Stabilizing Two-Wheeled Robot Using Linear Quadratic Regulator and States Estimation

Nur Uddin, Teguh Aryo Nugroho, and Wahyu Agung Pramudito
Department of Electrical Engineering
Universitas Pertamina
Jakarta, Indonesia
Email: nur.uddin@universitaspertamina.ac.id

***Abstract*—A control design for two-wheeled robot (TWR) stabilization using linear quadratic regulator (LQR) method and states estimation is presented. A two-wheeled robot is an unstable system such that a control system is required to stabilize it. LQR is a control design method where the control gain is calculated by minimizing a performance index. The LQR method results in a full states feedback control law. Applying the LQR method for TWR stabilization will require feedback from pitching angle and pitching rate angle measurements. However, the common available sensor in the TWR is only a rate gyro which measures the pitching rate angle. Theoretically, time integration of the pitching rate angle results in pitching angle but it is not practically applicable because the time integration of measurement data may result in an accumulation of error and noise. Instead of the time integration, a Luenberger observer is applied to estimate the pitching angle based on the pitching rate angle measurement. Simulations of the TWR stabilization using the designed controller are presented through two scenarios: 1) both required state feedback are assumed to be available, and 2) both required state feedback are obtained through estimation using Luenberger observer. The first scenario is representing an ideal condition, while the second scenario is representing a practical condition. Simulation results show that the TWR is stabilized for the both scenarios, where the control system performance in the second scenario is slightly less than the first scenario but still acceptable.**

Keywords: Two-wheeled robot, self-balancing robot, linear quadratic regulator, Luenberger observer, states estimation.

I. Introduction

Two-wheeled robot (TWR) is one of the interesting plants in control system study. The robot is quite cheap but it provide some challenge problems for control system study. The robot is statically unstable. It is not able to stand by self. The robot can be stabilized using a control system such that the robot has to be operate in a closed loop control system.

Structure of a TWR is basically consisted of a body supported by two wheels, where each wheel is driven by a motor. The two wheels support makes the robot to have high maneuverability which becomes the advantage of TWR compared to other wheeled robot (three or four-wheeled robot). The TWR is very potential to be a high maneuver ground vehicle with varying applications, for examples: transportation, surveillance, logistic, exploration, rescue, and leisure. Segway is an example of two wheel robot which is quite success in consumer market.

Dynamic model of a plant which is mathematics representation of the plant dynamic, is required in a control system design. The dynamic model can be obtained through system modeling or system identification. Modeling of TWR can be done by deriving the dynamic equations using the Newton's law as presented in [1], [2] or using the Euler-Lagrange method as presented in [3], [4]. The system modelings of TWR result in a non-linear dynamics system.

The non-linearity in the TWR dynamics becomes a challenge in the control system design. The TWR is stabilized for the whole operating region if the TWR closed loop system is global asymptotic stable (GAS). The GAS is achieved by satisfying the Lyapunov's stability theorem. The Lyapunov's stability theorems is the base of non-linear control system theories. Several studies of TWR stabilization using non-linear control theories has been presented. TWR stabilization using Lyapunov-based controller has been presented in [5]–[7]. Stabilizing TWR using backstepping control method has been presented in [2], [8]–[10]. Backstepping control method is a non-linear control method which provides a systematically procedure to obtain a control law including stability proof. However for some cases, the backstepping control method may result in a complex feedback control law which is very difficult to be implemented in a real system [11]. Another non-linear control design method using adaptive neural networks has been studied in [12]–[14].

Control system design using non-linear control theories is a complex and difficult task. On another hand, linear control theories provide a simpler task for the control system design. Unfortunately, the linear control theories are only applicable for linear system but not for a non-linear system. Thanks to linearization which is a method to approximate a non-linear system as a linear system such that the linear control design can be applied [15], [16]. This approximation is only valid for a limited region around an equilibrium point of the nonlinear system but it may be sufficient for some systems. Examples of control design of a non-linear system through linearization can be found in [3], [17], [18].

