

Trajectory Tracking Control Design for Mobile Robot Using Interval Type-2 Fuzzy Logic

Mochammad Dekcy Akmal¹, Adnan Rafi Al Tahtawi^{2,*}, Kartono Wijayanto³, Faisal Wahab⁴
^{1,2,3} Department of Electrical Engineering, Politeknik Negeri Bandung, Bandung Barat, Indonesia
⁴ Department of Mechatronics Engineering, Universitas Katolik Parahyangan, Bandung, Indonesia
Email: ¹ moch.dekcy.toi17@polban.ac.id, ² adnan.raf@polban.ac.id, ³ karwij@gmail.com, ⁴ faisal.wahab@unpar.ac.id
*Corresponding Author

Abstract— A two-wheeled mobile robot (TWMR) is a type of mobile robot that is widely used for several tasks, one of which is trajectory tracking. This task will be complicated for the robot to track the desired trajectory when there are disturbances in the robot, such as the presence of noise from the sensor feedback. Therefore, a robust control mechanism is needed so that the robot is able to track the trajectory properly. This study proposes an Interval Type-2 Fuzzy Logic Control (IT2FLC) for controlling TWMR which has noise parameters in the heading angle. IT2FLC Sugeno method is designed with error input from the heading angle and its changes, while the output is a voltage for the left and right motors of the robot. The input membership function is constructed using a triangular function with three different Footprint of Uncertainty (FoU) values as a comparison. This controller is then implemented to control the heading angle according to the trajectory of the robot using the Differential Drive Wheeled Mobile Robot (DDWMR) kinematic model. Several scenarios are then testing via MATLAB/Simulink to see the controller performance. The test results by including the noise parameter as disturbance show that IT2FLC can produce a better performance when compared to ordinary FLC. The widest FoU value of 0.3 can produce the smallest error value of up to 0.003 on the ISE criteria.

Keywords— IT2FLC; DDWMR; Heading Angle; FoU; Disturbance

I. INTRODUCTION

Nowadays, mobile robot technology is growing rapidly and has been widely applied in various fields. One type of mobile robot that is widely used is the Differential Drive Wheeled Mobile Robot (DDWMR). This robot moves based on the difference in the rotational speed of the wheels which are controlled separately. Both wheels are driven by an electric motor with a separate voltage source. This difference will result in the resultant direction of the robot's movement and the linear speed of the robot in carrying out its mission. Generally, DDWMR has two main wheels, each of which is driven by a separate drive and is equipped with a castor wheel that is placed in front or behind the robot as a counterweight. DDWMR can be modeled using kinematics and dynamics models. Kinematics models are often used because they have simpler equations when compared to the dynamic model. Besides, mobile robots usually use only the low speed of the motor to control the loop.

One of the most widely applied tasks in DDWMR is trajectory tracking. The robot must follow the desired trajectory with minimal error. Various control methods can be used in this task, such as Lyapunov-based control [1], back-stepping control [2], sliding mode control (SMC) [3],

[4], fuzzy logic control (FLC) [5]–[8], and model predictive control (MPC) [9]–[11]. These methods can be applied to both kinematic and dynamic models of the robot, and are usually performed to reject disturbance. The presence of disturbances in the robot such as that caused by measurement noise can reduce control performance. Thus, a robust control scheme is needed that can overcome disturbances to the robot. Several previous studies have designed robust control schemes for mobile robots. Research [12] and [13] designed an SMC for a DDWMR tracking system with various disturbances such as measurement noise parameters, frictional disturbances, and model uncertainty. The simulation results show that the proposed controller can produce good transient and steady-state responses when there are any disturbances. Then, research [14] designed an MPC on a mobile robot to reject disturbance. Two observers are designed to estimate the disturbance signal. The combination of the observer and SMC produces an asymptotically stable system if the harmonic frequency of the disturbance is known. A robust control approach with intelligent systems based on the neural network has also been carried out by [15]. The adaptive neural network tracking controller has been designed to estimate disturbances in a mobile robot. Then the intelligent system approach based on Z-number FLC has also been investigated by [16] to produce a robust tracking control. Lastly, robust control of mobile robots using a particle swarm optimization (PSO) algorithm approach has also been investigated by [17] for obstacle avoidance missions. Based on some of the studies above, the intelligent systems approach in the controller design is considered simpler when compared to the mathematical model approach because it does not involve a mathematical model equation of the plant to be controlled.

FLC is one of the intelligent control methods that are often used because of its simplicity in terms of design and implementation. In addition, FLC can be used as the basis for intelligent control by applying other artificial intelligence algorithms such as ant colony and PSO in the optimization process [18]. Currently, the Interval Type-2 Fuzzy Logic Controller (IT2FLC) is present as the development of the ordinary FLC. IT2FLC can produce robust control performance because it has a Footprint of Uncertainty (FoU) in its input membership function [19]. FoU is a range of values found at the foot of the linguistic set so that it is considered capable of overcoming the uncertainty of values that occur in the fuzzy membership function. This is proven in several applications, especially in the case of trajectory

tracking control on mobile robots. Research [20] designed IT2FLC to control mobile robot navigation with wheel slip and uncertainty parameters. The results of simulation and validation using the VREP simulator show that IT2FLC produces better performance when compared to ordinary FLC in overcoming parameter uncertainty. A dynamic-group particle swarm optimization (DGPSO) algorithm is proposed by [21] to provide navigation control and obstacle avoidance for a mobile robot. The results show that the proposed method is more efficient than other optimization methods in an unknown environment. IT2FLC has also been combined with SMC with a double-loop mechanism for trajectory tracking of the mobile robot [22]. IT2FLC is used to adjust the switching gain of the SMC when a disturbance occurs. The simulation results show that the proposed method produces good performance in terms of small errors and fast-tracking. In the intelligent systems approach, IT2FLC can also be integrated with a neural network in terms of parameter learning when the mobile robot is faced with a trajectory-tracking task [23]. Based on several studies related to the application of IT2FLC on mobile robots above, it can be concluded that IT2FLC can produce robust control performance when there is a disturbance on the robot. However, from these studies, the effect of FoU width on the IT2FLC input membership function on control performance was not investigated. Based on research [24], wider FoU will improve control performance on DC motor plant and water level tanks.

