Trajectory Control for Double-Linked Parallel Rotary Inverted Pendulum

Thanh-Truong Mai [1], Tuan-Thuong Le [2,*], Hong-Quang Le [3], Bao-Duy Than [4], Hoang-Thien-Hung Trinh [5], Dinh-Truc Nguyen [6], Hung-Thinh Tran [7], Tran-Quang-Huy Le [8], Tri-Dung Hoang [9], Sy-Luan Duong [10], Hoang-Chinh Tran [11], Ha-Duy Nguyen [12], Thi-My-Linh Dong [13]
[1,2,3,4,5,6,7,8,9,10,12,13] Ho Chi Minh City University of Technology and Education (HCMUTE), Vietnam
[11] Cao Thang Technical College, Vietnam
Email: [1] 22151319@student.hcmute.edu.vn, [2] 22151307@student.hcmute.edu.vn, [3] 21145644@student.hcmute.edu.vn, [4] 21145098@student.hcmute.edu.vn , [5] 21145155@student.hcmute.edu.vn, [6] 21145308@student.hcmute.edu.vn, [7] 21145652@student.hcmute.edu.vn, [8] 21145148@student.hcmute.edu.vn, [9] 21145100@student.hcmute.edu.vn, [10] 21145199@student.hcmute.edu.vn, [11] tranhoangchinh@caothang.edu.vn, [12] 21145095@hcmute.student.edu.vn, [13] 21145192@student.hcmute.edu.vn
*Corresponding Author

***Abstract*—The rotary inverted pendulum (RIP) is a benchmark nonlinear underactuated system commonly used in control research, with various extensions such as multi-link and parallel configurations developed to increase complexity and evaluate advanced controllers. This paper presents a hybrid control strategy combining Linear Quadratic Regulator (LQR) and a Genetic Algorithm (GA) for stabilizing and tracking control of a rotary double-linked parallel inverted pendulum (RDPIP), a nonlinear under-actuated single input-multi output (SIMO) system. The LQR controller is designed based on a linearized state-space model at the TOP-TOP equilibrium point. To enhance performance, the weighting matrices Q and R are optimized using GA with a fitness function minimizing trajectory error. Simulation results demonstrate that the GA-optimized controller (LQR 2) achieves superior performance compared to the trial-based LQR (LQR 1), with a reduced settling time of 0.5 seconds, lower oscillation amplitudes, and improved tracking of reference signals under sinusoidal and pulse disturbances. Specifically, the pendulums reached steady state within 2–3 seconds, and the arm settled within 6 seconds. These findings confirm the effectiveness of a hybrid strategy and robustness of the proposed hybrid approach for RDPIP control, laying a foundation for future implementation in real-world applications.**

***Keywords*—Rotary Double Parallel Inverted Pendulum; Nonlinear System; LQR Control; Genetic Algorithm; SIMO System**

I. INTRODUCTION

RIP has long served as a standard benchmark for the design and validation of nonlinear control strategies in under-actuated systems due to its intrinsic instability and coupled dynamics [1], [2]. While the basic single-link RIP has been thoroughly explored in both academic and laboratory environments [3], its multi-link variants, such as RDRIP, have attracted increasing attention for testing advanced control methods on higher-order systems [4], [5].

A further extension, RDPIP, features two pendulums mounted symmetrically at each end of a horizontal rotating arm. This configuration introduces a higher level of complexity in terms of coupling and control, modeling the system as a nonlinear, SIMO dynamic model [6], [7]. Despite its rich dynamics and relevance to real-world applications, the RDPIP has been comparatively less addressed in recent literature, especially with regard to robust trajectory tracking.

Existing studies on RDPIP control have primarily focused on basic stabilization or swing-up techniques, using PID, fuzzy logic, or energy-based control approaches [8]-[10]. More recent efforts have applied model-based strategies such as LQR and H-infinity control to improve stability and reduce energy consumption [11]-[13]. However, these methods often rely on manually tuned parameters, limiting their robustness and adaptability. To address this, heuristic optimization techniques like GA have been successfully employed to tune control gains and improve performance under nonlinear constraints [14], [15].