Linear quadratic regulator (LQR) is one of the powerful methods in the linear control theory. Using the LQR, a control gain of states feedback control law is calculated by minimizing a performance index which is represented by a quadratic performance index. Studies of applying LQR for stabilizing a TWR have presented. Control design using linear quadratic regulator for a TWR has been presented in [19]. A comparison of using PID control and LQR control to stabilize TWR has been presented in [20] and the study results show that both PID and LQR are able to stabilize the TWR, but the LQR controller has better performance compared to the PID controller. In the presented studies on stabilizing a TWR using LQR, the controller requires full states feedback which is assumed to be available. In fact, stabilizing a TWR requires at least two-states feedback: pitching angle and pitching rate angle of the robot. Gyro is a common sensor installed in a TWR to measure pitching movement of the robot. However, the gyro is only measure the rate angle. Time integration of the rate angle theoretically results in angle but the time-integration is not recommended in the real application, noise and error measurement will be accumulated by the running time and results in drifted measurement data. Applying the drifted data as states feedback in a control system may results in poor control performance or even more a failure.

Considering implementation of stabilizing TWR using LQR method, this paper is presenting LQR control design and state estimation. An observer is applied to estimate the unmeasured state (pitching angle) based on the measurement output of the robot pitching rate using gyro. By the state estimation, the designed control system will be better prepared for implementation than the assumption that all of the states feedback are available. This paper is organized as follows. Dynamics of a TWR is described in Section 2. LQR control design is presented in Section 3. Several LQR designs by varying the performance index are presented and simulated to show the weighting effect of each performance index. Section 4 presents an Luenberger observer to estimate the TWR system states and applying the estimated states as feedback for the LQR. Conclusion of this study is presented in Section 5.

II. Modeling of Two Wheels Robot

A two-wheeled robot (TWR) is basically consisted of a robot body supported by two wheels. Figure 1a shows a TWR model. The robot body is modeled by a linkage where the center of mass is located at the middle of linkage. It is assumed that both wheels of the robot are connected by a shaft and driven by a motor.

If the robot is disturbed, for an example by giving an initial clock wise pitching angle θ such that the robot is tilted to the right, the weight of the robot body gives a clockwise moment to the system and the robot will be fall. This is the reason why the TWR is statically unstable. A counter torque is required to stabilize the robot. When the motor is operating, it provides torque to rotate the wheels. Friction of the wheels and floor results in reaction torque (τ) to the body. This reaction torque can be applied as a counter torque for stabilizing the robot. Figure 1b shows free body diagram of the robot, where there are two working moments: moment due to the body weight and moment due to the reaction torque. Base on the free body diagram, the working torques on the robot body are evaluated at the wheel axis (point O). Applying the Newton's second law results in the following dynamic equations:

$$\begin{aligned} \Sigma M &= I\ddot{\theta} \\ \tfrac{1}{2}mglsin\theta - \tau &= I\ddot{\theta}. \end{aligned} \tag{1}$$

where M is the working moments, I is the inertia of the robot body, θ is the pitching angle, m is the mass of robot body, g is the gravity acceleration, l is the length of the robot body, $\ddot{\theta}$ is the pitching acceleration of the robot body, and τ is the input torque which is the counter torque to stabilize the robot body.

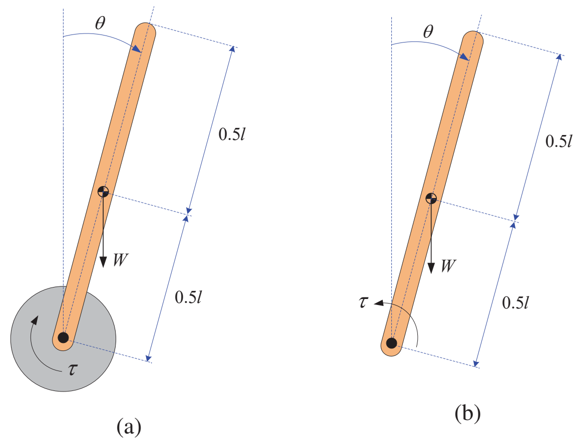

Fig. 1: (a) Two-wheeled robot (TWR) model. (b) Free-body diagram of the TWR model.