Based on previous research, IT2FLC can be applied for mobile robot control in terms of trajectory tracking [20]-[23]. Thus, this study aims to design a trajectory-tracking controller on a Two-Wheeled Mobile Robot (TWMR) using IT2FLC with various FoU values. IT2FLC is designed to overcome the presence of measurement noise parameters on the robot's heading angle with input in the form of heading angle error and its changes. The resulting control signal is a voltage for each motor of the robot. The control performance generated by IT2FLC is then compared with ordinary FLC and also performed with various FoU values. To test the proposed controller, a DDWMR kinematics model was built using MATLAB/Simulink for simplicity. The main contribution of this paper is to produce a robust mobile robot trajectory tracking performance and to see the effect of the various intervals of the FoU values on the designed controller. As a preliminary study, we use the IT2FLC Toolbox developed by [25] with the Karnik-Mendel (KM) type reducer method. This is due to the ease of design and simulation.

II. RESEARCH METHOD

A. Two-Wheeled Mobile Robot Kinematic Model

The TWMR can be modeled using the DDWMR model. A top view of the DDWMR geometry model used in this study is depicted in Fig. 1. The robot model consists of two coordinate frames, namely the reference frame and the vehicle frame. The reference frame (P_r) can be defined as.

$$P_r = [x \quad y \quad \theta]^T \quad (1)$$

while the vehicle frame (P_v) as.

$$P_v = [x_v \quad y_v \quad \theta_v]^T \quad (2)$$

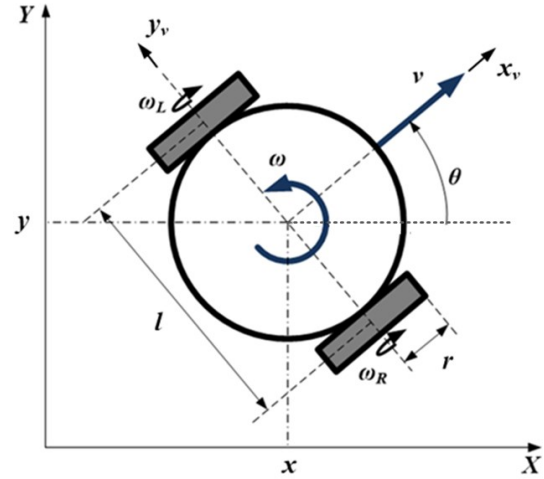


Fig. 1. DDWMR geometry model

The transformation of the robot coordinates from the vehicle frame (P_v) to the reference frame (P_r) satisfies the following equation.

$$P_r = \begin{bmatrix} \cos \theta & -\sin \theta & 0 \\ \sin \theta & \cos \theta & 0 \\ 0 & 0 & 1 \end{bmatrix} P_v \quad (3)$$

In the DDWMR model, several constraints must be met or are usually called non-holonomic constraints. These limitations are:

1. The robot trajectory is on a flat plane or zero potential energy
2. There is no lateral movement of the y-axis on the vehicle frame of the robot. This means $\dot{y}_v = 0$. Based on the inverse transformation matrix of P_r , it is obtained that

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

3. There is no slip on the wheel rotation. This constraint is formulated as

$$\dot{x} \cos \theta + \dot{y} \sin \theta + l\dot{\theta} = r\omega_R \quad (5)$$

$$\dot{x} \cos \theta + \dot{y} \sin \theta - l\dot{\theta} = r\omega_L \quad (6)$$

Equation (4) to equation (6) can be formed as

$$A(P)\dot{P} = 0 \quad (7)$$

$$\begin{bmatrix} -\sin \theta & \cos \theta & 0 & 0 & 0 \\ \cos \theta & \sin \theta & l & -r & 0 \\ \cos \theta & \sin \theta & -l & 0 & -r \end{bmatrix} \begin{bmatrix} \dot{x} \\ \dot{y} \\ \dot{\theta} \\ \dot{\phi}_R \\ \dot{\phi}_L \end{bmatrix} = 0 \quad (8)$$

where $A(P)$ is a non-holonomic constraint matrix. The DDWMR model can be formed using a kinematics approach. The kinematics model that is usually used in mobile robots is forward kinematics. This model aims to obtain the linear speed (v) and angular speed (ω) of the robot if the speed of the right (ω_R) and left wheels (ω_L) are known. The model can be formed as.

$$\begin{bmatrix} v \\ \omega \end{bmatrix} = \begin{bmatrix} \frac{r}{2} & \frac{r}{2} \\ r & -r \\ \frac{r}{2l} & -\frac{r}{2l} \end{bmatrix} \begin{bmatrix} \omega_R \\ \omega_L \end{bmatrix} \quad (9)$$

If the model is transformed in the reference frame, it can be written.

