

Experimental Swing-Up Control of Advanced Sliding and Energy-based Modes for Pendubot

Minh-Duy Tran¹, Minh-Phu Trinh², Nguyen-Son Do³, Thai-Chan Phan⁴, Tan-Bao Chau Ngo⁵, Viet-Thuan Nguyen⁶, Viet-Dung Ngo⁷, Ngoc-Quan Hoang⁸, Tan-Phong Trinh⁹, Thi-Hong-Lam Le^{10,*}
1, 2, 3, 4, 5, 6, 7, 8, 9, 10 Ho Chi Minh City University of Technology and Education (HCMUTE), Ho Chi Minh City, Vietnam
Email: ¹ 21151087@student.hcmute.edu.vn, ² 18151103@student.hcmute.edu.vn, ³ 18151113@student.hcmute.edu.vn, ⁴ 20151075@student.hcmute.edu.vn, ⁵ 21145081@student.hcmute.edu.vn, ⁶ 21142188@student.hcmute.edu.vn, ⁷ 21142507@student.hcmute.edu.vn, ⁸ 21142582@student.hcmute.edu.vn, ⁹ 21142572@student.hcmute.edu.vn, ¹⁰ lamth@hcmute.edu.vn
*Corresponding Author

Abstract—This study focuses on the implementation and comparative evaluation of two swing-up control strategies—Energy-Based Methods (EBM) and Advanced Sliding Mode Control (ASMC)—for pendubot, a nonlinear two-link robotic system. While previous research has extensively explored balancing algorithms for this model, swing-up strategies have primarily been analyzed through simulations, with limited application to real-world systems. This research addresses this gap by deploying both EBM and ASMC on a physical pendubot model. Practical results are presented to provide the most accurate evaluation of the control quality of each algorithm.

Keywords—LQR; Energy-based Method; Sliding Mode Control; Pendubot; Swing-up

I. INTRODUCTION

The pendubot is a typical nonlinear mechanical system consisting of two links connected to each other [1]. The first link is fixed to a rotating shaft controlled by a motor, while the second link can freely rotate around the other end of the first link. Many previous studies have focused on balancing the pendubot in an upright position using techniques such as the Linear Quadratic Regulator (LQR) for two-link systems [2], PID control [3], Model Predictive Control (MPC) [4], Fuzzy [5] and Sliding Mode Control (SMC) [6]. It is evident that while there are many studies on balancing algorithms, there has been less focus on swing-up algorithms. An important step in operating this system is the "swing-up" problem — the algorithm that enables the pendubot to automatically move from a resting or free-falling state to the upright position.

In the field of swing-up, various methods have been proposed. Research on swing-up for systems such as the acrobot [7] and the inverted pendulum [8] shows that this problem has been widely studied across different systems. Studies on swing-up for pendubots have also been carried out by many researchers, focusing on algorithms such as Energy-Based Methods (EBM) [9], Sliding Mode Control (SMC) [10], and Fuzzy Control [11]. The research team observed that the most popular method is EBM, while the fastest method is ASMC [12]. However, previous studies have primarily focused on theoretical analysis or simulations, while the practical effectiveness of these two methods on an actual pendubot has not been fully evaluated. This highlights the need for research on the

practical application of swing-up algorithms to real-world models.

In this study, we implemented and evaluated two swing-up methods, EBM and ASMC, on a real pendubot model. EBM was selected for its simplicity, ease of implementation, and adaptability to nonlinear dynamic systems. At the same time, ASMC was chosen for its advantages in handling disturbances and uncertainties, making it suitable for real systems with many non-ideal factors. The novelty of this study lies in its direct application to a real pendubot model, providing a comprehensive and practical perspective on these two methods. This research represents a fresh contribution, expanding the understanding of the practical application of modern control methods.