Several recent studies have explored advanced control strategies for underactuated systems. Mustafa et al. [23] introduced fractional order control for rotary double pendulums, while Chen and Huang [25] addressed adaptive control robustness under time-varying uncertainties. Sanjeewa and Parnichkun [24] proposed H∞ control for RDPIP but noted its dependence on precise modeling. To overcome such limitations, metaheuristic and soft computing methods have been increasingly adopted. Notably, Panjwani et al. [26] and Karthick et al. [27] applied NSGA II and SMC, respectively, to optimize LQR gains, improving stability and trajectory tracking. Yildiran [28] employed reinforcement learning to adapt LQR controllers, and Le et al. [29] analyzed how GA variant choices affect tuning outcomes. Nguyen et al. [30] demonstrated the effectiveness of PSO-LQR for rotary pendulum stabilization. These efforts support the growing trend toward hybrid intelligent control and reinforce the relevance of GA-based LQR optimization for complex systems such as RDPIP. In this paper, we propose a hybrid LQR-GA approach to optimize the state-feedback controller of RDPIP. The novelty of our study lies in combining optimal control design with evolutionary search techniques for automatic tuning of weighting matrices. This not only enhances system response but also strengthens disturbance rejection capability. Furthermore, we evaluate our method on the TOP-TOP equilibrium point, incorporating a reference tracking scenario with dynamic setpoints - a feature often neglected in earlier RDPIP works [16]-[19].

Compared to prior papers such as study [20], Fahmizal [22], which used ANFIS for control synthesis, and Dang et al. [21], which applied PID to simpler inverted pendulum models, our approach demonstrates superior transient performance and better generalizability. With detailed simulation results and quantitative metrics, this paper contributes a practical and extendable solution for high-order under-actuated control systems.

II. Mathematical Equations and Simulation

A. Mathematical Equations

Mathematical model of PRIP is illustrated in Fig. 1. A DC motor is mounted vertically on a fixed base, and its output shaft is connected to center of a rigid horizontal link, referred to as the arm. At both ends of arm, rotary encoders are attached. Each encoder serves as a pivot point for an attached link, referred to as a pendulum. These pendulums are allowed to rotate freely about axis of their respective encoders. The angular displacement of each pendulum is continuously measured by the corresponding encoder. Parameters of PRIP are listed in Table 1 and Table 2.

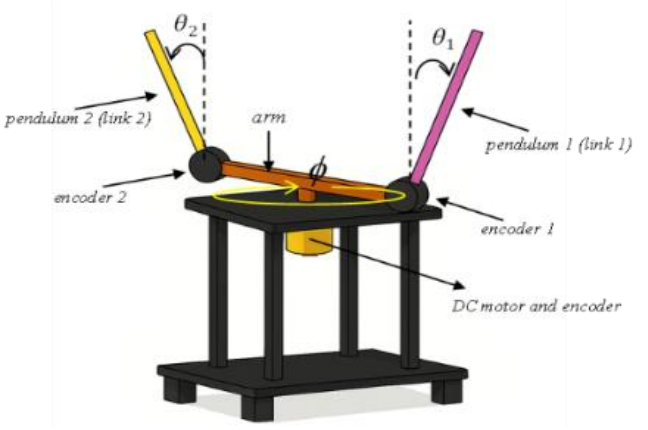

Fig. 1. Mathematical model of RDPIP

Table 1. System parameters of RDPIP (i=1, 2)

Parameters	Unit	Description
m_i	kg	Mass of pendulum i
l_{gi}	m	Distance to center of mass of pendulum i
J_i	$kg.m^2$	Moment of inertia of pendulum i
L	m	Length of arm
J_0	$kg.m^2$	Moment of inertia of the arm
g	m/s^2	Gravitational acceleration
C_i	$Nm.s$	Friction coefficient of pendulum i
C_0	$Nm.s$	Friction coefficient of the arm

Table 2. System variables of PRIP

Variables	Unit	Description
ϕ	rad	Angular displacement of the arm
$\dot{\phi}$	rad/s	Angular velocity of the arm
θ_1	rad	Angular displacement of pendulum 1
$\dot{\theta}_1$	rad/s	Angular velocity of pendulum 1
θ_2	rad	Angular displacement of pendulum 2
$\dot{\theta}_2$	rad/s	Angular velocity of pendulum 2

Dynamic equations of system based on Euler-Lagrange formulation are as follows [8]:

$$\frac{d}{dt}\frac{\partial L}{\partial \dot{q}_i} - \frac{\partial L}{\partial q_i} + \frac{\partial W}{\partial \dot{q}_i} = F_i \tag{1}$$