$$\begin{bmatrix} \dot{x} \\ \dot{y} \\ \dot{\theta} \end{bmatrix} = \begin{bmatrix} \cos \theta & 0 \\ \sin \theta & 0 \\ 0 & 1 \end{bmatrix} \begin{bmatrix} v \\ \omega \end{bmatrix} \quad (10)$$

If equation (9) is substituted into equation (10), it can be formed.

$$\begin{bmatrix} \dot{x} \\ \dot{y} \\ \dot{\theta} \end{bmatrix} = \begin{bmatrix} \frac{r}{2} \cos \theta & \frac{r}{2} \cos \theta \\ \frac{r}{2} \sin \theta & \frac{r}{2} \sin \theta \\ r & -r \\ \frac{r}{2l} & -\frac{r}{2l} \end{bmatrix} \begin{bmatrix} \omega_R \\ \omega_L \end{bmatrix} \quad (11)$$

B. IT2FLC Overview

The block diagram of IT2FLC is shown in Fig. 2. In general, the IT2FLC block diagram is almost the same as the type 1 fuzzy logic controller, which consists of the fuzzification process, inference mechanism, and defuzzification. The difference lies in the type reduction process, wherein in this process the output value of the inference mechanism is reduced so that the defuzzification process can be carried out.

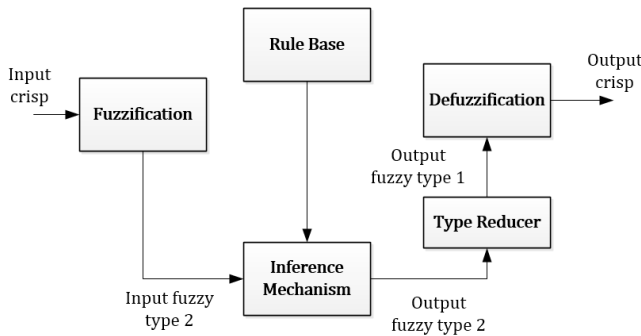


Fig. 2. The IT2FLC block diagram

As in ordinary FLC, the output of IT2FLC can be calculated in the following steps:

1. Define the rule structure of Sugeno type IT2FLC from N rules as follows

$$R^n: \text{if } x_1 \text{ is } \tilde{X}_1^n \text{ and } \dots \text{ and } x_l \text{ is } \tilde{X}_l^n, \text{ then } y \text{ is } Y^n \quad (12)$$

where \tilde{X}_i^n is the antecedent and $Y^n = [\underline{y}^n, \bar{y}^n]$ is the consequent membership function.

2. For crisp input vector $x' = [x'_1 \ x'_2 \ \dots \ x'_l]$, compute the membership degree of fuzzy sets for each x'_i and each \tilde{X}_i^n or can be written $[\underline{\mu}_{\tilde{X}_i^n}(x'_i), \bar{\mu}_{\tilde{X}_i^n}(x'_i)]$ for $i = 1, 2, \dots, l$ and $n = 1, 2, \dots, N$.
3. Calculate the rule firing interval as $F^n(x') = [\underline{f}^n, \bar{f}^n]$ for $n = 1, 2, \dots, N$ where

$$\underline{f}^n = [\underline{\mu}_{\tilde{X}_1^n}(x'_1) \times \dots \times \underline{\mu}_{\tilde{X}_l^n}(x'_l)] \quad (13)$$

$$\bar{f}^n = [\bar{\mu}_{\tilde{X}_1^n}(x'_1) \times \dots \times \bar{\mu}_{\tilde{X}_l^n}(x'_l)] \quad (14)$$

4. Combine $F^n(x')$ and the corresponding rule consequent using Type Reducer (TR). The TR commonly uses the center-of-sets which defined as

$$Y(x') = \bigcup_{\substack{f^n \in F^n(x') \\ y^n \in Y^n}} \frac{\sum_{n=1}^N y^n f^n}{\sum_{n=1}^N f^n} = [y_l, y_r] \quad (15)$$

where

$$y_l = \frac{\sum_{n=1}^L y^n \bar{f}^n + \sum_{n=L+1}^N y^n \underline{f}^n}{\sum_{n=1}^L \bar{f}^n + \sum_{n=L+1}^N \underline{f}^n} \quad (16)$$

$$y_r = \frac{\sum_{n=1}^R \bar{y}^n \underline{f}^n + \sum_{n=R+1}^N \bar{y}^n \bar{f}^n}{\sum_{n=1}^R \underline{f}^n + \sum_{n=R+1}^N \bar{f}^n}$$

R and L are switching points that can be found using several iterative algorithms.

5. Calculate the defuzzification crisp output as follows

$$u = \frac{y_l + y_r}{2} \quad (18)$$

The crisp output value (u) is an IT2FLC control signal that will be used to control the plant.

C. IT2FLC Design for Mobile Robot

In this section, IT2FLC will be designed to control the movement of the robot while tracking the trajectory. This study assumes the presence of noise in the sensor that detects the trajectory so that it affects the control performance. The proposed control scheme can be seen in Fig. 3.