II. MATHEMATICAL MODEL

Fig. 1 depicts a pendubot, a two-link robotic system commonly used in nonlinear dynamics and control studies. It consists of two links connected by joints: an active joint at the base, which is controlled by torque, and a passive joint, which moves freely without direct control. System parameters include lengths and masses of links, position of their centers of mass, and angular positions of links relative to horizontal axis. Gravity acts vertically downward. Pendubot is a classic under-actuated system, making it ideal for exploring advanced control strategies such as swing up and balance. System parameters are presented specifically in Table 1.

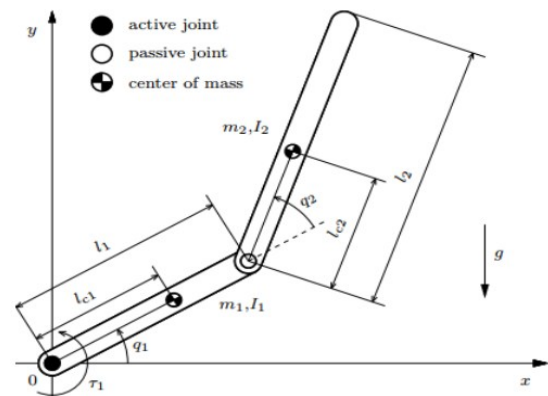


Fig. 1. Structure of the pendubot description [13]

Table 1. System parameters

Symbol	Description	Value	Unit
m_1	Mass of link 1	0.16	kg
l_1	Length of link 1	0.2	m
l_{c1}	Distance from active joint to center of mass of link 1	0.1	m
I_1	Moment of inertia of link 1	0.00222	kg.m ²
m_2	Mass of link 2	0.066	kg
l_2	Length of link 2	0.22	m
l_{c2}	Distance from passive joint to center of mass of link 2	0.11	m
I_2	Moment of inertia of link 2	0.00106	kg.m ²
K_t	Torque constant	0.0198	Nm
K_b	Back EFM constant (Force constant of motor)	0.0198	V
R_m	Resistor of rotor	6.835	Ω
J_m	Moment of inertia of rotor	0.000134	kg.m ²
C_m	Viscous friction constant	0.000048	Nm

Define the variables as follows to facilitate calculations in the subsequent equations:

$$\begin{aligned} \beta_1 &= m_1 l_{c1}^2 + m_2 l_1^2 + I_1; \beta_2 = m_2 l_{c2}^2 + I_2; \\ \beta_3 &= m_2 l_1 l_2; \beta_4 = m_1 l_1 + m_2 l_1; \beta_5 = m_2 l_2 \end{aligned} \quad (1)$$

In d, the dynamic equations of the Pendubot based on the Euler-Lagrange formulation are as follows [9]:

$$D(q)\ddot{q} + C(q, \dot{q})\dot{q} + G(q) = \tau \quad (2)$$

where:

$$D(q) = \begin{bmatrix} \beta_1 + \beta_2 + 2\beta_3 \cos q_2 & \beta_2 + \beta_3 \cos q_2 \\ \beta_2 + \beta_3 \cos q_2 & \beta_2 \end{bmatrix} \quad (3)$$

$$C(q, \dot{q}) = \begin{bmatrix} -\beta_3 \sin(q_2) \dot{q}_2 & -\beta_3 \sin q_2 \dot{q}_2 - \beta_3 \sin q_2 \dot{q}_1 \\ \beta_3 \sin(q_2) \dot{q}_1 & 0 \end{bmatrix} \quad (4)$$

$$G(q) = \begin{bmatrix} \beta_4 g \cos q_1 + \beta_5 \cos(q_1 + q_2) \\ \beta_5 \cos(q_1 + q_2) \end{bmatrix} \quad (5)$$

$$q = \begin{bmatrix} q_1 \\ q_2 \end{bmatrix} \quad \tau = \begin{bmatrix} \tau_1 \\ 0 \end{bmatrix} \quad (6)$$

