Minh Tuan, Nguyen Duc Tai, Tran Quang Huy, and Nguyen Truong Thinh *

Institute of Intelligent and Interactive Technologies, University of Economics Ho Chi Minh City–UEH, Ho Chi Minh City, Vietnam

Email: minhtuan3052001@gmail.com (P.M.T.); nguyenductai243@gmail.com (N.D.T.); huyquangtran08@gmail.com (T.Q.H.); thinhnt@ueh.edu.vn (N.T.T.)

*Corresponding author

***Abstract*—Mobile robots are increasingly used in service and industrial activities. For many previous decades, its navigation has always remained an open problem for researchers in the World. The target of this research is to develop a control method for mobile robots operating in dynamic and unknown environments with both static and moving obstacles. The method utilizes a fuzzy logic approach based on a Lidar sensor to gather information about the environment and make decisions for navigation. The study focuses on the use of mobile robots in material handling applications such as warehouses or libraries with flat floors. The results demonstrate the ability of the proposed method to navigate the mobile robot around both static and moving obstacles, enabling its use in various material handling applications. The study provides a potential solution for the open problem of mobile robot navigation in complex environments.**

***Keywords*—fuzzy controller, mobile robot, wheeled robot, path planning, trajectory**

I. INTRODUCTION

Nowadays, the service robots have abilities of working in complex environment as well as avoiding the moving obstacles and static one like as map. The mobile robot's navigation includes sensing of surrounding environment, localization, path planning, and motion control. *Lidar* sensor is used to percept its surrounding environment, determine its position and velocity of obstacles forward. Planning path in accordance with the task by using cognitive decision-making is a necessary stage before tracking to following the pre-determined trajectory by affordable controller.

Many researchers have expected that mobile robots undertake a variety of tasks in various fields, such as the supply of agricultural weeds, warehouses, construction sites and hospital etc. [1–4]. For applications in warehouses, mobile robots are used to handle the transport of materials from storage rooms to production sites or production lines and inventory various goods. For this reason, communication control needs to be suitable for the movement and navigation of a mobile robot in a predetermined environment. Indeed, the mobile robot's control requires the following good features: Self-organization, flexibility, improved result quality, and energy savings. In this paper, a fuzzy logic controller is used because it does not require a mathematical model of the controlled process. This will give the vehicle control with different mathematical models. In this paper, model of static and dynamic obstacle avoidance for the mobile robot is described with constant velocity of the moving obstacle. The robot can be able to flexibly move on the map in many different situations in order to get an optimal path.

In this work, two Fuzzy Logic Controller (FLC), using different membership functions, are used for mobile robot navigation in an unknown and dynamic environment. A Tracking Fuzzy Logic Controller (TFLC) is occurred to navigate the WMR to the target. An obstacle avoidance of fuzzy logic controller (OAFLC) is used to perform obstacle avoidance behavior. The main challenge in the navigation operation of the WMR is the dynamic environment and the insufficient information on the environment. The traditional mobile robot planning approaches remain not outstanding and unable to overcome challenges. Therefore, many reaction approaches have been introduced that allow the use of artificial intelligence techniques, including problem solving, learning, and reasoning. In this domain, fuzzy logic [5–7], neural networks [8–10] and other controlling techniques [11–13], have become the cornerstone of the navigation system in mobile robots. At stated in Pradhan *et al.* [14], a navigation approach for several mobile robots in an unknown environment using the fuzzy logic technique has been presented. In this paper, the Fuzzy Logic Controller (FLC) is used by the authors with a lot of different membership functions (three-membership functions and five-membership functions) to control their mobile robots in an unstructured environment. According to Oriolo *et al.* [15], A real-time

Manuscript received January 3, 2023; revised March 22, 2023; accepted June 25, 2023; published February 15, 2024.

fuzzy target tracking control scheme for autonomous mobile robot using infrared sensors is presented. It uses two robots, the first WMR is the moving target and the second is tracking to it. Li *et al.* [16] have presented indoor navigation using fuzzy logic. This paper proposed how to use FLC for the target tracking control of WMRs. As reported by Rashid *et al.* [17], they just use FLC for motion and do not use any FLC to avoid obstacles. Another indoor navigation system using fuzzy logic is expressed in [18]. The authors used the fuzzy logic and visual sensors to control the robot to the target. But this scheme just uses FLC for WMR motion, it does not use FLC to avoid the obstacles.