The kinetic energy K (N) of the RDPIP is then given by:

$$K = \frac{1}{2}J_0\dot{\phi}^2 + \frac{1}{2}J_1{\dot{\theta}_2}^2 + \frac{1}{2}m_1{v_1}^2 + \frac{1}{2}m_2{v_2}^2 \tag{2}$$

In this context, v_1 and v_2 represent the velocities of pendulum 1 and pendulum 2, respectively. The total kinetic energy of the system is then rewritten as follows:

$$\begin{aligned} K = {} & \frac{1}{2}J_0\dot{\phi}^2 + \frac{1}{2}J_1{\dot{\theta}_1}^2 + \frac{1}{2}J_2{\dot{\theta}_2}^2 + \\ & + \frac{1}{2}m_1(l_{g1}\sin\theta_1\,\dot{\phi})^2 + \frac{1}{2}m_1(L\dot{\phi})^2 + \\ & + \frac{1}{2}m_1(l_{g1}\dot{\theta}_1)^2 - m_1 l_{g1}L\cos\theta_1\,\dot{\theta}_1\dot{\phi} + \\ & + \frac{1}{2}m_2(l_{g2}\sin\theta_2\,\dot{\phi})^2 + \frac{1}{2}m_2(L\dot{\phi})^2 + \\ & + \frac{1}{2}m_2(l_{g2}\dot{\theta}_2)^2 - m_2 l_{g2}L\cos\theta_2\,\dot{\theta}_2\dot{\phi} \end{aligned} \tag{3}$$

The potential energy U (N) of RDPIP is expressed as:

$$U = m_1 g l_{g1}\cos\theta_1 + m_2 g\, l_{g\,2}\cos\theta_2 \tag{4}$$

The dissipated energy W (N) of the RDPIP is given by:

$$W = \frac{1}{2}c_0\dot{\phi}^2 + \frac{1}{2}c_1{\dot{\theta}_1}^2 + \frac{1}{2}c_2\dot{\theta}_2^2 \tag{5}$$

From (1) and (5), the Lagrangian function is computed as follows:

$$\begin{aligned} L &= K - U \\ &= \frac{1}{2}J_0\dot{\phi}^2 + \frac{1}{2}J_1{\dot{\theta}_1}^2 + \frac{1}{2}J_2\dot{\theta}_2^2 \\ &+ \frac{1}{2}m_1(l_{g1}\sin\theta_1\,\dot{\phi})^2 + \frac{1}{2}m_1(L\dot{\phi})^2 \\ &+ \frac{1}{2}m_1(l_{g1}\dot{\theta}_1)^2 - m_1 l_{g1}L\cos\theta_1\,\dot{\theta}_1\dot{\phi} \\ &+ \frac{1}{2}m_2(l_{g2}\sin\theta_2\,\dot{\phi})^2 + \frac{1}{2}m_2(L\dot{\phi})^2 \\ &+ \frac{1}{2}m_2(l_{g2}\dot{\theta}_2)^2 - m_2 l_{g2}L\cos\theta_2\,\dot{\theta}_2\dot{\phi} \\ &- m_1 g l_{g1}\cos\theta_1 - m_2 g l_{g2}\cos\theta_2 \end{aligned} \tag{6}$$

The RDPIP is actuated by a DC motor. The relationship between torque τ (N.m) and input voltage V_{in} (V) is given by the following equation:

$$\tau = (K_t V_{in} - K_t K_b\dot{\phi})/R_m \tag{7}$$

According to [16], the linearized dynamic equations of the RDPIP are derived from (6) and (7) as follows:

$$\begin{bmatrix} E_1 & F_1 & I_1 \\ E_2 & F_2 & I_2 \\ E_3 & F_3 & I_3 \end{bmatrix}\begin{bmatrix} \ddot{\phi} \\ \ddot{\theta}_1 \\ \ddot{\theta}_2 \end{bmatrix} + \begin{bmatrix} T_1 \\ T_2 \\ T_3 \end{bmatrix} = \frac{K_t}{R_m}\begin{bmatrix} V_{in} \\ 0 \\ 0 \end{bmatrix} \tag{8}$$

Where:

$E_1 = J_0 + m_1 l_{g1}^2\sin^2\theta_1 + m_1 L^2 + m_2 l_{g2}^2\sin^2\theta_2 + m_2 L^2;$

