PSO-OPTIMIZED LQR AND ENERGY-BASED SWING-UP CONTROL FOR REACTION WHEEL INVERTED PENDULUM

Dinh Dat Tran [1*], Binh Hau Nguyen [1], Bao Quoc Mai [1], Duc-Anh-Quan Nguyen [2]

[1] Posts and Telecommunications Institute of Technology (PTIT)
11-Nguyen Dinh Chieu Street, Da Kao Ward, District 1, Ho Chi Minh City, Vietnam.

[2] Ho Chi Minh City University of Technology and Education (HCMUTE)
01-VoVanNgan Str, Thu Duc city, Ho Chi Minh City, Vietnam

***** Corresponding author. E-mail: dattd@ptit.edu.vn

Abstract: The Reaction Wheel Inverted Pendulum (RWIP) system presents challenges in balancing and swing-up control due to its nonlinear, underactuated nature. This study introduces an energy-based control approach for the swing-up phase, along with an optimized Linear Quadratic Regulator (LQR) for stabilization. Through simulations and experiments, we demonstrate the effectiveness of these methods on an RWIP system with a reaction wheel mounted on a shaft supported by two ball bearings. Particle Swarm Optimization (PSO) is utilized to optimize the parameters of the LQR controller, aiming to improve both the stability and performance of the system. The control algorithms are implemented on an STM32F407 microcontroller, enabling real-time control and processing. Experimental results validate the control strategies under various external disturbances, proving their robustness and efficacy.
Keywords: RWIP, parameter optimization, energy-based control, swing-up mechanism, LQR, PSO, STM32F407 microcontroller.

1. Introduction

The RWIP is a system extensively studied in control engineering, particularly in applications requiring high-precision balancing, such as self-balancing vehicles, satellite attitude control, and gyroscopic stabilization systems. The control challenge arises from the system's nonlinear dynamics and underactuated configuration, where only the reaction wheel is actuated, and the pendulum's motion is indirectly controlled.

1.1. Control Challenges in RWIP Systems

The primary challenge in controlling the RWIP system lies in balancing the pendulum, which naturally tends to fall due to gravitational forces. When the pendulum is in its unstable downward position, it requires an external force to swing it up and stabilize it. This necessitates precise energy management to move the pendulum to the upright position and maintain equilibrium.

The nonlinear dynamics of the RWIP system [1] make it highly sensitive to external disturbances, further complicating the stabilization process. Conventional control methods, such as PID controllers [2][3], are typically inadequate for sustaining long-term equilibrium in such systems, particularly when faced with nonlinear behaviors and disturbances.

1.2. Proposed Solution

To address these challenges, this paper proposes an innovative control strategy that integrates following components:

- Energy-Based Swing-Up Control [4][5][6]: This approach injects energy into the system to transition the pendulum from its downward, unstable position to the upright position, where it can be stabilized.
- PSO-Optimized LQR: Once the pendulum is positioned upright, an LQR controller [7][8][9] is employed to maintain balance. LQR is a robust control technique, but its performance depends on precise tuning of its parameters. In this work, PSO [11][12][13][14] is utilized to optimize the LQR controller's parameters, enhancing the system's stability and overall performance.

1.3. Objective of the Paper

The primary objectives of this paper are:

- To model the dynamic behavior of the RWIP system using Euler-Lagrange Method.
- To develop an energy-based control strategy for the swing-up process.
- To design and optimize a LQR controller using PSO to improve system stability and performance.
- To validate the effectiveness of the proposed control strategy through simulations and real-world experiments, utilizing an STM32F407 microcontroller for real-time implementation.

1.4. Structure of the Paper

The structure of this paper is organized as follows:

- Section 2 presents the mathematical modeling of the RWIP.
- Section 3 describes the energy management-based swing-up control strategy.
- Section 4 discusses optimization of LQR using PSO.
- Section 5 presents the simulation and experimental results.
- Section 6 concludes the paper.

2. Mathematical Modeling of the RWIP

RWIP system, as illustrated in Fig. 1, consists of a pendulum attached to a shaft with a reaction wheel mounted on the same shaft. The aim is to regulate the angular velocity of the reaction wheel and maintain the pendulum in its upright balanced position. Due to the system's inherent nonlinear characteristics, it poses a significant control challenge.

To model this system, we employ Lagrangian mechanics to derive the equations of motion based on the system's kinetic and potential energy. This approach offers a systematic framework for accurately describing the dynamic behavior of the system.

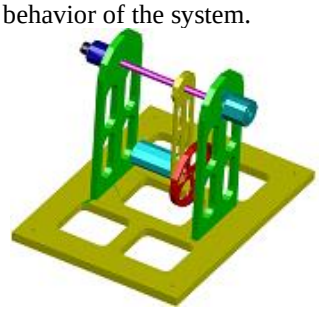

Fig. 1. The model of RWIP System.

2.1. Defining the System Variables

The parameters of the RWIP system, as detailed in Tab. 1 and illustrated in Fig. 2, are obtained through direct measurements of the model. The variables include θ, ω, and τ, where θ and ω represent the output signals, while τ denotes the input signal. These variables are initially unknown. The authors will establish the relationship between the input and output signals based on the developed algorithm.

Tab. 1 Parameters of the System

Parameters and variables	Described
m_p	Total mass of the pendulum and rotor
m_r	Mass of the rotor
I_b	Moment of inertia of the pendulum about the central axis
I_r	Moment of inertia of the rotor about its center
L_r	Distance from O to O_1 of the pendulum
L_p	Distance from O to O_2 of the pendulum
i	Current through the motor
ω_p	Relative angle between the pendulum and the fixed axis
ω_r	Relative angle between the pendulum and the rotor
$\theta_p = \omega_p$	Pendulum angle measured counterclockwise
$\theta_r = \omega_p + \omega_r$	Rotor angle measured counterclockwise
R	Radius of the rotating wheel
C_b	Rolling friction coefficient of the bearing (KBC 608 RS) in the gimbal shaft
B_m	Rolling friction coefficient inside the servo motor
τ_m	Instantaneous torque of the motor
g	Gravitational acceleration
$i_a(t)$	Node current
R_a	Node resistance
$e_b(t)$	Electromotive force (EMF)
$\tau_L(t)$	Frictional torque
L_a	Inductance
$e_a(t)$	Supply voltage
K_i	Twist stiffness constant
K_b	Electromotive force constant
Ø	Magnetic flux
$\omega_m(t)$	Angular velocity of the rotor
$\tau_m(t)$	Motor torque
J_m	Load inertia

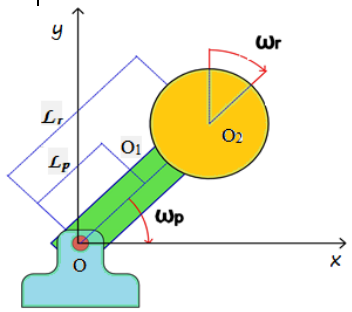

Fig. 2. The Mathematical model of the RWIP system.

2.2. Euler-Lagrange Method

Euler-Lagrange Method [10] is a robust framework for describing the dynamics of physical systems. The Lagrange function L is defined as the difference between the kinetic energy (T) and the potential energy (P) of the system:

$$L = K - P \quad (1)$$

where: L= Lagrange function; K= Kinetic energy; P= Potential energy.