II. Kinematic of Wheeled Mobile Robot

The mobile robot has to move from a start point from desired one along the pre-determined path and avoid the obstacles, which percept in working process. The controller decides the optimal path of combine between a planning path and obstacle-avoidance trajectory, it is collision-free path. This paper gives some assumptions like as: all of obstacles are presented by respective bounding circles; there is only an obstacle in considering area; no slip occurs on the pure-rolling wheel of mobile robot. The mobile robot is in this study a four-wheel platform with two separate driving wheels mounted on the same axle and the remaining castor wheels to balance the mobile robot during movement. These two active wheels are independently controlled by two actuators to reach desired position and orientation (Fig. 1). The dynamic model of mobile robot is described by the following nonlinear Eqs. (1)–(3) [16].

$$\dot{x}_{(t)} = v_{(t)} \cdot \cos\left(\theta_{(t)}\right) \tag{1}$$

$$\dot{y}_{(t)} = v_{(t)} \cdot \sin\left(\theta_{(t)}\right) \tag{2}$$

$$\dot{\theta}_{(t)} = \omega_{(t)} \tag{3}$$

where *x* and *y* are the coordinates of the position of the robot. θ is the direction of the mobile robot with respect to the positive direction X-axis. Besides, *v* is the linear velocity and ω is the angular velocity.

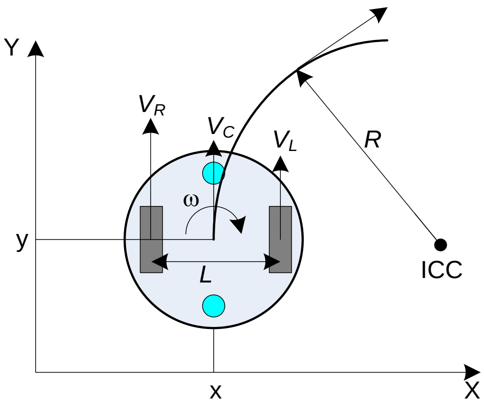


Fig. 1. Kinematics of mobile platform.

III. Dead-Reckoning

Dead-reckoning is used to estimate the position and direction of a WMR during motion. The dead-reckoning method is based on the previous position and direction at time $t = t$ (k) to calculate the next position and direction at $t = t$ (k+1). Using the Euler approximation, Eq. (1) is written as follows:

$$x_{(k+1)} = x_k + v_{(k)} \cdot T_s \cdot \cos\left(\theta_{(k)}\right) \tag{4}$$

$$y_{(k+1)} = y_k + v_{(k)} \cdot T_s \cdot \sin\left(\theta_{(k)}\right) \tag{5}$$

$$\theta_{(k+1)} = \theta_{(k)} + \omega_{(k)} \cdot T_s \tag{6}$$

where $T_s = t_{(k+1)} - t_{(k)}$ is the sampling time. It is preferable to use the distance travelled by the mobile robot during T_s which can be calculated by using pulses of the encoders on each motor of the robot DC wheel. This mathematical model is described by these following equations.

$$x_{(k+1)} = x_k + \frac{\pi \cdot D}{2} \cdot \frac{\left(\Delta r_L + \Delta r_R\right)}{r_w} \cdot \cos\left(\theta_{(k)}\right) \tag{7}$$

$$y_{(k+1)} = y_k + \frac{\pi \cdot D}{2} \cdot \frac{\left(\Delta r_L + \Delta r_R\right)}{r_w} \cdot \sin\left(\theta_{(k)}\right) \tag{8}$$

$$\theta_{(k+1)} = \theta_{(k)} + \frac{\pi \cdot D}{dst} \cdot \frac{\left(\Delta r_R - \Delta r_L\right)}{r_w} \tag{9}$$

where $\Delta T = T(k+1) - T(k)$ are the pulses of the encoders (L stands for left and R stands for right) during the sampling time *Ts*, *dst* is the distance between the leading wheels, *D* is the diameter of leading wheels, and Tw is the number of encoder pulses for a full rotation. It is known that this method is not accurate for fast dynamic motion of a WMR or slipped. However, using dept sensor lidar together can eliminate cumulative errors.

IV. Fuzzy Logic Controller

Two fuzzy logic controllers are developed and used to navigate the mobile robot from its original configuration to the target.

