LiDAR Based Trajectory-Tracking of an Autonomous Differential Drive Mobile Robot Using Fuzzy Sliding Mode Controller

MUHAMMAD QOMARUZ ZAMAN1, HSIU-MING WU2

1Graduate Institute of Manufacturing Technology, National Taipei University of Technology, Taipei, Taiwan (e-mail: me@zamzaman.com)
2Department of Intelligent Automation Engineering, National Taipei University of Technology, Taipei, Taiwan (e-mail: hmwu@mail.ntut.edu.tw)

Corresponding author: Hsiu-Ming Wu (hmwu@mail.ntut.edu.tw).

This research was funded by Taiwan’s Ministry of Science and Technology grant number MOST 110-2221-E-027-085.

ABSTRACT This study aims to achieve the trajectory-tracking of an autonomous differential drive mobile robot (ADDMR) in the presence of friction torques using the proposed fuzzy sliding-mode control (FSMC). First, the complete model of the ADDMR is established including kinematic, dynamic and actuator models. Then, the desired trajectory is planned and generated via Bézier curve combined with cubic time mapping. The localization of the ADDMR is performed using the Monte Carlo Localization (MCL) technique and LiDAR point cloud data. Subsequently, the proposed FSMC utilizes nonlinear sliding surfaces inferred based on fuzzy logic to asymptotically attain the pose convergence. To investigate the feasibility and robustness of the proposed FSMC, simulations and experiments are conducted under different operation conditions. The results show that the proposed FSMC using LiDAR data in the feedback leads to excellent control performance even in the presence of friction torques. It was also shown through simulations and experiments that the established ADDMR model fits the actual ADDMR model very well.

INDEX TERMS Autonomous differential drive mobile robot, Fuzzy sliding-mode control, Monte Carlo localization, Nonlinear sliding surface, Bézier curve.

I. INTRODUCTION

ONLINE shopping has been drastically increasing over the past year, especially during COVID-19 pandemics [1], [2]. Factories are racing to equip their warehouses with automated logistic handling technologies [3], [4] or contactless delivery service [5], [6]. For these reasons, there has been numerous research on the control of mobile robots. The commonly used configurations are robots driven by wheels speed differences and they are controlled to follow a prescribed path from initial to final position in a determinate time. The type of robots driven by wheel velocity differences are called Differential Drive Mobile Robot. The challenge of controlling Autonomous Differential Drive Mobile Robots (ADDMRs) comes from its nonlinearity. Specifically, its nonholonomic property constrains its side movements. Many control techniques are proposed to handle it, but the important issue is that most of them only consider kinematics [7]–[22]. It is understandable because the consideration of its dynamic model will increase the complexity and dimensionality of the model which leads to difficulties for a controller design standpoint. However, kinematics does not consider mass and when an ADDMR is driven at high speed, its dynamic model becomes a necessity in addition to the kinematic model. Thus, some controller design techniques have been taking the dynamic model of the ADDMR into consideration. For example, the works in [23]–[34] considered both kinematics and dynamics for their vehicle models.

A significant task for the above-mentioned works is trajectory tracking in which the robot must follow a predefined path. The studies in [24] and [28] have positioned their ADDMRs on the path initially. Keighobadi and Menhaj [24] stated that their proposed method could not fully compensate initial position outside the path. In [25]–[27], the results demonstrate tracking error convergence at arbitrary initial positions; however, the circular path just has a constant velocity reference. Xu et al. [27] showed that it is important
to consider speed transition of the robot movement. In most previous works, the ADDMR was given a path without time constraints; therefore, the ADDMR approaches each point in the path incrementally. This is an important issue that the time arrangement along the planned path can lead to speed variations that the ADDMR has to follow.

The most recent developments in trajectory-tracking using fuzzy logic [14]–[22], [24], [30], [33]–[35] and sliding mode control techniques [13], [29]–[32], [35]–[39] are the motivation behind the work presented herein, which takes advantage of findings from both methods. Therefore, the FSMC based on a LiDAR point cloud data is proposed here to control an ADDMR system to follow a trajectory at prescribed time from arbitrary initial positions under torque disturbances. The contributions of this work are as follows:

1) A comprehensive ADDMR model consists of the kinematic and dynamic models is presented here. The model is augmented with the actuator dynamics for a more realistic representation of the motion of the ADDMR which has not been addressed by Refs. [7]–[36].

