Sensors and Materials, Vol. 35, No. 3 (2023) 781–794
MYU Tokyo

S & M 3210

Navigation Control of Ackermann Steering Robot Using Fuzzy Logic Controller

Cheng-Jian Lin,[1*] Ming-Yu Chang,[1] Kuang-Hui Tang,[2,3] and Chuan-Kuei Huang[3]

[1]Department of Computer Science & Information Engineering, National Chin-Yi University of Technology, Taichung 411, Taiwan, ROC
[2]Department of Electronic Engineering, National Chin-Yi University of Technology, Taichung 411, Taiwan, ROC
[3]Department of Industrial Education and Technology, National Changhua University of Education, Changhua 500, Taiwan, ROC

(Received September 12, 2022; accepted November 24, 2022)

Keywords: Ackermann steering robot, LiDAR sensor, fuzzy logic controller, navigation control, collision

In this paper, a navigation control method is proposed for an Ackermann steering robot. In the proposed method, light detection and ranging (LiDAR) sensors are used to obtain the distance between an Ackermann steering robot and objects in an unknown environment. In accordance with the distances obtained by the LiDAR sensors, the navigation control system uses a behavior manager to switch between two types of behavior control, namely, toward-goal behavior control and wall-following behavior control (WFBC). If a wall or an obstacle is detected in the current path toward the target position, the behavior manager adopts WFBC to avoid the obstacle. To achieve WFBC, a fuzzy logic controller with three subfuzzy logic controllers—namely, a straight-based fuzzy logic controller, a right-based fuzzy logic controller, and a left-based fuzzy logic controller—is adopted. Switching between these three subcontrollers is achieved in accordance with the distance and angle between the robot and a wall (or an obstacle). The input signal of the proposed fuzzy logic controller is the distance between the robot and wall (or obstacle), which is determined by a LiDAR sensor at different angles, and the output of this controller is the steering angle of the Ackermann steering robot, which can move along a wall and avoid collisions with walls (or obstacles) in an environment. Experimental results indicated that the proposed fuzzy logic controller successfully implemented navigation control in two unknown environments.

1. Introduction

Because of their flexible and scalable characteristics, mobile robots have been widely used in many areas, such as industry,[1,2] hospitals,[3] and logistics.[4,5] Therefore, navigation control for mobile robots is a crucial research topic. Navigation control can be divided into two types: navigation in a known environment and navigation in an unknown environment. Navigation in a known environment is achieved through global path planning based on a map,[6,7] whereas navigation in an unknown environment is achieved by avoiding obstacles autonomously and

[*]Corresponding author: e-mail: cjlin@ncut.edu.tw
https://doi.org/10.18494/SAM4120

ISSN 0914-4935

moving toward a target. In both types of navigation control, crucial and dynamic obstacle avoidance functions prevent a robot from colliding with obstacles. The local planning method(8) and sensors are used to detect obstacles to achieve this dynamic avoidance of obstacles.

Common obstacle avoidance methods for robots are the artificial potential field method,(9) neural-network-based methods,(10) and evolutionary algorithms.(11) The artificial potential field method(9) involves considering a space as a virtual artificial potential field. The robot is considered to be attracted to and repulsed by the target and obstacles, respectively, and it moves along the direction of the resultant force of attraction and repulsion. This method is simple and easy to implement; however, it can fall into local solutions relatively easily. Neural networks(10) are machine-learning structures that mimic biological neural networks and are composed of many neurons. In neural-network-based obstacle avoidance, sensors are used to collect information regarding possible obstacles in an environment, and a model is then constructed through learning to guide a robot to avoid obstacles. Evolutionary algorithms(11) are search algorithms based on the concepts of elimination and evolution in nature. These algorithms have been successfully used to solve problems and optimize solutions. In many studies,(12–14) evolutionary algorithms have been used to achieve dynamic obstacle avoidance for robots. The optimal path can be generated through multiple iterations of inheritance, mutation, natural selection, and hybridization. These iterations prevent evolutionary algorithms from falling into locally optimal solutions and require a relatively long computation time. These three types of method require a relatively long computation time to achieve a high performance. To reduce the computation time, some scholars have used fuzzy logic control.

Fuzzy theory has been widely used in various uncertain, complex, and nonlinear applications and in automation control,(15) data classification,(16) and image vision.(17) This theory(18–20) was proposed by Zadeh in 1965. Fuzzy logic is a reasoning method similar to that of the human mind and is based on the observation that people make decisions on the basis of imprecise and nonnumerical information that is designed to replace Boolean logic with degree values. A fuzzy set is used for mathematically representing vague and imprecise information. Because fuzzy theory utilizes language rules to obtain automated control strategies, the knowledge and experience of experts or operators can be leveraged. Fuzzy logic is a universal approximation method in which the output changes in accordance with the input sensor data. The most well-known fuzzy logic is the Mamdani rule,(21) according to which all the input values are first fuzzified into fuzzy membership functions, the fuzzy output function is then calculated on the basis of all the applicable rules in the rule base, and the fuzzy output function is subsequently defuzzified to obtain the output value.

The common mobile robot chassis can be divided into three categories: differential drive systems,(22) omnidirectional chassis,(23) and Ackermann steering systems.(24) A differential drive system has a simple structure and low cost and can move in small spaces, such as indoor areas. Most mobile robots have this chassis structure. However, because of the differential structure of differential drive systems, each of their wheels must contain a drive motor. Therefore, the efficiency of differential drive systems is lower than that of other types of chassis. Omnidirectional chassis can be divided into two types: rollers and mecanum or spherical wheels. Because the wheels of omnidirectional chassis contain horizontal or oblique rollers, in contrast to a differential drive system, these chassis can move horizontally. Omnidirectional chassis are suitable for application in indoor or flat areas. The Ackermann steering architecture is mostly used in vehicles. This architecture has low flexibility and is usually used in outdoor environments. Mobile robots contain many sensors, such as vision systems, ultrasonic sensors, and light detection and ranging (LiDAR) sensors, which provide input signals for vehicle control algorithms.