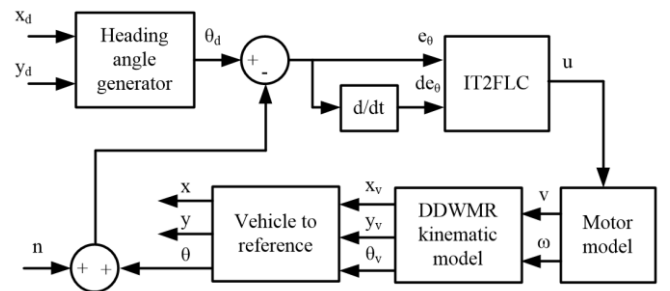


Fig. 3. Proposed control scheme

Given the desired robot posture $P_d = [x_d \ y_d]^T$ as reference trajectory, thus the desired heading angle (θ_d) can be calculated as follows

$$\theta_d = \tan^{-1} \frac{\delta_y}{\delta_x} \quad (19)$$

where

$$\delta_y = y_d(k) - y(k-1) \quad (20)$$

$$\delta_x = x_d(k) - x(k-1) \quad (21)$$

The resulting heading angle (θ_d) is then used as a controlled variable. We do not use trajectory coordinates (P_d) as a control setpoint because it will be more complex in FLC design as well as rule-based design. In this study, the controller is designed with the input error of the heading angle (e_θ) and its changes (δe_θ) which are defined as follows

$$e_\theta(k) = \theta_d(k) - \theta(k) \quad (22)$$

$$\delta e_\theta(k) = e_\theta(k) - e_\theta(k - 1) \quad (23)$$

with the membership function depicted in Fig. 4.

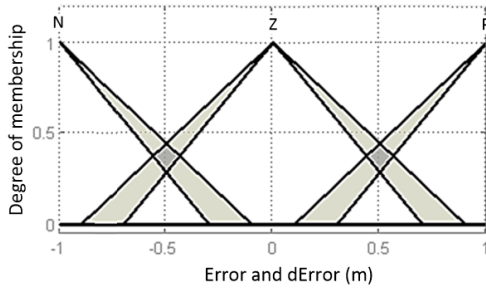


Fig. 4. Membership function of error and its changes

We use the same membership function for both input variables. As can be seen, the difference between IT2FLC and ordinary FLC is the presence of FoU in the input membership function. In this design, the two input variables are formed using three triangular linguistic sets, namely Negative (N), Zero (Z), and Positive (P) which have a width of FoU 0.2 with a range of values shown in Table 1.

Table 1. The values of input fuzzy sets with FoU 0.2

Sets	Values
N	Upper = [-1 -1 -0.1 1]
	Lower = [-1 -1 -0.3 1]
Z	Upper = [-0.9 0 0.9 1]
	Lower = [-0.7 0 0.7 1]
P	Upper = [0.1 1 1 1]
	Lower = [0.3 1 1 1]

The output of the controller is the voltage for both motors of the robot that is classified into seven crisp values Negative High (-12), Negative Medium (-8), Negative Low (-4), Zero (0), Positive Low (4), Positive Medium (8), and Positive High (12). Then the rule-base is defined as shown in Table 2.

Table 2. Fuzzy rule-base

E/dE	N	Z	P
N	NH	NM	NL
Z	NM	Z	PM
P	PL	PM	PH

The calculated control signal according to equations (12) to (18) is then given to the right motor (Rm) and the left motor (Lm) based on the algorithm as shown in Fig. 5. We use a DC motor model as an actuator that receives control signals [26]-[28]. These two DC motors will then move the mobile robot towards the desired trajectory.

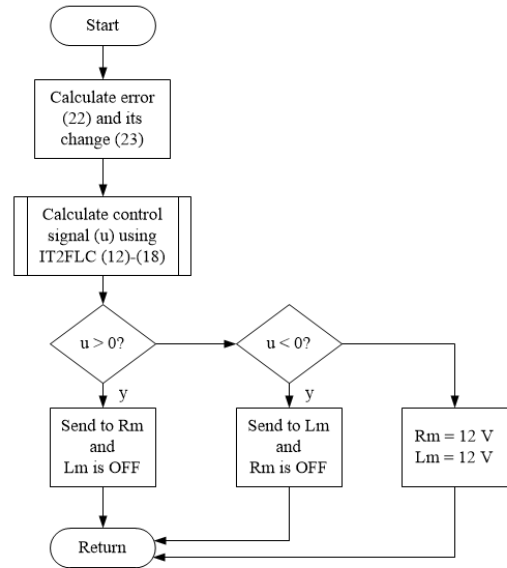


Fig. 5. The output IT2FLC algorithm

The control signal given to the motor will then cause the robot's translation speed and motor rotation speed. These two quantities then become input for the kinematic model in equation (10). Then, using the vehicle to reference the transformation matrix, heading angle data will be obtained which will be used as feedback.

III. RESULT AND DISCUSSION

In this section, the proposed controller will be tested to see the robot's response when moving on the desired trajectory. The robot model was built based on Fig. 3 using MATLAB/Simulink. The test is carried out by providing a disturbance signal as shown in Fig. 6 with a power parameter of 0.1 W and a frequency of 100 Hz. This disturbance signal represents noise in the detection of the robot heading angle sensor.

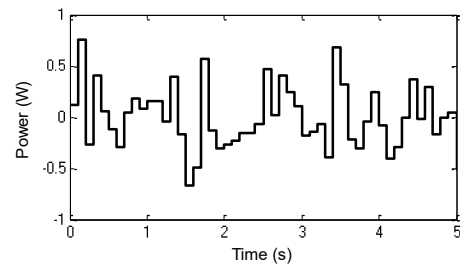


Fig. 6. Noise signal for testing

Three trajectory scenarios namely straight, curved, and sinusoidal are used in this test. In each test, the response of the head angle to the reference, the position of the robot in the reference frame, the translational speed, and the rotational speed of the robot will be seen and analyzed. Then the response to the control of the heading angle was compared to ordinary FLC. The error is then analyzed using the integral error method, namely Integral Absolute Error (IAE), Integral Time Absolute Error (IAE), Integral Square Error (ISE), and Integral Time Absolute Error (ITSE) using the following equations.

$$IAE = \int_0^{\infty} |e_{\theta}(t)| dt \quad (24)$$

$$ITAE = \int_0^{\infty} t |e_{\theta}(t)| dt \quad (25)$$

$$ISE = \int_0^{\infty} [e_{\theta}(t)]^2 dt \quad (26)$$

$$ITSE = \int_0^{\infty} t [e_{\theta}(t)]^2 dt \quad (27)$$

Finally, the effect of the various FoU values on the input membership function on control performance will also be investigated. For simulation requirements, the parameters used for the DDWMR model including the motor model are presented in Table 3. We use IT2FLC MATLAB Toolbox for designing the controller developed by [25] for simplicity. The controller is then implemented on the DDWMR model using MATLAB/Simulink.

