Sigma Journal of Engineering and Natural Sciences
Web page info: https://sigma.yildiz.edu.tr
DOI: 10.14744/sigma.2024.00157

Research Article

Modelling and control of a reaction wheel pendulum using a linear quadratic regulator

Haydar Kerem KARHAN[1], Tuğçe YAREN[2,*], Selçuk KİZİR[1]

[1]*Department of Mechatronic Engineering, Kocaeli, Kocaeli University, 41380, Türkiye*
[1]*Department of Electrical and Electronic Engineering, Balikesir University, Balikesir, 10145, Türkiye*

ARTICLE INFO

Article history
Received: 26 September 2023
Revised: 25 November 2023
Accepted: 01 January 2024

Keywords:
LQR; Model-Based Control; Mechatronics; Reaction Wheel Pendulum

ABSTRACT

This paper presents a tutorial-style approach to synthesizing a mechatronic control system from scratch, with a focus on mathematical modeling, real-world verification, model-based control using the Linear Quadratic Regulator (LQR), and rapid control prototyping. The system's equations of motion are derived through Lagrangian mechanics and subsequently linearized. Unknown parameters are estimated using optimization techniques. An LQR controller is designed and implemented on the STM32F4 microcontroller and its performance is rigorously tested against disturbances using MATLAB/Simulink. The Reaction Wheel Pendulum serves as the case study, demonstrating the successful implementation of the LQR controller, with the derived model verified through experimentation. A recovery angle of 20 degrees is obtained.

Cite this article as: Karhan HK, Yaren T, Kizir S. Modelling and control of a reaction wheel pendulum using a linear quadratic regulator. Sigma J Eng Nat Sci 2024;42(6):1907–1915.

INTRODUCTION

The inverted pendulum is a nonlinear and underactuated control problem that is used in developing new control algorithms and in teaching environments. Different types of this problem exist, such as the cart pendulum, the double inverted pendulum, the rotary pendulum, the damped driven pendulum, and the reaction wheel pendulum to name a few [1–4]. These differ from each other mainly by the amount of underactuated joints, and by the actuated rigid body that is used to control the system. Comparisons of the different types of pendulum structures and their potential applications are given in [5]. However, the control problem of these systems remains the same, swinging up the pendulum from its downright position and stabilizing the pendulum in its inverted position.

One of the many types of inverted pendulums is the reaction wheel pendulum (RWP), which was first introduced by Spong et al. [6]. The reaction wheel pendulum consists of two joints with parallel rotation axes, one of which is not actuated. The other joint is typically actuated by a DC motor and has a wheel attached to it which has a symmetric mass distribution, simplifying the analytical solution. The system is controlled using the wheel's inertia, whereby accelerating the wheel a reaction torque is produced.

*Corresponding author.
*E-mail address: tugce.yaren@balikesir.edu.tr

This paper was recommended for publication in revised form by Editor-in-Chief Ahmet Selim Dalkilic

Studies on modeling and control of the RWP have been conducted in detail. Spong et al. [6] were the first to introduce the system and proposed a stabilizing controller using an exact and approximate feedback linearization method, and a swing-up controller using partial feedback linearization on a physical system. Bapiraju et al. [7] proposed multiple linearization methods for the RWP using a fuzzy logic controller to perform stabilization and showed the results in an experimental setup. Srinivas and Behera [8] proposed two new swing-up controllers based on sinusoidal swing-up and interconnection and damping assignment-passivity-based control, implementing them in real-time. Jepsen et al. [9] developed a RWP and described the design process. In addition, swing-up control was implemented using a bang-bang controller, and stabilization was achieved through an observer-based feedback controller. Kadam and Seth [10] presented the modeling, simulation, and LQR control design for a one-wheel robot controlled by a reaction wheel.

Andrievsky [11] focused on the global stabilization of the RWP, ensuring stability for arbitrary initial conditions. An energy-based speed-gradient control scheme was used for the swing-up problem, and sliding mode control was used for stabilization, demonstrating the results through simulations. Gajamohan et al. [12] developed the Cubli, a cube equipped with a reaction wheel on three of its faces. The stabilization challenge involved stabilizing the cube on one of its edges. Similar to the RWP, the Cubli operates in three dimensions, making the balancing and swing-up problem more sophisticated. Nguyen et al. [13] modeled and controlled the RWP using LQR and a fuzzy logic controller, comparing the results through simulation and experimental validation. The system demonstrated stabilization from an initial position of 5 degrees. Trentin et al. [14] modeled and controlled a RWP equipped with two reaction wheels on opposite sides of the pendulum. They employed nonlinear proportional-derivative control and a sliding mode controller to stabilize the system. Montoya et al. [15] designed a controller based on the Lyapunov theory, ensuring stable operation through control input without additional calculations. The results were simulated, and asymptotic stability was guaranteed. Önen and Çakan [16] detailed the modeling and LQR control of the RWP system. Using a designed LQR controller, they simulated the system and presented the results. Cioaca and Flutur [17] designed a 3D-printed RWP system and stabilized it from an initial condition of 2 degrees. They presented results from both the physical system and the simulation.

For pendulum systems other than the RWP, many other methods such as PID, fuzzy logic, sliding mode control, LQR, adaptive control, neural network control, and model predictive control [18–22]. LQR is a fundamental control technique widely employed in various engineering disciplines to achieve optimal control of linear dynamic systems. It is a powerful tool that enables engineers to design controllers that minimize a quadratic cost function, effectively balancing control efforts and system performance. With its solid theoretical foundation and versatility, the LQR control strategy has found applications in diverse fields such as aerospace, robotics, process control, and economics [23].