In this paper, a navigation control method is proposed that enables an Ackermann steering robot to avoid obstacles autonomously and reach its target effectively in an unknown environment. The proposed navigation control method involves two types of behavior control, namely, toward-goal behavior control (TGBC) and wall-following behavior control (WFBC). The robot adopts the TGBC mode to navigate toward a target; however, when it detects an obstacle or a wall, it adopts the WFBC mode for dynamic obstacle avoidance. The mobile robot moves forward at a constant speed. The proposed fuzzy logic controller contains three subfuzzy logic controllers, namely, the straight-based fuzzy logic controller (SFLC), right-based fuzzy logic controller (RFLC), and left-based fuzzy logic controller (LFLC). The proposed fuzzy logic controller switches between the subfuzzy logic controllers depending on the situation. In summary, the contributions of this study are as follows:

1. Navigation control is realized for an Ackermann steering robot in an unknown environment.
2. The proposed navigation control method involves the TGBC and WFBC modes.
3. The proposed fuzzy logic controller for WFBC contains three subfuzzy logic controllers, namely, the SFLC, RFLC, and LFLC, to achieve dynamic obstacle avoidance.

The rest of this paper is organized as follows. Section 2 provides a detailed introduction of the architecture of the Ackermann steering robot and behavior control methods. Section 3 presents the experimental results obtained using the developed fuzzy logic controller. Section 4 provides the conclusions of this study and recommendations for future research.

2. Methods

This section describes the architecture of the Ackermann steering robot and the proposed navigation control method, which involves TGBC and WFBC.

2.1 Architecture of Ackermann steering robot

The Ackermann steering structure solves the problem of different steering angles caused by the different steering radii of the left and right wheels during steering. According to Ackermann's steering geometry, when a robot turns along a curve, the steering angle of its inner wheel can be increased to approximately 2–4° greater than that of its outer wheel by using the equal crank of the four-link structure. The center of the four-wheel paths can be approximately intersected with the extension line of the rear axle to obtain the steering center so that smooth turning can be achieved. Ackermann steering is associated with high-efficiency movement and a high payload. However, because of their large size, Ackermann steering structures cannot be used in small spaces and are mainly used in vehicles.

A self-manufactured Ackermann steering robot (Fig. 1) was used in this study. In the front axle of this robot, a four-link structure is employed as the steering axis, whereas in the rear axle, a DC motor is used as the driving wheel. The core controller comprises an Nvidia Jetson AGX Xavier module (CA, USA) to execute the robot's operating system. This controller communicates with the motor controller through controller area network bus communication technology to control the steering angle of the front axle and the wheel speed of the rear axle. The dimensions of the designed Ackermann steering robot are presented in Fig. 2. The maximum steering angle of the front axle is 25°, the wheels have a diameter of 21 cm, and the front and rear wheelbase is 86 cm. The aforementioned robot is designed for high-payload usage and therefore has a long wheelbase.

The Ackermann steering robot used in this study was equipped with a Velodyne VLP-16 LiDAR sensor. This sensor collects data in 16 channels and has a sensing angle of 360°, a vertical angle of 30°, and an effective range of 1–150 m.

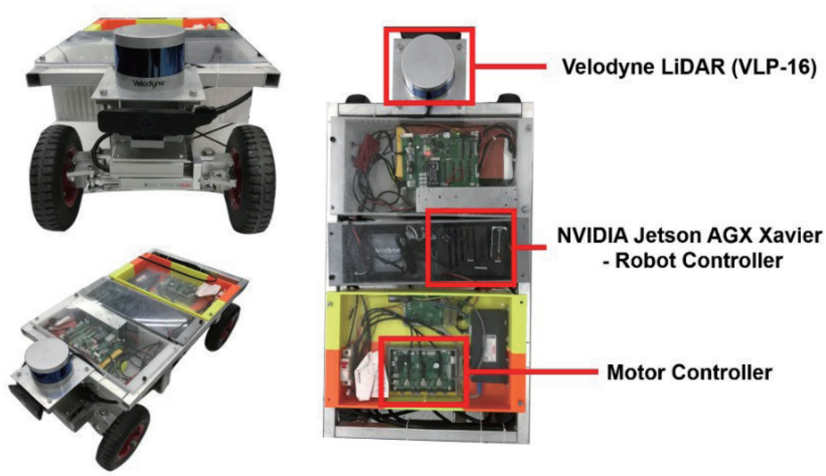


Fig. 1. (Color online) Image of the designed Ackermann steering robot.

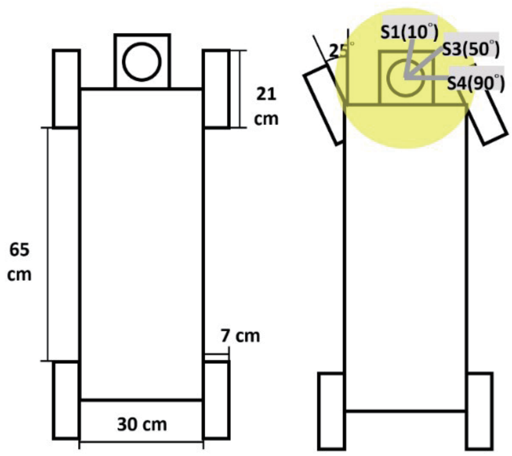


Fig. 2. (Color online) Dimensions of the designed Ackermann steering robot.

