Synthesis of Active Disturbance Rejection Controller via Extended State Observer Combined with LQR Controller for Two-Wheeled Line Tracking Robot

Nguyen Xuan Chiem [1], Nguyen Hoai Nam [2], Le Tran Thang [3*]
[1, 2] Department of Automation and Computing Techniques, Le Quy Don Technical University, Hanoi, Vietnam
[3] Controls, Automation in Production and Improvement of Technology Institute, Hanoi, Vietnam
Email: [1] chiemnx@mta.edu.vn, [2] namcoi12122001@gmail.com, [3] ltranthang@gmail.com
*Corresponding Author

Abstract—This paper presents a method of synthesizing control laws based on the LQR controller and ADRC method for a two-wheel differential line-following robot when the robot dynamics have uncertain factors. First, the mathematical model includes line-following kinematic and dynamic models. LQR controller is designed based on the linear model of the robot when coincident with the line. When the robot has uncertain factors such as model parameter uncertainty and impact noise, the LQR controller will not ensure the control quality of the system. To overcome this, two observers are designed to observe the linear velocity and angular velocity states of the robot. This ensures more complete and accurate information of the model states in the LQR control law. The effectiveness of the control law is demonstrated through numerical simulation results and compared with the LQR controller.

Keywords—*LQR Controller; ADRC Method; ESO; Mobile Robot; Track Following.*

I. INTRODUCTION

Differential 2-wheel robots have been widely used in various applications and environments due to their mobility and energy-saving characteristics. One of the main research directions of interest to researchers is the development of indoor autonomous robots such as factory robots, medical transport robots, delivery robots, and service robots [1]-[6]. These mobile robots all move along desired trajectories to perform required tasks. Therefore, the trajectory tracking control problem is an important issue in control. The content of this problem is how to generate the necessary linear velocity and angular velocity to allow the robot to follow a predetermined trajectory. That is, the error between the desired trajectory and the actual trajectory converges to zero [1]-[20].

The controller designed for Two-Wheeled Robot to follow a predefined path, as in many studies [5]-[17], can be broadly defined into two types: kinematic model-based and dynamic model-based. In the study [7]-[10], a channel-by-channel control method based on the PID controller was proposed. The results showed that the response of the robot's channels was quite good but did not take into account the cross-channel effects and external disturbances. The control method based on the LQR set was presented in the studies [12][13]. This controller can select the preferred response for

the robot states based on the choice of Q matrix and R matrix parameters. But using a linear model when the robot system has uncertain factors will reduce control quality or lose stability of the robot control system. To overcome the nonlinearity of the robot dynamics model, studies [18]-[27] have presented methods to design nonlinear control laws for this system. Studies [6][17][19][22] have generated desired trajectories based on Lyapunov functions and proposed a sliding mode control (SMC) method to control the robot on the desired path. The backstepping method is presented in studies [7][26][34][35] to design adaptive controllers for non-quadratic systems to deal with uncertain parameters. In studies [40]-[43] a hybrid fuzzy controller and its variants are presented. Controllers combined with neural networks are presented in studies [46][47]. The studies all give good results but do not take into account the input signals on real robots and the complexity of implementation on real models.

One trend in controller design for objects with model uncertainty and impact disturbances is to use ADRC [48]-[60]. ADRC is a robust control strategy, where the system model is extended with a new state boundary, which includes all the unknown dynamics and residual convolutions that are not considered in the conventional model. The direct estimation of the new state is performed using an Extended State Observer (ESO). This observer is responsible for monitoring and estimating the direct impact disturbances, and the errors of the modeled object compared to reality. In this way, although we have only a model with low accuracy, we can still design a good quality controller, robust against fluctuations for the real object, thereby indirectly simplifying the model. Any difference in the model will not affect the control mechanisms, including all the uncertainties in the extended state variables. In the studies [47] ADRC controller with PID feedback controller for line-following robot gives quite good results. However, there are still limitations of model-free controllers.

This paper studies the design of an ADRC controller th`rough two observers of linear velocity and angular velocity states, combining the LQR controller for mobile robots. In order to maintain the advantages of the LQR controller and overcome its disadvantages, such as being based on a linear model at a working point when robot dynamics have uncertain factors, simulation results and comparisons with the LQR control law show the superiority

of this method. In addition, the authors also implemented the controller on a real robot system to demonstrate the effectiveness of the proposed control law. This paper is organized as follows: Section 2 presents the kinematic and dynamic model of the Two-Wheeled Robot Line Tracking Robot. The synthesis of the nonlinear controller is presented in Section 3. Section 4 presents the simulation results on Matlab software and the experimental results on a real robot. Conclusions and future research directions are described in Section 5.

II. THE KINEMATIC AND DYNAMIC MODEL OF A TWO-WHEEL DIFFERENTIAL LINE-FOLLOWING ROBOT

To describe the behavior of a two-wheeled differential line tracking robot, it is necessary to define two different reference frames: the $Oxyz$ robot coordinate system and the $O'x'y'$ inertial coordinate system. Fig. 1 is a simplified schematic illustration of a Two-Wheeled Robot Line Tracking Robot in practice [1]-[3]. The figure shows and names the important elements of the model, as well as the forces acting on the robot, considered: an axial force and a horizontal force on each drive wheel (the forces acting on the free wheels are considered negligible). The curve $\sigma(q)$ represents the line that the robot needs to follow.
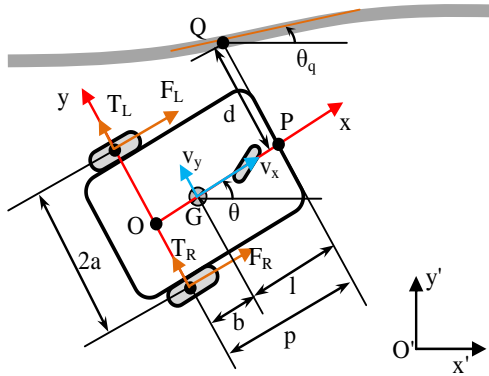

Fig. 1. Diagram of the robot and its path

A. Non-holonomic Constraints

The motion of a differentially driven mobile robot is characterized by non-holonomic constraint equations, which are achieved by several assumptions and constraints that must be defined before building the kinematic and dynamic model of the robot: (1) The motion of the robot on a flat surface means zero potential energy; (2) The rolling constraint is completely slip-free: then each wheel touches the ground at a single point. The mobile robot frame depends on three constraint equations. The first constraint is that the robot cannot move in a yaw direction:

$$-\dot{x}'\sin\theta + \dot{y}'\cos\theta = 0 \tag{1}$$

In which $\omega = \dot{\theta}$. The remaining two constraints are two non-slip rollers:

$$\begin{aligned}\dot{x}'\cos\theta + \dot{y}'\sin\theta + b\dot{\theta} - R\dot{\varphi}_R = 0 \\ \dot{x}'\cos\theta + \dot{y}'\sin\theta - b\dot{\theta} - R\dot{\varphi}_L = 0\end{aligned} \tag{2}$$

B. Track Following

As shown in Fig. 1, the parameter d represents the distance between the tracking point (P) and the point being tracked (Q). Equation (3) shows that the position of point Q can be determined by its components in the x'O'y' coordinate system, as well as by its belonging to the curve $\sigma(q)$.

$$\begin{pmatrix} x'_Q \\ y'_Q \end{pmatrix} = \begin{pmatrix} \sigma_{x'}(q) \\ \sigma_{y'}(q) \end{pmatrix} \tag{3}$$

Using the rotation matrix and known shape measurements of the vehicle, point Q can be placed relative to the center of mass as follows.

$$\begin{pmatrix} x'_Q \\ y'_Q \end{pmatrix} = \begin{pmatrix} x'_G \\ y'_G \end{pmatrix} + R(\theta)\begin{pmatrix} p \\ d \end{pmatrix} \tag{4}$$

With $R(\theta) = \begin{pmatrix} \cos\theta & -\sin\theta \\ \sin\theta & \cos\theta \end{pmatrix}$.