2) A desired trajectory is planned using Bézier curve. The $n + 1$ way-points on the trajectory are determined using an $n$th order Bézier curve. Unlike work in [27], the dynamic velocity command is embedded in the trajectory itself rather than in the control process. The advantage of the proposed approach is that it lends itself to possible combination with other control techniques in the closed-loop control design process. It also enables the trajectory to have zero velocity at the beginning and the end whereas previous works, such as those in [7]–[36] did not consider this. The close-loop performance of the ADDMR using the proposed control technique is validated experimentally using an 8th-order Bézier trajectory with cubic time mapping for the time arrangement along the path.

3) A fuzzy logic control design with 9 inference rules only combining a sliding mode control scheme leads to effective closed-loop control of the ADDMR. Fuzzy logic control offers multi-input, multi-output, model free controllers for trajectory tracking (see for example, [21] and [22]). In addition, the interpolative reasoning provides resilience against noisy measurement in the ADDMR system. However, unlike existing unlike existing techniques (e.g. [14]–[22], [30], [33]–[35]) where complex inference rules and membership functions are used , relatively simpler inference rules are used in the proposed control scheme. These lead to computational cost reduction and provide flexibility in the choice of computing devices used in the ADDMR testbed; a major advantage in the implementation phase.

4) LiDAR based Monte Carlo Localization (MCL) is used for feedback control to improve the control performance of the closed-loop system. The common approaches described in the literature (see for example [12], [29], [31]–[34]) use angular acceleration and torque feedback which renders the implementation a little more challenging. Instantaneous measurement of the orientation and position are easily accessible and implementable on an actual testbed [22]. Therefore, a LiDAR is used here for pose feedback due to its accuracy compared to GPS or odometry sensors by Refs. [40]–[43].

5) Experiments and simulations are conducted to validate the theoretical developments. Both the simulations and experiments are conducted using different initial conditions which have not been done in previous works (see for example [10], [12], [13], [16], [17], [19]–[21], [23], [24], [25], [33], [35], [36]).

The rest of the paper is arranged as follows: In Section 2, the system description, modelling, and problem formulation are given. In Section 3, the trajectory planning, pose measurement, and FSMC design are discussed. Discussions of the simulation and experiment results are presented in Section 4. Finally, concluding statements are made in Section 5.

II. SYSTEM DESCRIPTION, MODELING, AND PROBLEM FORMULATION

The hardware configuration for the ADDMR experimental platform is described here for which the equations of motion will be derived. The system modeling is divided into 3 subsystems, i.e., kinematic, dynamic, and actuator models. The details of each subsystem are explained in the following subsections. The problem formulation for the ADDMR is subsequently delineated using the equations of motion of all subsystems.

A. SYSTEM DESCRIPTION

The experimental platform of the ADDMR and its diagram are shown in Fig. 1. The main CPU from ADLINK AmITX-SL-G with Intel i5-6500TE and 8GB Random Access Memory (RAM) is where the proposed controller is computed and all the data from the STM32F microcontroller is stored. Two A4950 DMOS Full-Bridge motor drivers generate voltages to actuate two DC motors responsible for the motion of the robot. The provided sensors are Quadrature hall effect encoder and Inertial Measurement Unit (IMU) used to measure the corresponding position and orientation of the ADDMR; A LiDAR is also provided for measurements of distance around the robot. The wheels are centered and with center of mass is halfway in in between. In addition, two castor wheels are attached for the balance of the ADDMR.

B. SYSTEM MODELLING

The ADDMR is a type of mobile robots that the speed differences for two actuated wheels cause forward, backward, or turning movement. The illustration can be seen in the following Fig. 2. To establish geometric relationship of the ADDMR, an inertial frame $\{I\}$ and a body frame $\{B\}$ are given. Frame reference $\{I\}$ is attached to the floor and the
where \( \theta \) is the orientation of the ADDMR. Moreover, the relationship between the velocity of mass \( \{ B \} \) with respect to body frame and wheels’ angular velocities are given by:

\[
\begin{bmatrix}
\dot{\phi}_r \\
\dot{\phi}_l 
\end{bmatrix} = \frac{1}{R} \begin{bmatrix}
1 & L \\
1 & -L 
\end{bmatrix} \begin{bmatrix}
\dot{B} \dot{x} \\
\dot{\theta}
\end{bmatrix}
\]

In addition to the kinematic model of Eqs. (1) and (2), the Newton-Euler approach is used to build up the dynamic model of the robot. The ADDMR is assumed to have a unified rigid body and the equations of motion are determined with respect to reference \( \{ B \} \) as shown in Fig. 2. Moreover, two motors provide the ADDMR with longitudinal wheel forces \( (F_{lwx}, F_{rwx}) \) created by their corresponding torques \( (\tau_l, \tau_r) \) [45], [46], [49]. Hence, the linear and angular accelerations of the robot with respect to the body frame can be represented as

\[
\begin{align*}
\ddot{B} \dot{x} &= \frac{1}{MR} (\tau_l + \tau_r) \\
\ddot{\theta} &= \frac{L}{JR} (\tau_r - \tau_l)
\end{align*}
\]

That means that the dynamic model of the ADDMR generates linear velocity \( \dot{B} \dot{x} \) and angular velocity \( \dot{\theta} \) after integration when motor torques \( \tau_r \) and \( \tau_l \) are applied.