III. Control Design for Two-wheel Robot Stabilization

In order to stabilize the two-wheeled robot, a controller is required to drive a motor such that the motor generate a proper torque. Linear quadratic regulator (LQR) method is applied in designing the controller. The TWR dynamic equations (1) can be expressed in system states equation by defining system states as follows:

$$\begin{aligned} x_1 &= \theta \\ x_2 &= \dot{\theta}. \end{aligned} \tag{2}$$

Substituting (2) into (1) results in:

$$\begin{aligned} \dot{x}_1 &= x_2 \\ \dot{x}_2 &= \tfrac{mgl}{2I}sinx_1 + \tfrac{1}{I}\tau. \end{aligned} \tag{3}$$

Define constants $k_1 = \frac{mgl}{2I}$ and $k_2 = \frac{1}{I}$ and substituting into (3) results in:

$$\begin{aligned} \dot{x}_1 &= x_2 \\ \dot{x}_2 &= k_1 sinx_1 + k_2\tau. \end{aligned} \tag{4}$$

Equation (4) is the state equation of TWR system. The (4) is a non-linear system. In order to apply LQR in designing a controller for the non-linear system, transforming the system into a linear system is required. It can be done by linearizing the system around an equilibrium point. Linearizing the system around the origin ($\theta = 0$, $\dot{\theta} = 0$) results in:

$$\dot{x} = Ax + Bu, \tag{5}$$

where $x = \begin{bmatrix} x_1 \\ x_2 \end{bmatrix}$, $A = \begin{bmatrix} 0 & 1 \\ k_1 & 0 \end{bmatrix}$, $B = \begin{bmatrix} 0 \\ k_2 \end{bmatrix}$, and $u = \tau$.

A. Linear Quadratic Regulator (LQR)

LQR is a control design method, where a control gain is calculated by minimizing a quadratic cost function. For the system in (5), define a cost function or performance index:

$$J = \frac{1}{2}\int_0^\infty \left(x^TQx + Ru^2\right)dt, \tag{6}$$

TABLE I: SIMULATION PARAMETERS

Parameter	Symbol	Value	Unit
Mass of the robot body	m	0.1	kg
Length of the robot body	l	20	cm
Inertia of the robot body	I	13.408	$kg.cm^2$

where Q is a symetric positive semi definite matrix and R is a positive constant. Optimal control solution to minimize cost function (6) of system (5) is given by [21]:

$$u = -Kx, \tag{7}$$

where K is the control gain and defined by:

$$K = R^{-1}B^TP \tag{8}$$

with P is solution of algebraic Riccati equation:

$$A^TP + PA - PBR^{-1}B^TP + Q = 0. \tag{9}$$

The control law (7) is a full states feedback which requires feedback from pitching angle (x_1) and pitching rate angle (x_2) measurements.

Demonstration of robot stabilization using a controller designed using LQR method is presented through computer simulations. In the simulation, the robot position is initially at pitching angle $\theta = 20°$. Without any control action, the robot is definitely to be fall due to the gravity. The simulations are done based on robot parameters given in Table I. The simulation assumes an ideal condition where all of the system states are measured perfectly without any noise or error.

Simulations of four different controllers are presented to evaluate the control performance in stabilizing the robot. The four controllers are called Ctrl 1, Ctrl 2, Ctrl 3, and Ctrl 4 which are designed using LQR method and the cost function parameters (Q and R) are given in Table II. Figure 2 shows the simulation results. All of the controllers are able to stabilize the robot by returning the position from pitching at $20°$ to $0°$ such that the robot is steady at standing position. The four controllers shows different performances. The Ctrl 2 results in the best performance among them as shown by a shortest time to be steady at $\theta = 0$ (the fastest settling time). It shows that increasing the matrix element $Q(1,1)$ results in the faster response. The matrix element $Q(1,1)$ is related to the state x_1 (pitching angle) where choosing a higher number of $Q(1,1)$ means of giving more emphasis on x_1 and expecting the small value of x_1. The Ctrl 3 has a high number at matrix element $Q(2,2)$ where $Q(2,2)$ is related to x_2 (pitching rate angle). The Ctrl 3 gives a higher emphasis on x_2 and the simulation results in the least pitching rates among the four controllers. However, the Ctrl 3 results in the longest settling time on pitching angle. The long settling time is caused by the small pitching rates. In the Ctrl 4, both matrix elements of $Q(1,1)$ and $Q(2,2)$ are given the same big number which means to expect small pitching angle and small pitching rate angle. The simulation of using Ctrl 4 results in moderate response because the small pitching rate results in slow response in pitching angle.