From (1) to (6) we obtain:

$$\begin{bmatrix} \ddot{q}_1 \\ \ddot{q}_2 \end{bmatrix} = D(q)^{-1} \tau - D(q)^{-1} C(q, \dot{q}) \dot{q} - D(q)^{-1} g(q) \quad (7)$$

To match the experiment, we need an equation relating torque to voltage as follows:

$$u = u_1 \tau - u_2 \dot{q}_1 - u_3 \ddot{q}_1 \quad (8)$$

where $u_1 = \frac{K_t}{R_m}$; $u_2 = \frac{K_t K_b}{R_m}$; $u_3 = J_m$

III. CONTROL ALGORITHMS

A. Swing up control

Swing-up for pendubot refers to the process of bringing the system from a downward unstable position ($q_1 = -\frac{\pi}{2}$; $q_2 = 0$) to an upright ($q_1 = \frac{\pi}{2}$; $q_2 = 0$) and balanced state. The positions of pendubot are shown in Fig. 2. This is a challenging task because pendubot, being an under-actuated system, cannot direct way to apply force pendulum in upright position. The goal of the Swing-up process is to generate sufficient energy to flip pendulum upwards using the actuators, typically by exploiting dynamics of the system and carefully controlling the torque applied to the joints. Once pendubot reaches an upright position, it can then be stabilized using other control strategies, like LQR or SMC.

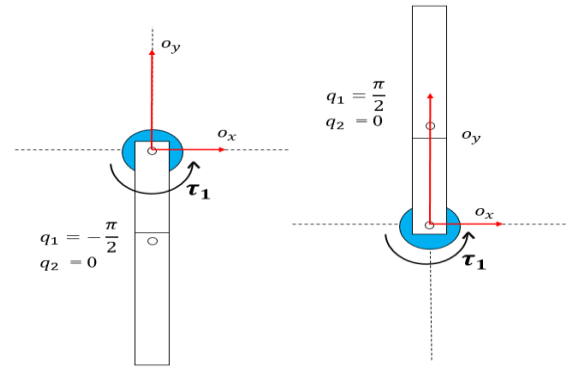


Fig. 2. The positions of pendubot

A swing-up controller works by first employing either an EBM or ASMC to build rotational kinetic energy in link 1. As system nears equilibrium position, it transitions to a balance controller once the predefined Switch Condition is satisfied. The integrated algorithm proposed in this paper combines EBM-LQR and ASMC-LQR.

Swing up process for the pendubot is shown in Fig. 3. The angle measurements of link 1 and link 2 are used as feedback to evaluate the Switch Condition. To define an appropriate Switch Condition, it is necessary to analyze the state variables near the equilibrium position and determine a condition that fits the system dynamics effectively.

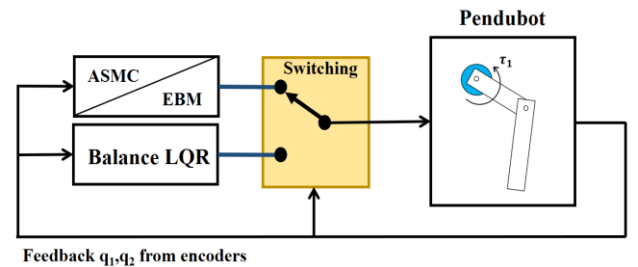


Fig. 3. Swing-up control structure

B. Energy-Based Method

EBM is an effective control strategy, especially for controlling under-actuated systems like pendubot. This method focuses on controlling the transformation of energy between different energy forms in the system, such as kinetic and potential energy, to achieve the desired control

objectives. One of the key advantages of this method is its ability to optimize energy usage, minimizing unnecessary energy consumption. Additionally, this approach is often simpler than other control techniques, as it does not require a detailed model of the system but relies on basic energy factors. This makes EBM an ideal choice for controlling systems with complex dynamics and limited direct control inputs, such as pendubot.