Combining the aforementioned kinematic with dynamic models, the actuator model is derived to complete the system’s equations of motion. However, designing the corresponding controller is very challenging due to the high dimensionality of the system’s model. The actuated system is comprised of two identical DC motors and the relations of voltage and torque of the motor are formulated as follows:

\[
\begin{align*}
V_a &= R_a i_a + L_a \frac{di_a}{dt} + e_a \\
e_a &= K_b \omega_m \\
\tau_m &= K_m i_a \\
\tau &= N \tau_m
\end{align*}
\]

where \( R_a, L_a, V_a, i_a \), and \( N \) are armature resistance, inductance, voltage, current, and gear ratio, respectively. The back emf \( e_a \) is proportional to the rotational velocity \( \omega_m \) with back emf constant \( K_b \). Moreover, the motor torque \( \tau_m \) is proportional to the armature current \( i_a \) with constant \( K_m \) and the output torque \( \tau \) is proportional to \( \tau_m \) with gear ratio \( N \). Based on Eq. (5), an actuator model can be obtained. In the subsystem, the two respective inputs are voltage \( V_{al} \) of the left motor and \( V_{ar} \) for the right motor. Here, \( \tau_r \) and \( \tau_l \) stand for torques of the left and right motors which are the outputs of the actuator model.

From the kinematic, dynamic, and actuator models, a comprehensive model is built for subsequently designing the trajectory-tracking controller. In addition, \( \tau_{dr} \) and \( \tau_{dl} \) which denote torque disturbances, are added to the dynamic model to investigate the robustness of the closed-loop control system when the proposed controller is applied.

### C. PROBLEM FORMULATION

The objective of the proposed FSMC is to control the ADDMR from an arbitrary initial position along a desired trajectory in the presence of torque disturbances \( \tau_{dr} \) and \( \tau_{dl} \). The pose information obtained by an MCL technique using LiDAR information to form nonlinear sliding surfaces. Then, the nonlinear sliding surfaces are reasoned to satisfy the reaching law (i.e., \( \sigma \dot{\sigma} < 0 \), where \( \sigma \) is a sliding surface) via the designed fuzzy inference rules. When control law is
applied to the ADDMR system, the tracking errors asymptotically converge to zero. The overall control block diagram is shown in Fig. 3.

III. TRAJECTORY PLANNING, POSE MEASUREMENT, AND FSMC DESIGN
A. TRAJECTORY PLANNING
It is common in the robotics community to differentiate between path and trajectory. A path is a spatial function that connects an initial pose to a final one where as a trajectory is a path which is a function of time \[50\]. In this work, a path is a spatial function of arc length \(s\). In this work, a path is a spatial function of arc length \(s\) as a trajectory is a function of time \(s\). One of the well-known ways to generate paths is Bézier curves where the path is described using a parametric equation. The \(n + 1\) parameter points \(P_i\) with \(i = 0, 1, \ldots, n\) are used to form an \(n\)th-order Bézier curve \(B(s)\) as:

\[
B(s) = \sum_{i=0}^{n} P_i B_i^n(s), \quad s \in [0, 1] \tag{6}
\]

where \(B_i^n(s)\) is a Bernstein polynomial given by

\[
B_i^n(s) = \binom{n}{i} s^i (1 - s)^{n-i}, \quad i = 0, 1, 2, \ldots, n \tag{7}
\]

and \(\binom{n}{i}\) is a polynomial coefficient given by

\[
\binom{n}{i} = \frac{n!}{i!(n-i)!} \tag{8}
\]

Because a Bézier curve only goes through the first and the last parameter points, the other points are used to determine the shape of the curve. These points \(V_i\) are spaced points in an equally divided arc length such that

\[
V_i = B(i/n), \quad i = 0, 1, 2, \ldots, n \tag{9}
\]