TABLE II: LQR Cost Function

Controller	Performance Index
Ctrl 1	$Q = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}$, R=1
Ctrl 2	$Q = \begin{bmatrix} 1000 & 0 \\ 0 & 1 \end{bmatrix}$, R=1
Ctrl 3	$Q = \begin{bmatrix} 1 & 0 \\ 0 & 1000 \end{bmatrix}$, R=1
Ctrl 4	$Q = \begin{bmatrix} 1000 & 0 \\ 0 & 1000 \end{bmatrix}$, R=1
Ctrl 5	$Q = \begin{bmatrix} 10000 & 0 \\ 0 & 1 \end{bmatrix}$, R=1
Ctrl 6	$Q = \begin{bmatrix} 10000 & 0 \\ 0 & 1 \end{bmatrix}$, R=10
Ctrl 7	$Q = \begin{bmatrix} 10000 & 0 \\ 0 & 1 \end{bmatrix}$, R=100

IV. States Estimation

In practical application, the most common sensor installed on a TWR to measure pitching motion is a rate gyro. The rate gyro is a sensor for measuring angular velocity such that the available measured data in the TWR is the pitching rate angle only. Pitching angle is theoretically the time integration of the pitching rate angle. However, time integration of measured data is not recommended in real application due to noise and error measurement such that the time-integration may result in an accumulated error. An observer is commonly used to estimate unmeasured states. The observer will be applied in estimating the pitching angle based on the rate gyro measurement output instead of integrating with respect to time.

Observer design for the pitching angle estimation is presented as follows. Recall the linearized system state equations (5),

$$\dot{x} = Ax + Bu$$

and the measurement output is given by:

$$y = Cx \tag{10}$$

where $C = \begin{bmatrix} 0 & 1 \end{bmatrix}$ which represents the pitching rate angle measurement. An observer known as the Luenberger observer

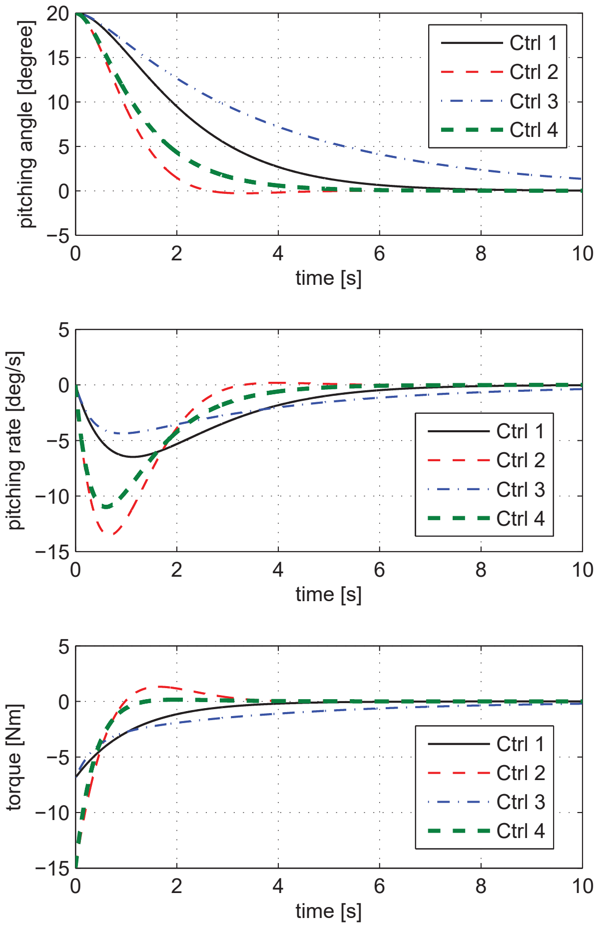

Fig. 2: COMPARISON OF ROBOT STABILIZATION USING SEVERAL LQR CONTROLLERS WHERE THE ROBOT IS INITIALLY PITCHING AT ANGLE $20°$.