In [14], energy calculations are presented as follows: Total energy of pendubot is:

$$E(q, \dot{q}) = \frac{1}{2} \dot{q}^T D(q) \dot{q} + P(q) \quad (9)$$

Energy of pendubot at upright position is

$$E_p = (\beta_4 + \beta_5)g \quad (10)$$

When $E = E_p$ the system reaches a stable state and moves to equilibrium from (9) and (10), we obtain:

$$\lim_{t \rightarrow \infty} E(q, \dot{q}) = E_p, \lim_{t \rightarrow \infty} (q_1) = \frac{\pi}{2}, \lim_{t \rightarrow \infty} (\dot{q}_1) = 0 \quad (11)$$

In [15], swing up controller is:

$$\tau_1 = \frac{-K_p \left(q_1 - \frac{\pi}{2} \right) + K_D B^T D^{-1} (C \dot{q} + G) - K_V \dot{q}_1}{\Psi} \quad (12)$$

Where

$$\Psi = K_E (E - E_p) + K_D B^T D^{-1} B \neq 0 \quad (13)$$

According to [15], switching condition is chosen as:

$$\left(\frac{\pi}{2} - q_1 \right) + 0.1 \dot{q}_1 + q_2 + 0.1 \dot{q}_2 < \xi \quad (14)$$

The control parameters following values: $K_e = 246.3$, $K_y = 187$, $K_p = 652.1$ and $K_d = 3$. The selection of the parameter ξ also plays a crucial role in determining the system's control quality. If ξ is set too high, the timing of the switch may hinder the LQR controller from stabilizing the system. This is because the angular displacement q_1 remains far from the desired value, and the angular velocity does not yet meet the conditions required for stable control. Conversely, a lower level of value can improve the system's behavior after the swing-up phase. However, setting ξ too low creates challenges in achieving the necessary conditions during the swing-up process. After many experiments on the model, this study determined the optimal value for $\xi = 1.8$.

C. Advanced Sliding Mode Controller

SMC offers several benefits compared to other control methods. One key advantage is that it does not rely on an exact model, making it well-suited to handling dynamic systems with modeling uncertainties and disturbances. As a nonlinear controller, it outperforms linear controllers in terms of efficiency. The advanced sliding mode controller builds upon the standard version by introducing a slight

improvement: it enables q_1 to track a predefined trajectory signal r for the swing-up motion. This design ensures greater stability compared to allowing q_1 to rotate at an undefined angle.

The state-space equations of the pendubot system are defined as follows:

$$\dot{x}_1 = x_2; \dot{x}_2 = f_1 + g_1 \tau; \dot{x}_3 = x_4; \dot{x}_4 = f_2 + g_2 \tau \quad (15)$$

The state variables are set as follows:

$$x_1 = q_1 - \frac{\pi}{2}; x_2 = \dot{q}_1; x_3 = q_2; x_4 = \dot{q}_2 \quad (16)$$

The error e is now defined as follows [12]:

$$e = x_1 + x_3 - r; \dot{e} = \dot{x}_1 + \dot{x}_3 - \dot{r} \quad (17)$$

Construct the sliding surface based on (16) and (17):

$$s = ce + \dot{e} = c(x_1 + x_3 - r) + (\dot{x}_1 + \dot{x}_3 - \dot{r}) \quad (18)$$

At the TOP position, the reference signal r , which q_1 needs to track, is given by (19) and represented in Fig. 4:

$$r = \begin{cases} 3.3 \sin(4.35t) - \frac{\pi}{2} : t < \frac{\pi}{8.7} \\ \frac{\pi}{2} : t \geq \frac{\pi}{8.7} \end{cases} \quad (19)$$

From (15) to (19) and follow as [12] we obtain:

$$\tau = \frac{-K \text{sign}(s) - cx_2 - cx_4 + c\dot{r} + \dot{r} - f_1 - f_2}{g_1 + g_2} \quad (20)$$

Control parameters are as follows: $c = 4$ and $K = 9547$. For ASMC, the switching condition is chosen based on the angles of link 1 and link 2 with the conditions $|q_1| < 0.5$ & $|q_2| < 0.5$. After many experiments on the model, the value 0.5 was chosen to be the most suitable for triggering the switch.