$F_1 = -m_1 L l_{g1}\cos\theta_1\,;$

$I_1 = -m_2 L l_{g2}\cos\theta_2\,; I_2 = 0;$

$T_1 = m_1 l_{g1}^2\dot{\theta}_1\dot{\phi}\sin(2\theta_1) + m_1 l_{g1}L\dot{\theta}^2{}_1\sin\theta_1 + c_0\dot{\phi} +$

$m_2 l_{g2}^2\dot{\theta}_2\dot{\phi}\sin(2\theta_2) + (c_0 + \frac{K_t K_b}{R_m})\dot{\phi};$

$E_2 = -m_1 L l_{g1}\cos\theta_1\,; F_2 = J_1 + m_1{l_{g1}}^2; I_2 = 0;$

$$T_2 = -m_1 {l_{g1}}^2 \dot{\phi}^2 \sin\theta_1 \cos\theta_1 - m_1 g l_{g1} \sin\theta_1 + c_1 \dot{\theta}_1;$$
$$E_3 = -m_2 L l_{g2} \cos\theta_1 ; F_3 = 0; I_3 = J_2 + m_2 {l_{g2}}^2;$$
$$T_3 = -m_2 {l_{g2}}^2 \dot{\phi}^2 \sin\theta_2 \cos\theta_2 - m_2 g l_{g2} \sin\theta_2 + c_2 \dot{\theta}_2;$$

B. Control Algorithm

1. Linear Quadratic Regulator (LQR)

The Linear Quadratic Regulator (LQR) is utilized to stabilize the RDPIP system based on its linearized state-space model Fig. 2. The controller computes an optimal state-feedback gain matrix that minimizes a quadratic cost function involving both state deviations and control effort. The resulting control law $u = -Kx$ enables the system to maintain stability around the equilibrium point while reducing oscillations. Appropriate selection of weighting matrices Q and R ensures a balance between performance and energy consumption. This approach is well-suited for RDPIP due to the system's high sensitivity and strong coupling dynamics.

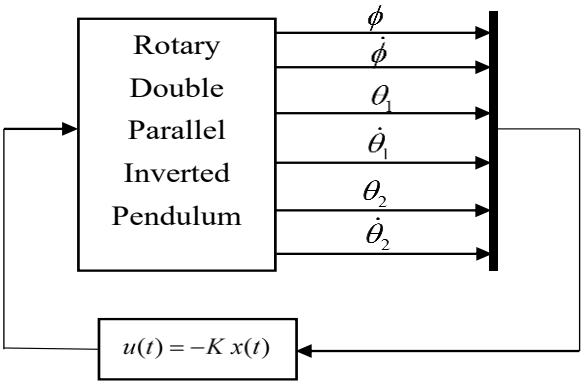

Fig. 2. Block diagram of the principle of the system controlled by the LQR controller

The state variables of the system are defined as follows:

$$\begin{aligned} x_1 &= \phi x_4 = \theta_1 x_7 = \theta_2 \\ x_2 &= \dot{\phi} x_5 = \dot{\theta}_1 x_8 = \dot{\theta}_2 \\ x_3 &= \ddot{\phi} x_6 = \ddot{\theta}_1 x_9 = \ddot{\theta}_2 \end{aligned} \tag{9}$$

Equation (8) is converted into the following system of equations:

$$\begin{cases} x_3 = \ddot{\phi} = h_1(\phi, \dot{\phi}, \theta_1, \dot{\theta}_1, \theta_2, \dot{\theta}_2) \\ x_6 = \ddot{\theta}_1 = h_2(\phi, \dot{\phi}, \theta_1, \dot{\theta}_1, \theta_2, \dot{\theta}_2) \\ x_9 = \ddot{\theta}_2 = h_3(\phi, \dot{\phi}, \theta_1, \dot{\theta}_1, \theta_2, \dot{\theta}_2) \end{cases} \tag{10}$$

From Equation (8), combined with the "solve" function in MATLAB referred in Fig. 3, the result for Equation (10) is derived.

```
H=M1*[x3;x6;x9]+M2-M3;
h1=H(1);
h2=H(2);
h3=H(3);
[x3 x6 x9] = solve(h1,h2,h3,x3,x6,x9)
```