Tracking Fuzzy Logic Control (TFLC) and Obstacle Avoidance Fuzzy Logic Control (OAFLC) are combined to move the mobile robot to the target along a collision path. In the first, the algorithm starts with TFLC. If any obstacles appear inside safe zone of the mobile robot, continue checking to see if obstacles are in the avoiding zone. If the avoiding zone is compromised, then the controlling algorithm switches to the OAFLC. Otherwise, it will continue to be in TFLC algorithm. The output of TFLC and OAFLC are the left and right velocities of each leading wheel. Fig. 2 illustrates the fuzzy controller of both TFLC and OAFLC. Because the robot do not always need to avoid obstacles. If collision does not occur there is no need to start with the definition of a safe zone (*S*) first. Safe zone is an area that surrounds the robot and the area predicted in the future (at a time Δ*t*) the robot able to collide with obstacles (Fig. 3).

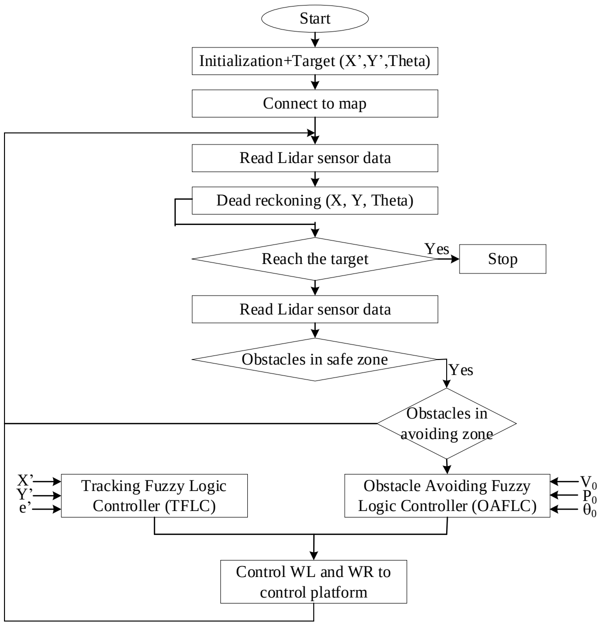


Fig. 2. Flowchart of the fuzzy logic algorithm.

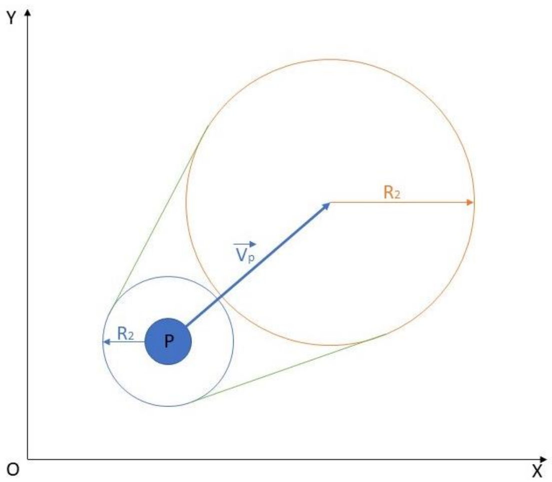


Fig. 3. Simulation of safe zone.

The region is based on the input factors related to *Vp* which is the instantaneous velocity of the robot in m/s. *Rp* is the radius of the robot, R_0 is the radius of the obstacle and V_0 is the maximum velocity whose obstacle can be achieved. For example, obstacles moving at a speed too fast will not be considered. The safe zone is defined by drawing a circle with radius R_1 centered at the center of the robot (P). R_1 is equal to the sum of R_1 radius, obstacle radius *Rp* and safety factor γ are selected from 50 mm to 200 mm.

$$R_1 = R_0 + R_p + \gamma \tag{10}$$

Next is drawing a circle with radius R_2. The center of this circle is about a distance *X* from the center of (P, R1) equal to the magnitude of the velocity of the robot *Vp* multiple the collision prediction time Δt. R_2 represents the maximum magnitude of the obstacle velocity (Vo) which multiplied by the collision prediction time (Δt).

$$X = V_p \cdot \Delta t \tag{11}$$

$$R_2 = V_{0\max} \cdot \Delta t \tag{12}$$

After that, draw two tangent segments of the two circles just drawn. The area of the total of the closed shapes is the safe zone need to find:

$$\begin{aligned} S = {} & R_1^2 \cdot \arccos\left(\frac{\sqrt{X^2-(R_2-R_1)^2}}{X}\right) + \\ & +R_2^2 \cdot \left(1-\arccos\left(\frac{\sqrt{X^2-(R_2-R_1)^2}}{X}\right)\right) + \\ & +(R_1+R_2)\cdot\sqrt{X^2-(R_2-R_1)^2} \end{aligned} \tag{13}$$

After checking if the obstacle is in the safe zone, continue to check in the avoiding zone (A). Basically, the Avoiding zone is only different from the safe zone in R_2, which means the Avoiding zone will have the second circle whose radius will be R_2'. The R_2' is the value of the instantaneous velocity of the Vo obstruction multiplied by the time of collision prediction Δt.