From equation (4), we have:

$$\begin{pmatrix} \dot{x}'_G \\ \dot{y}'_G \end{pmatrix} + \frac{d}{dt}R(\theta)\begin{pmatrix} p \\ d \end{pmatrix} + R(\theta)\begin{pmatrix} 0 \\ \dot{d} \end{pmatrix} = \dot{q}\begin{pmatrix} \frac{\partial}{\partial q}\sigma_{x'}(q) \\ \frac{\partial}{\partial q}\sigma_{y'}(q) \end{pmatrix} \tag{5}$$

Since the coordinate system x'O'y' is fixed, or $\dot{x}'_G = v_{x'}, \dot{y}'_G = v_{y'}, \dot{\theta} = \omega$. Equation (5) can be expressed as:

$$\begin{pmatrix} v_{x'} \\ v_{y'} \end{pmatrix} + \frac{d}{dt}R(\theta)\begin{pmatrix} p \\ d \end{pmatrix} + R(\theta)\begin{pmatrix} 0 \\ \dot{d} \end{pmatrix} = \dot{q}\begin{pmatrix} \cos\theta_q \\ \sin\theta_q \end{pmatrix} \tag{6}$$

Transforming equation (6) we get as follows:

$$\begin{cases} \dot{q} = \dfrac{v_{x'}\cos\theta + v_y\sin\theta - \omega d}{\cos\theta_e} \\ \dot{d} = v_{x'}\sin\theta - v_{y'}\cos\theta - \omega p + \dot{q}\sin\theta_e \end{cases} \tag{7}$$

In which $\theta_e = \theta - \theta_q$.

Since equations (7) are represented in the x'O'y' coordinate system, the rotation matrix $R(\theta)$ is used to find the expression of $\dot{d}$ and $\dot{q}$ in the xOy coordinate system with $\begin{bmatrix} v_{x'} & v_{y'} \end{bmatrix}^T = R^{-1}(\theta)\begin{bmatrix} v_x & v_y \end{bmatrix}^T$, we get the following result:

$$\begin{cases} \dot{d} = -v_y - \omega p - \tan\theta_e(v_x - \omega d) \\ \dot{q} = \dfrac{v_x - \omega d}{\cos\theta_e} \end{cases} \tag{8}$$

Since the robot does not slip, the relationship between v_y and ω in (2) and using the parametric relationship $l = p - b$, we have:

$$\dot{d} = -l\omega - \tan\theta_e(v_x - \omega d) \tag{9}$$

Derivative θ_e, we get.

$$\dot{\theta}_e = \dot{\theta} - \dot{\theta}_q = \dot{\theta} - c(q)\dot{q} \tag{10}$$

Where $c(q) = \partial\theta_q / \partial q$ is the curvature of the path for every q. Substituting in equation (8) we get (11):

$$\dot{\theta}_e = \omega - c(q)\frac{v_x - \omega d}{\cos\theta_e} \quad (11)$$

C. Dynamic Model

When considering the robot as a rigid body, its dynamics can be studied by applying the principles of Conservation of Momentum and Conservation of Angular Momentum. The velocities and accelerations have been calculated in the general reference frame fixed on the ground $x'O'y'$, but the vectors are represented in the mobile xOy frame, which is attached to the robot and has its angular velocity (Fig. 1).

Applying the law of conservation of momentum, the following equations can be found:

$$\begin{aligned} F_R + F_L &= ma_x \\ T_R + T_L &= ma_y, \end{aligned} \quad (12)$$

where m (kg) is the mass of the robot, a_x and a_y are the corresponding acceleration components on the xOy coordinate system. The acceleration vector $a = (\,a_x,\,a_y,\,0)$ and the velocity vector $v =(\,v_x,\,v_y,\,0)$ are represented in the xOy coordinate system in the form:

$$\begin{pmatrix} a_x \\ a_y \\ 0 \end{pmatrix} = \dot{v} = \begin{pmatrix} \dot{v}_x \\ \dot{v}_y \\ 0 \end{pmatrix} + \begin{pmatrix} 0 \\ 0 \\ \dot{\theta} \end{pmatrix} \wedge \begin{pmatrix} v_x \\ v_y \\ 0 \end{pmatrix} = \begin{pmatrix} \dot{v}_x - \dot{\theta}v_y \\ \dot{v}_y + \dot{\theta}v_x \\ 0 \end{pmatrix} \quad (13)$$

and replaced according to (13), equation (12) can be expressed as follows

$$\begin{aligned} m\left(\dot{v}_x - \omega v_y\right) &= (F_R + F_L) \\ m\left(\dot{v}_y + \omega v_x\right) &= (T_R + T_L) \end{aligned} \quad (14)$$

Applying the Theorem of Conservation of Angular Momentum about the center of mass, we have the following equation of rotation of the robot:

$$I_Z\dot{\omega} = b(T_R + T_L) + a(F_R - F_L) \quad (15)$$

where I_z is the robot's moment of inertia about the longitudinal axis.

Since there is no slip, the only longitudinal forces present in the model are those generated by the electric motors driving each wheel. Under the assumption that the two wheels and their motors are identical and the wheels have a radius r (m). If the moments of inertia of the wheels are negligible, the dynamics of the wheel-motor assembly can be described as follows:

$$\begin{aligned} F_R &= \frac{\tau_R}{r} - \frac{B_f}{r}\omega_R \\ F_L &= \frac{\tau_L}{r} - \frac{B_f}{r}\omega_L \end{aligned} \quad (16)$$

where B_f is the total viscous friction coefficient between the rotor and the wheel. Since slip is not considered, the angular velocity of each wheel can be calculated from the velocity of the wheel center in the x direction, i.e.:

$$\omega_R = \frac{v_{R,x}}{r}; \quad \omega_L = \frac{v_{L,x}}{r} \quad (17)$$

Since the robot is considered a rigid body, the velocity of the wheels is related to the velocity at the center of mass and the angular velocity of the robot:

$$v_R = \begin{pmatrix} v_{R,x} \\ v_{R,y} \\ 0 \end{pmatrix} = \begin{pmatrix} v_x + \omega a \\ v_y + \omega b \\ 0 \end{pmatrix} \quad v_L = \begin{pmatrix} v_{L,x} \\ v_{L,y} \\ 0 \end{pmatrix} = \begin{pmatrix} v_x - \omega a \\ v_y - \omega b \\ 0 \end{pmatrix} \quad (18)$$

Substituting (18), and (17) into (16), the vertical force of the robot can be expressed as (19).

$$\begin{aligned} F_R &= \frac{\tau_R}{r} - \frac{B_f}{r^2}(v_x + \omega a) \\ F_L &= \frac{\tau_L}{r} - \frac{B_f}{r^2}(v_x - \omega a) \end{aligned} \quad (19)$$