This yields a Bézier curve that will always go through all the \(V_i\). The polynomial form of the Bézier curve in Eq. (6) can be expanded into a matrix form of

\[
A(s) = \begin{pmatrix} s^0 & s^1 & \ldots & s^n \end{pmatrix}
\begin{pmatrix}
B_0^n & B_1^n & \ldots & B_n^n \\
B_0^{n-1} & B_1^{n-1} & \ldots & B_n^{n-1} \\
\vdots & \vdots & \ddots & \vdots \\
B_0^n & B_1^n & \ldots & B_n^n
\end{pmatrix}
\begin{pmatrix}
P_0 \\
P_1 \\
\vdots \\
P_n
\end{pmatrix} = \begin{pmatrix}
B_0(1/n) \\
B_1(1/n) \\
\vdots \\
B_n(1/n)
\end{pmatrix}
\begin{pmatrix}
P_0 \\
P_1 \\
\vdots \\
P_n
\end{pmatrix}
\]

where \(A(s)\) is an \(1 \times (n + 1)\) column vector that maps parameter points \(P_i\) to Bézier points at arc length \(s\). The corresponding \(P_i\) to the designed \(V_i\) can be calculated by expanding (9) as follows:

\[
\begin{pmatrix}
V_0 \\
V_1 \\
\vdots \\
V_n
\end{pmatrix} = \begin{pmatrix}
B_0(1/n) \\
B_1(1/n) \\
\vdots \\
B_n(1/n)
\end{pmatrix}^{-1} \begin{pmatrix}
P_0 \\
P_1 \\
\vdots \\
P_n
\end{pmatrix}
\]

To form a smooth trajectory from the planned Bézier curve path, a time mapping function of the arc length \(s(t)\) is designed using a cubic polynomial as follows:

\[
s(t) = a_0 t^3 + a_1 t^2 + a_2 t + a_3 \tag{12}
\]

\[
\dot{s}(t) = 3a_0 t^2 + 2a_1 t + a_2
\]

where the coefficients \(a_i, i \in \{0, 1, 2, 3\}\) are calculated based on the intended initial and final conditions of \(s\) and \(s\). By combining the path with a cubic time mapping, the desired trajectory is obtained for tracking control by the ADDMR system.

B. POSE MEASUREMENT
To localize and control the ADDMR, its pose measurement for feedback is required. In the experiment, in addition to odometry, MCL is used to improve the accuracy of the pose measurement via a LiDAR. The MCL technique is inspired by a recursive Bayes filter based on importance sampling to estimate probability of robot poses using sensory measurement data \[51\]. The states of the robot can be defined in a probabilistic approach with the help of two distribution models i.e. motion and observation models \[52\]. The motion model \(p(x_t | x_{t-1}, u_t)\) represents the transition of probability distribution of state \(x\) at time \(t\) based on the
previous state $x_{t-1}$ and the control input $u_t$. The observation model $p(z_t \mid x_t)$ incorporates readings of sensors $z_t$ to calculate the likelihood. The whole MCL algorithm is described in Table 1. The continuous belief of states $bel(x_t)$ in MCL is represented by a discrete number of $M$ particles $X_t = \{x^{(i)}, w^{(i)}\}_{i=1}^{M}$ [53]. The weight $w^{(i)}$ is approximately proportional to the probability of robot at states $x_t^{(i)}$. The initial belief $bel(x_0)$ is uniformly distributed. Hence, the initial weights $w_0$ are also uniform and the states $x_0$ are initially sampled from all possible state values with uniform distribution [54]. Line 4 in Table 1 propagates the particles using the system dynamics to generate hypothetical state $x_t$. This process represents a sampling from distribution $p(x_t \mid x_{t-1}, u_t)$. Line 5 in Table 1 calculates the likelihood representing measurement probability $z_t$ from particle $x_t^{(i)}$ that is $p(z_t \mid x_t^{(i)})$. In this study, LiDAR beams are utilized to measure the likelihood of each particle. Line 6 in Table 1 assigns the states and likelihoods to a temporary particles $\hat{X}_t$. Line 8 in Table 1 commonly referred to as re-sampling step in which $j \in [1, M]$ integers are drawn proportionally to the likelihoods $w_t$. The newly selected particles are assigned to the reference of $j$ in Line 10. Particles with bigger likelihood are more likely to be drawn multiple times. Hence, more particles are distributed to the most probable states.