The Euler-Lagrange equations of motion can be expressed in terms of generalized coordinates, facilitating the description of specific reference frames. The number of independent moments is determined by the number of components within the system. The generalized equation is reformulated according to Euler-Lagrange method as follows:

$$\frac{d}{dt}\frac{\partial L}{\partial q'} - \frac{\partial L}{\partial q} = \tau_q, q = \begin{bmatrix} \omega_p \\ \omega_r \end{bmatrix} \quad (2)$$

Where τ_i represents the total external forces (inertial forces, frictional forces, etc.).

Let $\vec{R}_1$ and $\vec{R}_2$ be the position vectors of the pendulum rod and the rotating wheel, respectively, in the coordinate system (xOy). We then have:

$$\begin{cases} \overrightarrow{R_1} = L_p \sin \omega_p \vec{x} + L_p \cos \omega_p \vec{y} & (3) \\ \overrightarrow{R_2} = L_r \sin \omega_r \vec{x} + L_r \cos \omega_r \vec{y} & (4) \\ \theta_p = \omega_p + \omega_r & (5) \end{cases}$$

The kinetic energy of the pendulum rod is expressed as follows:

$$K_1 = \frac{1}{2} I_p \omega_p'^2 - \frac{1}{2} I_r' \omega_p^2 + \frac{1}{2} m_p \left(\vec{R}_{1x} \vec{R}_{1y}\right) \quad (6)$$

The kinetic energy of rotating wheel is given by:

$$K_2 = \frac{1}{2} I_r \theta_r'^2 + \frac{1}{2} m_r \left(\vec{R}_{2x} \vec{R}_{2y}\right) \quad (7)$$

The combined kinetic energy of the system, due to the kinetic energy of the pendulum rod and the wheel, and using (3) - (7), leads to:

$$\begin{aligned} K &= K_1 + K_2 = \\ &\frac{1}{2} I_p \omega_p'^2 - \frac{1}{2} I_r' \omega_p^2 + \frac{1}{2} m_p \left(\vec{R}_{1x} \vec{R}_{1y}\right) \\ &+ \frac{1}{2} I_r \theta_r'^2 + \frac{1}{2} m_r \left(\vec{R}_{2x} \vec{R}_{2y}\right) \\ &= \frac{1}{2} I_p \omega_p'^2 - \frac{1}{2} I_r' \omega_p^2 + \frac{1}{2} m_p L_p^2 \omega_p'^2 \\ &+ \frac{1}{2} I_r \theta_r'^2 + \frac{1}{2} m_r L_r^2 \omega_r'^2 \end{aligned} \quad (8)$$

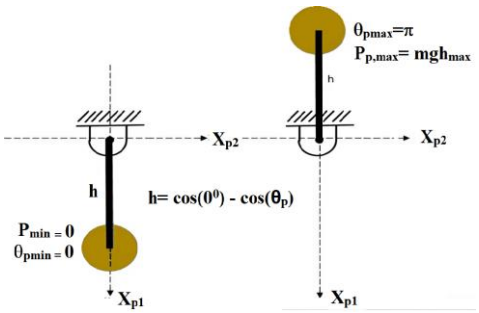

Fig. 3. [1] Schematic maximum and minimum position of the pendulum

The potential energy of the pendulum rod, based on Fig. 3, is given by:

$$P_1 = m_p L_p g(1 - \cos(\omega_p)) \quad (9)$$

The potential energy of the rotating wheel, as shown in Fig. 3, is given by:

$$P_2 = m_r L_r g(1 - \cos(\omega_p)) \quad (10)$$

The potential energy of the system, derived from equations (9) to (10), is given by:

$$\begin{aligned} P &= P_1 + P_2 \\ &= (m_p L_p + m_r L_r) g(1 - \cos(\omega_p)) \end{aligned} \quad (11)$$

By combining the equations from (8) and (11) and substituting into (1), the Lagrangian is obtained as:

$$\begin{aligned} L = &\frac{1}{2} I_p \omega_p'^2 + \frac{1}{2} m_p L_p^2 \omega_p'^2 \\ &- \frac{1}{2} I_r \omega_p'^2 + \frac{1}{2} I_r \theta_r'^2 + \frac{1}{2} m_r L_r^2 \omega_p'^2 \\ &- \left(m_p L_p + m_r L_r\right) g(1 - \cos(\omega_p)) \end{aligned} \quad (12)$$

Using the Euler-Lagrange method from equation (12), the differential equation of motion for the system can be derived, taking into account the bearing friction at both ends and the sliding friction of the motor:

$$\begin{cases} \frac{\partial L}{\partial \omega_p'} = (I_p + m_p L_p^2 + m_r L_r^2)\omega_p' - I_r \omega_p' + I_r \theta_r' & (13.1) \\ \frac{d}{dt}\left(\frac{\partial L}{\partial \omega_p'}\right) = (I_p + m_p L_p^2 + m_r L_r^2)\omega_p'' + I_r \omega_r'' & (13.2) \\ \frac{\partial L}{\partial \omega_p} = -\left(m_p l_p + m_r l_r\right) g \sin(\omega_p) & (13.3) \\ \frac{\partial L}{\partial \omega_r'} = I_r \theta_r' & (13.4) \\ \frac{d}{dt}\left(\frac{\partial L}{\partial \omega_r'}\right) = I_r \omega_p'' + I_r \omega_r'' & (13.5) \\ \frac{\partial L}{\partial \omega_r} = 0 & (13.6) \end{cases}$$

Substituting equations (13.1) to (13.6) sequentially into equation (2), we obtain:

$$\begin{cases} \left(I_P + L_p^2 m_p + L_r^2 m_r\right)\omega_p'' + I_r \omega'' & (14.1) \\ - g\left(L_p m_p + L_r m_r\right) \sin\omega_p = 0 & \\ I_r \omega_p'' + I_r \omega_r'' = \tau & (14.2) \end{cases}$$

Substituting equations (14.1) and (14.2), we obtain the system of equations:

$$\begin{cases} \left(I_P - I_r + L_p^2 m_p + L_r^2 m_r\right)\omega_p'' \\ - g\left(L_p m_p + L_r m_r\right) \sin\omega_p = -\tau \\ I_r \omega_p'' + I_r \omega_r'' = \tau \end{cases} \quad (15)$$

$$\begin{cases} \omega_p'' = \dfrac{g\left(L_p \mathrm{m}_p + L_r \mathrm{m}_r\right) sin\omega_p - \tau}{\left(I_P - I_r + L_p{}^2 \mathrm{m}_p + L_r{}^2 \mathrm{m}_r\right)} \\ \omega_r'' = \dfrac{\left(I_P - I_r + L_p{}^2 \mathrm{m}_p + L_r{}^2 \mathrm{m}_r\right)\tau}{I_r\left(I_P - I_r + L_p{}^2 \mathrm{m}_p + L_r{}^2 \mathrm{m}_r\right)} \\ \quad - \dfrac{I_r\left(g\left(L_p \mathrm{m}_p + L_r \mathrm{m}_r\right) sin\omega_p - \tau\right)}{I_r\left(I_P - I_r + L_p{}^2 \mathrm{m}_p + L_r{}^2 \mathrm{m}_r\right)} \end{cases} \quad (16)$$

2.3. Mathematical Equation for the Rotating Wheel Motor

A fundamental DC motor operates as a torque converter, converting electrical energy into mechanical energy. The torque generated on the motor shaft is directly proportional to the magnetic flux and the armature current. In Fig. 4, a conductive wire is situated within a magnetic field defined by the flux, and this wire is located at a distance r from the axis of rotation. The relationship between the instantaneous torque, the friction at the bearings on both ends of the motor, and the frictional force acting on the pendulum shaft bearing is expressed as follows:

$$\tau = \tau_m - 2C_b\omega_p' - B_m\omega_r' + J_m u \quad (17)$$

Where:

- C_b is the friction coefficient of the pendulum shaft bearing.
- B_m is the friction coefficient of the DC motor bearings.
- J_m is the moment of inertia of the motor.
- τ_m is the torque of the motor.
- u is the instantaneous supply voltage value.