$$R_2' = V_0 \cdot \Delta t \tag{14}$$

$$\begin{aligned} A = {} & R_1^2 \cdot \arccos\left(\frac{\sqrt{X^2-(R_2'-R_1)^2}}{X}\right) + \\ & +R_2^2 \cdot \left(1-\arccos\left(\frac{\sqrt{X^2-(R_2'-R_1)^2}}{X}\right)\right) + \\ & +(R_1+R_2')\cdot\sqrt{X^2-(R_2'-R_1)^2} \end{aligned} \tag{15}$$

A. *Tracking Fuzzy Logic Controller (TFLC)*

TFLC is proposed to move the robot to its target smoothly. The inputs of TFLC are the angle between the robot and the target (error angle), and the distance from the robot to its target. The outputs of TFLC are the velocities of the left and right wheels. Besides, each input of TFLC is performed with seven membership functions as illustrated in Fig. 4(a) and Fig. 4(b). The linguistic variables used for the angle between the robot and the target (error angle) are: N: Negative, SN: Small Negative, NNZ: Near Negative Zero, Z: Zero, NPZ: Near Positive Zero, SP: Small Positive, and P: Positive. The variables used for input distance are: Z: Zero, NZ: Near Zero, N: Near, M: Medium, NF: Near Far, F: Far, and VF: Very Far.

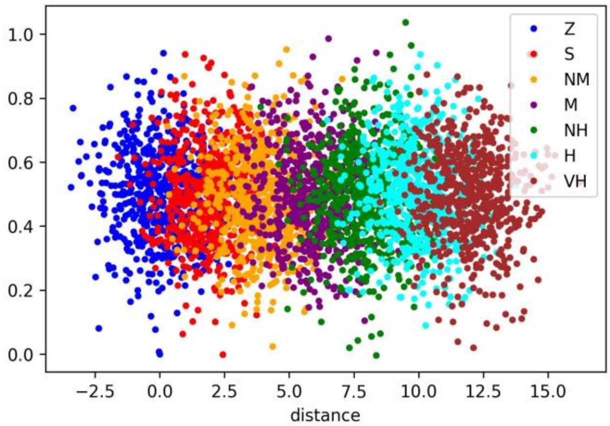
LE. Angle= {*N, SN, NNZ, Z, NPZ, SP, P*}

LDistance= {*Z, NZ, N, M, NF, F, VF*}

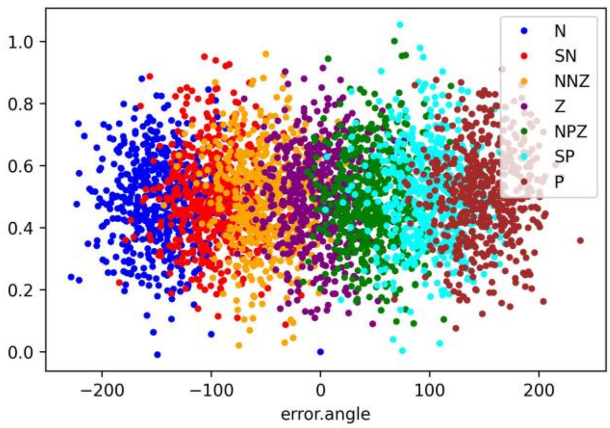
TFLC is implemented with seven membership functions for each output (Left Velocity LV and Right

Velocity RV of the wheel) as illustrated in Fig. 4(c). The variables used for the fuzzy rules of TFLC are: Z: Zero, S: Slow, NM: Near Medium, M: Medium, NH: Near High, H: High, and VH: Very High.

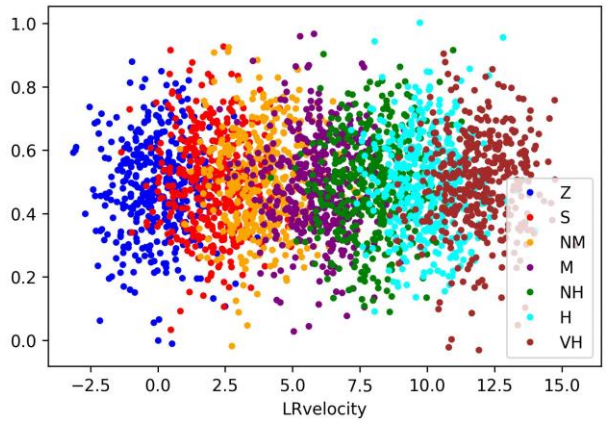
LLV= LRV= {*Z, S, NM, M, NH, H, VH*}



(a)



(b)



(c)

Fig. 4. Membership functions for (a) the distance, (b) the error in angle, and (c) left and right velocities.