Table 3. Simulation parameter

Parameters	Values
Distance between the drive wheel and the axis of symmetry (l)	5.6 m
Wheel radius (r)	0.2 m
Motor resistance (R)	5 Ohm
Inductance (L)	0.0054 H
Moment of inertia (J)	0.00015 Kg.m ²
Rotor friction (b)	0.0001 N.m.s
Motor constant (K)	0.1

A. System Testing with Straight Trajectory

The first test is carried out on a straight line or $x_d = 2t$ and $y_d = 0$. Thus, according to equation (19) $\theta_d = 0$ is obtained. The test results can be seen in Fig. 7.

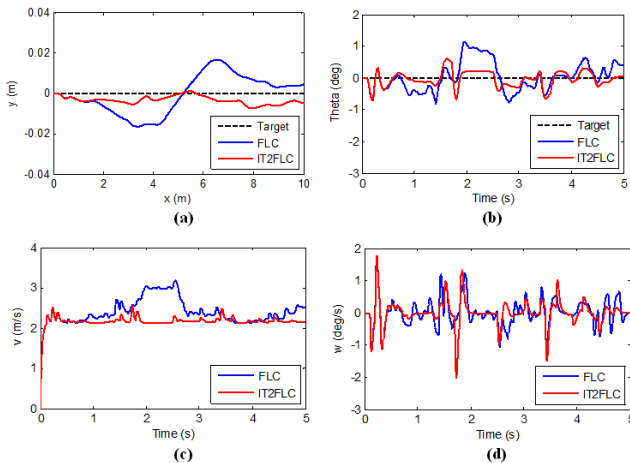


Fig. 7. Responses at straight trajectory: (a) robot position, (b) robot heading angle, (c) translation speed, and (d) rotational speed

Based on the test results, it appears that IT2FLC produces better performance than FLC. Fig. 7(a) shows the movement of the robot when it is on a straight trajectory where IT2FLC produces a smaller deviation than ordinary FLC. Fig. 7(b) is the resulting robot heading angle where the deviation from IT2FLC is also smaller when compared to FLC. Fig. 7(c) is

the translational speed of the robot. It can be seen that IT2FLC produces robot speed with smaller oscillation than FLC. Fig. 7(d) shows the rotational speed of the robot when moving on a trajectory where IT2FLC produces an oscillation frequency lower than FLC.

B. System Testing with Curved Trajectory

The second test is done using a curved trajectory. The desired trajectory is described as $x_d = 2t$ and $y_d = 0.2t^2 - 4$ with resulted heading angle $\theta_d = 10t - 0.44$. The test result for this trajectory is depicted in Fig. 8.

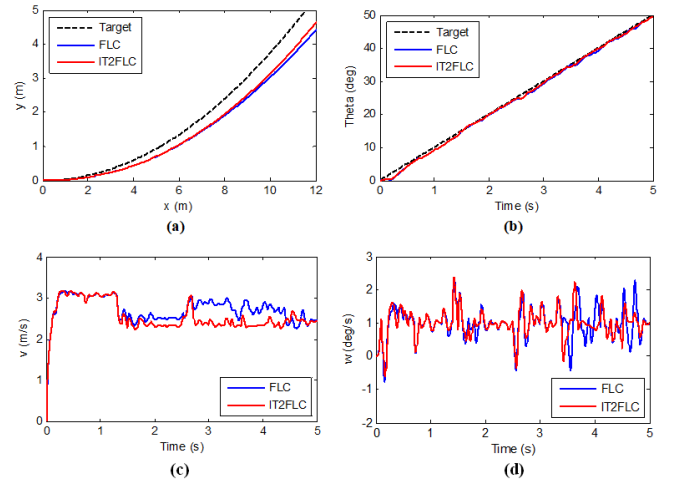


Fig. 8. Responses at curved trajectory: (a) robot position, (b) robot heading angle, (c) translation speed, and (d) rotational speed

In the second scenario testing, IT2FLC also produces better control performance than ordinary FLC. Fig. 8(a) shows the movement of the robot where IT2FLC produces a smaller deviation than ordinary FLC when the system is disturbed. Fig. 8(b) shows the resulting robot heading angle on the second scenario testing. Fig. 8(c) shows the translational speed of the robot where IT2FLC produces robot speed with smaller oscillation. Fig. 8(d) shows the rotational speed of the robot where IT2FLC produces a lower oscillation frequency in the last simulation time.