The significance of disturbance rejection in inverted pendulum systems cannot be overstated. Real-world applications, such as segways, can be accurately modeled as inverted pendulums, and they often encounter disturbances when navigating diverse terrains. While previous studies have conducted minor disturbance tests, this study aims to undertake a more aggressive disturbance rejection test. The evaluation will be based on the recovery angle, representing the system's ability to return to a stable position after being disturbed, without relying on a swing-up structure. Implementing this disturbance-rejecting controller enhances the system's suitability for more challenging environments where disturbances are frequent.

The sections are set up as follows. The equations of motion are derived using Lagrangian mechanics and, the system is linearized in section 2. The modeled system is verified and unmodeled parameters are estimated in section 3. The LQR controller is designed in section 4. In section 5, the hardware used in the system is specified and the designed controller is tested against disturbances.

EQUATIONS OF MOTION AND DYNAMICS

The system diagram is given in Figure 1. The system's stable equilibrium is in the downright position. Throughout this paper, the pendulum angle is taken with reference to the upright position, which is the unstable equilibrium point,

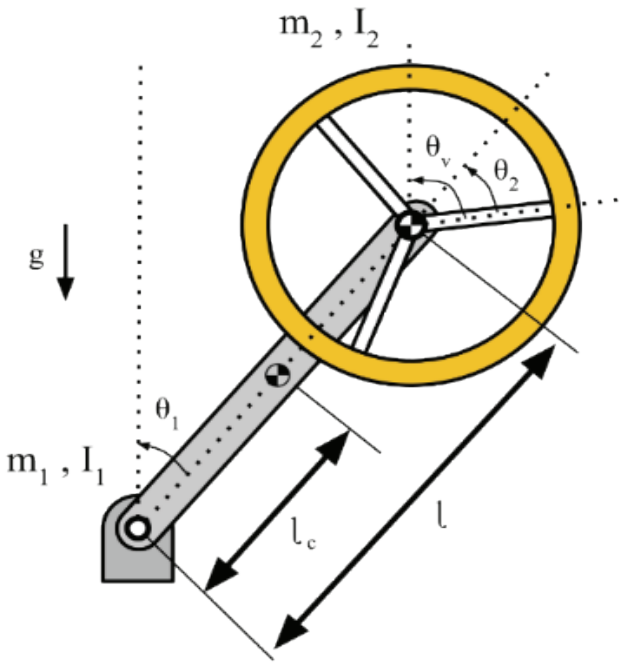

Figure 1. Reaction wheel pendulum diagram.

Table 1. System parameters

Parameter	Description
m_1	Pendulum arm mass
m_2	Reaction wheel mass
l_1	Pendulum arm moment of inertia
l_2	Reaction wheel moment of inertia
θ_1	Pendulum arm angle
θ_2	Reaction wheel angle (with respect to the pendulum arm)
θ_v	Reaction wheel angle (with respect to vertical reference)
l_c	Pendulum arm center of mass
l	Reaction wheel center of mass

around which stabilization is required. All system parameters and the symbols used for them are presented in Table 1.

Mathematical Modeling

The pendulum can be thought of as a planar robot arm with two links the first link is the pendulum arm, and the second link is the reaction wheel. The positions and linear velocities of the center of masses are given in Eqs. (1) and (2) respectively. The linear velocities are obtained by taking the time derivative of the positions.

$$p_1 = \begin{bmatrix} l_c \sin\theta_1 \\ l_c \cos\theta_1 \end{bmatrix} \quad p_2 = \begin{bmatrix} l \sin\theta_1 \\ l \cos\theta_1 \end{bmatrix} \tag{1}$$

$$\dot{p_1} = \begin{bmatrix} l_c\dot{\theta_1} \cos\theta_1 \\ -l_c\dot{\theta_1} \sin\theta_1 \end{bmatrix} \quad \dot{p_2} = \begin{bmatrix} l\,\dot{\theta_1} \cos\theta_1 \\ -l\,\dot{\theta_1} \sin\theta_1 \end{bmatrix} \tag{2}$$

The Lagrangian is the difference between the kinetic and potential energy of the system, which are given in Eqs. (3) and (4) respectively. The kinetic energy of the system (T) consists of linear and angular components. The potential energy (U) only depends on the pendulum arm's angle, since only that parameter affects the vertical displacement of the system.

$$T = \frac{1}{2}\dot{p_1}^T m_1 \dot{p_1} + \frac{1}{2} I_1 \dot{\theta_1}^2 + \frac{1}{2}\dot{p_2}^T m_2 \dot{p_2} + \frac{1}{2} I_2 \dot{\theta_v}^2 \tag{3}$$

$$U = \cos(\theta_1)\,(m_1 g\, l_c + m_2 g\, l) \tag{4}$$

The reaction wheel's angular velocity $\dot{\theta}_v$ must be written with respect to the pendulum arm, since in the real system, it will be measured in that way. Eq. (5) shows how the angle of the pendulum arm and reaction wheel added together, is equal to the reaction wheel angle with respect to the vertical reference. Taking the time derivative of Eq. (5) and substituting it in Eq. (3) yields Eq. (6) which is the system's total kinetic energy. The Lagrangian (L) can thus be found and is given in Eq. (7).

$$\theta_v = \theta_1 + \theta_2 \tag{5}$$

$$T = \frac{1}{2}\dot{\theta_1}^2 (m_1 l_c^2 + m_2 l^2 + I_1 + I_2) + I_2 \dot{\theta_1}\dot{\theta_2} + \frac{1}{2} I_2 \dot{\theta_2}^2 \tag{6}$$

$$L = T - U = \frac{1}{2}\dot{\theta_1}^2 (m_1 l_c^2 + m_2 l^2 + I_1 + I_2) + I_2 \dot{\theta_1}\dot{\theta_2} + \frac{1}{2} I_2 \dot{\theta_2}^2 - \cos(\theta_1)\,(m_1 g\, l_c + m_2 g\, l) \tag{7}$$