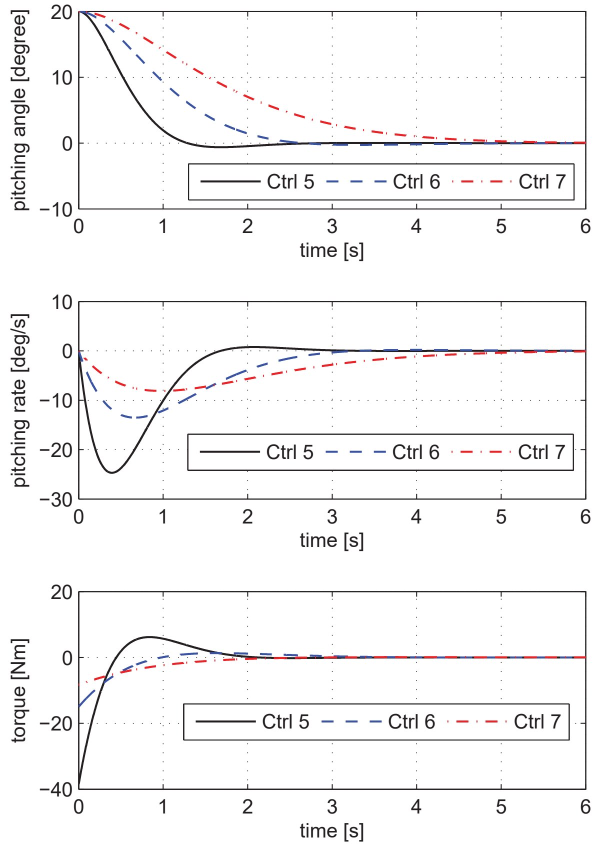

Fig. 3: ROBOT STABILIZATION OF GIVING INITIAL PITCHING ANGLE $20°$.

is given by the following equations:

$$\dot{\hat{x}} = A\hat{x} + Bu + L(y - \hat{y}) \tag{11}$$
$$\hat{y} = C\hat{x} \tag{12}$$

where $\hat{x}$ is the estimated states, $\hat{y}$ is the estimated output, and L is the observer gain. The states-estimation error is defined by:

$$e = x - \hat{x} \tag{13}$$

and the error dynamic is

$$\begin{aligned} \dot{e} &= \dot{x} - \dot{\hat{x}} \\ &= Ax - A\hat{x} - L(y - \hat{y}) \\ &= Ae - L(Cx - C\hat{x}) \\ &= (A - LC)e. \end{aligned} \tag{14}$$

If the matrix $(A - LC)$ is Hurwitz, the states-estimation error will converge to zero by the running time. The matrices A and C are already known but matrix L is unknown. Matrix L can be obtained using linear control theories, and pole placement is one of the applicable methods which is applied in this study. Pole placement is a method to find L such that the poles (eigenvalues) of $(A - LC)$ will be located close to the desired poles location. In this case, the eigenvalues of the of $(A - LC)$ is placed to be close to -8 and -10.

The designed observer is evaluated through computer simulation. Figure 4 shows the simulation result of the states estimation. In the simulation, the robot is initially at the pitching position $20°$, while the initial states of the observer are assumed to be zero. The simulation shows that the estimated states converge to the actual states. This means the observer is able to estimate the actual states. However, in the beginning of pitching angle estimation, it appears a big difference between the estimation and the actual. The difference is due to the different initial values of the actual system and the observer.

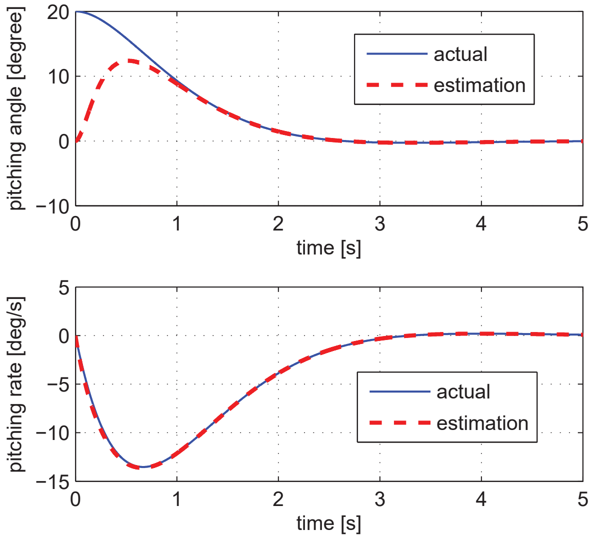

Fig. 4: ESTIMATION OF THE ROBOT PITCHING ANGLE AND THE ROBOT PITCHING RATE.