C. System Testing with Sinusoidal Trajectory

The last testing is carried out using a sinusoidal trajectory. The desired trajectory is generated as $x_d = 2t$ and $y_d = 0.2 - 0.2 \cos 2t$ with heading angle $\theta_d = 10 \sin 2t - 0.06 \cos 2t$. The test result is depicted in Fig. 9. The last scenario testing result shows that IT2FLC also produces better control performance. Fig. 9(a) shows the movement of the robot on the reference frame, where IT2FLC produces a smaller deviation significantly. Fig. 9(b) shows the comparison of the robot heading angle between FLC and IT2FLC. Fig. 9(c) is the translational speed of the robot where IT2FLC gives the best response if compared with FLC. Fig. 9(d) is the rotational speed of the robot where IT2FLC also gives the smallest oscillation.

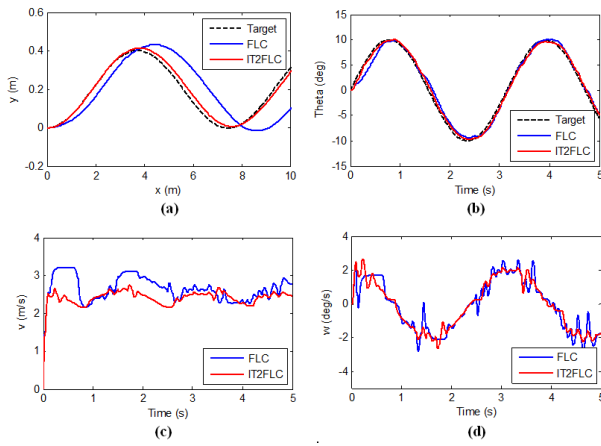


Fig. 9. Responses at sinusoidal trajectory: (a) robot position, (b) robot heading angle, (c) translation speed, and (d) rotational speed

D. Comparison of Different FoU

In the last section, a test was conducted on IT2FLC with different FoU. The test aims to see the effect of the FoU width on the input membership function on control performance. The test is carried out by changing the value of FoU 0.2 to be smaller (0.1) and larger (0.3) with a range of values for each FoU can be seen in Table 4. Then, the response of the robot’s movement and the heading angle of the robot are observed and the error values are compared for each FoU.

Table 4. The values of input fuzzy sets with different FoU

Sets	FoU = 0.1	FoU = 0.3
N	Upper = [-1 -1 -0.15 1] Lower = [-1 -1 -0.25 1]	Upper = [-1 -1 -0.05 1] Lower = [-1 -1 -0.35 1]
Z	Upper = [-0.85 0 0.85 1] Lower = [-0.75 0 0.75 1]	Upper = [-0.95 0 0.95 1] Lower = [-0.65 0 0.65 1]
P	Upper = [0.15 1 1 1] Lower = [0.25 1 1 1]	Upper = [0.05 1 1 1] Lower = [0.35 1 1 1]

The test result of the robot’s movement and its heading angle for three scenarios testing with different FoU is shown in Fig. 10. Based on the test results, it can be seen that FoU = 0.3 produces better control performance than FoU = 0.2 and FoU = 0.1. This can be seen from the smaller robot deviation at FoU = 0.3 when given a disturbance.

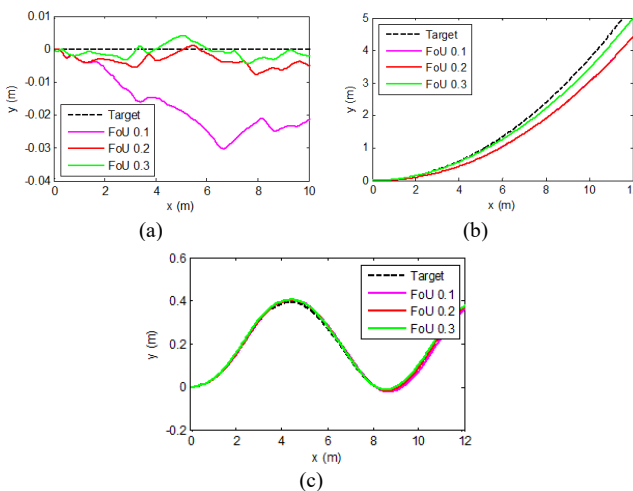


Fig. 10. Response comparison with various FoU: (a) straight trajectory, (b) curved trajectory, and (c) sinusoidal trajectory

To prove it, the error integral calculation is repeated in all test scenarios. The results are shown in Fig. 11. The result of the comparison of integral error with different FoU for each trajectory scenario shows that the wider the FoU, the better the robustness of the response. This means that the FoU on IT2FLC represents a range of uncertainty, one of which is caused by measurement noise. However, this result cannot be proven mathematically. In addition, the reliability of the computing device needs to be considered since IT2FLC has a wide FoU, and iterations on the type reduction algorithm are quite complex.

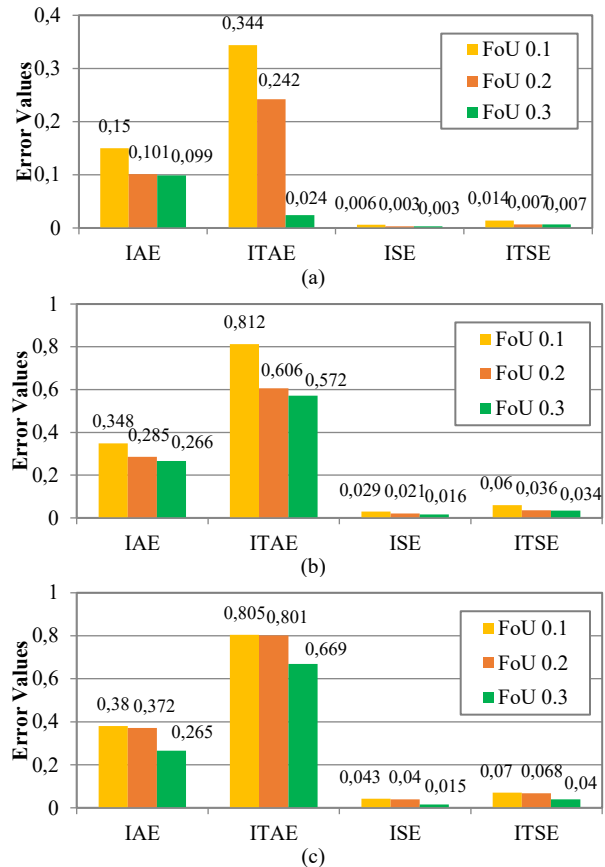


Fig. 11. Integral error comparison with various FoU: (a) straight trajectory, (b) curved trajectory, and (c) sinusoidal trajectory