Eq. (8) shows the Lagrange-Euler equation where q is a generalized coordinate system and Q is a generalized force vector. Since the Lagrangian is energy-based, the remaining dynamics such as friction, constraints, and/or contact forces can't be modeled directly, hence the use of Q. The generalized coordinates in this system are $q = [\theta_1; \theta_2]$. Therefore, the output of Eq. (8) must be a 2x1 vector. Substituting Eq. (7) in Eq. (8), the equations of motion are presented in Eq. (9).

$$\frac{d}{dt}\frac{\partial L}{\partial \dot{q}} - \frac{\partial L}{\partial q} = Q \tag{8}$$

$$\begin{bmatrix} \alpha\ddot{\theta_1} + I_2\ddot{\theta_2} \\ I_2\ddot{\theta_1} + I_2\ddot{\theta_2} \end{bmatrix} - \begin{bmatrix} \beta \sin\theta_1 \\ 0 \end{bmatrix} = \begin{bmatrix} \tau_1 \\ \tau_2 \end{bmatrix} \tag{9}$$

where $\alpha = (m_1 l_c^2 + m_2 l^2 + I_1 + I_2)$, $\beta = m_1 g\, l_c + m_2 g\, l$. τ_1 and τ_2 are the net joint torques. However, the system is only actuated in τ_2. This results in τ_1 only consisting of the friction in joint 1, which is written as b_1. The control input will be written as τ, and the friction in joint 2 (which depends on the motor) is written as b_2.

Note that Eq. (9) contains coupled dynamic equations, since $\ddot{\theta_1}$ and $\ddot{\theta_2}$ cannot be expressed independently. To uncouple the equations of motion, Eq. (9) is rewritten in the form of Eq. (10).

$$M\begin{bmatrix} \ddot{\theta_1} \\ \ddot{\theta_2} \end{bmatrix} = \begin{bmatrix} -b_1\,\dot{\theta_1} \\ \tau - b_2\,\dot{\theta_2} \end{bmatrix} + \begin{bmatrix} \beta \sin\theta_1 \\ 0 \end{bmatrix}, \quad M = \begin{bmatrix} \alpha & I_2 \\ I_2 & I_2 \end{bmatrix} \tag{10}$$

This is the matrix form of this system, here M is the mass matrix. To uncouple the equations, both sides can be multiplied by the inverse mass matrix. Doing this, yields $\ddot{\theta_1}$ and $\ddot{\theta_2}$ separately which are given in Eqs. (11) and (12) respectively.

$$\ddot{\theta_1} = \frac{1}{I_2 - \alpha}\left(-\beta \sin\theta_1 + b_1\dot{\theta_1} - b_2\dot{\theta_2}\right) + \frac{1}{I_2 - \alpha}\tau \tag{11}$$

$$\ddot{\theta_2} = -\frac{1}{I_2 - \alpha}\left(-\beta \sin\theta_1 + b_1\dot{\theta_1} - \frac{\alpha}{I_2} b_2\dot{\theta_2}\right) + \frac{\alpha}{\alpha\, I_2 - I_2^2}\tau \tag{12}$$

These equations are the system's dynamic equations for a general control input. The nonlinearity of the system comes from the $sin(\theta_1)$ term. Thus, the system can easily be linearized using the Taylor Series expansion, which will be done further in the paper.

The control torque will be supplied by a DC motor, which will be driven over an H-bridge. Therefore, expressing the control input as voltage rather than torque will make it more intuitive for the user. Eqs. (13) and (14) show the torque equation and current equation for a DC motor respectively. Table 2 shows the symbols and their definitions.

$$\tau = n\,K_T\,i \tag{13}$$

$$\mathrm{i} = \frac{V - nK_V\,\dot{\theta}_2}{R} \tag{14}$$

The armature inductance is ignored, since in small power DC motors it is very small. Substituting Eq. (13) in Eq. (12), yields the motor torque with the voltage as the input. This can be used as our control input and substituted in the equations of motion. This will yield the final equations of motion given in Eqs. (15) and (16) with the motor voltage as the control input.

Table 2. Motor parameter definitions

Parameter	Description
V	Input voltage
i	Armature current
R	Armature resistance
K_T	Motor torque constant
K_V	Motor velocity constant
n	Gear ratio

Table 3. System parameter values

Parameter	Description	Value
m_1	Pendulum arm mass	0.043 [kg]
m_2	Reaction wheel mass	0.042 [kg]
l_1	Pendulum arm moment of inertia	0.00012 [$kg\,m^2$]
l_2	Reaction wheel moment of inertia	0.002005 [$kg\,m^2$]
l	Pendulum arm center of mass	0.03 [m]
l_c	Reaction wheel center of mass	0.093 [m]
R	Armature resistance	6.6 [Ω]
K_T	Motor torque constant	0.0931676 [Nm/A]
K_V	Motor velocity constant	0.0139746 [$V/(rad/s)$]
n	Gear ratio	9.68

$$\ddot{\theta}_1 = \frac{1}{I_2 - \alpha}\left(-\beta \sin\theta_1 + b_1\dot{\theta}_1 - \gamma\,\dot{\theta}_2\right) + \frac{nK_T}{R(I_2 - \alpha)}V \tag{15}$$

$$\ddot{\theta}_2 = -\frac{1}{I_2-\alpha}\left(-\beta \sin\theta_1 + b_1\dot{\theta}_1 - \frac{\alpha}{I_2}\gamma\,\dot{\theta}_2\right) + \frac{nK_T\alpha}{\mathrm{R}(\alpha\,I_2 - I_2^2)}V \quad \left(\gamma = b_2 + \frac{n^2K_TK_V}{R}\right) \tag{16}$$

The parameters such as link lengths, center of mass location, mass and moments of inertia were obtained using Solidworks. All system parameters and their values are given in Table 3.