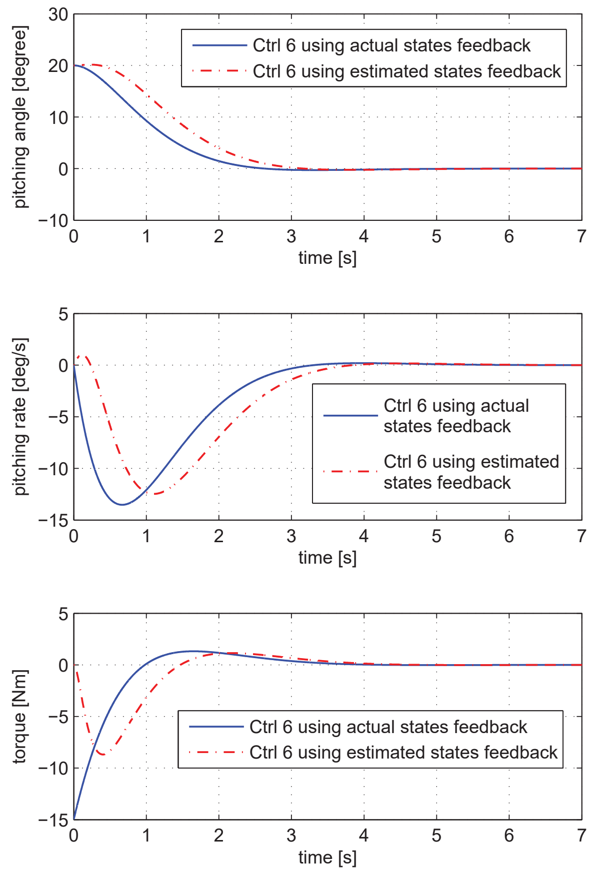

Fig. 5: COMPARISON OF CONTROL SYSTEM PERFORMANCE USING ESTIMATED STATES FEEDBACK AND THE ONE USING ACTUAL STATE FEEDBACK .

V. Control System Performance Using Estimated States

As shown in the previous section that the observer results in good estimation of the actual states, the estimated states can be applied to stabilize the robot. Feedback control law using the estimated states is defined as:

$$\hat{u} = -K\hat{x}, \tag{15}$$

where the $\hat{u}$ is the calculated control action based on the estimated states. Simulation of the robot stabilization using the estimated states is presented and the result is shown in Figure 5. The simulation result shows that the feedback control using estimated states is able to stabilize the robot and requires less torque than the one using actual states feedback. The less torque requirement is because initial value of the estimated states are zero such that the calculation in the feedback control law results in zero, while for the feedback control law using actual states requires torque about $-15Nm$ as the initial actual pitching angle is $20°$. Considering the closed loop response, the performance of using estimated states feedback is slower than the one using actual states feedback, but it is still acceptable.

VI. Conclusion

Control system design of two-wheeled robot (TWR) using linear quadratic regulator (LQR) method has been presented. Using the LQR method, the closed loop control system performance can be adjusted by tuning the performance index and it has been demonstrated in computer simulation. Applying the LQR method results in a full states feedback control which requires measurements of pitching angle and pitching rate angle. In practical application, the pitching rate angle measurement using rate gyro is the most common available measurement in TWR and the pitching angle measurement is not available. Luenberger observer was applied to estimate the unmeasured state (pitching angle) base on the pitching rate angle measurement. It has been demonstrated that the observer is able to estimate the pitching angle. The estimated states can be used in the states feedback control to stabilize the robot as demonstrated in the simulation. The Luenberger observer can be a solution for the unavailable pitching angle sensor in the TWR.

VII. Future Works

This work is a part of research project on autonomous robot. It will be continued to build a two wheels robot and implementing the designed control system. Control performance will be examined in a real-time system.

Acknowledgment

The authors acknowledge the financial support of Universitas Pertamina through internal research grant for research project on Autonomous Robot in 2017.

References

[1] J. Li, X. Gao, Q. Huang, Q. Du, and X. Duan, "Mechanical design and dynamic modeling of a two-wheeled inverted pendulum mobile robot," in *Automation and Logistics, 2007 IEEE International Conference on*. IEEE, 2007, pp. 1614–1619.

[2] T. Nomura, Y. Kitsuka, H. Suemitsu, and T. Matsuo, "Adaptive backstepping control for a two-wheeled autonomous robot," in *ICCAS-SICE, 2009*. IEEE, 2009, pp. 4687–4692.

[3] J. Akesson, A. Blomdell, and R. Braun, "Design and control of yaipan inverted pendulum on two wheels robot," in *Computer Aided Control System Design, 2006 IEEE International Conference on Control Applications, 2006 IEEE International Symposium on Intelligent Control, 2006 IEEE*. IEEE, 2006, pp. 2178–2183.