2.2 Navigation control

In the proposed method, navigation control is achieved through TGBC and WFBC. First, the obstacle sensing area is divided into four parts, namely, R0, R1, R2, and R3 (Fig. 3). If an obstacle is judged to be between R0 and R2, the WFBC mode is activated. If an obstacle is judged to be at R3 or if no obstacle is detected, the TGBC mode is activated, and the judgment is repeated until the robot reaches the target position.

The switching between TGBC and WFBC is illustrated in Fig. 4. When only considering whether an obstacle is in the detection area for switching between the two controller modes, the mobile robot bypasses the obstacle through WFBC. When the navigation control system switches back to TGBC, the mobile robot moves toward the obstacle again, which leads to the problem of the robot falling into a dead zone. Therefore, in the proposed method, the shortest distance between the mobile robot and the goal is calculated. The WFBC mode continues to be active until the shortest distance between the mobile robot and the goal is achieved. The achievement of this distance indicates that the dead zone has been successfully escaped from. At this point, the TGBC mode is activated.

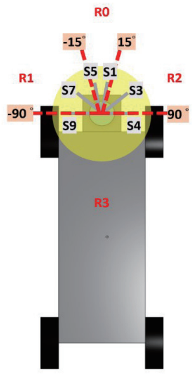


Fig. 3. (Color online) Obstacle sensing area.

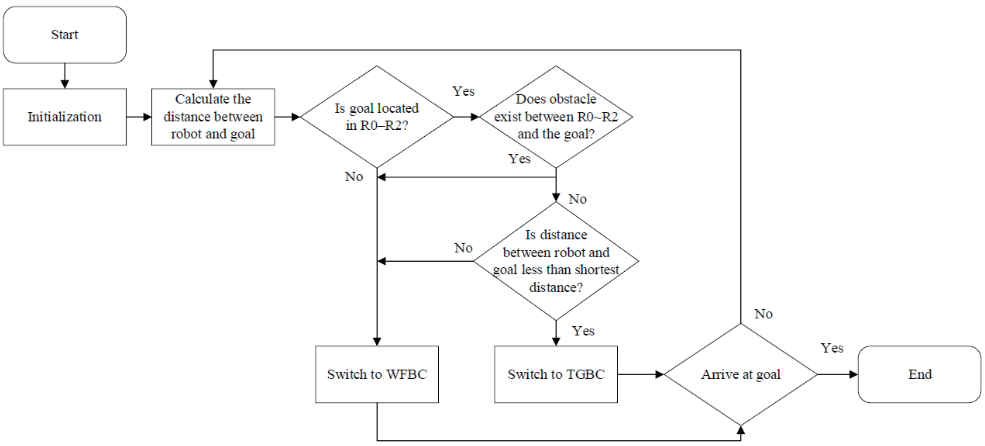


Fig. 4. Flowchart of the switching between WFBC and TGBC in the proposed method.

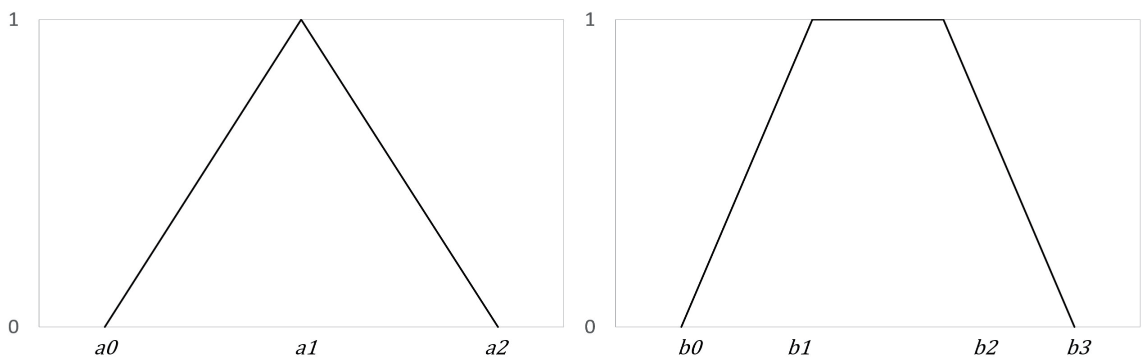
2.2.1 TGBC

If an obstacle is judged to be in R3 or if no obstacle is detected, the TGBC mode is activated. The judgment is repeated until the mobile robot reaches the target position.

2.2.2 WFBC

In general, a fuzzy control system can be divided into four parts, namely, a fuzzification system, a knowledge base, an inference engine, and a defuzzification system. The fuzzification system transforms the crisp values of an input signal into fuzzy sets and assigns the input signal to the operation of fuzzy sets with a certain degree of membership. The degree of membership function is in the interval [0, 1], where 0 indicates that the function does not belong to the fuzzy set and 1 indicates that the function completely belongs to the fuzzy set. Triangular and trapezoidal membership functions are the most commonly used membership functions (Fig. 5). The membership function of fuzzy set A is denoted μ_A, and $\mu_A(x)$ represents the membership degree of an input variable x mapped to fuzzy set A. The knowledge base (i.e., *IF–THEN* rules) is used to obtain output membership functions from input linguistic variables. The knowledge base comprises expert experience and engineering knowledge. The output of a fuzzy control system is a crisp value obtained from a fuzzy conclusion. In the present study, the center of area method was used to perform defuzzification. This method is expressed as

$$y = \frac{\sum_i \mu_A(y_i) * y_i}{\sum_i \mu_A(y_i)}, \tag{1}$$



$$\mu_A = \begin{cases} 0 & for\ x \le a0 \\ \dfrac{x - a0}{a1 - a0} & for\ a0 \le x \le a1 \\ \dfrac{a2 - x}{a2 - a1} & for\ a1 \le x \le a2 \\ 0 & for\ a2 \le x \end{cases}$$

(a)

$$\mu_A = \begin{cases} 0 & for\ x \le b0 \\ \dfrac{x - b0}{b1 - b0} & for\ b0 \le x \le b1 \\ 1 & for\ b1 \le x \le b2 \\ \dfrac{b3 - x}{b3 - b2} & for\ b2 \le x \le b3 \\ 0 & for\ b3 \le x \end{cases}$$

(b)

Fig. 5. (a) Triangular and (b) trapezoidal membership functions.

where y represents the output of the fuzzy control system, y_i is the output of the consequent part of the ith fuzzy rule, and $\mu A(y_i)$ is the membership function of fuzzy set A calculated for the input variables.