IV. CONCLUSION

Trajectory tracking control for the TWMR kinematics model using IT2FLC has been successfully designed and tested. Three different FoU values were tested in three robotic trajectory scenarios. The results show that IT2FLC has better robustness when compared to ordinary FLC when the robot is given noise parameters. Furthermore, the larger the FoU width on IT2FLC results in better control performance where the resulting integral error is smaller. In further research, optimization of FoU selection can be applied so that a response that is considered optimal can be produced.

REFERENCES

[1] M. Elsayed, A. Hammad, A. Hafez, and H. Mansour, “Real Time Trajectory Tracking Controller based on Lyapunov Function for Mobile Robot,” *Int. J. Comput. Appl.*, vol. 168, no. 11, pp. 1–6, 2017, <https://doi.org/10.5120/ijca2017914540>.
 [2] M. Kalyoncu and F. Demirbaş, “Differential Drive Mobile Robot Trajectory Tracking With Using Pid and Kinematic Based

- Backstepping Controller,” *Selcuk Univ. J. Eng., Science Technol.*, vol. 5, no. 1, pp. 1–15, 2017, <https://doi.org/10.15317/Scitech.2017.65>.
- [3] N. H. Hadi and K. K. Younus, “Path tracking and backstepping control for a wheeled mobile robot (WMR) in a slipping environment,” *IOP Conf. Ser. Mater. Sci. Eng.*, vol. 671, no. 1, 2020, <https://doi.org/10.1088/1757-899X/671/1/012005>.
- [4] N. Esmaili, A. Alfi, and H. Khosravi, “Balancing and Trajectory Tracking of Two-Wheeled Mobile Robot Using Backstepping Sliding Mode Control: Design and Experiments,” *J. Intell. Robot. Syst. Theory Appl.*, vol. 87, no. 3–4, pp. 601–613, 2017, <https://doi.org/10.1007/s10846-017-0486-9>.
- [5] A. Adib and B. Masoumi, “Mobile robots navigation in unknown environments by using fuzzy logic and learning automata,” *7th Conf. Artif. Intell. Robot. IRANOPEN 2017*, pp. 58–63, 2017, <https://doi.org/10.1109/RIOS.2017.7956444>.
- [6] N. Y. Allagui, D. Ben Halima Abid, and N. Derbel, “Fuzzy PI controller for mobile robot navigation and tracking,” *2018 15th Int. Multi-Conference Syst. Signals Devices, SSD 2018*, pp. 1178–1183, 2018, <https://doi.org/10.1109/SSD.2018.8570477>.
- [7] M. S. Abood, I. K. Thajeel, E. M. Alsaedi, M. M. Hamdi, A. S. Mustafa, and S. A. Rashid, “Fuzzy Logic Controller to control the position of a mobile robot that follows a track on the floor,” *4th Int. Symp. Multidiscip. Stud. Innov. Technol. ISMSIT 2020 - Proc.*, 2020, <https://doi.org/10.1109/ISMSIT50672.2020.9254417>.
- [8] R. Hartono and T. N. Nizar, “Speed Control of a Mobile Robot Using Fuzzy Logic Controller,” *IOP Conf. Ser. Mater. Sci. Eng.*, vol. 662, no. 2, 2019, <https://doi.org/10.1088/1757-899X/662/2/022063>.
- [9] C. Wang, X. Liu, X. Yang, F. Hu, A. Jiang, and C. Yang, “Trajectory tracking of an omnidirectional wheeled mobile robot using a model predictive control strategy,” *Appl. Sci.*, vol. 8, no. 2, 2018, <https://doi.org/10.3390/app8020231>.
- [10] T. P. Nascimento, C. E. T. Dórea, and L. M. G. Gonçalves, “Nonholonomic mobile robots’ trajectory tracking model predictive control: A survey,” *Robotica*, vol. 36, no. 5, pp. 676–696, 2018, <https://doi.org/10.1017/S0263574717000637>.
- [11] M. R. Azizi and J. Keighobadi, “Point stabilization of nonholonomic spherical mobile robot using nonlinear model predictive control,” *Rob. Auton. Syst.*, vol. 98, 2019, pp. 347–359, 2017, <https://doi.org/10.1016/j.robot.2017.09.015>.
- [12] N. K. Goswami and P. K. Padhy, “Sliding mode controller design for trajectory tracking of a non-holonomic mobile robot with disturbance,” *Comput. Electr. Eng.*, vol. 72, pp. 307–323, 2018, <https://doi.org/10.1016/j.compeleceng.2018.09.021>.
- [13] Y. Guo, L. Yu, and J. Xu, “Robust Finite-Time Trajectory Tracking Control of Wheeled Mobile Robots with Parametric Uncertainties and Disturbances,” *J. Syst. Sci. Complex.*, vol. 32, no. 5, pp. 1358–1374, 2019, <https://doi.org/10.1007/s11424-019-7235-z>.