[4] Y. Kim, S.-H. Lee, and D. H. Kim, "Dynamic equations of a wheeled inverted pendulum with changing its center of gravity," in *Control, Automation and Systems (ICCAS), 2011 11th International Conference on*. IEEE, 2011, pp. 853–854.

[5] N. Uddin, "Lyapunov-based control system design of two-wheeled robot," in *2017 International Conference on Computer, Control, Informatics and its Applications (IC3INA)*, Jakarta, Indonesia, Oct. 2017.

[6] S. Kimura, T. Nakai, H. Nakamura, T. Ibuki, and M. Sampei, "Finite-time control of two-wheeled mobile robot via generalized homogeneous locally semiconcave control lyapunov function," in *Society of Instrument and Control Engineers of Japan (SICE), 2016 55th Annual Conference of the*. IEEE, 2016, pp. 1643–1648.

[7] S. Blažič, "Two approaches for nonlinear control of wheeled mobile robots," in *Control & Automation (ICCA), 2017 13th IEEE International Conference on*. IEEE, 2017, pp. 946–951.

[8] N. G. M. Thao, D. H. Nghia, and N. H. Phuc, "A pid backstepping controller for two-wheeled self-balancing robot," in *Strategic Technology (IFOST), 2010 International Forum on*. IEEE, 2010, pp. 76–81.

[9] C.-C. Tsai and S.-Y. Ju, "Trajectory tracking and regulation of a self-balancing two-wheeled robot: A backstepping sliding-mode control approach," in *SICE Annual Conference 2010, Proceedings of*. IEEE, 2010, pp. 2411–2418.

[10] N. Esmaeili, A. Alfi, and H. Khosravi, "Balancing and trajectory tracking of two-wheeled mobile robot using backstepping sliding mode control: Design and experiments," *Journal of Intelligent & Robotic Systems*, vol. 87, no. 3, pp. 601–613, Sep 2017. [Online]. Available: https://doi.org/10.1007/s10846-017-0486-9

[11] N. Uddin and J. T. Gravdahl, "Active compressor surge control using piston actuation," in *Proc. of the ASME Dynamics System and Control Conference*, Virginia, 2011.

[12] M. Boukens, A. Boukabou, and M. Chadli, "Robust adaptive neural network-based trajectory tracking control approach for nonholonomic electrically driven mobile robots," *Robotics and Autonomous Systems*, vol. 92, pp. 30–40, 2017.

[13] A. A. Saputra, I. A. Sulistijono, and N. Kubota, "Multimodal recurrent neural network (mrnn) based self balancing system: Applied into two-wheeled robot," in *International Conference on Intelligent Robotics and Applications*. Springer, 2016, pp. 596–608.

[14] C.-C. Tsai, H.-C. Huang, and S.-C. Lin, "Adaptive neural network control of a self-balancing two-wheeled scooter," *IEEE Transactions on Industrial Electronics*, vol. 57, no. 4, pp. 1420–1428, 2010.

[15] J.-J. E. Slotine, W. Li *et al.*, *Applied nonlinear control*. Prentice hall Englewood Cliffs, NJ, 1991.

[16] H. K. Khalil, *Noninear Systems*, 3rd ed. Prentice-Hall, New Jersey, 2002.

[17] N. Uddin and J. T. Gravdahl, "Piston-actuated active surge control of centrifugal compressor including integral action," in *Proc. of the 11th International Conference on Control Automation and System*, 2011, pp. 991–996.

[18] F. Grasser, A. D'arrigo, S. Colombi, and A. C. Rufer, "Joe: a mobile, inverted pendulum," *IEEE Transactions on industrial electronics*, vol. 49, no. 1, pp. 107–114, 2002.

[19] M. O. Asali, F. Hadary, and B. W. Sanjaya, "Modeling, simulation, and optimal control for two-wheeled self-balancing robot," *International Journal of Electrical and Computer Engineering (IJECE)*, vol. 7, no. 4, pp. 2008–2017, 2017.

[20] A. Nasir, M. Ahmad, and R. R. Ismail, "The control of a highly nonlinear two-wheels balancing robot: A comparative assessment between lqr and pid-pid control schemes," *World Academy of Science, Engineering and Technology*, vol. 70, pp. 227–232, 2010.

[21] D. S. Naidu, *Optimal control systems*. CRC press, 2002.