The velocities (LV and RV) of the leading wheels of the robots are calculated using the defuzzification step. Note that in this work the robot have adopted the weighted average defuzzification technique. Further, if an obstacle is detected around (from 0° to 360°) the mobile robot, the control algorithm switches over to the OAFLC.

B. Obstacle Avoidance Fuzzy Logic Controller (OAFLC)

OAFLC is proposed to generate a control signal (LV and LR) so as to avoid obstacles in unknown dynamic environments. The input of OAFLC will be taken from the Lidar sensor and then processed by the data to give 3 input coefficients which are the velocity value of the obstacle (V_o), the relative position of the obstacle with the robot (P_o) and deflection angle of the velocity vector of the obstacle with the velocity vector of the robot (ϑ_o), as illustrated in Fig. 5.

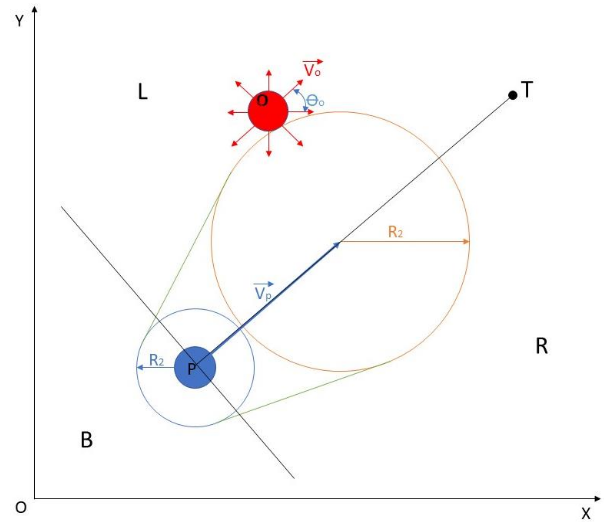
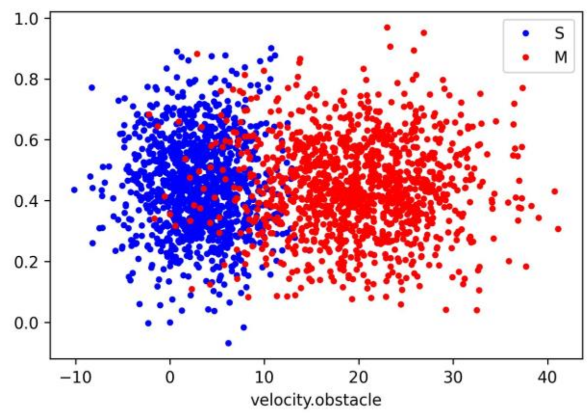
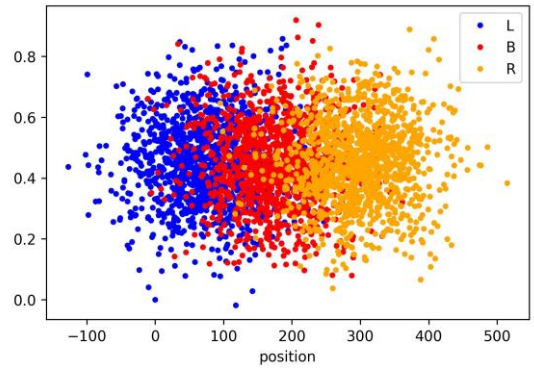


Fig. 5. Simulation for the input variables of both robot and obstacle.

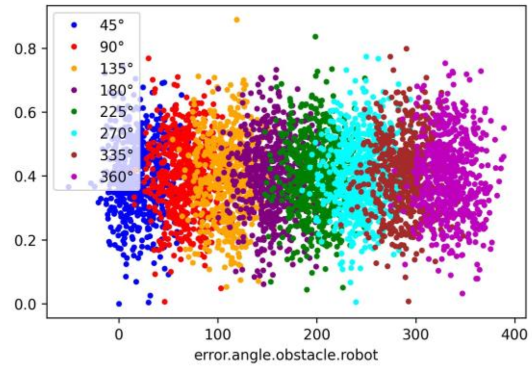
These values are generated through the processing of signals from the Lidar sensor. The output of OAFLC is the velocities of the left and right wheels (Fig. 6(a)). The value of the obstacle velocity is divided into 2 components (S, M) which are static and mobile. The relative position of the obstacle is divided into 3 members (L, R, B): Left, Right and Behind, shown in Fig. 6(b). The deflection angle of the obstacle velocity vector and the robot velocity vector are divided into 8 members corresponding to the rotation angle of 45° around the center of the obstacle (45°, 90°, 135°, 180°, 225°, 270°, 315°, 360°), as shown in Fig. 6 (c).