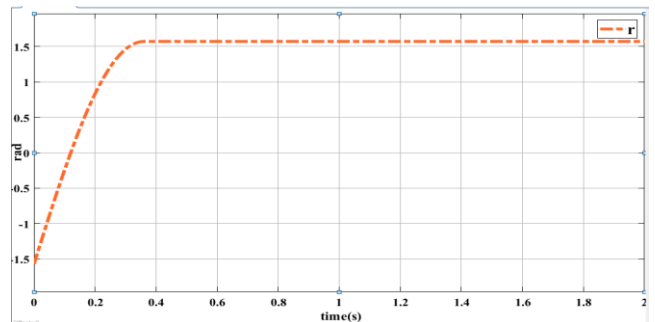


Fig. 4. The reference signal r (rad)

D. LQR

LQR control algorithm is a widely used optimization method for stabilizing nonlinear systems around an equilibrium point. For pendubot, LQR calculates the optimal control signal by minimizing a cost function that balances state error and control effort. LQR is often combined with other algorithms, such as Sliding Mode

Control or energy-based strategies, to perform the swing-up process before stabilizing the system at the desired position. This combination leverages the strengths of both methods to ensure high performance and stability. Calculate the matrix K in MATLAB:

$$K = dlqr(Ad, Bd, Q, R) \quad (21)$$

Matrix A_d and B_d as follows:

$$Ad = \begin{bmatrix} 1.0023 & 0.01 & -0.0004 & 0 \\ 0.4615 & 1.0023 & -0.0704 & -0.0004 \\ -0.0019 & 0 & 1.0029 & 0.01 \\ -0.4789 & -0.002 & 0.3974 & 1.0023 \end{bmatrix} \quad (22)$$

$$Bd = \begin{bmatrix} 0.0112 \\ 2.2351 \\ -0.0175 \\ -3.4961 \end{bmatrix} \quad (23)$$

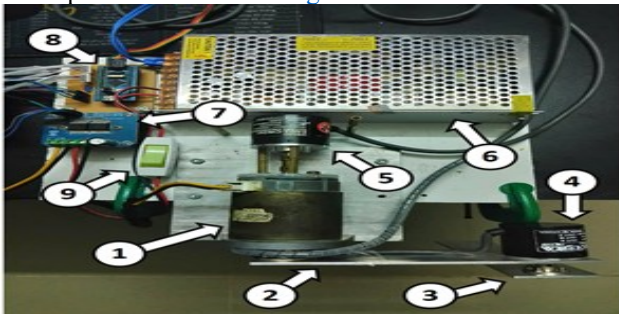
The selection of R and Q parameters in the LQR controller is a critical step in achieving optimal control performance. The authors employ the Genetic Algorithm (GA) to optimize these matrices. From (21) to (23), the gain matrix K is determined as follows:

$$K = [-6.3453 \quad -1.3245 \quad -5.7979 \quad -0.9292]$$

IV. EXPERIMENT

4.1 Experimental model

Description of numbers in Fig. 5 is:



- | | |
|-----------------------------|--------------------------|
| 1- Motor | 5- Power supply |
| 2- Link 1 | 6- Motor drives BTS 7960 |
| 3- Link 2 | 7- STM32F103C8T6 |
| 4- Encoder to measure q_2 | 8- Electric switch |
| Encoder to measure q_1 | |

Fig. 5. The pendubot real model

4.2. Experimental results

In this experiment, the authors use the parameters q_1 , q_2 , and the voltage u supplied to the motor to evaluate the control performance of the algorithms applied for swing-up. The overshoot is analyzed exclusively for q_1 since only link 1 directly receives the primary torque from the motor. The experiments for both methods are conducted over a duration of 20 s to ensure consistency in terms of time. The experimental results are presented using graphs with two axes. The x-axis(Time) represents the time during the experiment. In Fig. 6 and Fig. 8, the y-axis indicates the values of the angles q_1 and q_2 during the swing-up process, measured in radians. Meanwhile, in Fig. 7 and Fig. 9, the y-axis represents the voltage supplied to the motor,

measured in Volts (V), and is constrained within the range of -24 V to 24 V.