The actuator is chosen as a 12 V Pololu #4862 brushed DC motor with a built-in encoder and gearbox. Therefore, the control input V is saturated to [-12, 12]. This needs to be included in the simulation, or the controller can output extreme voltage values.

State Space Representation and Linearization

In order to simulate the system, it is useful to represent the nonlinear system in state variables. The system is expressed with state variables in Eqs. (17), (18) and (19). Note that the reaction wheel angle is omitted, since it never appears in the equations of motion, thus $x_3 = \dot{\theta}_2$.

$$\dot{x_1} = x_2 \tag{17}$$

$$\dot{x_2} = \frac{1}{I_2 - \alpha}(-\beta \sin x_1 + b_1 x_2 - \gamma\,x_3) + \frac{nK_T}{R(I_2 - \alpha)}V \tag{18}$$

$$\dot{x_3} = -\frac{1}{I_2 - \alpha}\left(-\beta \sin x_1 + b_1 x_2 - \frac{\alpha}{I_2}\gamma\,x_3\right) + \frac{nK_T\alpha}{\mathrm{R}(\alpha\,I_2 - I_2^2)}V \tag{19}$$

The system should be linearized, in order to use linear control methods. To linearize the system, the Taylor Series Expansion of the sine term is given in Eq. (20) and can be substituted in Eqs. (18) and (19).

$$\sin(x_1) = x_1 - \frac{x_1^3}{3!} + \frac{x_1^5}{5!} - \frac{x_1^7}{7!} + \cdots \tag{20}$$

Since the goal of the system is to control it in the upright position, x_1 will be close to 0. Therefore $sin(x_1) \approx x_1$, which gets rid of the nonlinearity of the equations. The state space representation of the system is given in Eq. (21).

$$\begin{bmatrix}\dot{x_1}\\ \dot{x_2}\\ \dot{x_3}\end{bmatrix} = \begin{bmatrix}0 & 1 & 0\\ \frac{-\beta}{I_2-\alpha} & \frac{b_1}{I_2-\alpha} & \frac{-\gamma}{I_2-\alpha}\\ \frac{\beta}{I_2-\alpha} & \frac{-b_1}{I_2-\alpha} & \frac{\alpha\,\gamma}{I_2^2-\alpha I_2}\end{bmatrix}\begin{bmatrix}x_1\\ x_2\\ x_3\end{bmatrix} + \begin{bmatrix}0\\ \frac{nK_T}{R(I_2-\alpha)}\\ \frac{nK_T\alpha}{\mathrm{R}(\alpha\,I_2-I_2^2)}\end{bmatrix}V \qquad y = \begin{bmatrix}1 & 0 & 0\\ 0 & 1 & 0\\ 0 & 0 & 1\end{bmatrix}\begin{bmatrix}x_1\\ x_2\\ x_3\end{bmatrix} \tag{21}$$

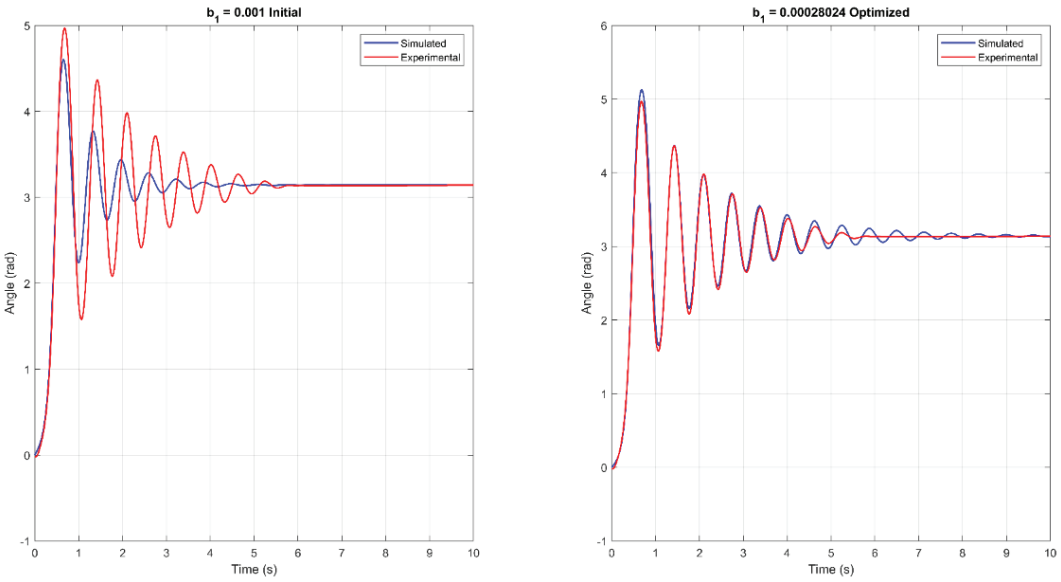

Figure 2. Optimization results.

Using Eq. (21), linear controllers can be designed and tested in simulation and on the real system. The control input V is saturated according to the motor operating limits and/or the supply voltage limits.

Parameter Estimation

In order to use model-based controller design methods, the real system and simulated system should perform as similarly as possible. All measurable parameters were given in Table 3. However, the equations of motion also include two friction terms.

The viscous friction coefficient of the DC motor can be found parametrically by using the DC motor's specifications and is found as b_2 = 0.0012. The viscous friction coefficient of the pendulum arm, however, has to be experimentally verified. In order to do this, the pendulum was manually stabilized in the upright position. Afterwards, a small perturbation was applied to move the pendulum, after which the angle of the pendulum arm was recorded using an optical encoder. This was performed four times and similar results were obtained.