Substituting into equations (14) and (15) we have:

$$\begin{cases} m(\dot{v}_x - \omega v_y) = \dfrac{1}{r}(\tau_R + \tau_L) - 2\dfrac{B_f}{r^2}v_x \\ m(\dot{v}_y + \omega v_x) = (T_R + T_L) \\ I_z\dot{\omega} = b(T_R + T_L) + \dfrac{a}{r}(\tau_R - \tau_L) - 2\dfrac{B_f a^2}{r^2}\omega \end{cases} \quad (20)$$

T_R and T_L are the forces that prevent the wheels from moving on the y-axis, but their values are unknown, so another equation is needed to get a deterministic system. Assuming the robot obeys non-holonomic constraints and assuming that the two wheels and their motors are identical [27]. From the system of equations (20) and with the above assumptions, we have the following dynamic system of the robot:

$$\begin{cases} \dot{v}_x = \dfrac{1}{mr}(\tau_R + \tau_L) - 2\dfrac{B_f}{mr^2}v_x - b\omega^2 \\ \dot{\omega} = \dfrac{1}{I_z + mb^2}\left(\dfrac{a}{r}(\tau_R - \tau_L) - 2\dfrac{B_f a^2}{r^2}\omega + mb\omega v_x\right) \end{cases} \quad (21)$$

Let $x =(x_1, x_2, x_3, x_4)^T=(d,\theta_e, v_x,\omega)$, $\tau_v = \tau_R + \tau_L$ and $\tau_\omega = \tau_R - \tau_L$, from equations (9), (11) and (15) we have the state equation when there is noise and uncertainty about the robot's dynamics in the form:

$$\begin{cases} \dot{x}_1 = -lx_4 - \tan x_2(x_3 - x_4 x_1) \\ \dot{x}_2 = x_4 - c(q)\dfrac{x_3 - x_4 x_1}{\cos x_2} \\ \dot{x}_3 = \dfrac{1}{mr}\tau_v - 2\dfrac{B_f}{mr^2}x_3 - bx_4^2 + d_v(t) \\ \dot{x}_4 = \dfrac{1}{I_z + mb^2}\left(\dfrac{a}{r}\tau_\omega - 2\dfrac{B_f a^2}{r^2}x_4 + mbx_4 x_3\right) + d_\omega(t) \end{cases} \quad (22)$$

where $d_v(t)$ and $d_\omega(t)$ are the external disturbance and uncertainty of the robot dynamics model. The parameters of the robot model are relatively calculated from the experimental model.

III. CONTROLLER SYNTHESIS FOR TWO-WHEELED LINE TRACKING ROBOT

Two control structures are commonly used for MIMO objects, called decentralized control and centralized control.

In decentralized control, the control law for each independent channel is simple where each channel is controlled as a SISO system. The coupling effects between joints, arising from the configuration changes during the motion, are treated as disturbances. Centralized control takes into account the entire robot structure, which is more difficult to design, but if the system model is given, the control performance can be significantly increased compared to decentralized control [60].

To control the robot to follow the predetermined line. The linear feedback control law is when the robot dynamics are uncertain and the impact of disturbances is difficult to ensure the control quality. The paper proposes a solution based on ADRC as an approach to control system design. The control structure is determined to combine decentralized and centralized control. The controller includes an LQR controller designed based on a linear model for 2 robot channels. Two EOS observers are designed to observe each channel and are used in the 2 LQR controllers. This control strategy considers the effects of cross-linking, disturbances, and parameter uncertainties through the values of the observed states on 2, but still ensures a simple and effective control structure in this case.

A. LQR Controller Synthesis

The LQR method is an optimal controller design method based on a linear model of a quadratic objective function [13][14][30]. From the linearized system of equations (22) at the operating point $x_{lqr} = (0,0, v_d,0)$ with $c(q)=0$, we get an equation of the form:

$$\dot{x} = Ax + B\tau \qquad (23)$$

where

$$A = \begin{bmatrix} 0 & -v_d & 0 & -l \\ 0 & 0 & 0 & 1 \\ 0 & 0 & -2\dfrac{B_f}{mr^2} & 0 \\ 0 & 0 & 0 & \dfrac{-2B_f a^2}{(I_z+mb^2)r^2}+\dfrac{mbv_d}{I_z+mb^2} \end{bmatrix}$$

$$B = \begin{bmatrix} 0 & 0 \\ 0 & 0 \\ \dfrac{1}{mr} & 0 \\ 0 & \dfrac{a}{(I_z+mb^2)r} \end{bmatrix}; \tau = \begin{bmatrix} \tau_v \\ \tau_\omega \end{bmatrix}$$

We need to find the matrix K of the optimal control value: $\tau_0=[\tau_{0v},\tau_{0\omega}]^T=-Kx(t)$ that satisfies the quality criterion J to reach the minimum value: $J = \int_0^\infty (x^T Qx + \tau_0^T R\tau_0)dt$. In which Q is a positive definite (or positive semi-definite) matrix, R is a positive definite matrix. The optimal matrix K is determined to have the form: $\tau_0^* = -R^{-1}B^T Px = -Kx$. The matrix P must then satisfy the equation from the Riccati equation:

$$PA_1 + A_1^T P + Q - PBR^{-1}B^T P = 0 \qquad (24)$$

B. ADRC Method

ADRC (Active Disturbance Rejection Control) is a control method in which the system model is extended with a new state variable [49]-[54]. This state variable includes all unknown dynamics and system disturbances that are not taken into account in the conventional object description. Estimation of this new state variable is performed using an Extended State Observer (ESO). This observer has the role of estimating the direct impact disturbances and errors of the object modeling compared to reality. In that way, although there is only a model with not too high accuracy, we can still design a good quality controller, thereby indirectly helping to simplify the model. The general structure of ADRC includes the following basic components: (1)-TD (Tracking Differentiator) is the component that determines the steady speed and the output response trajectory of the object. TD component can be extracted from the desired trajectory of the system. Control quality criteria will be used to calculate; (2)-ESO (Extended State Observer): the component that estimates disturbance and model error. If the above two components are determined, we can control the desired object; (3)- Nonlinear error feedback controller, in the special case of ADRC error feedback can be a linear controller or PID controller. The control structure diagram of the Two-Wheeled Line Tracking Robot with LQR controller and ADRC is shown in the Fig. 2.
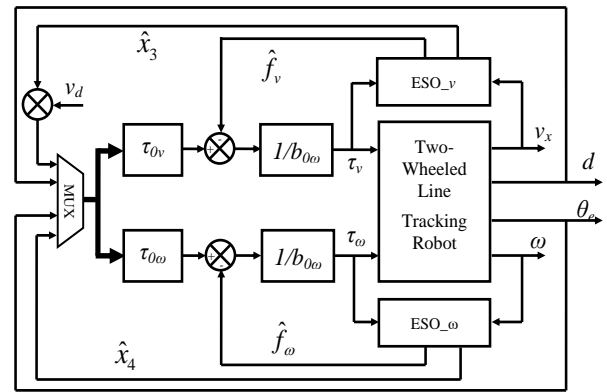

Fig. 2. Control structure based on LQR and ADRC controllers

The design of an ADRC controller for a robot is done on two separate channels. From the state equation (22), we separate the last two equations into two subsystems. These two subsystems are first-order objects, so we will synthesize the control law for each object. Synthesize the control law for the subsystem v_x.