**TABLE 1. Monte Carlo Localization algorithm.**

| Line | Description |
|------|-------------|
| 1    | Input: $X_{t-1}$, $u_t$, $z_t$ |
| 2    | Output: $\hat{X}_t$ |
| 3    | for $i = 1$ to $M$ do |
| 4    | $x_t^{(i)} = \text{motionModel}(x_{t-1}^{(i)}, u_t)$ |
| 5    | $w_t^{(i)} = \text{observationModel}(z_t, x_{t-1}^{(i)})$ |
| 6    | $\hat{X}_t = \text{temporaryParticle}(x_{t-1}^{(i)}, w_t^{(i)})$ |
| 7    | endfor |
| 8    | draw new indices $j = \{1, 2, \ldots, M\}$ with probability $\propto w_t$ |
| 9    | for $j = 1$ to $M$ do |
| 10   | assign $x_j^{(i)}$ to $\hat{X}_t$ |
| 11   | endfor |

Here, a simulated example is displayed. The result shown in Fig. 4 indicates particles (blue dots) convergence history. The robot moves in a counterclockwise direction and the position is marked in green dashed line as a ground truth. The mean position of particles is taken as a predicted location and marked by an orange line. As seen from the figure, the particles gradually concentrate and converge to the real location of the robot.

### C. FUZZY SLIDING-MODE CONTROL DESIGN

Based on the established comprehensive model and localization of the ADDMR, the proposed FSMC is designed to control an ADDMR to track the desired trajectory. The proposed FSMC has two inputs; i.e., heading error $e_\theta$ and position error $e_p$, and two outputs; i.e., $V_{al}$ and $V_{ar}$. The details are further described in the following subsections.

1) Nonlinear sliding surfaces design

Prior to designing the FSMC, nonlinear sliding surfaces are first designed. The distance error $e_p$ and heading error $e_\theta$ information are defined to construct nonlinear sliding surfaces. To clearly understand how both errors are calculated, a geometrical schematic shown in Fig. 5 describes the relation between the inertial frame $\{I\}$, robot frame $\{B\}$, and the planned Bezier trajectory $T = B(s(t))$. Vector $^I\!P_B$ and $^I\!P_T$ are the robot position and trajectory position relative to the inertial frame. The trajectory position relative to the body frame is denoted by $^B\!P_T$. In this way, it can be calculated by
utilizing homogeneous transformation matrix $^B\mathbf{T}_p \in \text{SE}(2)$

$$\begin{bmatrix} ^B\mathbf{P}_T \\ 1 \end{bmatrix} = ^B\mathbf{T}_p^{-1} [^I\mathbf{P}_T]$$

$$= \begin{bmatrix} \cos \theta & -\sin \theta & ^I\mathbf{x}_p \\ \sin \theta & \cos \theta & ^I\mathbf{y}_p \\ 0 & 0 & 1 \end{bmatrix}^{-1} \begin{bmatrix} ^I\mathbf{x}_T \\ ^I\mathbf{y}_T \\ 1 \end{bmatrix}$$

(13)

and the position error is the magnitude of $^B\mathbf{P}_T$ as follows:

$$e_p = \sqrt{{^B\mathbf{x}_T}^2 + {^B\mathbf{y}_T}^2}$$

(14)

Using Eq. (13), the heading error $e_{\theta}$ is

$$e_{\theta} = \text{atan2}(^B\mathbf{y}_T, ^B\mathbf{x}_T)$$

(15)

where $\text{atan2}(\cdot)$ is a full-quadrant inverse tangent with output between $[-\pi, \pi]$ [55]–[58]. The function $\text{atan2}(\cdot)$ is preferred over the $\tan^{-1}(\cdot)$ function for converting Cartesian coordinates to their corresponding polar coordinates.

The control objective is to make $e_p$ and $e_{\theta}$ asymptotically converge to zero. Firstly, the following nonlinear sliding surfaces $s_p$ and $s_{\theta}$ are constructed by tracking errors are designed. Here, the sliding surface $s_p$ corresponds to trajectory tracking and $s_{\theta}$ corresponds to robot’s heading tracking represented as

$$s_p = e_p + k_p e_p + f_p \text{sign}(e_p)$$

$$s_{\theta} = e_{\theta} + k_{\theta} e_{\theta} + f_{\theta} \text{sign}(e_{\theta})$$

(16)

where $k_p$, $k_{\theta}$, $f_p$, and $f_{\theta}$ are positive constants and $\text{sign}(\cdot)$ is a signum function [13], [15], [21], [26], [29], [31], [32], [35], [36]. Eq. (16) can be proved that once the sliding surfaces approach to zero, the corresponding tracking errors also approach to zero. Then, the fuzzy logic is proposed to regulate nonlinear sliding surfaces towards zero and the details are described in next subsection.