With the experimental results the model can be simulated using Eqs. (11) and (12) and optimized to fit the simulated response as closely to the experimental results as possible. This optimization is done using the nonlinear least squares method, with a sum squared error cost function. The error is defined in Eq. (22).

$$e(t) = y_{exp}(t) - y_{sim}(t) \tag{22}$$

MATLAB Simulink is used to solve the optimization problem. An inital guess of 0.001 for b_1 was used. The results are given in Figure 2. The optimized result gives a mean absolute percentage error of 94.1%, which is the error in Eq. (22) divided by the experimental output and summed over all time steps. Note that the simulated response reaches steady state later than the experimental result in the optimized response. This is due to unmodelled nonlinear effects in the real system, such as the tension that occurs in the motor's electrical wires. This is irrelevant, however, since the simulated and experimental results are very similar up to 4 seconds, which is significantly more than the actuator's response time.

Controller Design

Now that an accurate system model is obtained, it can be used to design a controller. A Linear Quadratic Regulator (LQR) is chosen as the system's controller in this paper. LQR is an optimal control method based on full state feedback, which means all states must be accessible. This method tries to minimize a cost function J, given in Eq. (23).

$$J = \int_0^\infty (x^T Q x + u^T R u) dt \tag{23}$$

Q is the state-cost weighted matrix, R is the input-cost weighted matrix and u is the control input $u = -Kx$. This optimal control problem's goal is to find an appropriate K vector that minimizes the cost function. To determine the Q and R matrices, a good first step is Bryson's rule [24] which is given in Eq. (24).

$$Q_{ii} = \frac{1}{x_{i_{max}}^2}, i = 1,2,\dots,n \qquad R = \frac{1}{u_{max}^2} \tag{24}$$

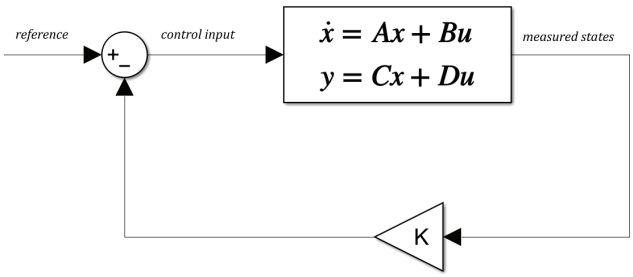

Figure 3. LQR controller block diagram.

Q_{ii} are the elements of a diagonal matrix. $x_{i_{max}}$ is the maximum error for state i. u_{max} is the maximum control input.

The max state errors were defined as 0.005°, 0.01 rpm and 0.1 rpm for x_1, x_2 and x_3 respectively. These were obtained by iteratively testing the system using different Q and R matrices, where the best performing ones are given in Eq. (25). The maximum control input is the saturated 12 Volts of the motor.

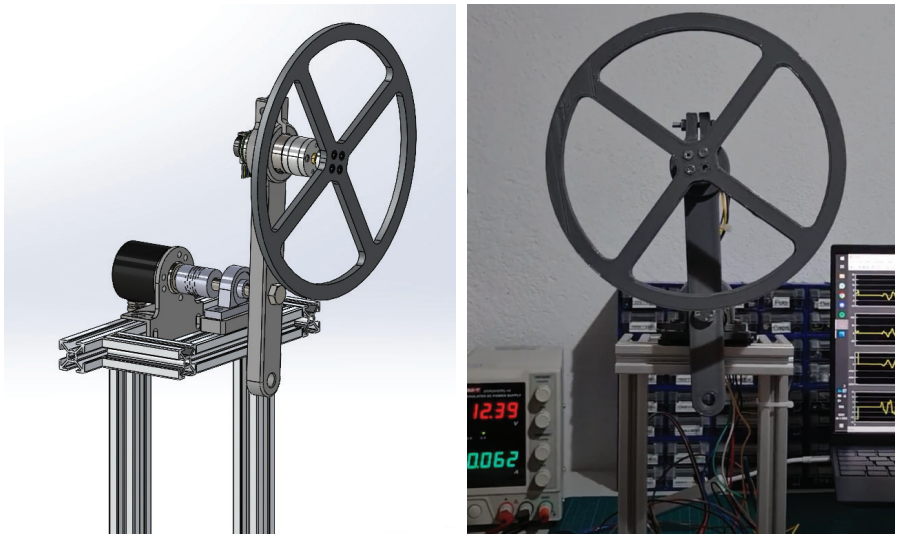

Figure 4. CAD design and real system.