Convert the state equation of the linear velocity of the robot to the following form:

$$\frac{dx_3}{dt} = f_v(t) + b_{0v}\tau_v \qquad (25)$$

In which $f_v(t) = -2\dfrac{B_f}{mr^2}x_3 - bx_4^2 + d_v(t) + \Delta b_{0v}.\tau_v(t)$; $d_v(t)$ is the noise and uncertain parameter part of the robot on the robot linear velocity channel; $b_v= b_{0v}+ \Delta b_{0v}$, with $b_{0v}=1/mr$ is a known component in the object's model; Δb_{0v} - uncertainty of parameters m, r.

If we can find a value of $\hat{f}_v \approx f_v(t)$, with just one more integration step, we can solve for x_3. For this reason, an extended state observer (ESO) was introduced. The ESO is also the key to any other ADRC controller. This observer will help us estimate the approximate value of $\hat{f}_v$.

To construct ESO, first, we rewrite equation (25) in the form of a system of state space equations as follows:

$$\begin{cases} \begin{pmatrix} \dot{v}_1(t) \\ \dot{v}_2(t) \end{pmatrix} = \begin{pmatrix} 0 & 1 \\ 0 & 0 \end{pmatrix} \cdot \begin{pmatrix} v_1(t) \\ v_2(t) \end{pmatrix} + \begin{pmatrix} b_{0v} \\ 0 \end{pmatrix} u(t) + \begin{pmatrix} 0 \\ 1 \end{pmatrix} \dot{\hat{f}}_v(t) \\ x_3(t) = \begin{pmatrix} 1 & 0 \end{pmatrix} \cdot \begin{pmatrix} v_1(t) \\ v_2(t) \end{pmatrix} \end{cases} \quad (26)$$

In the system of equations (26), there is an additional virtual input $\hat{f}_v(t)$, which cannot be measured directly, but we can estimate it based on the ESO. The estimation is based only on the measurement and processing of the robot's inputs $\tau_v(t)$ and outputs $x_3(t)$. The equation of the extended state observer is given as (27):

$$\begin{aligned} \begin{pmatrix} \dot{\hat{v}}_1(t) \\ \dot{\hat{v}}_2(t) \end{pmatrix} &= \begin{pmatrix} 0 & 1 \\ 0 & 0 \end{pmatrix} \begin{pmatrix} \hat{v}_1(t) \\ \hat{v}_2(t) \end{pmatrix} + \begin{pmatrix} b_{0v} \\ 0 \end{pmatrix} \tau_v(t) + \begin{pmatrix} l_{1v} \\ l_{2v} \end{pmatrix} (x_3(t) - \hat{v}_1(t)) \\ &= \begin{pmatrix} -l_{1v} & 1 \\ -l_{2v} & 0 \end{pmatrix} \begin{pmatrix} \hat{v}_1(t) \\ \hat{v}_2(t) \end{pmatrix} + \begin{pmatrix} b_{0v} \\ 0 \end{pmatrix} \tau_v(t) + \begin{pmatrix} l_{1v} \\ l_{2v} \end{pmatrix} x_3(t) \end{aligned} \quad (27)$$

In the above equation $\hat{v}_1(t) = \hat{x}_3(t)$; $\hat{v}_2(t) = \hat{f}_v(t)$, based on this estimation, the disturbance rejection is performed through the following control law (28).

$$\tau_v(t) = \frac{\tau_{0v}(t) - \hat{f}_v(t)}{b_{0v}} \quad (28)$$

where: $\tau_{0v}(t)$ is the input control law found based on the LQR method in the previous section.

Substitute (28) into (25):

$$\dot{x}_3(t) = f_v(t) + b_{0v} \frac{\tau_{0v}(t) - \hat{f}_v(t)}{b_{0v}} \quad (29)$$

$$= \left(f_v(t) - \hat{f}_v(t) \right) + \tau_{0v}(t) \approx \tau_{0v}(t)$$

When $f_v(t) \approx \hat{f}_v(t)$, we have:

$$\dot{x}_3(t) \approx \tau_{0v}(t) \quad (30)$$

Finally, the remaining task is to find the two parameters of the extended state observer l_{1v} , l_{2v} in equation (27). Finding l_{1v} , l_{2v} will be based on the principle of assigning reasonable poles to the observer. For the controller to work well, its observer must have fast enough dynamics; in other words, the poles of the observer must be such that its dynamics are faster than the dynamics of equation (30).

Synthesizing the ADRC control law for the ω subsystem is done similarly to the v channel, we have the following control law:

$$\tau_\omega(t) = \frac{\tau_{0\omega}(t) - \hat{f}_\omega(t)}{b_{0\omega}} \quad (31)$$

where $f_\omega(t) = \frac{1}{I_z + mb^2} \left(-2\frac{B_f a^2}{r^2} x_4 + mb x_4 x_3 \right) + d_\omega(t) + \Delta b_{0\omega} \cdot \tau_v(t)$; $d_\omega(t)$ is the noise and uncertain parameter part of the robot on the robot angular velocity channel; $b_\omega = b_{0\omega} + \Delta b_{0\omega}$, with $b_{0\omega} = a/(r(I_z + mb^2))$ is the known component in the model of the object; Δb_{0v}- the uncertainty of the parameters m, r, I_z. Substitute (21) into (25) with $f_\omega(t) \approx \hat{f}_\omega(t)$, we have:

$$\dot{x}_4(t) \approx \tau_{0\omega}(t) \quad (32)$$

From equations (30) and (32), we see that if the observed states are close to the actual values of the robot, the LQR control law ensures the stability of the system.

IV. SIMULATION AND EXPERIMENTAL RESULTS

The performance and efficiency of the proposed controller for the line-tracking robot are verified through the results obtained from simulations performed in Matlab/Simulink and the low-cost hardware platform developed.

A. Building a Model of a 2-Wheel Robot with a Line-Tracking Differential

The robot model is designed in the form of 2 differential wheels as shown in Fig. 3. The model parameters are determined relatively as follows: $m=10(kg)$; $a=0.18(m)$; $b=0.3(m)$; $l=0.068(m)$; $r=0.035(m)$; $L=0.25(m)$; $W=0.18(m)$; $I_z=9.292*10^{-3}(kg/m^2)$; $B_f=1.543*10^{-4}$.
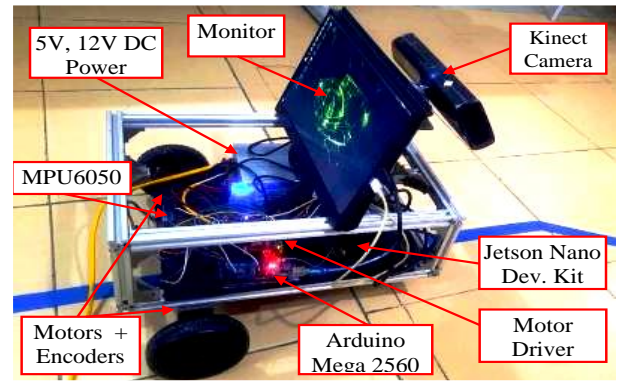