A. EBM

Fig. 6 shows the output response of pendubot using EBM-LQR algorithm for swing-up control. The system oscillates continuously from 0 to 4 s and begins to stabilize after 4 s. It is noticeable that initially, the amplitude of q_1 gradually increases with each oscillation. This aligns with energy-based swing-up principle, where system continuously adjusts the control torque to increase both kinetic and potential energy. The gradual increase in q_1 oscillation amplitude over time minimizes instability during the swing-up process, with an overshoot of only 3%. Meanwhile, q_2 oscillates alternately with a smaller amplitude compared to q_1 , maintaining dynamic relationship between the links. Once swing-up goal is achieved, the system transitions to stabilization using the LQR controller. Oscillations of q_1 and q_2 decrease and remain close to desired values, with a steady-state error of ± 0.04 (rad). System stabilizes after approximately 4(s).

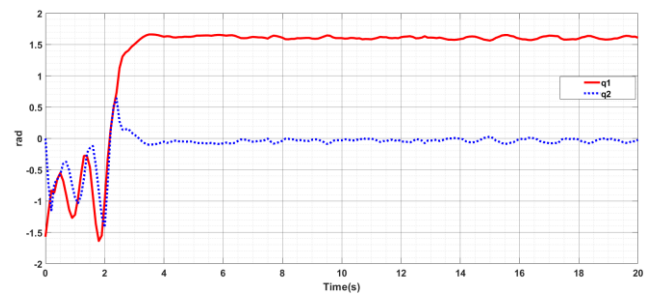


Fig. 6. Angles of links under EBM control

An analysis based on Fig. 7, during the time interval from 0 to 4 s, there is a significant fluctuation in voltage, with alternating positive and negative values. This indicates that EBM continuously adjusts the voltage to provide energy to the system, enabling the "swing up" of the robot arm in pendubot. After 4 s, the system switches to LQR control, and the voltage no longer oscillates with large amplitudes as it did in the initial phase. Instead, oscillations become smaller, with values ranging between -10V and 10V. This demonstrates that the LQR method is attempting to maintain the pendulum in a balanced state. These small oscillations compensate for disturbances or minor errors in the system. Voltage oscillations in this phase are neither high in amplitude nor highly variable in frequency because LQR focuses on stabilizing the system and minimizing deviations with high precision rather than generating large dynamics like EBM.

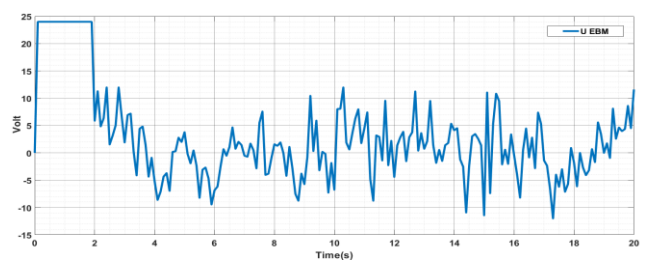


Fig. 7. Voltage supplied to the motor by ASMC

B. ASMC

Fig. 8 shows the output response of the pendubot under the ASMC-LQR algorithm for swing-up control. The system exhibits strong oscillations from 0 to 2 s and begins to stabilize after 2 s. Initially, due to control requirements, q_1 receives a large torque to track reference signal r , moving toward an upright balanced position with an overshoot of 25.4%. After achieving the swing-up goal, the system transitions to stabilization using the LQR controller. Oscillations of q_1 and q_2 decrease and remain close to desired values, with a steady-state error of ± 0.03 (rad). The system stabilizes after approximately 2 s.