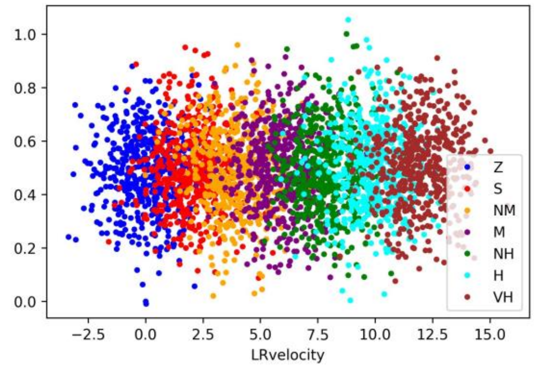
(a)



(b)



(c)



(d)

Fig. 6. Outputs for (a) The velocity of obstacle, (b) relative position of obstacle with the robot (c) error in angle of obstacle to robot, (d) the left and right velocity of OAFLC.

$$V_0 = \{S, M\}$$

$$P_0 = \{L, B, R\}$$

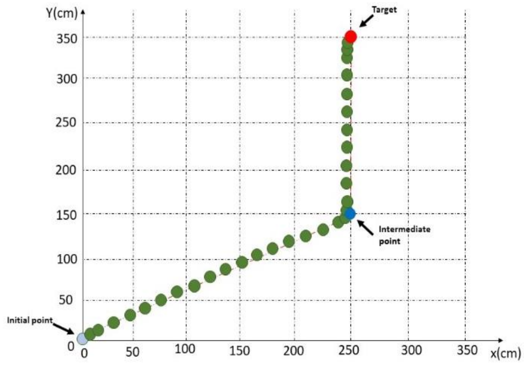
$$\vartheta_0 = \{45, 90, 135, 180, 225, 270, 315, 360\}$$

For the outputs, OAFLC is implemented with seven membership functions for each output (left velocity LV and right velocity RV) as illustrated in Fig. 6(d). The variables which are used for the fuzzy rules of OAFLC are: Z: Zero, S: Slow, NM: Near Medium, M: Medium, NH: Near High, H: High and VH: Very High.

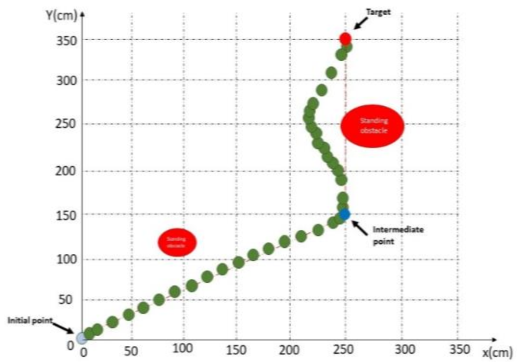
$$LLV= LRV=\{Z, S, NM, M, NH, H, VH\}$$

V. Experiments and Discussions

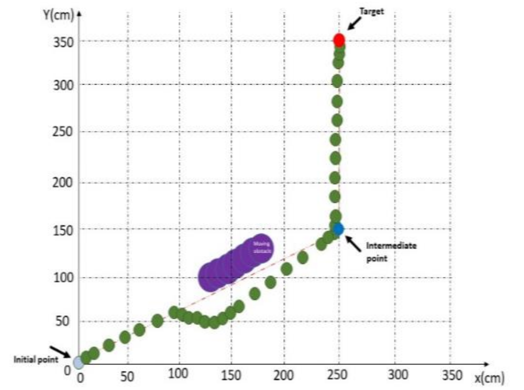
The platform of mobile robot used is our own research and design to suit the purpose. Robots are used in a wheel-design platform system that allows them to navigate an unreasonably flat environment. Using the Lidar RPLIDAR A2 2D 360° sensor, it is able to detect the depth of the right scanned object and can rotate 360 degrees to rotate the center to get results from different directions. In addition, it uses the C++ interface as a programming language. In the past experiment, the recommended sizes for using obstacles are: V_{0max} = 0.3 m /s, R_0 = 0.8 m, γ = 0.2 m, Rp = 0.2 m, Δt = 2 s and T_w = 1600 (pulses/round). Based on the experiments of research on this issue, we have divided it into three scenarios. In the first scenario, the robot will move in an environment without obstacles and has not been built before in the programming system, as shown in Fig. 7(a). In the second case, the robot system is moved in the same environment.