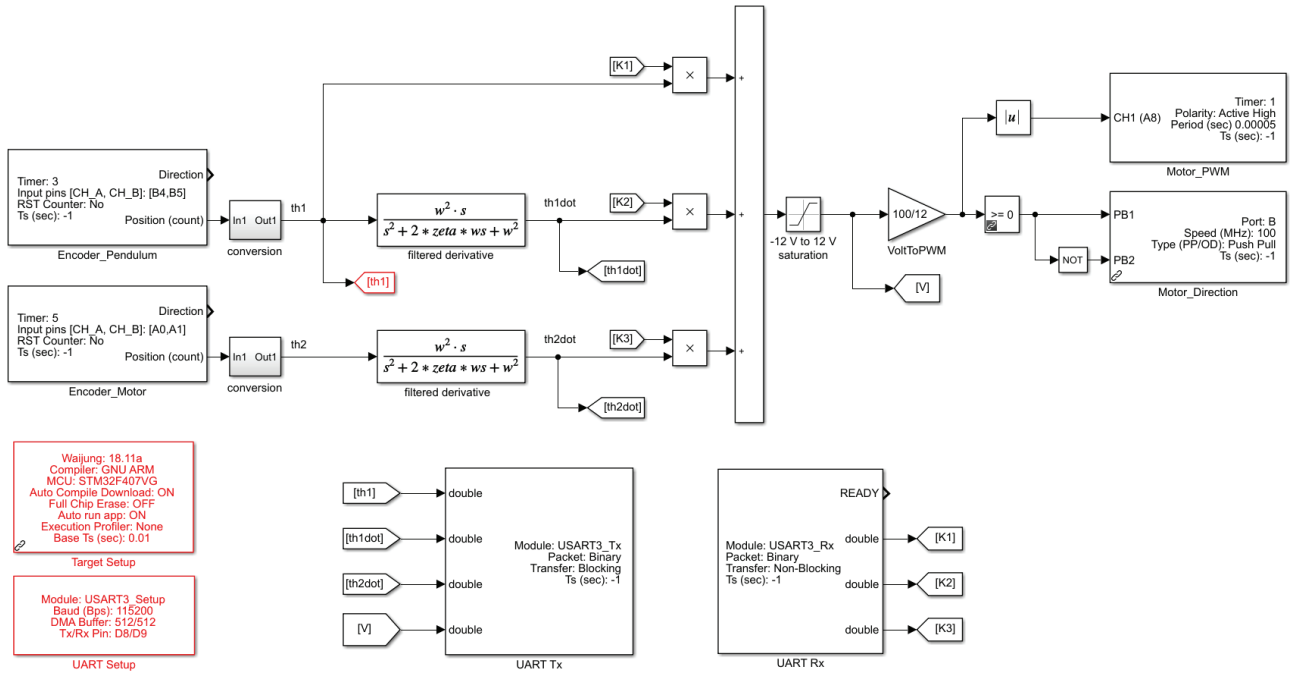

Figure 5. Simulink block diagram using Waijung Blockset.

$$Q = 10^6 \begin{bmatrix} 131.31 & 0 & 0 \\ 0 & 0.9119 & 0 \\ 0 & 0 & 0.0091 \end{bmatrix}, \quad R = 0.0069 \quad (25)$$

Using these Q and R matrices, the K gain matrix can be computed using MATLAB and used as a feedback gain on all the measured states. K was found as [-158.62, -15.02, -1.30]. In Figure 3 the system is represented in a block diagram. The reference in the modeled RWP system is zero since stabilization is intended around this point.

HARDWARE AND EXPERIMENTAL RESULTS

The CAD design in SOLIDWORKS and real system is given in Figure 4. The STM32F4 discovery board was used as the microcontroller. The code was generated using the Waijung Blockset by Aimagin [25] in MATLAB Simulink. The motor is a Pololu #4862 geared brushed DC motor. The motor was driven using a L298N dual channel H-bridge driver. The pendulum arm angle was read using a 2048 CPR optical encoder connected to the shaft via a coupler. The Simulink block diagram is given in Figure 5. The filtered derivative block is a transfer function block where $\omega = 10\frac{rad}{s}$ and $\zeta = 0.7$. This Simulink block diagram was directly embedded on the microcontroller.

The disturbance rejection experiment results are given in Figure 6 and 7, respectively. The peaks in the pendulum arm angle (θ_1), are the points where the system was hit by an object. The stabilization of the system has an oscillatory behavior of 8 degrees. Furthermore, the maximum recovery angle is around 20 degrees as can be seen in Figure 7. The controller exhibits strong disturbance-rejecting performance, as it can stabilize the system even when it deviates significantly from the linearization point, despite being designed based on a linearized model. The behavior of the system against disturbances can be seen in the following video link https://www.youtube.com/watch?v=bb33BNByxlQ.

Nevertheless, enhancing the stability performance is feasible by adopting an alternative controller structure, as demonstrated in [9] with the implementation of an observer-based feedback controller. Directly taking derivatives can result in noise amplification. To mitigate this, a second-order low-pass filter is incorporated with the derivative, introducing additional delays. Therefore, the use of an observer-based feedback controller has the potential to improve stabilization performance.