Fig. 3. Line-tracking robot model

The block diagram connecting the components of the robot is shown in Fig. 4. The two motors used are JGB37-545 geared motors with encoders attached to the rotor shaft of the motor. The two motors are controlled via a 7A Dual DC Motor Driver Module, which can control two independent motors up to 7A per channel or less than 160W. The ATmega2560 microcontroller is used as the central controller, implementing the control law, observing and receiving data from sensors, and receiving and transmitting data with the Jetsion Nano board. The program execution cycle on the microcontroller is 30 (μs). Two relative encoders with a resolution of 328 (pulses/rev) mounted on the motor are used to calculate the angular velocity of the two wheels, the translational velocity of each wheel is calculated from formula (17). The translational velocity of the robot is calculated as the average of the translational velocities of the two wheels. The MPU6050 sensor is used to determine the rotational velocity ω of the robot by measuring the angular velocity at the z-axis on the sensor. The distance d and angle θ_e are determined through the image captured from the Kinect XBOX 360 camera. The method for determining the distance d and angle θ_e is shown in Fig. 5. Because the camera is mounted so that the vertical axis of the captured image coincides with the vertical axis of the robot. So the distance d is determined from the center of the line to the center of the bottom edge image in pixels and converted to m through the conversion formula built from an experiment. The angle $\theta_e = \theta - \theta_q$, because it is not possible to directly measure the two values θ, θ_q from the sensors mounted on the robot. But in Fig. 5 we see that the angle θ_e is the angle formed by the tangent to the line at point Q with

the vertical axis of the image or the vertical axis of the robot. The system is powered by a DC source with a voltage of 12 (V), a current of 5 (A), and a voltage of 5 (V) current of 10 (A). The data acquisition software is built on the Jetsion Nano board connected to the Atmega 2560 microcontroller via a serial communication port.
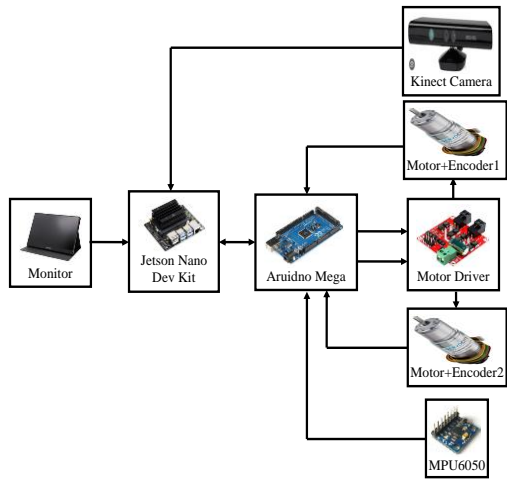

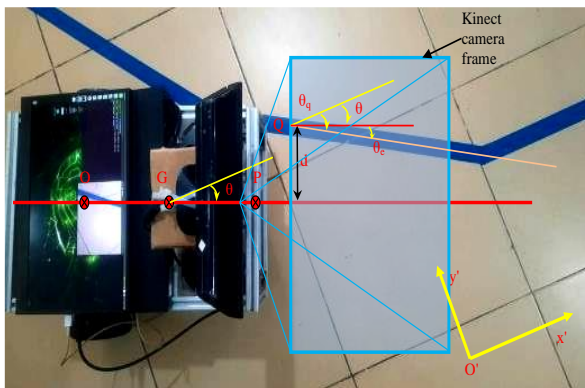Fig. 4. Connection diagram of line-tracking robot

Fig. 5. Method of calculating distance d and θ_e via Kinect camera frame

B. Numerical Simulation

The numerical simulation scenario is conducted under the condition that the robot will follow a path with a fixed small curvature c=0.01, the robot's translational velocity is stable at 0.5 (*m/s*) when there is noise in 2 channels in the form d_v=-0.2+random(-0.4,0.4); d_ω=0.2-random(-0.4,0.4), where the fixed component represents the uncertainty in the model and model parameters, and the random component represents the external noise. The initial state values of the robot are as follows: d_0=0.3 (*m*), θ_{e0}=0.5 (*rad*.), v_{x0}=0 (*m/s*), ω_0=0 (*rad./s*). The parameters of the LQR control law found with the linearized model are at the point x_{lqr} =(0, 0, 0.3, 0)^T. The matrix K found by the above command in the software has the following value:

$$K = \begin{bmatrix} 0 & 0 & 0.9944 & 0 \\ -1 & 1.3539 & 0 & 2.2441 \end{bmatrix} \quad (33)$$

The parameters of the two selected observers have the following values: l_{1v}=10, l_{2v}=2, $l_{1\omega}$=10, $l_{2\omega}$=100.

The simulation results performed with the proposed control law are shown in Fig. 6, Fig. 7, Fig. 8, and Fig. 9. From the results in Fig. 6, it can be seen that with the LQR

controller, there is always a static error away from the line of $d{\approx}0.1(m)$. The proposed controller makes the robot follow the line exactly after 11.5 (*s*). The LQR controller cannot achieve the linear velocity at 0.5 (*m/s*) shown in Fig. 8. This is due to the uncertainty component of the model that the linear controller cannot overcome. The proposed controller brings the linear velocity to the desired value within 15 (*s*) but is still affected by random noise. The response of the robot to the yaw angle θ_e and angular velocity ω with the two controllers is shown in Fig. 7 and Fig. 9. From the results, it can be seen that the response of the proposed controller is faster than that of the LQR controller, but this also makes the response of the proposed controller have a higher overshoot. The statistical results based on the simulation data are shown in Table I.
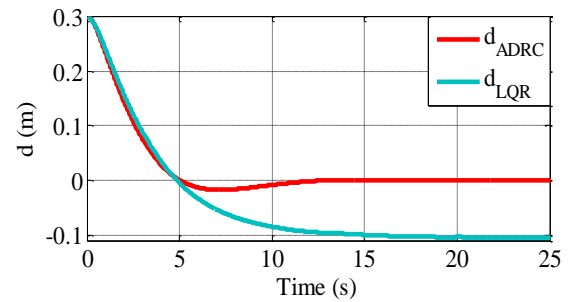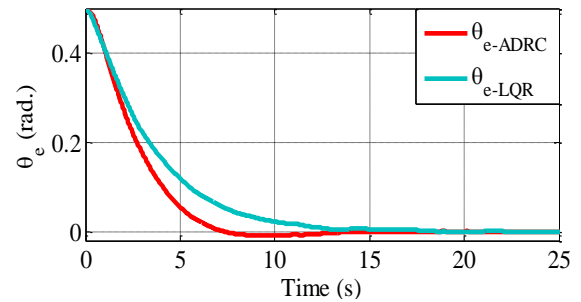

Fig. 6. Response distance error d of robot
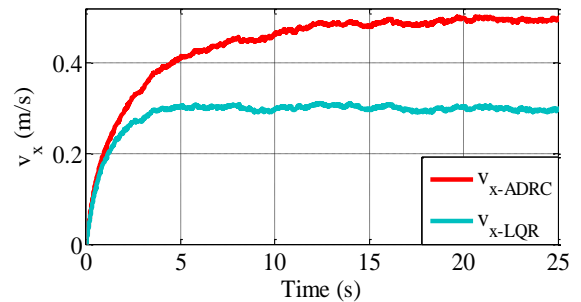

Fig. 7. Response to the robot's angular error θ_e.