2) Fuzzy inference logic design

In fuzzy inference logic design, the sliding surfaces are inputs and the actuator voltages are outputs to control the left and right wheels. In the beginning, sliding surfaces are normalized using gains $k_{np}$ and $k_{n\theta}$ yielding $\hat{s}_p = k_{np}s_p$ and $\hat{s}_{\theta} = k_{n\theta}s_{\theta}$. The normalized values are then evaluated using triangular fuzzy membership functions shown in Fig. 6. For each sliding surface, there are 3 membership functions comprising $\{N : \text{Negative}, Z : \text{Zero}, P : \text{Positive}\}$. In addition to the fuzzified inputs, this work also uses fuzzified voltage outputs for each actuator. The fuzzification of the outputs are the triangular type with 3 linguistic voltage representations $\{N : \text{Negative}, Z : \text{Zero}, P : \text{Positive}\}$ shown in Fig. 7. Then, fuzzy rules are designed in the inference engine as follows:

$$R_1 : \text{if}(\hat{s}_p = N \text{ AND } \hat{s}_{\theta} = N) \text{ then}(V_{f_r} = N, V_{f_l} = P)$$

$$R_2 : \text{if}(\hat{s}_p = N \text{ AND } \hat{s}_{\theta} = Z) \text{ then}(V_{f_r} = P, V_{f_l} = P)$$

$$\vdots$$

$$R_9 : \text{if}(\hat{s}_p = P \text{ AND } \hat{s}_{\theta} = P) \text{ then}(V_{f_r} = N, V_{f_l} = P)$$

(17)

The t-norm $\text{AND}$ operator in this work is $\text{MIN}(\cdot)$. The consequence of every rule is aggregated using t-conorm $\text{MAX}(\cdot)$ function as

$$R = \bigcup_{i=1}^{9} R_i$$

(18)

The complete list of the rules is displayed in the following Table 2. A crisp value from each fuzzified voltage is obtained by means of the centroid method expressed as follows:

$$V_r = \begin{cases} \sum_{V_{f_r}} V_{f_r} \mu_{f_r}(V_{f_r}) \\ \sum_{V_{f_r}} \mu_{f_r}(V_{f_r}) \end{cases}$$

(19)

Based on the proposed FSMC design, discussion of the corresponding simulation and experiment results is presented in the next section.
IV. DISCUSSION OF SIMULATION AND EXPERIMENT RESULTS

The proposed method applied to the ADDMR system is tested to follow the planned Bézier trajectory. The trajectory is designed using the following via points as

\[
V = \begin{pmatrix}
0 & 0 \\
0.5 & -0.5 \\
0 & -1.0 \\
-0.5 & -1.5 \\
0 & -2.0 \\
0.5 & -2.5 \\
0 & -3.0 \\
-0.5 & -3.5 \\
0 & -4.0 \\
\end{pmatrix}
\]

(20)

The via points are chosen to mimic a sinusoidal path. Using Eq. (11), we find the parameter points and the corresponding matrix form of the Bézier curve as follows:

\[
B(s) = (s^0 \ s^1 \ldots \ s^8) \cdot
\begin{pmatrix}
1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
-8 & 8 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
28 & -56 & 28 & 0 & 0 & 0 & 0 & 0 & 0 \\
-56 & 168 & -168 & 56 & 0 & 0 & 0 & 0 & 0 \\
70 & -280 & 320 & \ldots & \ldots & \ldots & \ldots & \ldots & \ldots \\
-56 & 280 & -560 & \ldots & \ldots & \ldots & \ldots & \ldots & \ldots \\
28 & -168 & 420 & \ldots & \ldots & \ldots & \ldots & \ldots & \ldots \\
-8 & 56 & -168 & \ldots & \ldots & \ldots & \ldots & \ldots & \ldots \\
1 & -8 & 28 & -56 & 70 & -28 & 28 & -8 & 1 \\
\end{pmatrix}
\]

(21)

The resulting path can be seen in Fig. 8. To follow a speed variation in the trajectory, the path is designed to end at 60 s with a nonlinear arc length function as follows:

\[
\begin{align*}
\tilde{s}(t) &= -9.3 \cdot 10^{-6}t^3 + 8.3 \cdot 10^{-4}t^2 - 3.2 \cdot 10^{-18}t \\
\dot{\tilde{s}}(t) &= -27.9 \cdot 10^{-6}t^2 + 16.6 \cdot 10^{-4}t - 3.2 \cdot 10^{-18}
\end{align*}
\]