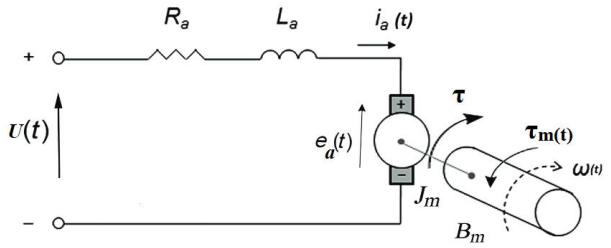

Fig. 4. [14] DC motor schematic diagram

In addition to the torque shown in Fig. 4, when the conductor moves within the magnetic field, an induced electromotive force V (flux current) is generated across its terminals. The generated voltage is directly proportional to the shaft's rotational speed and acts to counter the current flow. This relationship between the voltage, $e_a(t)$, and shaft speed is expressed as follows [14]:

$$U(t) = K_m\theta_r(t) \quad (18)$$

Referring to the circuit diagram illustrated in Fig. 4, the regulation of the DC motor is implemented at the armature terminals through an applied voltage, denoted as $e_a(t)$. For the purpose of linear analysis, it is postulated that the torque produced by the motor is directly proportional to both the flux current and the circuit current. Thus, [14]:

$$\tau_m(t) = K_m(t).i_a(t) = K_i.\, i_a(t) \quad (19)$$

Applying Kirchhoff's law to the DC motor with the input control voltage e_a (t), and calculating the motor torque according to Fig. 4, we have:

$$\begin{cases} \dfrac{di_a(t)}{dt} = \dfrac{1}{L_a}e_a(t) - \dfrac{R_a}{L_a}i_a(t) - \dfrac{1}{L_a}e_a(t) & (20.1) \\ \tau_m(t) = K_i i_a(t) & (20.2) \\ e_b(t) = K_b\dfrac{d\theta_m(t)}{dt} = K_b\omega_m'(t) & (20.3) \\ \dfrac{d^2\theta_m(t)}{dt^2} = \dfrac{1}{J_m}\tau_m(t) - \dfrac{1}{J_m}\tau_L(t) - \dfrac{B_m}{J_m}\dfrac{d\theta_m(t)}{dt} & (20.4) \end{cases}$$

From the differential equations (20.1) to (20.4), the DC motor torque equation can be rewritten as:

$$\tau_m(t) = K_i.i_a(t) = \frac{K_i(e_a(t) - K_b\omega_m'(t))}{R_a} \quad (21)$$

With e_a = u (Fig. 4), K_b=1.356K_i [5], from equation (21), we obtain:

$$\tau_m = K_i.i_a = \frac{K_i(u - 1.356K_i\omega_r')}{R_a} \quad (22)$$

By substituting equation (22) into (17), we obtain:

$$\tau = \frac{K_i(u - 1.356K_i\omega_r')}{R_a} - 2C_b\omega_p' - B_m\omega_r' + J_m u \quad (23)$$

By substituting equation (23) into (16), we obtain:

$$\begin{cases} \omega_p'' = \dfrac{g(L_p m_p + L_r m_r)\sin\omega_p - \dfrac{K_i(u - 1.356K_i\omega_r')}{R_a}}{(I_p - I_r + L^2{}_p m_p + L^2{}_r m_r)} \\ \qquad + \dfrac{2C_b\omega_p' + B_m\omega_r' - J_m u}{(I_p - I_r + L^2{}_p m_p + L^2{}_r m_r)} \\ \omega_r'' = \dfrac{\left(\dfrac{K_i(u - 1.356K_i\omega_r')}{R_a} - 2C_b\omega_p' - B_m\omega'\right) + J_m u}{I_r} \\ \qquad - \left(\dfrac{g(L_p m_p + L_r m_r)\sin\omega_p - \dfrac{K_i(u - 1.356K_i\omega_r')}{R_a}}{(I_p - I_r + L^2{}_p m_p + L^2{}_r m_r)}\right) \\ \qquad - \left(\dfrac{2C_b\omega_p' + B_m\omega_r' - J_m u}{(I_p - I_r + L^2{}_p m_p + L^2{}_r m_r)}\right) \end{cases} \quad (24)$$

2.4. Mathematical Equation RWIP

To formulate the mathematical model, the following variables are defined:

$$\begin{cases} x = \begin{bmatrix} x_1 & x_2 & x_3 & x_4 \end{bmatrix}^T \\ \quad = \begin{bmatrix} \omega_p & \omega_p' & \omega_r & \omega_r' \end{bmatrix}^T \\ \dot{x}_1 = x_2 \\ \dot{x}_2 = \omega_p'' \\ \dot{x}_3 = x_4 \\ \dot{x}_4 = \omega_r'' \end{cases} \quad (25)$$

From equations (25) and (24), we can derive

$$\begin{Bmatrix} \dot{x}_1 \\ \dot{x}_2 \\ \dot{x}_3 \\ \dot{x}_4 \end{Bmatrix} = \begin{bmatrix} x_2 \\ \dfrac{g(L_p m_p + L_r m_r)\sin\omega_p - \dfrac{K_i(u - 1.356K_i\omega_r')}{R_a}}{(I_p - I_r + L^2{}_p m_p + L^2{}_r m_r)} \\ \quad + \dfrac{2C_b\omega_p' + B_m\omega_r' - J_m u}{(I_p - I_r + L^2{}_p m_p + L^2{}_r m_r)} \\ x_4 \\ \dfrac{(\dfrac{K_i(u - 1.356K_i\omega_r')}{R_a} - 2C_b\omega_p' - B_m\omega') + J_m u}{I_r} \\ \quad - (\dfrac{g(L_p m_p + L_r m_r)\sin\omega_p - \dfrac{K_i(u - 1.356K_i\omega_r')}{R_a}}{(I_p - I_r + L^2{}_p m_p + L^2{}_r m_r)}) \\ \quad - (\dfrac{2C_b\omega_p' + B_m\omega_r' - J_m u}{(I_p - I_r + L^2{}_p m_p + L^2{}_r m_r)}) \end{bmatrix} \quad (26)$$