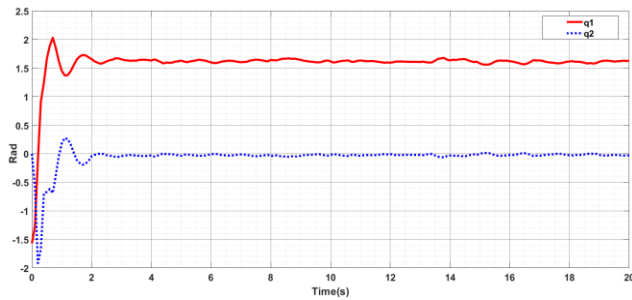


Fig. 8. Angles of links under ASMC control

Analysis based on Fig. 9, during the first 2 s, the voltage fluctuates strongly. This shows that ASMC is operating with high sensitivity, making significant voltage adjustments to quickly bring the pendubot's pendulum to the target position. The high voltage amplitude and rapid oscillations are characteristic of ASMC, which is designed to effectively correct deviations and handle uncertainties in the system. ASMC prioritizes rapid and accurate responses over smoothness, which explains the strong voltage changes during this phase. After 2 s, the voltage gradually stabilizes within the range of -10 V to 10 V. This reflects that LQR is controlling the motor to maintain the pendulum in a stable balanced state, with smaller and smoother adjustments to minimize errors. The voltage oscillations decrease in amplitude and occur at lower frequencies because LQR focuses on optimizing energy consumption and maintaining system stability.

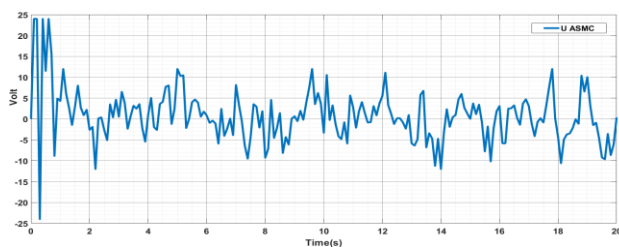


Fig. 9. Voltage supplied to the motor by ASMC

Table 2. Comparison of results

Algorithm	Overshoot	Settling time (s)
ASMC	25,4%	2
EBM	3%	4

Based on Table 2, the amplitude of SMC oscillations is higher, and the duration of SMC usage is shorter 2 s compared to 4 s in EBM-LQR). SMC responds more

quickly and can bring the system to the target state in a shorter time. However, its high overshoot may destabilize the system if not well-controlled. Both methods eventually switch to LQR, but the graph shows that in SMC-LQR, the voltage stabilizes earlier due to SMC rapidly bringing the pendulum closer to the target position.

V. CONCLUSION

EBM-LQR and ASMC-LQR both achieve swing-up and stabilization objectives for pendubot. But, each method has its own advantages. EBM-LQR ensures a stable swing-up process with low overshoot but requires 4 s to fully stabilize. In contrast, ASMC-LQR responds faster, stabilizing within 2 s, but exhibits higher overshoot, requiring careful control to avoid instability. ASMC-LQR excels in speed, while EBM-LQR prioritizes stability during the transition process. Through comparison between EBM-LQR and ASMC-LQR, for swinging-up pendubot, each method has its unique strengths and weaknesses, depending on specific evaluation criteria. From these results, we can entirely aim to develop the algorithm further when applying it to the 3-link pendubot or other higher-order nonlinear systems with similar characteristics.