In this paper, an effective LiDAR-based method is proposed for achieving WFBC (Fig. 6). The proposed WFBC system includes three subfuzzy logic controllers, namely, the SFLC, RFLC, and LFLC. Each subfuzzy controller controls a certain movement of the Ackermann steering robot and uses a switch to change the behavior of the mobile robot. The SFLC is used to control the movement of the robot in the target direction and to maintain a fixed distance between the robot and the wall (or an obstacle) by enabling the robot to turn by a small angle. The LFLC and RFLC control the left and right turning movements of the robot, respectively. When the robot turns left or right, the LFLC or RFLC subcontroller, respectively, maintains a fixed distance between the mobile robot and the wall (or obstacle). In the proposed WFBC system, the SFLC and LFLC subcontrollers have two inputs and one output each, whereas the RFLC subcontroller has one input and one output.

The mobile robot's movement must be maintained parallel to the wall during behavior control with the proposed WFBC system. That is, the input of the proposed SFLC is 50° (i.e., S3) and 90° (i.e., S4) from the LiDAR sensor (Fig. 2). A decrease in the sensing value for S3 indicates that the mobile robot is gradually approaching a wall. At this time, the mobile robot must turn away from the wall. Conversely, an increase in the sensing value for S3 indicates that the mobile robot is far from the wall and must turn toward the wall. The membership functions of S3 and S4 are depicted in Fig. 7. Figure 7 indicates that five fuzzy rules are used in the proposed SFLC. The control surface obtained in accordance with the output of the designed SFLC is shown in Fig. 8.

The left-turn behavior of the mobile robot using the LFLC depends on the angle between the mobile robot and the wall (or obstacle). If the S1 distance of the LiDAR sensor is longer than its S3 distance, the turning angle is less than 90°, and movement control with a fixed distance from

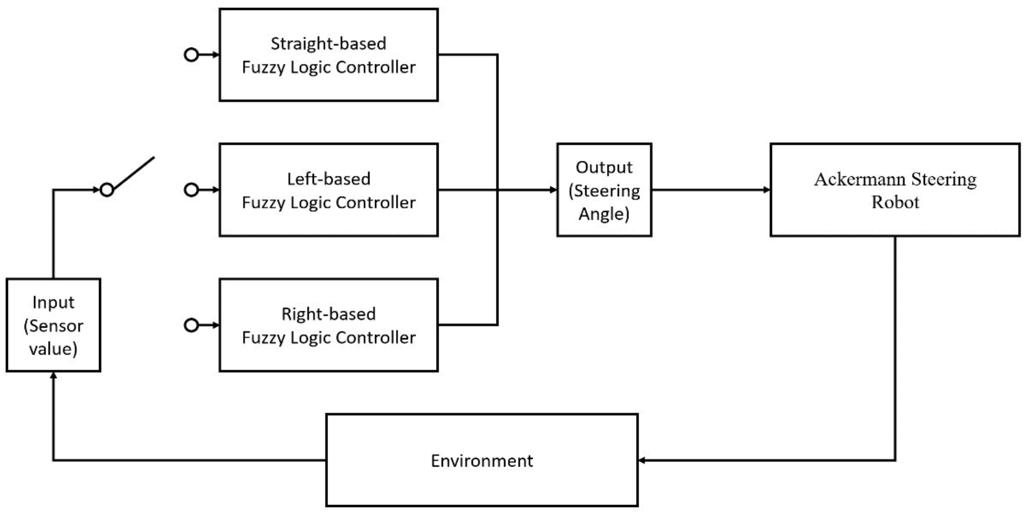
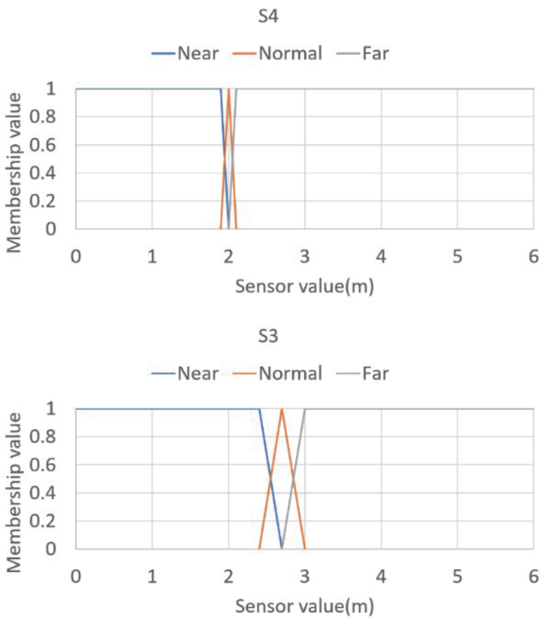


Fig. 6. Proposed WFBC system.



Straight-based Fuzzy Logic Controller		
INPUTS(AND)		Output
S4(m)	S3(m)	Angle(degree)
Any	Near	-10
Near	Normal	-5
Normal	Normal	0
Far	Normal	5
Any	Far	10

Fig. 7. (Color online) Membership functions of S3 and S4 for the SFLC as well as the five fuzzy rules of the SFLC.