We set the parameters for simplification:

$$\begin{cases} H_1 = g(L_p m_p + L_r m_r) \\ H_2 = (I_p - I_r + L^2{}_p m_p + L^2{}_r m_r) \\ H_3 = \dfrac{K_i}{R_a} \end{cases}$$

The from (26) we change to (27):

$$\begin{Bmatrix} \dot{x}_1 \\ \dot{x}_2 \\ \dot{x}_3 \\ \dot{x}_4 \end{Bmatrix} = \begin{bmatrix} x_2 \\ \dfrac{H_1\sin\omega_p - H_3(u - 1.356K_i\omega_r') + 2C_b\omega_p' + B_m\omega_r' - J_m u}{H_2} \\ x_4 \\ \dfrac{H_2((H_3(u - 1.356K_i\omega_r') - 2C_b\omega_p' - B_m\omega') + J_m u)}{H_2 I_r} \\ -\dfrac{I_r((H_1\sin\omega_p - H_3(u - 1.356K_i\omega_r') + (2C_b\omega_p' + B_m\omega_r' - J_m u))}{H_2 I_r} \end{bmatrix} \quad (27)$$

On the other hand, we observe that in the range where ω_p is very small, $\sin(\omega_p)$ is approximately equal to ω_p and then from (27), we isolate u and transform the value x follow (25) to obtain the state equation in the form: x'=Ax +Bu, we have:

$$\begin{Bmatrix} \dot{x}_1 \\ \dot{x}_2 \\ \dot{x}_3 \\ \dot{x}_4 \end{Bmatrix} = \begin{bmatrix} 0 & 1 & 0 & 0 \\ \dfrac{H_1}{H_2} & 2C_b & 0 & \dfrac{1.356H_3K_i + B_m}{H_2} \\ 0 & 0 & 0 & 1 \\ -\dfrac{H_1}{H_2 I_r} & \dfrac{-(2C_b H_2 + 2C_b I_r)}{H_2 I_r} & 0 & \dfrac{1.356H_3K_i(H_2 + I_r) - (B_m H_2 + B_m I_r)}{H_2 I_r} \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \\ x_3 \\ x_4 \end{bmatrix} + \begin{bmatrix} 0 \\ -\dfrac{H_3 + J_m}{H_2} \\ 0 \\ \dfrac{(H_3 H_2 - I_r H_3 + J_m I_r}{H_2 I_r} \end{bmatrix} u \quad (28)$$

2.5. Simulation Model of the System in Matlab

Based on equation (24), the model is constructed as follows in Matlab.

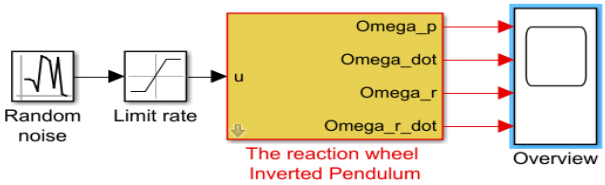

Fig. 5. Simulation Model of RWIP in MATLAB

3. Energy Control Algorithm for Swing-Up of RWIP

3.1. Energy Basic Control

All experiments begin under the initial conditions $\theta_p(0)=0$ (rad), $\dot{\theta}_p(0)=0$, $\theta_r(0)=0$, $\dot{\theta}_r(0)=0$, This indicates that the pendulum begins in a downward orientation. Given that the control goal requires the wheel to track a time-varying trajectory while keeping the pendulum in an unstable upright position, a swing-up strategy is implemented. We will also provide an energy command to sustain the swing-up process until the angle θp reaches a threshold of ± 0.3491 rad ($\pm 20^0$). The control algorithm used to elevate the pendulum is the one introduced in [1][3], which is clearly outlined, and the strategy employed is based on energy control.

First, the energy of the pendulum is given by:

$$E = K + P \quad (29)$$

Where:

- E: mechanical energy of the pendulum;
- K: potential energy obtained from equation (8);
- P: kinetic energy obtained from equation (11).

We obtain:

$$\begin{aligned} E = & \frac{1}{2} I_p \omega_p'^2 + \frac{1}{2} m_p L^2{}_p \omega_p'^2 \\ & - \frac{1}{2} I_r \omega_p'^2 + \frac{1}{2} I_r \theta_r'^2 + \frac{1}{2} m_r L^2{}_r \omega_p'^2 \\ & + \left(m_p L_p + m_r L_r\right) g(1 - \cos(\omega_p)) \end{aligned} \quad (30)$$

Our objective is for the pendulum to be at the equilibrium position at the top, where the angle $\theta_p=\pi$, and the energy will be completely converted into potential energy:

$$\begin{aligned} E_{target} &= P \\ &= (m_p L_p + m_r L_r) g(1-\cos(\pi)) \\ &= 2(m_p L_p + m_r L_r) g \end{aligned} \quad (31)$$

Let ΔE_0 be the amount of energy consumed to move from the position $\theta_p=0$ (rad) to $\theta_p=\pi$ (rad), which gives us:

$$\begin{aligned} \Delta E_0 &= E_{Target} - E \\ &= \left(m_p L_p + m_r L_r\right) g(1-\cos(\omega_p)-2) \\ &\quad -(\frac{1}{2} I_p \omega_p'^2 + \frac{1}{2} m_p L_p^2 \omega_p'^2 \\ &\quad -\frac{1}{2} I_r \omega_p'^2 + \frac{1}{2} I_r \theta_r'^2 + \frac{1}{2} m_r L_r^2 \omega_p'^2) \\ &= \left(m_p L_p + m_r L_r\right) g(\cos(\omega_p)-1) \\ &\quad -(\frac{1}{2} I_p \omega_p'^2 + \frac{1}{2} m_p L_p^2 \omega_p'^2 \\ &\quad -\frac{1}{2} I_r \omega_p'^2 + \frac{1}{2} I_r \theta_r'^2 + \frac{1}{2} m_r L_r^2 \omega_p'^2) \end{aligned} \quad (32)$$

From equations (32) and (27), we obtain the simplified expression for ΔE_0:

$$\Delta E_0 = H_1(\cos(\omega_p)-1) - (\frac{H_2}{2} \omega_p'^2 + \frac{1}{2} I_r \theta_r'^2) \quad (33)$$

Thus, to enable the RWIP to reach the position ±0.3491 rad, a stable energy source of ΔE_0 must be provided. This implies that a necessary torque τ must be applied to add energy to the system, which will depend on the energy difference ΔE_0 and the direction of the pendulum's motion (represented by the sign of θ'_p):

$$\tau = K_u \Delta E sign(\theta_p') \quad (34)$$

Where:

- K_u: control gain coefficient.
- sign(θ'_p): determines the direction of the torque based on the pendulum's motion.