Fig. 8. Robot linear velocity response v_x
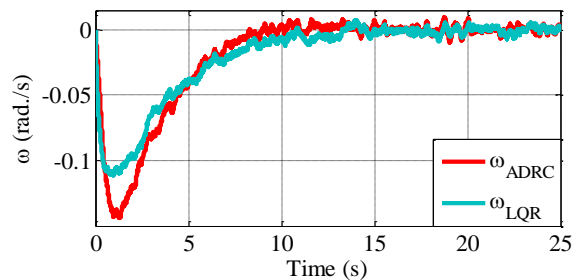

Fig. 9. Angular velocity response ω of the robot

TABLE I.　COMPARISON PARAMETERS OF TWO CONTROLLERS WHEN SIMULATING

Variable	LQR				ADRC			
	d	θ_e	v_x	ω	d	θ_e	v_x	ω
Settling Time (s)	∞	9.8	∞	10.6	8.3	6	14.9	9.1
Overshoot (%)	33.3	0	0	11.5	5.5	2	0	14
Static error (m, rad, m/s, rad./s)	0.1	0	0.2	0	0	0	0	0
IAE (m, rad, m/s, rad./s)	2.76	2.3	5.9	0.52	1.02	1.85	2.02	0.54

C. Experimental Results

Experiments were conducted to validate the results of the control law built according to the proposed method. The control program was written in C programming language on Arduino IDE software. The image processing program was written in Python language on the Linux operating system. The experimental data were saved on the Jetson Nano embedded computer. The robot was run along the line, with the initial condition being 0.5 (*m*) off the line and the deviation angle being 0.6 (*rad*.). The structural parameters of the line-tracking robot are given in the simulation section. The parameters of the control law are given in the simulation section. The response results of the line-tracking robot are shown in Fig. 10 to Fig. 13, showing that the response on the real system is quite similar to the response of the system with the simulation in the above section, which shows that the results of the proposed control law are valid. The linear velocity of the robot with the LQR controller (Fig. 10) does not reach the desired value of 0.5 (*m/s*), this is because the motor model is not fully recognized. The proposed controller has overcome the disadvantage of the LQR controller when the observer has observed the uncertainty of the model. Therefore, on the same distance of the same length, with the proposed controller the robot travels in 17.0 (*s*) while the LQR controller is 23.5 (*s*). The response of the robot angular velocity ω of both cases gives quite a good response (Fig. 11). The response of the distance error d (Fig. 12) of both controllers goes to 0 but fluctuates around the line, with the proposed controller having a lower overshoot at the bends. The angular error response θ_e of both cases has the largest overshoot at the curve (Fig. 13) but still ensures the robot follows the line. The experimental statistical results based on the initial data when the vehicle starts to adhere to the line are shown in Table II.
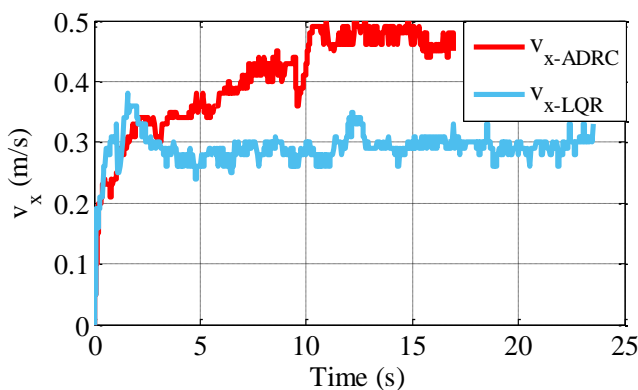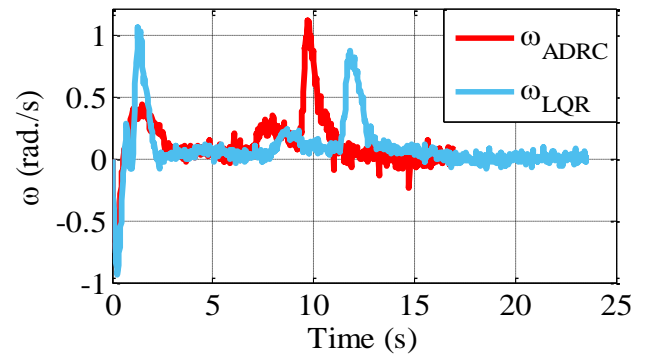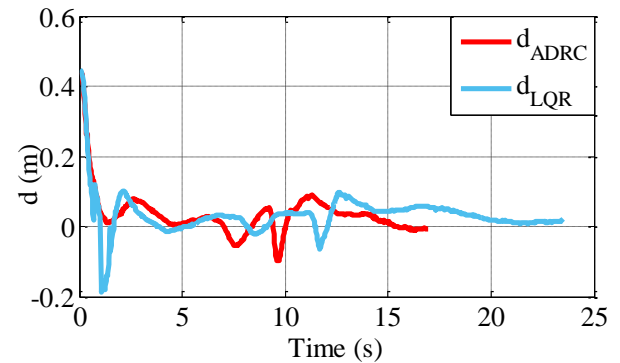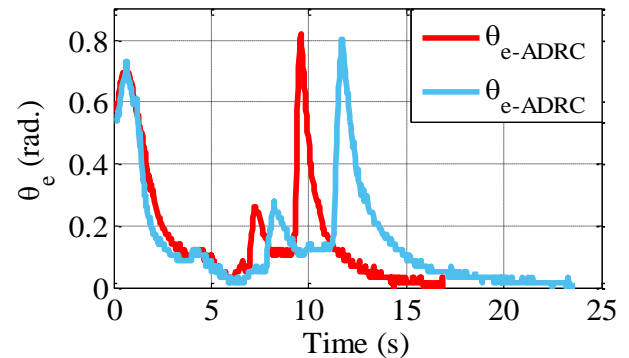

Fig. 10.　Robot linear velocity response v_x

Fig. 11.　Angular velocity response ω of the robot

Fig. 12.　Response distance error d of robot

Fig. 13.　Response to the robot's angular error θ_e

TABLE II.　COMPARISON PARAMETERS OF TWO CONTROLLERS WHEN EXPERIMENTING

Variable	LQR				ADRC			
	d	θ_e	v_x	ω	d	θ_e	v_x	ω
Settling Time (s)	5.0	6.0	∞	3.5	5.0	6.0	10	3.8
Overshoot (%)	44.4	0	0	150	0	0	0	55
Static error (m, rad, m/s, rad./s)	0.01	0	0.2	0	0.01	0	0.01	0
IAE (m, rad, m/s, rad./s)	1.75	2.62	5.13	2.63	1.31	2.93	3.24	2.25

V.　CONCLUSION

In this article, we have synthesized the ADRC controller for mobile robot line-following based on the ADRC method and the LQR controller. The control law for the robot is combined with 2 observers at 2 channels of linear velocity and angular velocity of the robot to observe uncertain factors in the robot dynamics model. This will provide more information for the LQR controller in responding to uncertain factors and noise. In addition, although it is necessary to add 2 more observers compared to the

traditional LQR controller, the calculation of the parameters of the observer is quite simple and can be easily deployed in practice. The simulation results show the superiority of the proposed control method when the evaluation quality indicators are mostly better. The experimental results show that the proposed control method ensures that the real system operates to meet the requirements in the control task. In particular, it shows that the proposed control law ensures that the system is stable at the desired linear velocity. In future studies, the proposed method will also be studied in combination with nonlinear feedback controllers, fuzzy controllers, and objects with more complex structures.

REFERENCES

[1] S. G. Tzafestas. *Introduction to mobile robot control*. Elsevier, 2013.

[2] G. Klančar, A. Zdešar, S. Blažič, and I. Škrjanc. *Wheeled Mobile Robotics*. Butterworth-Heinemann, 2017.

[3] L. Jaulin. *Wheeled Mobile Robotics*. ISTE Press – Elsevier, 2017.

[4] A. V. Savkin, A. S. Matveev, M. Hoy, and C. Wang. *Safe robot navigation among moving and steady obstacles*. Butterworth-Heinemann, 2015, doi: 10.1016/C2014-0-04743-0.