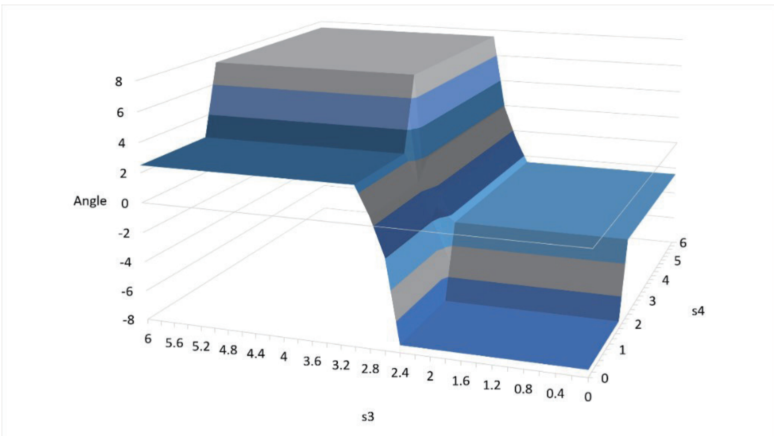
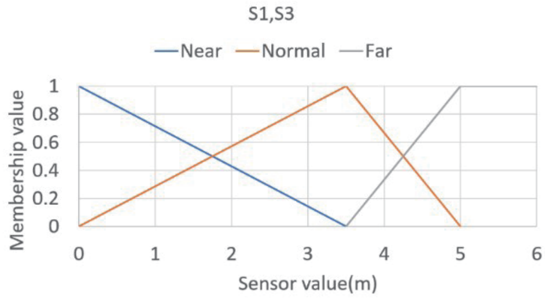


Fig. 8. (Color online) Control surface obtained using the SFLC.

the wall is adopted. By contrast, if the S3 distance is longer than the S1 distance, the turning angle is higher than 90°, and the mobile robot must turn to avoid entering the dead zone. The membership functions of S1 and S3 as well as the five fuzzy rules of the LFLC are depicted in Fig. 9. The control surface obtained in accordance with the output of the designed LFLC is depicted in Fig. 10.

The right-turn behavior of the mobile robot using the RFLC depends on the S4 signal of the LiDAR sensor. If the S4 distance of this sensor is too long, the mobile robot's movement is not maintained at a fixed distance from the wall. At this time, the robot must turn right to avoid collision with other walls (or obstacles). By contrast, if the S4 distance of the LiDAR sensor is



Left-based Fuzzy Logic Controller		
INPUTS(AND)		Output
S1(m)	S3(m)	Angle(degree)
Near	Any	-25
Normal	Near	-25
Normal	Normal	-20
Normal	Far	-15
Far	Any	-15

Fig. 9. (Color online) Membership functions of S1 and S3 for LFLC as well as the five fuzzy rules of the LFLC.

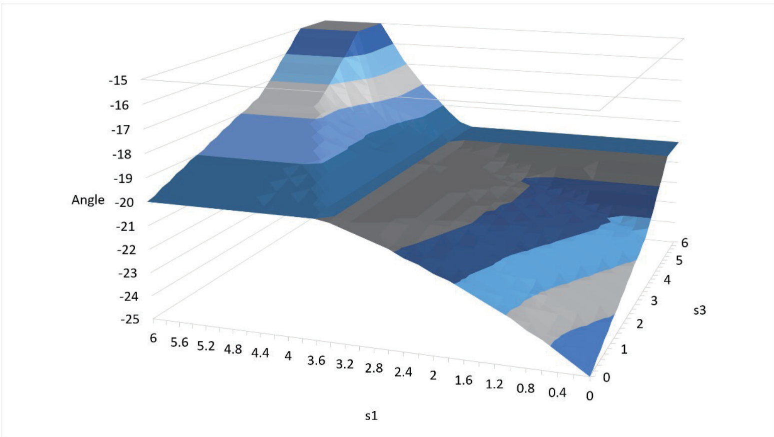


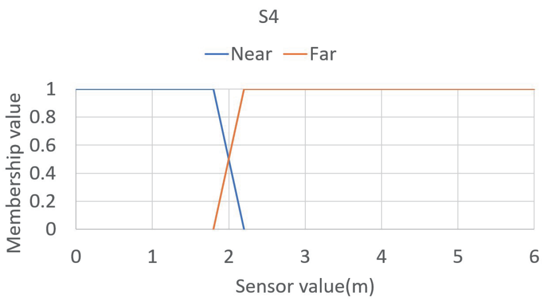
Fig. 10. (Color online) Control surface obtained using the LFLC.

too short, the mobile robot needs to turn right. At this time, a fixed distance must be maintained between the robot and the wall. The proposed RFLC has only two membership functions (i.e., Near and Far), as displayed in Fig. 11. The control curve obtained in accordance with the output of the designed RFLC is depicted in Fig. 12.

The control curve and surfaces displayed in Figs. 8, 10, and 12 indicate that the LiDAR sensing value is higher when the robot is further away from a wall or an obstacle. In this scenario, the mobile robot turns toward the wall or obstacle. A low LiDAR sensing value indicates that the mobile robot is too close to a wall or an obstacle. In this case, the mobile robot turns away from the wall or obstacle.

3. Experimental Results

Two sets of experiments were conducted in this study. In the first set of experiments, the control of the designed mobile robot when it moved along a wall (or an obstacle) in two unknown environments was examined. In the second set of experiments, the navigation control of the mobile robot was investigated in two unknown environments. The mobile robot moved at a constant speed in all the experiments.



Right-based Fuzzy Logic Controller	
INPUTS(AND)	Output
S4(m)	Angle(degree)
Near	5
Far	25

Fig. 11. (Color online) Membership function of S4 for the RFLC and the two fuzzy rules of the RFLC.