Therefore, from equation (21) and substituting (34), the necessary voltage to generate the appropriate torque is derived as follows:

$$\begin{aligned} K_u \Delta E sign(\theta_p') &= \frac{K_i(u - 1.356 K_i \omega_r')}{R_a} \\ \Rightarrow u &= \frac{K_u R_a \Delta E sign(\theta_p')}{K_i} + 1.356 K_i \omega_r' \end{aligned} \quad (35)$$

From equation (35), we observe that the sign of u depends on θ'_p . Based on [5][6][7], we neglect the velocity term ω'_r, thus (33 and (34) becomes as shown in Fig. 6:

$$\begin{cases} u = \dfrac{K_u R_a (H_1(\cos(\omega_p)-1) - (\frac{H_2}{2} \omega_p'^2 + \frac{1}{2} I_r \theta_r'^2)) sign(\theta_p')}{K_i} \\ u > 0; (\theta_p' < 0 \,\&\, E < E_{target} \,||\, \theta_p' >= 0 \,\&\, E >= E_{target}) \\ u < 0; \textit{For the remaining cases} \end{cases} \quad (36)$$

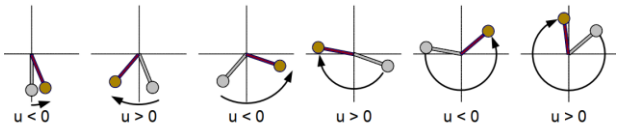

Fig. 6. The state of switching the sign of u in Swing-up Energy control

3.2. Swing-Up Model of RWIP

Based on the calculations from equation (35), we construct the swing-up model for RWIP.

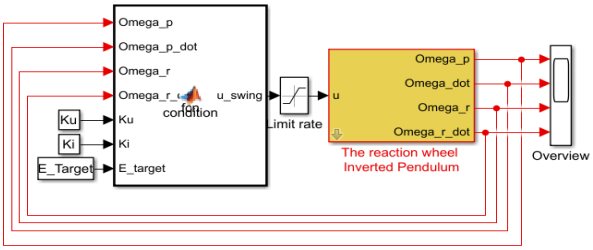

Fig. 7. Simulation Model of Swing-Up

4. LQR Algorithm

4.1. LQR Mathematical Formulation

We establish the state equations of the RWIP system based on [7], [8]:

$$x' = Ax + Bu$$

As follows (28):

$$A = \begin{bmatrix} 0 & 1 & 0 & 0 \\ \frac{H_1}{H_2} & 2C_b & 0 & \frac{1.356 H_3 K_i + B_m}{H_2} \\ 0 & 0 & 0 & 1 \\ -\frac{H_1}{H_2 I_r} & \frac{-(2C_b H_2 + 2C_b I_r)}{H_2 I_r} & 0 & \frac{1.356 H_3 K_i (H_2 + I_r) - (B_m H_2 + B_m I_r)}{H_2 I_r} \end{bmatrix},$$

$$B = \begin{bmatrix} 0 \\ -\frac{H_3 + J_m}{H_2} \\ 0 \\ \frac{(H_3 H_2 - I_r H_3 + J_m I_r}{H_2 I_r} \end{bmatrix} \quad (37)$$

Adaptive Function is determined as (40):

$$I = \int_0^\infty (x^T Q x + u^T R u + 2 x^T N u) dt \quad (38)$$

As follows: Q is matrix with n x n size and R is weighting matrix:

$$\begin{cases} Q = \begin{bmatrix} q_1 & 0 & 0 & 0...0 \\ 0 & q_2 & 0 & 0...0 \\ ... & & & \\ 0 & 0 & 0 & 0...q_n \end{bmatrix} \\ R = r \end{cases} \quad (40)$$

The control signal is provided through feedback by LQR , as shown below:

$$u = -kx \quad (41)$$

As follows:

$$k = R^{-1}(B^T P + N^T) \quad (42)$$

P is determined using the Riccati function as follows:

$$A^T P + PA - (PB + N)R^{-1}(B^T P + N^T) + Q = 0 \quad (43)$$

In each situation, configuring N=0 facilitates a simple computation of the Riccati function. To determine k, the authors utilize the command shown below in MATLAB:

$$k = dlqr(A, B, Q, R) \quad (44)$$

As follows matrices A and B are determined at (37), Q and R we define by [9]:

$$\begin{cases} Q = \begin{bmatrix} 3600 & 0 & 0 & 0 \\ 0 & 3600 & 0 & 0 \\ 0 & 0 & 3600 & 0 \\ 0 & 0 & 0 & 10 \end{bmatrix}, \\ R = 5 \end{cases} \quad (46)$$

4.2. Simulation and Experimental Model

We build the simulink in matlab Fig. 8:

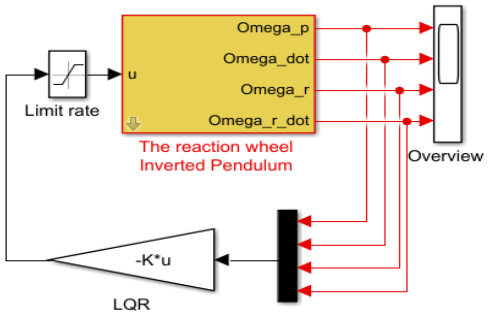

Fig. 8. Scheme of LQR for the RWIP system

Experimental model is constructed in Fig. 9.

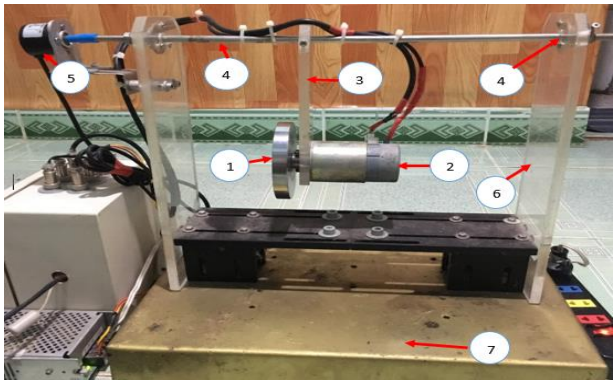

Fig. 9. Experimental Model of the RWIP

Where:

1. Flywheel
2. Servo motor
3. Pendulum rod
4. Bearing
5. Encoder
6. Model support frame
7. Model base

With parameters of model in Tab. 2:

Tab. 2 Parameters of the Real System

Parameters	Values	Units
m_p	0.92856	kg
m_r	1.97856	kg
I_p	6.9642*10-3	kg·m2
I_r	2.4732*10-3	kg·m2
R_D	0.15	m
L_p	0.15	m
L_r	0.15	m
K_b	0.088062864	N.m/A
C_b	0.12*10-3	
B_m	0.00004813	
g	9.80665	m/s2
J_m	$1.341727*10^{-4}$	kg·m2
L_a	4.4659	Henri
R_a	6.83572	

And then, with parameters are at Tab. 2, The matrices A and B of the state-space model are computed as follows:

$$A = \begin{bmatrix} 0 & 1 & 0 & 0 \\ 61 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 \\ -24736 & -1 & 0 & 0 \end{bmatrix},$$

$$B = \begin{bmatrix} 0 \\ -0.1378 \\ 0 \\ 3.707 \end{bmatrix}, \quad (47)$$

From (44), (46), (47) and (48) we have:

$$k = 1.0e+03 *[-7.169507635031356 \ -0.2286939963617901 \ 0.026832815729997 \ 0.004110509371807038] \quad (48)$$

Hence LQR is designed with above vector k.