[5] C. C. De Wit, "Quasicontinuous stabilizing controllers for nonholonomic systems: Design and robustness considerations," in *Proc. of 3rd European Control Conference,* pp. 2630-2635, 1995.

[6] J. Guldner and V. I. Utkin, "Stabilization of non-holonomic mobile robots using Lyapunov functions for navigation and sliding mode control," *Proceedings of 1994 33rd IEEE Conference on Decision and Control*, pp. 2967-2972, 1994, doi: 10.1109/CDC.1994.411340.

[7] F. Demirbaş and M. Kalyoncu, "Differential drive mobile robot trajectory tracking with using pid and kinematic based backstepping controller," *Selçuk Üniversitesi Mühendislik, Bilim ve Teknoloji Dergisi*, vol. 5, no. 1, pp. 1-15, 2017.

[8] R. R. Carmona, H. G. Sung, Y. S. Kim, and H. A. Vazquez, "Stable PID Control for Mobile Robots," *2018 15th International Conference on Control, Automation, Robotics and Vision (ICARCV)*, pp. 1891-1896, 2018, doi: 10.1109/ICARCV.2018.8581132.

[9] A. Barsan, "Position control of a mobile robot through PID controller," *Acta Universitatis Cibiniensis. Technical Series*, vol. 71, no. 1, pp. 14-20, 2019, doi: 10.2478/aucts-2019-0004.

[10] C. S. Shijin and K. Udayakumar, "Speed control of wheeled mobile robots using PID with dynamic and kinematic modelling," *2017 International Conference on Innovations in Information, Embedded and Communication Systems (ICIIECS)*, pp. 1-7, 2017, doi: 10.1109/ICIIECS.2017.8275962.

[11] T. P. Nascimento, C. E. T. Dórea, and L. M. G. Gonçalves, "Nonlinear model predictive control for trajectory tracking of nonholonomic mobile robots: A modified approach," *International Journal of Advanced Robotic Systems*, vol. 15, no. 1, 2018.

[12] F. Lin, Z. Lin, and X. Qiu, "LQR controller for car-like robot," *2016 35th Chinese Control Conference (CCC)*, pp. 2515-2518, 2016, doi: 10.1109/ChiCC.2016.7553742.

[13] J. Fang, "The LQR Controller Design of Two-Wheeled Self-Balancing Robot Based on the Particle Swarm Optimization Algorithm," *Mathematical Problems in Engineering*, vol. 2014, no. 1, p. 729095, 2014, doi: 10.1155/2014/729095.

[14] C. Samson, P. Morin, and R. Lenain, "Modeling and control of wheeled mobile robots," *Springer handbook of robotics*, pp. 1235-1266, 2016.

[15] I. Reguii, I. Hassani, and C. Rekik, "Mobile robot navigation using planning algorithm and sliding mode control in a cluttered environment," *Journal of Robotics and Control (JRC)*, vol. 3, no. 2, pp. 166-175, 2022, doi: 10.18196/jrc.v3i2.13765.

[16] I. A. Hassan, I. A. Abed, and W. A. Al-Hussaibi, "Path planning and trajectory tracking control for two-wheel mobile robot," *Journal of Robotics and Control (JRC)*, vol. 5, no. 1, pp. 1-15, 2024, doi: 10.18196/jrc.v5i1.20489.

[17] C. E. Martínez-Ochoa, I. O. Benítez-González, A. O. Cepero-Díaz, J. R. Nuñez-Alvarez, C. G. Miguélez-Machado, and Y. E. Llosas-Albuerne, "Active disturbance rejection control for robot manipulator," *Journal of Robotics and Control (JRC)*, vol. 3, no. 5, pp. 622-632, 2022.

[18] N. A. Alawad, A. J. Humaidi, and A. S. Alaraji, "Observer sliding mode control design for lower exoskeleton system: Rehabilitation case," *Journal of Robotics and Control (JRC)*, vol. 3, no. 4, pp. 476-482, 2022.

[19] P. Victerpaul, D. Saravanan, S. Janakiraman, and J. Pradeep, "Path planning of autonomous mobile robots: A survey and comparison," *Journal of Advanced Research in Dynamical and Control Systems*, vol. 9, no. 12, pp. 1535-1565, 2017.

[20] D. Liu, Q. Gao, Z. Chen, and Z. Liu, "Linear Active Disturbance Rejection Control of a Two-Degrees-of-Freedom Manipulator," *Mathematical Problems in Engineering*, vol. 2020, no. 1, 2020.

[21] M. A. Faraj, B. Maalej, and N. Derbel, "Optimal sliding mode controller for lower limb rehabilitation exoskeleton in constrained environments," *Indonesian Journal of Electrical Engineering and Computer Science*, vol. 30, no. 3, pp. 1458-1469, 2023.

[22] S. B. Messaoud, M. Belkhiri, A. Belkhiri, and A. Rabhi, "Active disturbance rejection control of flexible industrial manipulator: A MIMO benchmark problem," *European Journal of Control*, vol. 77, p. 100965, 2024.

[23] M. Khamar and M. Edrisi, "Designing a backstepping sliding mode controller for an assistant human knee exoskeleton based on nonlinear disturbance observer," *Mechatronics*, vol. 54, pp. 121-132, 2018.

[24] A. A. Sneineh and W. A. Salah, "Development of a Sensor-Based Glove-Controlled Mobile Robot for Firefighting and Rescue Operations," *International Journal of Robotics and Control Systems*, vol. 4, no. 4, pp. 1641-1655, 2024.

[25] I. Hassani and C. Rekik, "Backstepping controller for mobile robot in presence of disturbances and uncertainties," *International Journal of Robotics and Control Systems*, vol. 3, no. 4, pp. 934-954, 2023.

[26] I. Hassani, I. Ergui, and C. Rekik, "Turning Point and Free Segments Strategies for Navigation of Wheeled Mobile Robot," *International Journal of Robotics and Control Systems*, vol. 2, no. 1, pp. 172-186, 2022.

[27] R. Rajamani. *Vehicle Dynamics and Control*. Springer, 2006.

[28] W. E. Dixon, D. M. Dawson, E. Zergeroglu, and A. Behal. *Nonlinear Controlof Wheeled Mobile Robots*. Springer, 2001.

[29] N. X. Chiem, "Synthesis of LQR Controller Based on BAT Algorithm for Furuta Pendulum Stabilization," *Journal of Robotics and Control (JRC)*, vol. 4, no. 5, pp. 662-669, 2023, doi: 10.18196/jrc.v4i5.19661.

[30] R. S. Ali, A. A. Aldair, and A. K. Almousawi, "Design an Optimal PID Controller using Artificial Bee Colony and Genetic Algorithm for Autonomous Mobile Robot," *International Journal of Computer Applications*, vol. 100, no. 1, pp. 8-16, 2014.

[31] Dhaouadi, R., Hatab, A. A., 2013, ‚Dynamic Modelling of Differential-Drive Mobile Robots using Lagrange and Newton-Euler Methodologies: A Unified Framework', Research Article, Advances in Robotics & AutomationTechnology, Vol. 2 (2), pp. 1-7.

[32] A. E. Cabrera. *Rapid Prototyping of Mobile Robot Control Algorithms*, Degree of Master of Science in Faculty of Electrical Engineering, Department of Control Engineering, Czech Technical University In Prague, 2014.

[33] R. Fierro and F. L. Lewis, "Control of A Nonholonomic Mobile Robot: Backstepping Kinematics Into Dynamics," in *Proceedings of 34th IEEE Conference on Decision and Control*, pp. 3805-3810, 1995.