ACKNOWLEDGEMENT

This contribution belongs to the project funded by Ho Chi Minh City University of Technology and Education (HCMUTE), for students of HCMUTE for year 2025. We also want to give thanks to PhD. Van-Dong-Hai Nguyen and Ms. Eng. Thi-Thanh-Hoang Le (HCMUTE) who helped us in theory. Video links of experimental operation are shown in:

- EBS method: <https://www.youtube.com/shorts/x4YUfdK4ycc>
- ASMC method: <https://www.youtube.com/shorts/L-slZ2nZrPU>

REFERENCES

- [1] D. J. Block *et al.*, "Mechanical Design and Control of the Pendubot," *SAE Transactions*, vol. 104, pp. 36-43, 1995, <https://doi.org/10.2991/eaeme-15.2015.108>.
- [2] Trong-Bang Tran *et al.*, "Trajectory Tracking using LQR Control for Pendubot: Simulation and Experiment", *JFSC*, Vol.2, No.1, 2007, <https://doi.org/10.59247/jfsc.v2i1.163>.
- [3] P. Y> Chen, PY *et al.*, "A PSO method with nonlinear time-varying evolution for optimal design of PID controllers in a pendubot system," *Artif Life Robotics*, vol. 14, pp. 58-61, 2009, <https://doi.org/10.1007/s10015-009-0628-7>.
- [4] M. Gulan *et al.*, "Real-time MHE-based nonlinear MPC of a Pendubot system," *IFAC-Papers Online*, vol. 48, no. 23, pp. 422-427, 2015, <https://doi.org/10.1016/j.ifacol.2015.11.315>.
- [5] M.-D. Tran *et al.*, "A Comparative Study between Sliding Mode and Fuzzy Sliding Mode Control for Pendubot," *Serbian Journal of Electrical Engineering*, vol. 20, pp. 361-390, 2023, <https://doi.org/10.2298/SJEE2303361T>.
- [6] Dong Sang Yoo, "Balancing control for the pendubot using sliding mode," *IEEE ISR 2013*, pp. 1-4, 2013, <https://doi.org/10.1109/ISR.2013.6695735>.
- [7] M.W.Spong, "Swing up control of the Acrobot" *Proceedings of the 1994 IEEE International Conference on Robotics and Automation*, 1994, <https://doi.org/10.1109/ROBOT.1994.350934>.
- [8] K. Yoshida, "Swing-up control of an inverted pendulum by energy-based methods," *Proceedings of the 1999 American Control Conference*, 1999, <https://doi.org/10.1109/ACC.1999.786297>.

- [9] I. Fantoni *et al.*, "Energy Based Control of the Pendubot," *IEEE Transactions on Automatic Control*, vol. 45, 2000, <https://doi.org/10.1109/9.847110>.
- [10] D. Qian *et al.*, "Hierarchical Sliding Mode Control to Swing up a Pendubot," *Proc. of the American Control Conf.*, pp. 5254-5259, 2007, <https://doi.org/10.1109/ACC.2007.4282176>.
- [11] X. Q. Ma *et al.*, "A new fuzzy approach for swing up control of Pendubot," *Proceedings of American Control Conference*, 2002, <https://doi.org/10.1109/ACC.2002.1023149>.
- [12] C. V. Kien *et al.*, "Swing Up and Balancing Implementation for the Pendubot Using Advanced Sliding Mode Control," *International Conference on Artificial Intelligence and Control Automation*, 2015, <https://doi.org/10.2991/eame-15.2015.108>.
- [13] Martin Gulan *et al.*, "Tracking Control of Unforced and Forced Equilibrium Positions of the Pendubot System: A Nonlinear MHE and MPC Approach," *Actuators*, vol. 12, no. 9, 2023. <https://doi.org/10.3390/act12090343>.
- [14] Xin Xin *et al.*, "New analytical results of energy-based swing-up control for the Pendubot," *International Journal of Non-Linear Mechanics*, vol. 52, 2013, pp. 110-118. <https://doi.org/10.1016/j.ijnonlinmec.2013.02.003>.
- [15] G. B. Hong, "Comparison between two swing-up algorithms: partial feedback linear and energy based method," *Journal of Technical Education Science*, vol. 14, no. 5, pp. 1-6., 2019. <https://jte.edu.vn/index.php/jte/article/view/161>.