5. PSO Algorithm

5.1. Introduction to the PSO Algorithm

From equation (46), it is evident that selecting the parameters for the matrices Q and R is challenging, even though the parameters for matrix k have been obtained. However, it is certain that these parameters are not yet optimized. In control engineering, there are several algorithms available to optimize the matrices Q and R, such as GA and optimization using the Least Squares tool in MATLAB. However, the authors focus on a newly applied method in recent years, namely PSO.

PSO is an optimization technique that draws inspiration from the collective behavior observed in birds and fish. This method was first proposed by R. Eberhart and J. Kennedy in 1995. The algorithm initializes a group of random particles (potential solutions) in the search space. Each particle adjusts its position based on its own experience and the experience of neighboring particles. The algorithm iteratively updates the velocity and position of each particle using the following equations:

$$\begin{cases} v_i^{t+1} = \omega v_i^{t+1} + c_1 r_1 (pi - x_i^t) + c_2 r_2 (g - x_i^t) \\ x_i^{t+1} = x_i^t + v_i^{t+1} \end{cases} \quad (49)$$

where:

- v_i^t denotes the velocity of particle i at iteration t.
- x_i^t denotes the position of particle i at iteration t.
- p_i denotes the best position found by particle i.
- g denotes the best position found by the entire swarm.
- ω is the inertia weight, controlling the impact of the previous velocity.
- c_1 and c_2 are acceleration coefficients
- r_1 and r_2 are random numbers in Swarm

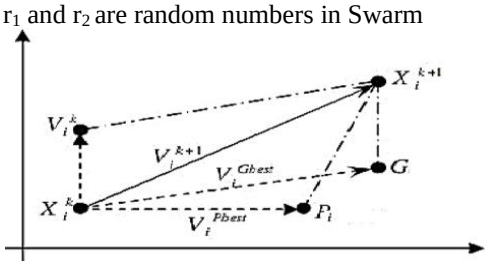

Fig. 10. Illustration of PSO

This Fig. 10 illustrates the operational framework of the PSO algorithm. It demonstrates how particles navigate through the search space, adjusting their positions based on personal and collective experiences to converge toward optimal solutions.

In the figure, each particle is represented by a point in the multidimensional search space, while arrows indicate the direction and velocity of movement. The best-known position for each particle is marked, as well as the overall best position found by the swarm.

The implementation procedure for optimizing the LQR [12][13] is depicted in Fig. 11. Following that is the block diagram representing the PSO-based LQR for the RWIP system. The design steps for the PSO-based LQR control aimed at balancing the RWIP are outlined as follows.

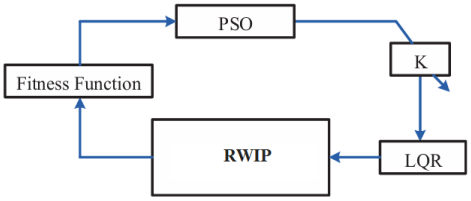

Fig. 11 Block diagram of PSO based LQR on inverted pendulum system

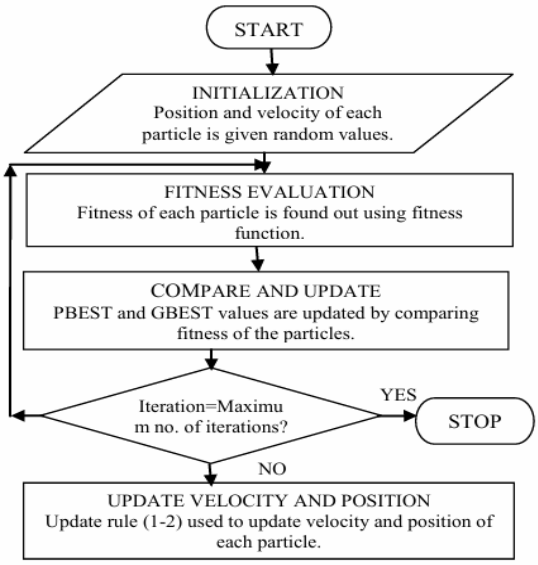

Fig. 12. [13] Flowchart of PSO Algorithm

The flowchart illustrated in Figure 12 presents the key steps involved in PSO algorithm. It commences with the initialization of particles, followed by the evaluation of fitness, the updating of velocities and positions, and the determination of the global best solution. The iterative nature of the algorithm is emphasized, highlighting the convergence process towards optimal solutions within the defined search space.

5.2. Application of the Algorithm to the Model.

Based on flowcharts in Fig. 11., Fig. 12., and equation (49), the Simulink model is constructed, and the optimization of LQR begins by optimizing the gain matrix K. The parameters are set and selected as follows:

- k= [K1 K2 K3 K4]

- n = 100 %This parameter represents the population size, indicating the total number of particles in the swarm.
- bird_step = 100 %This denotes the maximum number of steps for the optimization process.
- Dim = 4 %This indicates the number of weights to be optimized, specifically K_1, K_2, K_3, and K_4.
- C_2 = 1.2 %This is the PSO parameter C_1, which influences the cognitive component of the particle's movement.
- C_1 = 0.12 % This is the PSO parameter C_2, which affects the social component of the particle's behavior

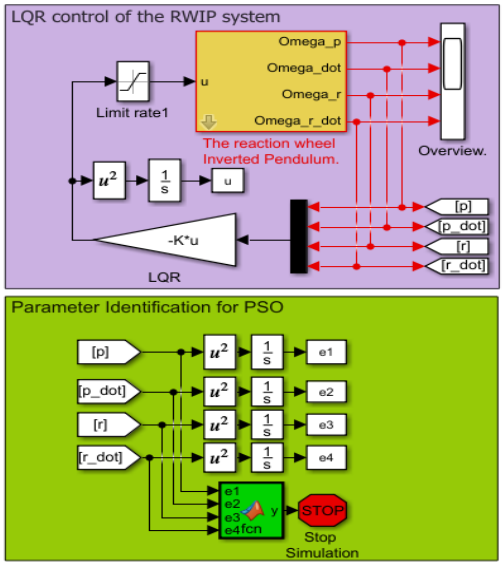

Fig. 13.a Model for Optimizing LQR using PSO for RWIP

After running the simulation and identification, the results obtained are as follows:

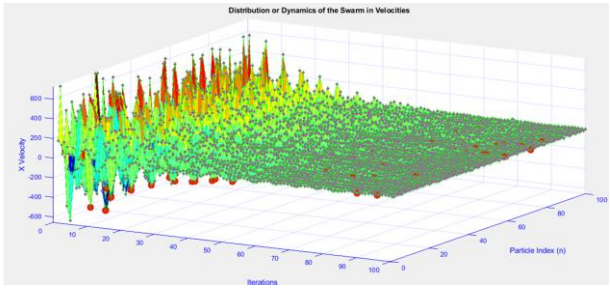

Fig. 13.b Distribution Map of the Swarm's Region or Dynamics

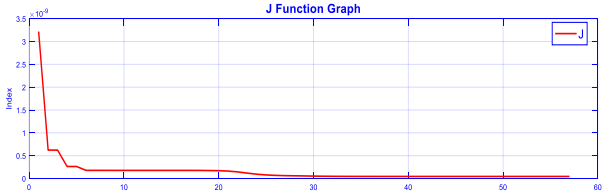

Fig. 14. Graph of the J Function

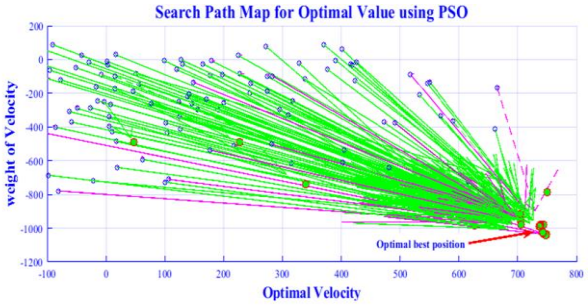

Fig. 15. PSO Swarm Activity Chart

PSO algorithm demonstrates a rapid operational efficiency in converging towards the optimal K values, effectively balancing exploration and exploitation within the search space. From Fig. 13, 14, and 15, we observe that the PSO method consistently outperforms traditional optimization techniques in terms of convergence speed and accuracy. As a result, we obtained the optimal parameter set K in equation (48), and the evaluation will be conducted in Section 6.