(22)

and the evolution of \(s(t)\) and \(\dot{s}(t)\) can be seen in Fig. 9. In practical situation, there are some torque disturbances generated from frictions or uneven floors. In order to investigate this effect, friction torque \(\tau_f\) and uncertainty torque \(\tau_u\) are applied to the ADDMR control system. Both disturbances are formulated as follows:

\[
\begin{align*}
\tau_d &= \tau_f + \tau_u \\
\tau_f &= T_v \dot{\varphi} + \left[ T_c + (T_s - T_c) e^{-\frac{\varphi}{\varphi_s}} \right] \text{sign} (\dot{\varphi}) \\
\tau_u &= [0.1 + 0.1 \sin(10\pi t) + 0.01 \cos(62\pi t)] \text{sign}(\dot{\varphi})
\end{align*}
\]

(23)

The friction torque is based on friction model with stribeck effect, where \(T_v, T_c, T_s, \text{ and } \varphi_s\) are viscous friction, coulomb friction, stick friction, and stribeck velocity constants, respectively. The friction model gives variation of disturbance torque with respect to angular velocity of the wheels \(\dot{\varphi}\). Furthermore, the uncertainty torque gives variation of disturbance torque with respect to time which includes low and high frequency components. The system and control parameters in the simulation are listed in Table 3 and the illustration of the disturbance is shown in Fig. 10. During simulations and experiments, the ADDMR is initially placed outside the path. The initial positions for each case are \((x_{ci}, y_{ci}) = \{(-1.2, 0.8), (0.5, 1), (-0.5, 1.5)\}\) and the initial orientation is \(\theta_{ci} = 90^\circ\) for all cases.

| Parameter | Value | Unit |
|-----------|-------|------|
| \(R\)     | 40    | \(\text{mm}\) |
| \(L\)     | 25    | \(\text{cm}\) |
| \(K_p\)   | \(0.93\) | \(\text{V/\text{rad/s}}\) |
| \(K_t\)   | \(0.03\) | \(\text{N/\text{rad/s}}\) |
| \(M\)     | \(8.5\) | \(\text{kg}\) |
| \(J\)     | \(0.098\) | \(\text{kgm}^2\) |
| \(T_v\)   | 0.058 | \(\text{N/\text{rad/s}}\) |
| \(T_s\)   | 0.120 | \(\text{Nm}\) |
| \(\varphi_s\) | 0.04 | \(\text{rad/s}\) |

It is clearly seen from Fig. 11 that the proposed FSMC can achieve excellent trajectory-tracking of the ADDMR despite...
the presence of disturbance torques. Similarly, the orientation tracking is also good and displayed in Fig. 12 where the rise time of $e_p$ is 2 s and the rise time of $e_\theta$ is 1.26 s. Moreover, Fig. 13 shows that the sliding surface of case 1 converges to zero as the robot successfully follows the planned trajectory. The provided simulations validate that the proposed method is capable of controlling the ADDMR to follow the planned trajectory under torque disturbances.

In the experiments, the proposed FSMC is implemented on ADLINK NeuronBot. The robot follows the same trajectory at the same initial conditions as the simulations. The implementation result can be seen in Fig. 14 where the trajectory-tracking is satisfactory despite the presence of the floor disturbances. Compared to the simulations shown in Fig. 11, the curvatures are smaller when approaching the desired trajectory. This is because the robot operates indoors and its speed limit is needed for safety consideration. As seen in Fig. 15, the linear velocity of the robot does not exceed 0.4 m/s. As a result, $e_\theta$ is reduced much faster than $e_p$ as shown in Fig. 16 where the robot made a sharper move compared to the simulations. Furthermore, it can be seen from error responses that the robot completes the trajectory-tracking in a specified time.
V. CONCLUSIONS

In this study, the FSMC technique is proposed to control an ADDMR subject to torque disturbances. In addition, the comprehensive model of the ADDMR consisting of kinematics, dynamics, and actuator dynamics is established. The trajectory is planned via Bézier curve. In addition, the pose measurement of the ADDMR is measured through MCL technique based on a LiDAR point cloud data. The numerical and experimental validation results prove the effectiveness and robustness of the proposed control scheme.

REFERENCES

[1] H.-H. Chang and C. D. Meyerhoefer, “Covid-19 and the demand for online food shopping services: Empirical evidence from Taiwan,” American Journal of Agricultural Economics, vol. 103, no. 2, pp. 448–465, 2021.

[2] J. Grashuis, T. Skevas, and M. S. Segovia, “Grocery shopping preferences during the covid-19 pandemic,” Sustainability, vol. 12, no. 13, p. 5369, 2020.

[3] A. Bolu and O. Koçak, “Adaptive task planning for multi-robot smart warehouse,” IEEE Access, vol. 9, pp. 27 346–27 358, 2021.