(a)



(b)



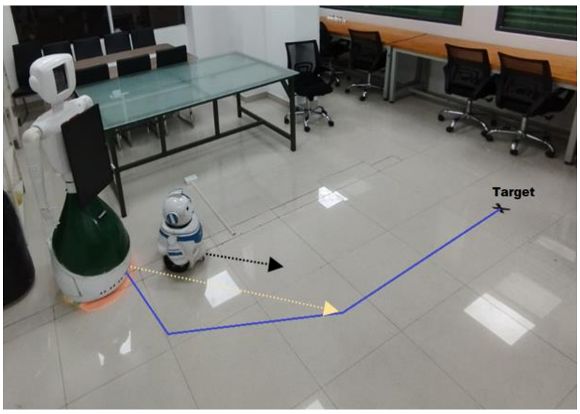
(c)

Fig. 7. The simulating trajectories of the robots in three scenario (a) First scenario (environment without obstacles). (b) Second scenario (environment has static obstacles). (c) Third scenario (workspace with mobile obstacle).

But static obstacle suddenly appears on the trajectory of the path of the robot. When scanning in the safe zone where obstacles are present, the robot will continue to process in the area to be avoided in order to provide the best obstacle avoidance plan, shown in Fig. 7(b). In the third case, On the trajectory, there is a mobile obstacle, and the robot relies on the input data to provide an appropriate evasion direction, as shown in Fig. 7(c). In all three scenarios to move to the target, the robot moves through the intermediate target to add an objective part in the research process. In Fig. 7(a), the simulation of the simple path finder algorithm to navigate to the target was designed by the fuzzy logic controller. Note that this data is generated using the dead calculation algorithm.

(a)

(b)



(c)

Fig. 8. The movements of the robots in three scenario (a) First scenario (environment without obstacles). (b) Second scenario (environment has static obstacles). (c) Third scenario (workspace with mobile obstacle).

When the speed of the mobile robot is slow, the algorithm provides precise positioning and orientation for the robot. The robot moves from the initial point to the intermediate point, and then ends at the target point. In the second scenario, two obstacles suddenly appear without prior notice. As shown in Fig. 8(b), the first obstacle appears in a safe zone. But after the robot continues processing in avoiding zone, obstacle is outside of avoiding zone so it does not need to change the trajectory of the path. Therefore, the control algorithm returns to TFLC algorithm. When the robot continues to move from intermediate point to target, there is second obstacle. But now, the obstacle appears in both the safe zone and the avoiding zone so the moving algorithm is changed from TFLC to OAFLC to avoid obstacles. As being seen from Figs. 7(a) and 7(b), the movement of the robot in the first scenario is much different from the second. This difference comes from the second obstacle that the robot scanned and need to change the moving trajectory to be able to avoid obstacle. Fig. 8(b) illustrates the actual results of the second scenario, where two obstacles were added in the unknown environment. Fig. 7(c) also shows us that the robot has overcome both obstacles.

The experiment created in Fig. 8(a) creates the path in Fig. 7(a). And the experiment in Fig. 8(b) gives the path in Fig. 7(b). In Fig. 8(c), show the mobile obstacle with another robot moving directly towards the robot with a velocity $V_0 = 0.15$ m/s. With the proposed algorithm and the input parameters are set before, the robot will judge the collision in the future $\Delta t = 2$ s. The robot then devised a plan to avoid obstacles. Looking at Fig. 7(c), we see the solution of the robot to avoid obstacles relatively simple and effective.

VI. CONCLUSIONS

This paper presents a proposal to use fuzzy logic to control mobile robots and avoid both static and mobile obstacles. The goal is to demonstrate the mobility of self-propelled robots in flat and unknown environments, such as warehouses or libraries, where traditional methods of detecting colored lines are not applicable. The use of self-propelled robots offers great potential for flat environments that are not overly complex and where obstacles may appear unexpectedly. The proposed fuzzy logic algorithms allow the robot to determine its own behavior and provide optimal solutions for obstacle avoidance. The research results indicate that the robot is capable of effectively and smoothly navigating both structured and unstructured environments using fuzzy algorithms. In the future, research work can focus on further improving the efficiency and reliability of the fuzzy logic algorithms, as well as exploring their potential applications in more complex and dynamic environments. Additionally, it would be beneficial to investigate the integration of additional sensors or systems, such as vision-based obstacle detection, to enhance the performance and capabilities of the self-propelled robots.