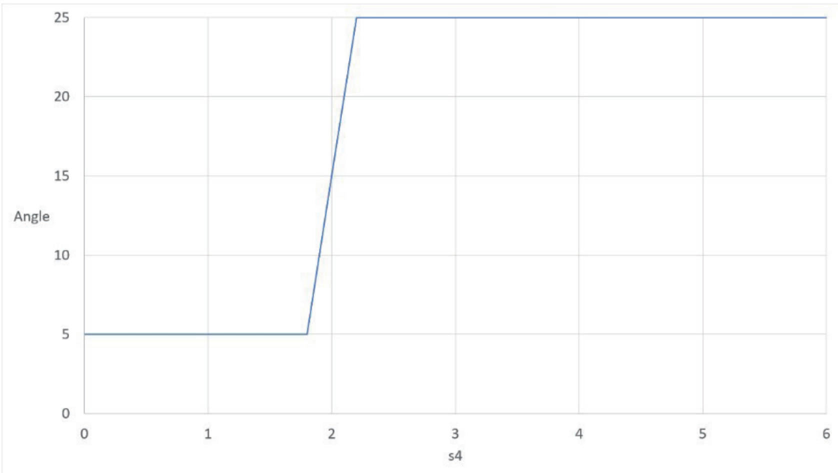


Fig. 12. (Color online) Control curve obtained using the RFLC.

3.1 Results obtained for WFBC

To verify the feasibility of the proposed WFBC method, two unknown environments were established (Figs. 13 and 14). The red, green, and blue lines in Figs. 13 and 14 represent the experimental results obtained using the designed LFLC, RFLC, and SFLC, respectively. The experimental results indicated that the mobile robot effectively switched between the three subfuzzy logic controllers and moved at an appropriate distance from the wall. However, when the angle between the obstacle and the robot was 90°, the movement of the robot was limited by the minimum turning radius of the Ackermann steering structure; therefore, the mobile robot could not come close to the obstacle. In this case, the mobile robot had to turn in advance, and the experimental results indicated that the mobile robot still moved around the obstacle without colliding with it.

3.2 Results of navigation control

To validate the proposed navigation control method, two unknown environments were established (Figs. 15 and 16). As depicted in Fig. 15, to avoid the problem of the robot falling into

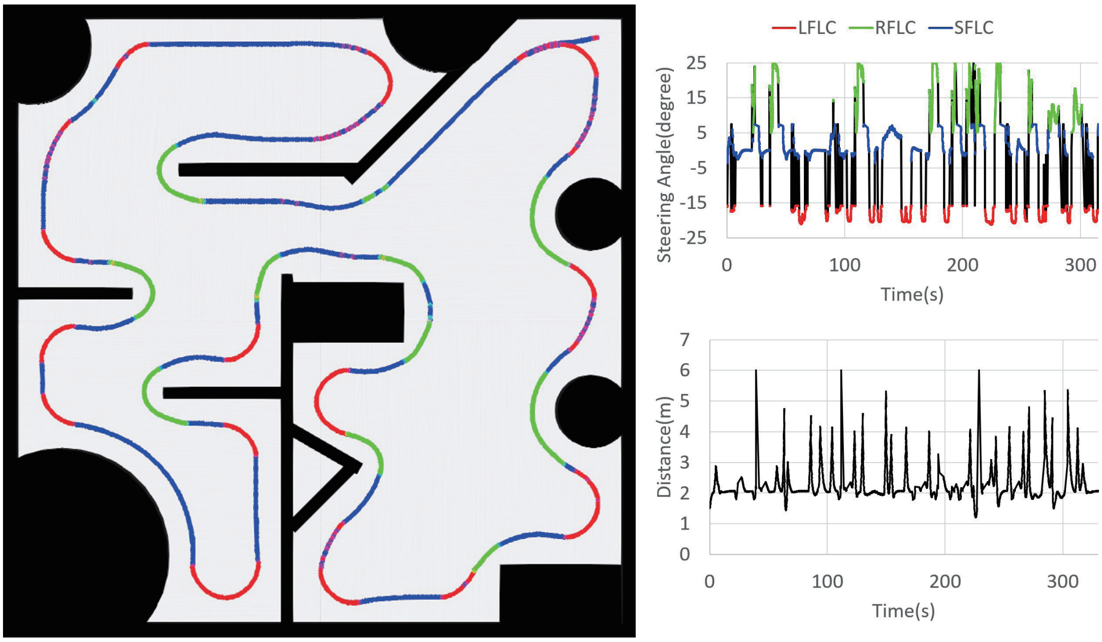


Fig. 13. (Color online) Results obtained with the proposed WFBC method in environment 1.