[34] E. J. Hwang, H. S. Kang, C. H. Hyun, and M. Park, "Robust backstepping control based on a Lyapunov redesign for skid-steered wheeled mobile robots," *International Journal of Advanced Robotic Systems*, vol. 10, no. 1, p. 26, 2013.

[35] Y. Kanayama, Y. Kimura, F. Miyazaki and T. Noguchi, "A stable tracking control method for an autonomous mobile robot," *Proceedings., IEEE International Conference on Robotics and Automation*, pp. 384-389, 1990, doi: 10.1109/ROBOT.1990.126006.

[36] M. Oubbati, M. Schanz and P. Levi, "Kinematic and dynamic adaptive control of a nonholonomic mobile robot using a RNN," *2005 International Symposium on Computational Intelligence in Robotics and Automation*, pp. 27-33, 2005, doi: 10.1109/CIRA.2005.1554250.

[37] R. Solea, A. Filipescu, A. Filipescu, E. Minca and S. Filipescu, "Wheelchair control and navigation based on kinematic model and iris movement," *2015 IEEE 7th International Conference on Cybernetics and Intelligent Systems (CIS) and IEEE Conference on*

Robotics, Automation and Mechatronics (RAM), pp. 78-83, 2015, doi: 10.1109/ICCIS.2015.7274600.

[38] E.-J. Hwang, H.-S. Kang, C.-H. Hyun, and M. Park, "Robust Backstepping Control Based on a Lyapunov Redesign for Skid-Steered Wheeled Mobile Robots," *International Journal of Advanced Robotic Systems*, vol. 10, 2013, doi: 10.5772/55059.

[39] S. M. Swadi, M. A. Tawfik, E. N. Abdulwahab, and H. A. Kadhim, "Fuzzy-Backstepping Controller Based on Optimization Method for Trajectory Tracking of Wheeled Mobile Robot," *2016 UKSim-AMSS 18th International Conference on Computer Modelling and Simulation (UKSim)*, pp. 147-152, 2016, doi: 10.1109/UKSim.2016.52.

[40] R. Martínez, O. Castillo, and L. T. Aguilar, "Optimization of Interval Type-2 Fuzzy Logic Controllers for a Perturbed Autonomous Wheeled Mobile Robot Using Genetic Algorithms," *Inf. Sci.,* vol. 13, pp. 2158-2174.

[41] T. T. Mac, C. Copot, R. De Keyser, T. D. Tran, and T. Vu, « MIMO Fuzzy Control for Autonomous Mobile Robot," *Journal of Automation and Control Engineering*, vol. 4, no. 1, pp. 65-70, 2016.

[42] N. X. Chiem, N. D. Anh, A. D. Lukianov, P. D. Tung, H. D. Long, and N. D. Linh, "Design real-time embedded optimal PD fuzzy controller by PSO algorithm for autonomous vehicle mounted camera," *AIP Conference Proceedings,* vol. 2188, p. 030008, 2019, doi: 10.1063/1.5138401.

[43] R. D. Puriyanto and A. K. Mustofa, "Design and implementation of fuzzy logic for obstacle avoidance in differential drive mobile robot," *Journal of Robotics and Control (JRC)*, vol. 5, no. 1, pp. 132-141, 2024, doi: 10.18196/jrc.v5i1.20524.

[44] K. Lee, D. Y. Im, B. Kwak, and Y. J. Ryoo, "Design of fuzzy-PID controller for path tracking of mobile robot with differential drive," *International Journal of Fuzzy Logic and Intelligent Systems*, vol. 18, no. 3, pp. 220-228, 2018, doi: 10.5391/IJFIS.2018.18.3.220.

[45] N. Van Tinh, "Neural network-based adaptive tracking control for a nonholonomic wheeled mobile robot subject to unknown wheel slips," *Journal of Computer Science and Cybernetics*, vol. 33, no. 1, pp. 70-85, 2017, doi: 10.15625/1813-9663/33/1/8914.

[46] S. A. Ahmed and M. G. Petrov, "Trajectory control of mobile robots using type-2 fuzzy-neural PID controller," *IFAC-PapersOnLine*, vol. 48, no. 24, pp. 138-143, 2015.

[47] X. Hai *et al*., "Mobile Robot ADRC With an Automatic Parameter Tuning Mechanism via Modified Pigeon-Inspired Optimization,"

in *IEEE/ASME Transactions on Mechatronics*, vol. 24, no. 6, pp. 2616-2626, Dec. 2019, doi: 10.1109/TMECH.2019.2953239.

[48] J. Han, "From PID to active disturbance rejection control," *IEEE Transactions on Industrial Electronics*, vol. 56, pp. 900–906, 2009.

[49] B.-Z. Guo and Z.-L. Zhao, "Active disturbance rejection control: theoretical perspectives," *Communications in Information and Systems*, vol. 15, no. 3, pp. 361–421, 2015.

[50] L. Guo and S. Cao. Anti-disturbance control theory for systems with multiple disturbances: a survey," *ISA Transactions*, vol. 53, no. 4, pp. 846–849, 2014.

[51] S. Li, J. Yang, W.-H. Chen, and X. Chen. *Disturbance observerbased control: methods and applications*. CRC Press, 2014.

[52] M. Makarov *et al.*, "Modeling and preview H control design for motion control of elastic-joint robots with uncertainties*," IEEE Trans. Ind. Electron.*, vol. 63, no. 10, pp. 6429–6438, Oct. 2016.

[53] L. H. Wang, "Adaptive control of robot manipulators with uncertain kinematics and dynamics," *IEEE Trans. Autom. Control*, vol. 62, no. 2, pp. 948–954, Feb. 2017.

[54] B. Brahmi *et al*., "Adaptive tracking control of an exoskeleton robot with uncertain dynamics based on estimated time-delay control," *IEEE/ASME Trans. Mechatronics*, vol. 23, no. 2, pp. 575–585, Apr. 2018.

[55] M. Van, M. Mavrovouniotis, and S. Ge, "An adaptive backstepping nonsingular fast terminal sliding mode control for robust fault tolerant control of robot manipulators," *IEEE Trans. Syst., Man, Cybern*., vol. 49, no. 7, pp. 1448–1458, Jul. 2019.

[56] J. Mukherjee, S. Mukherjee, and I. N. Kar, "Sliding mode control of planar snake robot with uncertainty using virtual holonomic constraints," *IEEE Robot. Autom. Lett*, vol. 2, no. 2, pp. 1077–1084, 2017.

[57] N. Martínez-Fonseca *et al*., "Robust disturbance rejection control of a bipedrobotic system using high-order extended state observer," *ISA Trans.*, vol. 62, pp. 276–286, May 2016.

[58] B. Ahi and A. Nobakhti, "Hardware implementation of an ADRC controller on a gimbal mechanism," *IEEE Trans. Control Syst. Technol.*, vol. 26, no. 6, pp. 2268–2275, Nov. 2018.

[59] G. Herbs. *A Simulative Study on Active Disturbance Rejection Control as a Control Tool for Practitioners*. In Siemens AG, Clemens-Winkler-Strabe 3, Germany. 2013.

[60] Z. Chu, Y. Sun, C. Wu, and N. Sepehri, "Active disturbance rejection control applied to automated steering for lane keeping in autonomous vehicles," *Control Eng. Pract*., vol. 74, pp. 13–21, 2018.