Conflict of Interest

The authors declare no conflict of interest.

Author Contributions

P.M.T. is the first author - methodology, wring original draft, visualization, validation, configured. N.D.T. and T.Q.H. conducted the research. N.T.T is correspondent, project administration, review and editing, methodology. All authors had approved the final version. All authors had approved the final version

Acknowledgment

This research is funded by University of Economics Ho Chi Minh City—UEH, Vietnam.

References

[1] N. T. Thinh and N. D. X. Hai, "Telemedicine mobile robot-robots to assist in remote medical," *International Journal of Mechanical Engineering and Robotics Research*, vol. 10, no. 6, pp. 337–342, 2021.

[2] K. Kittiwad *et al.*, "Design and development of service robot Based human-robot interaction (HRI)," in *Proc. 2017 International Conference on System Science and Engineering (ICSSE)*, 2017.

[3] N. D. X. Hai, L. H. T. Nam, and N. T. Thinh, "Remote healthcare for the elderly, patients by tele-presence robot," in *Proc. 2019 International Conference on System Science and Engineering (ICSSE)*, 2019.

[4] N. T. Thinh, T. P. Tho, and N. D. X. Hai, "Adaptive fuzzy control for autonomous robot under complex environment," in *Proc. International Journal of Mechanical Engineering and Robotics Research*, 2021, vol. 10, no. 5, pp. 216–223.

[5] L. A. Zadeh, "Fuzzy sets," *Information and Control*, vol. 8, no. 3, pp. 338–353, 1965.

[6] L. A. Zadeh, "Outline of a new approach to the analysis of complex systems and decision processes," *IEEE Transactions on Systems, Man, and Cybernetics*, vol. 1, pp. 28–44, 1973.

[7] D. Driankov, H. Hellendoorn, M. Reinfrank, *An Introduction to Fuzzy Control*, Springer Science & Business Media, 2013.

[8] B. Kosko, *Neural Network and Fuzzy Systems: A Dynamical Systems Approach to Machine Intelligence*, Prentice Hall International, Inc, 2012.

[9] C. Ricardo, E. F. Camacho, and D. Patino, "A neural network based feedforward adaptive controller for robots," *IEEE Transactions on Systems, Man, and Cybernetics*, vol. 25, no. 9, 1995, pp. 1281–1288.

[10] F. W. Lewis, S. Jagannathan, and A. Yesildirak, *Neural Network Control of Robot Manipulators and Non-linear Systems*, CRC press, 1998.

[11] P. P. Thanh *et al.*, "Design of adaptive fuzzy sliding mode controller for mobile robot," *International Journal of Mechanical Engineering and Robotics Research*, vol. 10, no. 2, pp. 54–59, 2021.

[12] L. P. Gia and N. T. Thinh, "Hybrid genetic algorithm based smooth global-path planning for a mobile robot," *Mechanics Based Design of Structures and Machines*, vol. 51, no. 3, pp. 1758–1774, 2023.

[13] L. P. Gia and N. T. Thinh, "Real-time hybrid navigation system-based path planning and obstacle avoidance for mobile robots," *Applied Sciences*, vol. 10, no. 10, 3355. 2020.

[14] P. S. K. Pradhan, D. R. Parhi, and A. K. Panda, "Fuzzy logic techniques for navigation of several mobile robots," *Applied Soft Computing*, vol. 9, no. 1, pp. 290–304, 2009.

[15] O. Giuseppe, A. D. Luca, and M. Vendittelli, "WMR control via dynamic feedback linearization: design, implementation, and experimental validation," *IEEE Transactions on Control Systems Technology*, vol. 10, no. 6, pp. 835–852, 2002.

[16] T. H. Li, S. J. Chang, and W. Tong, "Fuzzy target tracking control of autonomous mobile robots by using infrared sensors," *IEEE Transactions on Fuzzy Systems*, vol. 12, no. 4, pp. 491–501, 2004.

[17] R. Razif *et al.*, "Differential drive Wheeled Mobile Robot (WMR) control using fuzzy logic techniques," in *Proc. 2010 Fourth Asia International Conference on Mathematical/Analytical Modelling and Computer Simulation*, 2010.

[18] V. Raudonis and R. Maskeliunas, "Trajectory based fuzzy controller for indoor navigation," in *Proc. 2011 IEEE 12th International Symposium on computational Intelligence and Informatics (CINTI)*, IEEE, 2011.