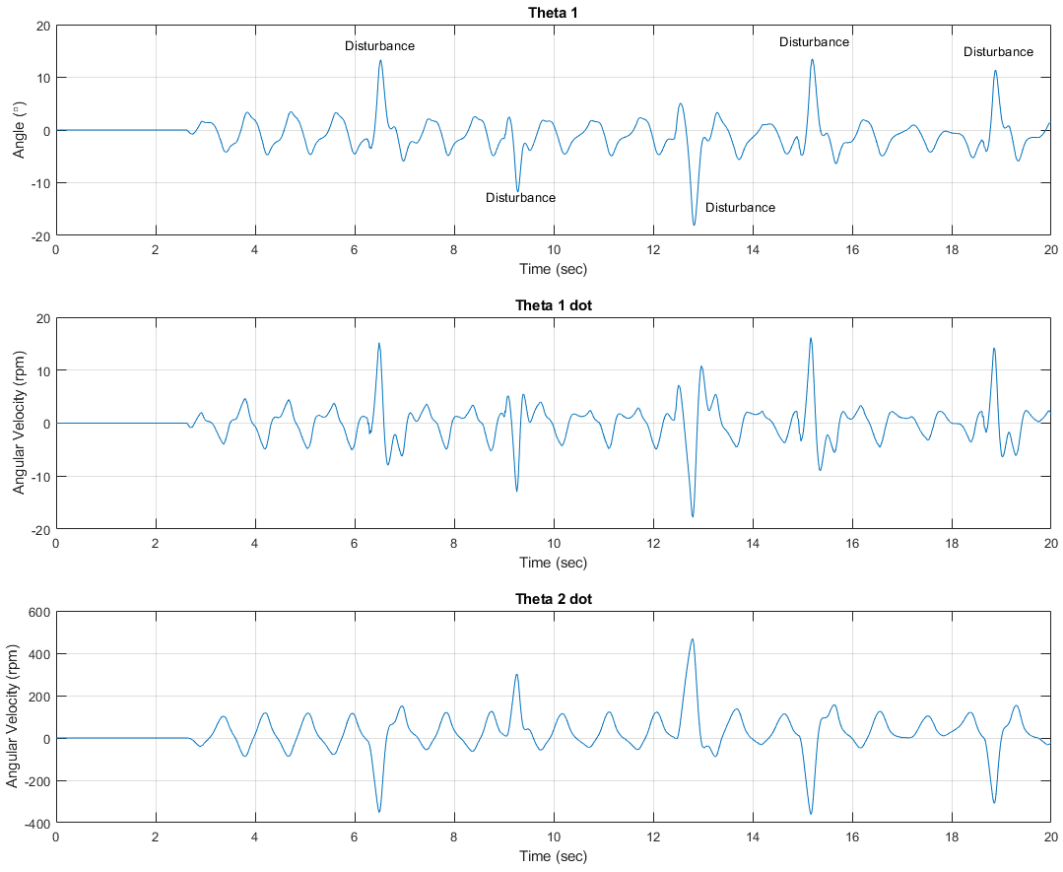

Figure 6. Disturbance test-1.

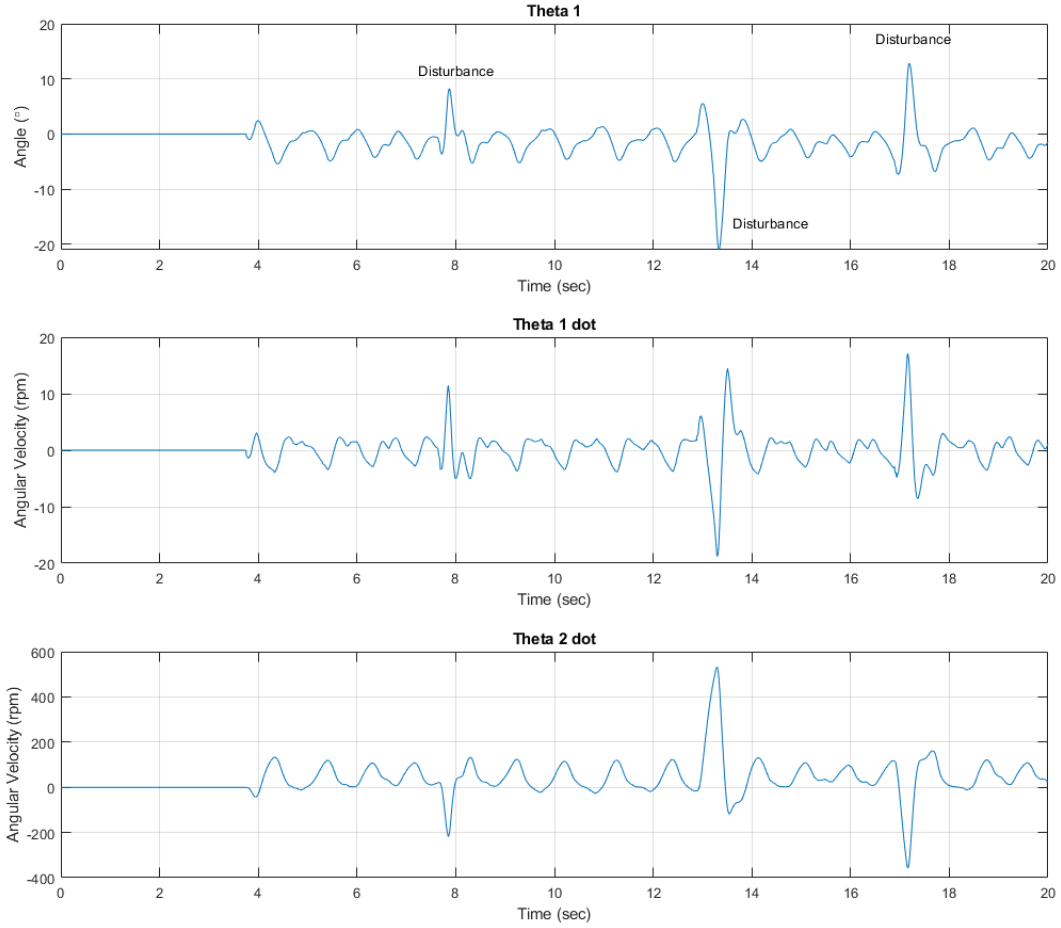

Figure 7. Disturbance test-2.

CONCLUSION

In this paper, the Reaction Wheel Pendulum was mathematically modeled and the system's equations of motion were derived. The system was linearized and was represented in state space. Furthermore, the only unmodeled parameter, friction in the pendulum arm, was estimated using optimization techniques by measuring the system's natural response without any control inputs and fitting the simulated response to this. Using the complete and verified model, an LQR controller was designed and implemented on a microcontroller, after which a disturbance test was done to validate the controller's performance against the supplied disturbances. The system reached a maximum recovery angle of 20 degrees, and had oscillatory steady state behavior of 8 degrees.

The controller can be enhanced by eliminating the oscillatory steady state response, which could be achieved using different Q and R matrices, thus different K gains. Also, using a different controller structure can improve the stabilization performance. Furthermore, adding imbalances to the reaction wheel (such as small weights on one side) and designing and testing controllers against these can be possible areas of improvement. Compared to previous literature, this paper presents a new metric to characterize performance against disturbances, the recovery angle, which is the maximum angle the system can reach before it can return to its stable equilibrium without a swing-up procedure.