$$\begin{cases} k = [K1 \quad K2 \quad K3 \quad K4] \\ \quad = [322.7167 \quad -524.0909 \quad 14.4182 \quad 5.8882] \\ J = 7.0307*10^{-26} \end{cases} \tag{48}$$

6. Simulation and Experimental Results

To conduct experiments with the algorithm, we implemented our approach on the physical model shown in Fig. 9, using the STM32F407 Discovery control board and the H-Bridge board depicted in Figures 16 and 17. The connections are illustrated in Figure 18. Subsequently, we developed a control program that was uploaded from MATLAB, as shown in Fig. 19.

Fig. 16. STM32F407 Processing Board

Fig. 17. H-Bridge Board

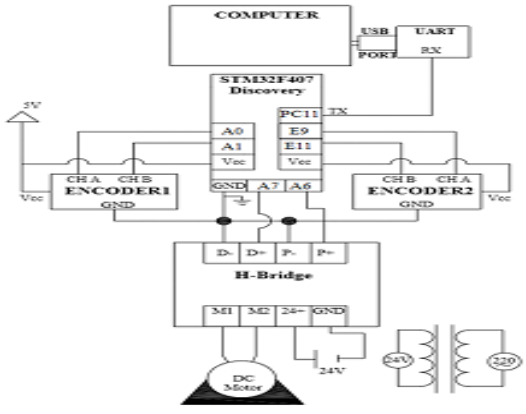

Fig. 18. Electrical Control Diagram.

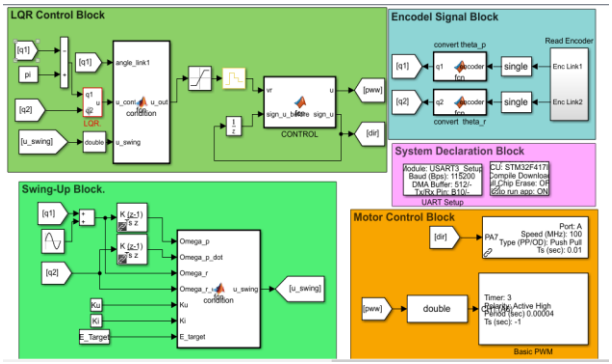

Fig. 19. Control Program for the Real System.

6.1. Simulation Results

Angle of pendulum after PSO compared to basic.

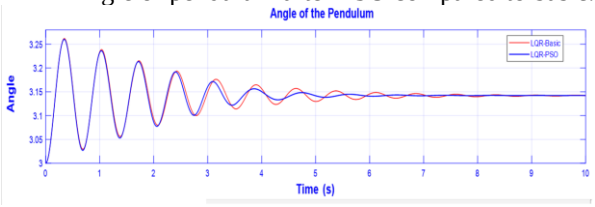

Fig. 20. Angle of Pendulum response (Simulation)

The Angle of the Flywheel Optimized by PSO Compared to basic.

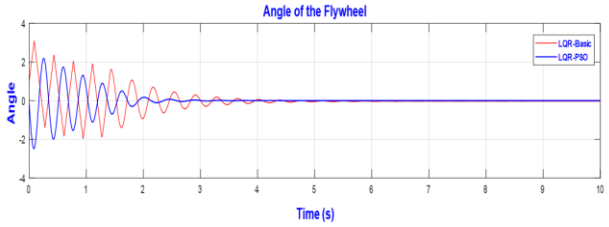

Fig. 21. Angle of the Flywheel.

Motor Voltage Before and After Optimization.

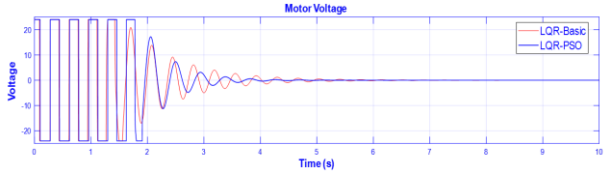

Fig. 22. Motor Voltage.

Simulation of combined LQR-PSO with Swing-Up control.

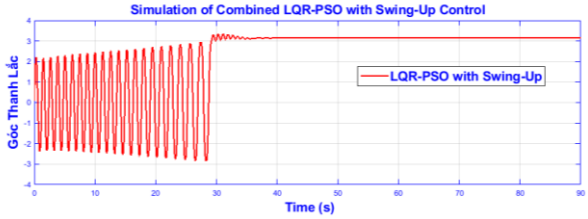

Fig. 23. Swing up and LQR-PSO

From Fig. 20 to 22, it is evident that the PSO leads to more stable LQR parameters, with improved response and settling time. This optimization enhances the overall performance of the control system. Additionally, Fig. 23 illustrates the effective energy management during the swing-up process, resulting in successful positioning of the RWIP system at $\theta_p = \pi$, showcasing the potential of the proposed control strategy.

6.2. Results of the Experiments

The RWIP is effectively stabilized using an LQR controller that has been optimized through PSO. The outcomes are illustrated in Figures 24-25, along with the initial values of the system, where:

$$\begin{cases} \theta_p = \pm 0.3491(rad) \ \left(\pm 20^0\right) \\ \theta_r = 0(\text{rad}) \ \left(0^0\right) \\ \dot{\theta}_p = 0(\text{rad/s}) \\ \theta_r' = 0(\text{rad/s}) \end{cases}$$