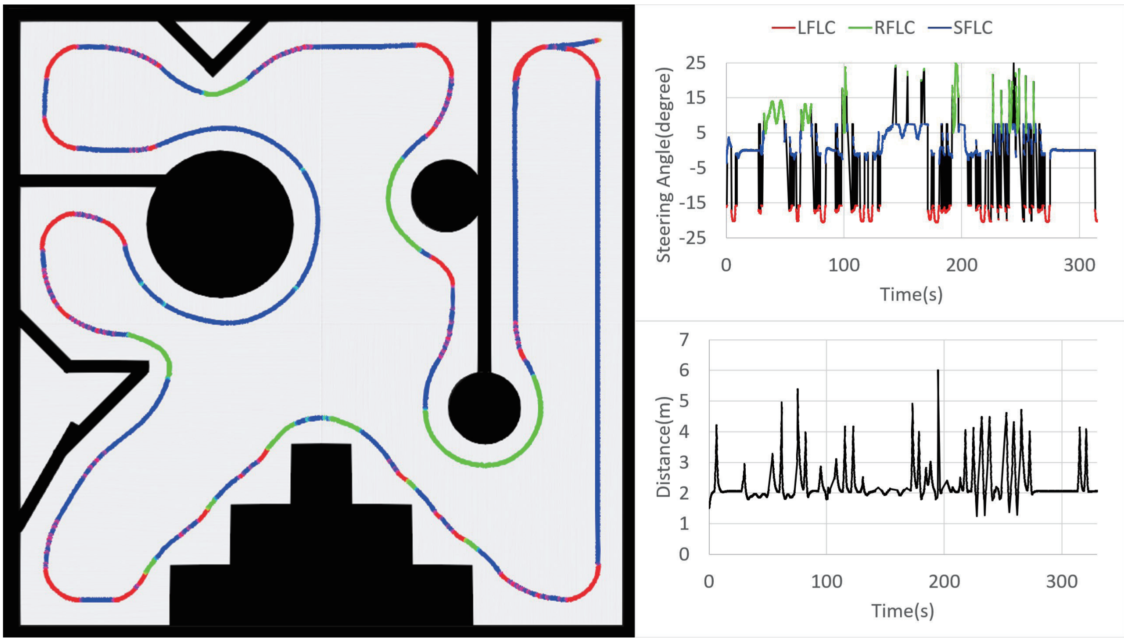


Fig. 14. (Color online) Results obtained with the proposed WFBC method in environment 2.

a dead zone, the mobile robot used the WFBC mode until the shortest distance between the robot and obstacle was achieved. Subsequently, the robot switched back to the TGBC mode. Figure 16 indicates that the robot effectively switched between the WFBC and TGBC modes to move toward the target. It used the WFBC mode (i.e., the fuzzy logic controller) to avoid obstacles.

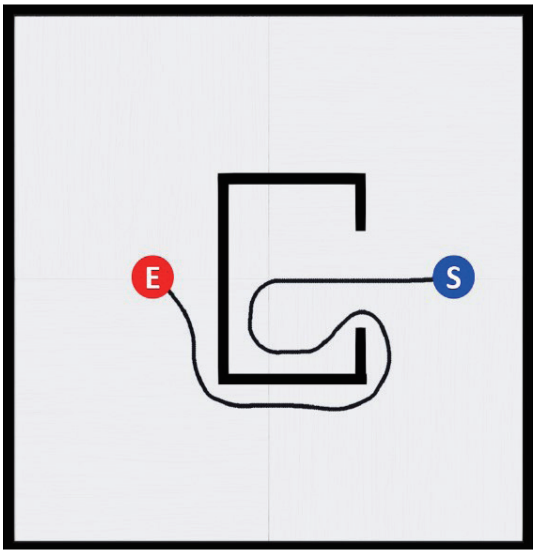


Fig. 15. (Color online) Results of navigation control in environment 1.

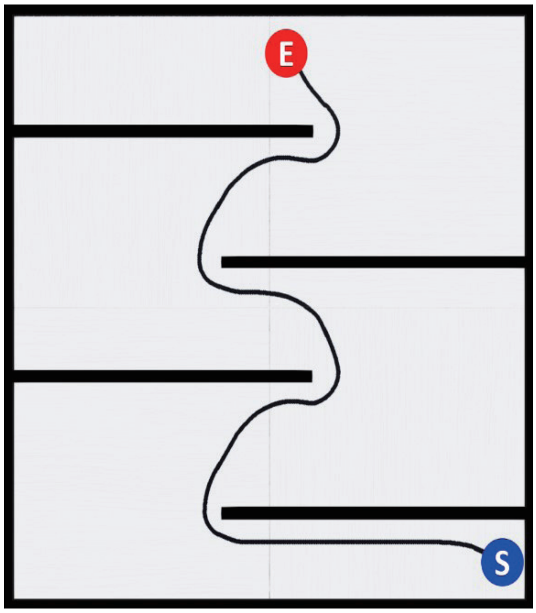


Fig. 16. (Color online) Results of navigation control in environment 2.

4. Conclusions

In this paper, a navigation control method is proposed for an Ackermann steering robot. In the proposed navigation method, a behavior manager switches between TGBC and WFBC in accordance with the distance between an obstacle and the robot, which is obtained by a LiDAR sensor. If an object or obstacle is detected in the current path of movement toward the target point, the behavior manager switches to WFBC to enable the robot to avoid the obstacle. In the proposed method, WFBC is achieved using a fuzzy logic controller that comprises three subfuzzy logic controllers: an SFLC, an RFLC, and an LFLC. Switching occurs between these three subcontrollers in accordance with the distance and angle between the robot and obstacle in the environment. The proposed navigation control method can prevent the problem of the robot falling into a dead zone. Experimental results indicated that the designed fuzzy logic controller successfully performed navigation control in two unknown environments.

The proposed fuzzy logic controller spends considerable time adjusting the positions of membership functions. Therefore, machine-learning methods will be used in future research to enable this controller to adjust the positions of membership functions automatically.

References

1 F. G. Lopez, J. Abbenseth, C. Henkel, and S. Dörr: 2017 Eur. Conf. Mobile Robots (ECMR 2017) 1–6. https://doi.org/10.1109/ECMR.2017.8098677
2 C. Hajdu, J. Hollósi, R. Krecht, Á. Ballagi, and C. R. Pozna: 2020 IEEE 3rd Int. Conf. and Workshop in Óbuda on Electrical and Power Engineering (CANDO-EPE 2020) 000155–000160. https://doi.org/10.1109/CANDO-EPE51100.2020.9337786
3 M. Takahashi, T. Suzuki, H. Shitamoto, T. Moriguchi, and K. Yoshida: Rob. Auton. Syst. **58** (2010) 889. https://doi.org/10.1016/j.robot.2010.03.010