AUTHORSHIP CONTRIBUTIONS

Authors equally contributed to this work.

DATA AVAILABILITY STATEMENT

The authors confirm that the data that supports the findings of this study are available within the article. Raw data that support the finding of this study are available from the corresponding author, upon reasonable request.

CONFLICT OF INTEREST

The author declared no potential conflicts of interest with respect to the research, authorship, and/or publication of this article.

ETHICS

There are no ethical issues with the publication of this manuscript.

REFERENCES

[1] Yaren T, Kizir S. Power-based modelling and control: experimental results on a cart-pole double inverted pendulum. Turk J Electr Eng Comput Sci 2021;29:1736-1750. [CrossRef]

[2] Chinelato CIG, Neves GPD, Angelico BA. Safe control of a reaction wheel pendulum using control barrier function. IEEE Access. 2020;8:160315-24. [CrossRef]

[3] Saleem O, Abbas F, Iqbal J. Complex fractional-order LQIR for inverted-pendulum-type robotic mechanisms: design and experimental validation. Mathematics. 2023;11:913. [CrossRef]

[4] Şahin H. Design of experiments optimization application in physics: a case study of the damped driven pendulum experiment. Sigma J Eng Nat Sci. 2021;39:322-330. [CrossRef]

[5] Hazem ZB, Bingül Z. Comprehensive review of different pendulum structures in engineering applications. IEEE Access. 2023;11:42862-42880. [CrossRef]

[6] Spong MW, Corke P, Lozano R. Nonlinear control of the reaction wheel pendulum. Automatica 2001;37:1845-1851. [CrossRef]

[7] Bapiraju B, Srinivas KN, Kumar PP, Behera L. On balancing control strategies for a reaction wheel pendulum. In: Proceedings of the IEEE INDICON; 2004. p. 199-204. [CrossRef]

[8] Srinivas KN, Behera L. Swing-up control strategies for a reaction wheel pendulum. Int J Syst Sci. 2006;39:1165-1177. [CrossRef]

[9] Jepsen F, Soborg A, Pedersen AR, Yang Z. Development and control of an inverted pendulum driven by a reaction wheel. In: Proceedings of the International Conference on Mechatronics and Automation; 2009. p. 2829-2834. [CrossRef]

[10] Kadam SN, Seth B. LQR controller of one wheel robot stabilized by reaction wheel principle. In: Proceedings of the International Conference on Instrumentation Control and Automation; 2011. p. 299-303. [CrossRef]

[11] Andrievsky BR. Global stabilization of the unstable reaction-wheel pendulum. Autom Remote Control. 2011;72:1981-1993. [CrossRef]

[12] Gajamohan M, Muehlebach M, Widmer T, D'Andrea R. The Cubli: a reaction wheel-based 3D inverted pendulum. In: Proceedings of the IEEE European Control Conference (ECC); 2013. p. 268-274. [CrossRef]

[13] Nguyen BH, Cu MP, Nguyen MT, Tran MS, Tran HC. LQR and fuzzy control for reaction wheel inverted pendulum model. Robot Manag 2019;24:19-23.

[14] Trentin JFS, Da Silva S, De Souza RJM, Schaub H. Inverted pendulum nonlinear controllers using two reaction wheels: design and implementation. IEEE Access 2020;8:74922-74932. [CrossRef]

[15] Montoya OD, Gil-González W. Nonlinear analysis and control of a reaction wheel pendulum: Lyapunov-based approach. Eng Sci Technol Int J 2020;23:21-29. [CrossRef]

[16] Önen Ü, Çakan A. Multibody modeling and balance control of a reaction wheel inverted pendulum using LQR controller. Int J Robot Control Syst. 2021;1:84-89. [CrossRef]

[17] Cioaca RA, Flutur C. Reaction wheel control of a 3D printed inverted pendulum. In: Proceedings of the IEEE International Conference on Control Systems and Computer Science (CSCS); 2021. p. 57-61. [CrossRef]

[18] Gani A, Keçecioğlu ÖF, Açıkgöz H, Şekkeli M. Fuzzy logic controller design based on Sugeno inference method for nonlinear inverted pendulum dynamical system. Sigma J Eng Nat Sci 2017;8:19-30.

[19] Aydın M, Yakut O, Alli H. Implementation of sliding mode control with artificial neural network to the rotary inverted pendulum system. J Eng Nat Sci 2013;5:39-50.

[20] Rabah M, Rohan A, Kim SH. Comparison of position control of a gyroscopic inverted pendulum using PID, fuzzy logic, and fuzzy PID controllers. Int J Fuzzy Log Intell Syst. 2018;18:103-110. [CrossRef]

[21] Zabihifar SH, Yushchenko AS, Navvabi H. Robust control based on adaptive neural network for rotary inverted pendulum with oscillation compensation. Neural Comput Appl 2020;32:14667-14679. [CrossRef]

[22] Chu TD, Chen CK. Design and implementation of model predictive control for a gyroscopic inverted pendulum. Appl Sci 2017;7:1272. [CrossRef]

[23] Yaren T, Kizir S. Impact analysis of LQR controller parameters on system dynamics: double inverted pendulum. Konya J Eng Sci 2020;8:175-191. [CrossRef]

[24] Kizir S. Real-time full state feedback control of a seesaw system based on LQR. J Polytechnic 2019;22:1023-1030. [CrossRef]

[25] Kizir S, Yaren T, Kelekçi E. Matlab Simulink destekli gerçek zamanlı kontrol: teori ve mühendislik uygulamaları. Ankara: Seçkin Yayıncılık; 2019.