[4] M.-K. Ng, Y.-W. Chong, K.-m. Ko, Y.-H. Park, and Y.-B. Leau, “Adaptive path finding algorithm in dynamic environment for warehouse robot,” Neural Computing and Applications, vol. 32, no. 17, pp. 13 155–13 171, 2020.

[5] A. Pani, S. Mishra, M. Goliás, and M. Figliozzi, “Evaluating public acceptance of autonomous delivery robots during covid-19 pandemic,” Transportation research part D: transport and environment, vol. 99, p. 102600, 2020.

[6] Y. Shen, D. Guo, F. Long, L. A. Mateos, H. Ding, Z. Xiu, R. B. Hellman, A. King, S. Chen, C. Zhang et al., “Robots under covid-19 pandemic: A comprehensive survey,” IEEE Access, pp. 1590–1615, 2020.

[7] Y.-H. Lee, S.-G. Kim, T.-Y. Kue, J.-K. Park, S.-H. Ji, Y.-S. Moon, and Y.-J. Cho, “Virtual target tracking of mobile robot and its application to formation control,” International Journal of Control, Automation and Systems, vol. 12, no. 2, pp. 390–398, 2014.

[8] A. Morro, A. Sgorbissa, and R. Zaccaria, “Path following for unicycle robots with an arbitrary path curvature,” IEEE Transactions on Robotics, vol. 27, no. 5, pp. 1016–1023, 2011.

[9] F. Heidari and R. Fotouhi, “A human-inspired method for point-to-point and path-following navigation of mobile robots,” Journal of Mechanisms and Robotics, vol. 7, no. 4, pp. 041025, 2015.

[10] B. Anushree and B. S. Prasad, “Design and development of novel control strategy for trajectory tracking of mobile robot: Featured with tracking error minimization,” in 2016 IEEE Annual India Conference (INDICON). IEEE, 2016, pp. 1–6.

[11] M. Asai, G. Chen, and I. Takami, “Neural network trajectory tracking of tracked mobile robot,” in 2019 16th International Multi-Conference on Systems, Signals & Devices (SSD). IEEE, 2019, pp. 225–230.

[12] N. Y. Allagui, D. B. Abid, and N. Derbel, “Autonomous navigation of mobile robot with combined fractional order pi and fuzzy logic controllers,” in 2019 16th International Multi-Conference on Systems, Signals & Devices (SSD). IEEE, 2019, pp. 78–83.

[13] Y. Su and Y. Liang, “Research on double closed-loop trajectory tracking control algorithm of wheeled mobile robot based on global stability,” in 2020 12th International Conference on Intelligent Human-Machine Systems and Cybernetics (IHMSC), vol. 1, 2020, pp. 180–183.

[14] M. Faisal, M. Alghabri, B. M. Abdelkader, H. Dhahri, and M. M. Al Rahhal, “Human expertise in mobile robot navigation,” IEEE Access, vol. 6, pp. 1694–1705, 2018.

[15] D. Chwa and J. Boo, “Adaptive fuzzy output feedback simultaneous posture stabilization and tracking control of wheeled mobile robots with kinematic and dynamic disturbances,” IEEE Access, vol. 8, pp. 228 863–228 878, 2020.

[16] M. H. Falsafi, K. Alipour, and B. Tarvirdizadeh, “Tracking-error fuzzy-based control for nonholonomic wheeled robots,” Arabian Journal for Science and Engineering, vol. 44, no. 2, pp. 881–892, 2019.
MUHAMMAD QOMARUZ ZAMAN received bachelor’s and master’s degrees in electrical engineering from Institut Teknologi Sepuluh Nopember, Surabaya, Indonesia, in 2018. From 2015 to 2018, he was a Research Assistant in Microcontroller Laboratory at Institut Teknologi Sepuluh Nopember. His research interests include robotics and IoT. He is currently pursuing a Ph.D. degree in National Taipei University of Technology.

HSIU-MING WU received Ph.D. degree from department of electrical engineering, National Taiwan University of Science and Technology in 2009. Next, he served the Opto-Mechatronics Technology center, National Taiwan University of Science and Technology as a postdoctoral fellow. In early 2013, he worked in Texas A&M University at Qatar as a postdoctoral fellow for 4 years. In early 2018, he joined in the department of mechanical engineering at National Chin-Yi University of Technology in Taichung for two years. Currently, he is an assistant professor in the department of intelligent automation engineering, National Taipei University of Technology. His research interests include mechatronics, nonlinear control, and robotic control.