4 M. Song and S. Xin: 2021 IEEE Int. Conf. Computer Science, Electronic Information Engineering and Intelligent Control Technology (CEI 2021) 80–83. https://doi.org/10.1109/CEI52496.2021.9574514
5 Y. Sun: Adv. Multimedia **2021** (2021). https://doi.org/10.1155/2021/4090990
6 P. E. Hart, N. J. Nilsson, and B. Raphael: IEEE Trans. Syst. Sci. Cybern. **4** (1968) 100. https://doi.org/10.1109/TSSC.1968.300136
7 Z. Jian, S. Zhang, S. Chen, Z. Nan, and N. Zheng: IEEE Robot. Autom. Lett. **6** (2021) 5349. https://doi.org/10.1109/LRA.2021.3074878
8 T. Zha, J. Wen, Y. Li, and L. Sun: 2021 6th Int. Conf. Control, Robotics and Cybernetics (CRC 2021) 80–84. https://doi.org/10.1109/CRC52766.2021.9620114
9 O. Khatib: Proc. 1985 IEEE Int. Conf. Robotics and Automation (1985) 500–505. https://doi.org/10.1109/ROBOT.1985.1087247
10 S. G. Tzafestas: 1995 Proc. IEEE Int. Symp. Industrial Electronics **1** (1995) 12–20. https://doi.org/10.1109/ISIE.1995.496471
11 Y. Hu and S. X. Yang: Proc. 2004 IEEE Int. Conf. Robotics and Automation 5 (ICRA 2004) 4350–4355. https://doi.org/10.1109/ROBOT.2004.1302402
12 H. Wei and Q. Ye: 2020 4th Int. Conf. Robotics and Automation Sciences (ICRAS 2020) 6–10. https://doi.org/10.1109/ICRAS49812.2020.9134923
13 M. Lin, J. Xiaoming, and Q. Fei: 2018 11th Int. Conf. Intelligent Computation Technology and Automation (ICICTA 2018) 327–331. https://doi.org/10.1109/ICICTA.2018.00081
14 H. Chen and M. Zhuang: 2022 14th Int. Conf. Measuring Technology and Mechatronics Automation (ICMTMA 2022) 215–222. https://doi.org/10.1109/ICMTMA54903.2022.00049
15 A. Sagdatullin: 2021 3rd Int. Conf. Control Systems, Mathematical Modeling, Automation and Energy Efficiency (SUMMA 2021) 715–718. https://doi.org/10.1109/SUMMA53307.2021.9632251
16 H. Thakkar, V. Shah, H. Yagnik, and M. Shah: Clinical eHealth **4** (2021) 12–23. https://doi.org/10.1016/j.ceh.2020.11.001
17 F. Faizah, A. Triwiyatno, and R. R. Isnanto: 2021 IEEE Int. Conf. Communication, Networks and Satellite (COMNETSAT 2021) 57–63. https://doi.org/10.1109/COMNETSAT53002.2021.9530815
18 L. A. Zadeh: Inf. Control. **8** (1965) 338. https://doi.org/10.1016/S0019-9958(65)90241-X
19 U. Farooq, A. Khalid, M. Amar, A. Habiba, S. Shafique, and R. Noor: 2010 2nd Int. Conf. Signal Processing Systems **2** (2010) 740–746. https://doi.org/10.1109/ICSPS.2010.5555781
20 Fahmizal and C. Kuo: 2013 Int. Conf. Fuzzy Theory and Its Applications (iFUZZY 2013) 7–12. https://doi.org/10.1109/iFuzzy.2013.6825401
21 E. H. Mamdani: Proc. Inst. Electr. Eng. **121** (1974) 1585–1588. https://doi.org/10.1049/PIEE.1974.0328
22 G. Campion, G. Bastin, and B. Dandrea-Novel: IEEE Robot. Autom. **12** (1996) 47. https://doi.org/10.1109/70.481750
23 S. Fujisawa, K. Ohkubo, T. Yoshida, N. Satonaka, Y. Shidama, and H. Yamaura: Proc. 36th IEEE Conf. Decision and Control **4** (1997) 3654–3656. https://doi.org/10.1109/CDC.1997.652423
24 J. Hrbáček, T. Ripel, and J. Krejsa: Proc. 14th Int. Power Electronics and Motion Control Conf. (EPE-PEMC 2010) 46–51. https://doi.org/10.1109/EPEPEMC.2010.5606853

About the Authors

Cheng-Jian Lin received his B.S. degree in electrical engineering from Ta Tung Institute of Technology, Taipei, Taiwan, R.O.C., in 1986 and his M.S. and Ph.D. degrees in electrical and control engineering from National Chiao Tung University, Taiwan, R.O.C., in 1991 and 1996, respectively. Currently, he is a chair professor of the Computer Science and Information Engineering Department, National Chin-Yi University of Technology, Taichung, Taiwan, R.O.C., and dean of Intelligence College, National Taichung University of Science and Technology, Taichung, Taiwan, R.O.C. His current research interests are machine learning, pattern recognition, intelligent control, image processing, intelligent manufacturing, and evolutionary robots.
(cjlin@ncut.edu.tw)

Ming-Yu Chang received his B.S. degree from the Department of Computer Science and Information Engineering, National Chin-Yi University of Technology, Taichung, Taiwan, in 2022, where he is currently pursuing his M.S. degree. His current research interests include deep learning, neural fuzzy systems, LiDAR sensors, mobile robots, and computer vision and applications. (kamisama0406@gmail.com)

Kuang-Hui Tang received his M.S. degree in telecommunication engineering from Da-Yeh University, Taiwan, in 2005. Currently, he is an assistant professor of National Chin-Yi University of Technology, Taiwan. His current research interests are image processing and recognition, communication electronics, electromagnetic compatibility, and solar control systems.

Chuan-Kuei Huang received his Ph.D. degree from the Graduate School of Engineering Science and Technology, National Yunlin University of Science and Technology, Taiwan, in 2004. Currently, he is a professor at the Department of Industrial Education and Technology, National Changhua University of Education, Taiwan. His current research interests are image processing, chaotic system application, power electronics, and electric vehicle control.