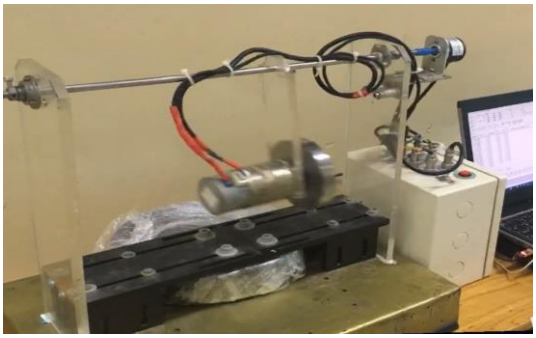

Fig. 24. Unstable state model of the RWIP system

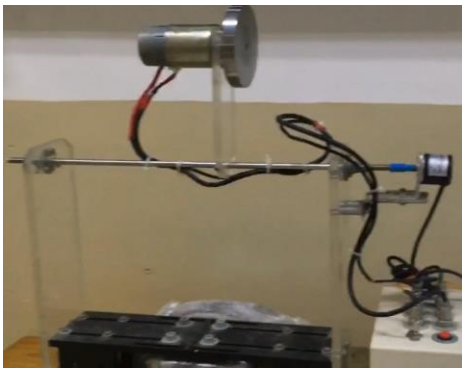

Fig. 25. Stable state model of the RWIP system

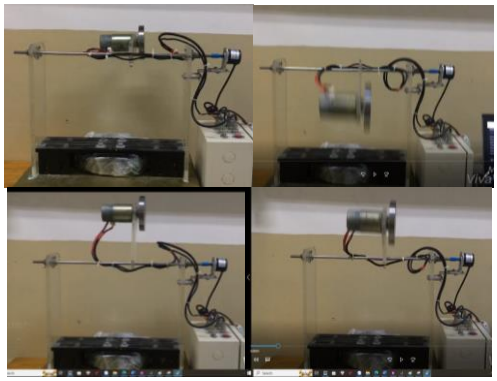

Fig. 26. Practical Swing-Up Control.

The experiment yielded the angle of the pendulum when at the position $\theta_p = \pi$:

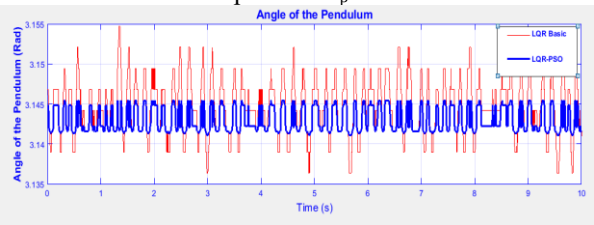

Fig. 27. Angle of Pendulum Response (Experiment)

We obtained the angle of the pendulum during the swing-up process.

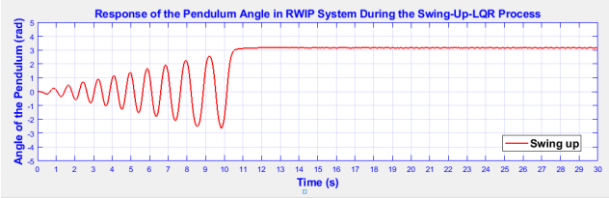

Fig. 28. Swing-Up and Balancing in Experimental

7. Conclusions

This study has presented the optimization of the LQR controller for RWIP through a practical energy control solution. Experimental results from Fig. 23 to Fig. 28 demonstrate that the application of PSO algorithm not only enhances the stability of the system but also improves responsiveness during the swing-up process.

Specifically, the optimized LQR parameters enable the system to quickly achieve the desired equilibrium position while minimizing energy consumption. The energy control during the swing-up process is clearly illustrated through the angular and voltage metrics of both the flywheel and the pendulum, indicating the effectiveness of this method in controlling the complex dynamics of the system.

Future Directions: These results provide valuable insights into the control capabilities of the RWIP and pave the way for future research in the fields of nonlinear control and energy optimization. Future studies may focus on the application of alternative optimization algorithms and the integration of machine learning techniques to further enhance control performance.

8. References

[1] Valenzuela J.M. , Avelar C.A.: "Motion Control of Underactuated Mechanical Systems," Part of the book series: Intelligent Systems, Control and Automation: Science and Engineering (ISCA, volume 88), 2018. DOI:10.1007/978-3-319-58319-8

[2] Medina I., Santiago L., Hernández Gómez J.J., Castillo R., Castaned C.C.:" Speed PID controller simulation of a reaction wheel for CubeSat orientation applications," Journal of Physics Conference Series 1723(1):012013. DOI:10.1088/1742-6596/1723/1/012013

[3] Astrom K., Hagglund T.: "PID controllers: theory, design and tuning", Instrument Society of America; 1995.

[4] Astrom K.J. and Furuta K.:'' Swinging up a pendulum by energy control," Automatica Volume 36, Issue 2, Pages 287-295, February 2000. DOI: 10.1016/S0005-1098(99)00140-5

[5] Sowman J., Laila D. S. and Longo S.:" Real-Time Approximate Explicit Nonlinear Model Predictive Control for the Swing-Up of a Reaction Wheel Pendulum," 2015 IEEE 54th Annual Conference on Decision and Control (CDC) , 12/2015. Osaka, Japan. DOI: 10.1109/CDC.2015.7402891

[6] Srinivas K.N., Behera L.:" Swing-up control strategies for a reaction wheel pendulum," International Journal of Systems Science Vol. 39, No. 12, 1165–1177, 12/2008. DOI: 10.1080/00207720802095137

[7] Andrievsky B.R.:" Global stabilization of the unstable Reaction-Wheel Pendulum", Autom Remote Control 72, 1981–1993, Copyright 2011. DOI: 10.1134/S0005117911090189

[8] Kiattisin K.: " Lqr and MPC Controller Design and Comparison for a Stationary Self-Balancing Bicycle Robot with a ReactionWheel", Kybernetika 51 no. 1, 173-191, 2015. DOI: 10.14736/kyb-2015-1-0173

[9] Murdock D.D. and Taylor D.G.:" Balancing a Reaction Wheel Pendulum with PM Synchronous Motor Actuation", IECON 2014 - 40th Annual Conference of the IEEE Industrial Electronics Society, 2014. DOI: 10.1109/IECON.2014.7048483

[10] David M.:" in Chapter 6, The Lagrangian Method," Harvard University, Copyright 2007.

[11] Benedetti M., Azaro R., Franceschini D. and Massa A.:" PSO-Based Real-Time Control of Planar Uniform Circular Arrays," IEEE Antennas and Wireless Propagation Letters, Vol. 5, 2006 DOI: 10.1109/LAWP.2006.887553

[12] Howimanporn S., Thanok S., Chookaew S., and Sootkaneung W.:" Design and Implementation of PSO Based LQR Control for Inverted Pendulum through PLC," Proceedings of the 2016 IEEE/SICE International Symposium on System Integration, Sapporo Convention Center, Sapporo, Japan,December 13-15, 2016. DOI: 10.1109/SII.2016.7844075

[13] Juneja M., Nagar S.K.:" Particle swarm optimization algorithm and its parameters: A review," International Conference on Control, Computing, Communication and Materials (ICCCCM), 2016. DOI: 10.1109/ICCCCM.2016.7918233

[14] Farnaz Z, Sajith H.S., Abeysekara A.D., Welikala S., Binduhewa P.J., Ekanayake P.B., Samaranayake L.:"Dc Motor Torque Control Using State Estimation.", Control and Intelligent Systems, Vol. 44, No. 3, 2016. DOI: 10.2316/Journal.201.2016.3.201-2762

[15] Spong M.W., Corke P., Lozano R.:" Nonlinear control of the Reaction Wheel Pendulum," Automatica Volume 37, Issue 11, Pages 1845-185, November 2001. DOI:10.1016/S0005-1098(01)00145-5

Acknowledgment:

This research received funding from the Posts and Telecommunications Institute of Technology, Vietnam, under grant number 05-2024-HV-KTĐT2.