- [14] Z. Sun, Y. Xia, L. Dai, K. Liu, and D. Ma, “Disturbance rejection MPC for tracking of wheeled mobile robot,” *IEEE/ASME Trans. Mechatronics*, vol. 22, no. 6, pp. 2576–2587, 2017, <https://doi.org/10.1109/TMECH.2017.2758603>.
- [15] M. Boukens, A. Boukabou, and M. Chadli, “Robust adaptive neural network-based trajectory tracking control approach for nonholonomic electrically driven mobile robots,” *Rob. Auton. Syst.*, vol. 92, pp. 30–40, 2017, <https://doi.org/10.1016/j.robot.2017.03.001>.
- [16] M. Abdelwahab, V. Parque, A. M. R. Fath Elbab, A. A. Abouelsoud, and S. Sugano, “Trajectory tracking of wheeled mobile robots using Z-Number based fuzzy logic,” *IEEE Access*, vol. 8, pp. 18426–18441, 2020, <https://doi.org/10.1109/ACCESS.2020.2968421>.
- [17] H. S. Dewang, P. K. Mohanty, and S. Kundu, “A Robust Path Planning for Mobile Robot Using Smart Particle Swarm Optimization,” *Procedia Comput. Sci.*, vol. 133, pp. 290–297, 2018, <https://doi.org/10.1016/j.procs.2018.07.036>.
- [18] F. Olivas, L. Amador-Angulo, J. Perez, C. Caraveo, F. Valdez, and O. Castillo, “Comparative study of type-2 fuzzy Particle swarm, Bee Colony and Bat Algorithms in optimization of fuzzy controllers,” *Algorithms*, vol. 10, no. 3, 2017, <https://doi.org/10.3390/a10030101>.
- [19] M. A. Sanchez, O. Castillo, and J. R. Castro, “Generalized Type-2 Fuzzy Systems for controlling a mobile robot and a performance comparison with Interval Type-2 and Type-1 Fuzzy Systems,” *Expert Syst. Appl.*, vol. 42, no. 14, pp. 5904–5914, 2015, <https://doi.org/10.1016/j.eswa.2015.03.024>.
- [20] Y. Saidi, A. Nemra, and M. Tadjine, “Robust mobile robot navigation using fuzzy type 2 with wheel slip dynamic modeling and parameters uncertainties,” *Int. J. Model. Simul.*, vol. 40, no. 6, pp. 397–420, 2020, <https://doi.org/10.1080/02286203.2019.1646480>.
- [21] J. Y. Jhang, C. J. Lin, C. T. Lin, and K. Y. Young, “Navigation Control of Mobile Robots Using an Interval Type-2 Fuzzy Controller Based on Dynamic-group Particle Swarm Optimization,” *Int. J. Control. Autom. Syst.*, vol. 16, no. 5, pp. 2446–2457, 2018, <https://doi.org/10.1007/s12555-017-0156-5>.
- [22] S. Dian *et al.*, “Double Closed-Loop General Type-2 Fuzzy Sliding Model Control for Trajectory Tracking of Wheeled Mobile Robots,” *Int. J. Fuzzy Syst.*, vol. 21, no. 7, pp. 2032–2042, 2019, <https://doi.org/10.1007/s40815-019-00685-z>.
- [23] A. Bencherif and F. Chouireb, “A recurrent TSK interval type-2 fuzzy neural networks control with online structure and parameter learning for mobile robot trajectory tracking,” *Appl. Intell.*, vol. 49, no. 11, pp. 3881–3893, 2019, <https://doi.org/10.1007/s10489-019-01439-y>.
- [24] E. Ontiveros, P. Melin, and O. Castillo, “Impact study of the footprint of uncertainty in control applications based on interval type-2 fuzzy logic controllers,” *Stud. Comput. Intell.*, vol. 749, pp. 181–197, 2018, doi: 10.1007/978-3-319-71008-2_15.
- [25] A. Taskin and T. Kumbasar, “An open source Matlab/Simulink toolbox for interval type-2 fuzzy logic systems,” *Proc. - 2015 IEEE Symp. Ser. Comput. Intell. SSCI 2015*, pp. 1561–1568, 2015, https://doi.org/10.1007/978-3-319-71008-2_15.
- [26] R. T. Yunardi, D. Arifianto, F. Bachtiar, and J. I. Prananingrum, “Holonomic implementation of three wheels omnidirectional mobile robot using dc motors,” *Journal of Robotics and Control (JRC)*, vol. 2, no. 2, pp. 65–71, 2021, <https://doi.org/10.18196/jrc.2254>.
- [27] B. Raafi’u, P. A. Darwito, A. Fauzan 'Adziimaa, H. S. Hadi, S. N. Patrialova, and M. Raditya, “Comparative Study of Fuzzy-PID and Fuzzy-PI Control Systems on DC Motor Speed for Four-Wheeled Mobile Robotic,” *2019 International Conference on Advanced Mechatronics, Intelligent Manufacture and Industrial Automation (ICAMIMIA)*, Batu, Indonesia, 2019, pp. 129–133, <https://doi.org/10.1109/ICAMIMIA47173.2019.9223420>.
- [28] M. Dziubek, M. Klimowicz, and R. Grądzki, “Evaluation of temperature changes of a loaded DC motor in autonomous mobile robots,” *2019 20th International Carpathian Control Conference (ICCC)*, pp. 1–5, 2019, <https://doi.org/10.1109/CarpathianCC.2019.8766058>.