Fig. 3. Illustrates the computation performed using MATLAB

The LQR controller is linear, whereas the RDPIP system is inherently nonlinear. Therefore, the system must be linearized around an operating point, specifically the TOP-TOP equilibrium point.

$$\phi \approx \dot{\phi} \approx \theta_1 \approx \dot{\theta}_1 \approx \theta_2 \approx \dot{\theta}_2 \approx 0; V_{in} \approx 0; \tag{11}$$

$$x = [\phi\ \dot{\phi}\ \theta_1\ \dot{\theta}_1\ \theta_2\ \dot{\theta}_2]^T \tag{12}$$

The linearized state-space equation of the RDPIP is given by:

$$\dot{x} = Ax + Bu \tag{13}$$

Where:

$$A = \begin{bmatrix} \frac{\partial(f_1)}{\partial\phi} & \frac{\partial(f_1)}{\partial\dot{\phi}} & \frac{\partial(f_1)}{\partial\theta_1} & \frac{\partial(f_1)}{\partial\dot{\theta}_1} & \frac{\partial(f_1)}{\partial\theta_2} & \frac{\partial(f_1)}{\partial\dot{\theta}_2} \\ \frac{\partial(f_2)}{\partial\phi} & \frac{\partial(f_2)}{\partial\dot{\phi}} & \frac{\partial(f_2)}{\partial\theta_1} & \frac{\partial(f_2)}{\partial\dot{\theta}_1} & \frac{\partial(f_2)}{\partial\theta_2} & \frac{\partial(f_2)}{\partial\dot{\theta}_2} \\ \frac{\partial(f_3)}{\partial\phi} & \frac{\partial(f_3)}{\partial\dot{\phi}} & \frac{\partial(f_3)}{\partial\theta_1} & \frac{\partial(f_3)}{\partial\dot{\theta}_1} & \frac{\partial(f_3)}{\partial\theta_2} & \frac{\partial(f_3)}{\partial\dot{\theta}_2} \\ \frac{\partial(f_4)}{\partial\phi} & \frac{\partial(f_4)}{\partial\dot{\phi}} & \frac{\partial(f_4)}{\partial\theta_1} & \frac{\partial(f_4)}{\partial\dot{\theta}_1} & \frac{\partial(f_4)}{\partial\theta_2} & \frac{\partial(f_4)}{\partial\dot{\theta}_2} \\ \frac{\partial(f_5)}{\partial\phi} & \frac{\partial(f_5)}{\partial\dot{\phi}} & \frac{\partial(f_5)}{\partial\theta_1} & \frac{\partial(f_5)}{\partial\dot{\theta}_1} & \frac{\partial(f_5)}{\partial\theta_2} & \frac{\partial(f_5)}{\partial\dot{\theta}_2} \\ \frac{\partial(f_6)}{\partial\phi} & \frac{\partial(f_6)}{\partial\dot{\phi}} & \frac{\partial(f_6)}{\partial\theta_1} & \frac{\partial(f_6)}{\partial\dot{\theta}_1} & \frac{\partial(f_6)}{\partial\theta_2} & \frac{\partial(f_6)}{\partial\dot{\theta}_2} \end{bmatrix} \tag{14}$$

$$B = \left[\frac{\partial f_1}{V_{in}} \quad \frac{\partial f_2}{V_{in}} \quad \frac{\partial f_3}{V_{in}} \quad \frac{\partial f_4}{V_{in}} \quad \frac{\partial f_5}{V_{in}} \quad \frac{\partial f_6}{V_{in}}\right]^T \tag{15}$$

Where:

$f_1 = \dot{x}_1 = x_2;\ f_2 = \dot{x}_2 = x_3;\ f_3 = \dot{x}_4 = x_5$
$f_4 = \dot{x}_5 = x_6;\ f_5 = \dot{x}_7 = x_8;\ f_6 = \dot{x}_8 = x_9$
Weighting matrix Q:

$$Q = \begin{bmatrix} Q_1 & 0 & 0 & 0 & 0 & 0 \\ 0 & Q_2 & 0 & 0 & 0 & 0 \\ 0 & 0 & Q_3 & 0 & 0 & 0 \\ 0 & 0 & 0 & Q_4 & 0 & 0 \\ 0 & 0 & 0 & 0 & Q_5 & 0 \\ 0 & 0 & 0 & 0 & 0 & Q_6 \end{bmatrix} \tag{16}$$

Energy coefficient R (R > 0).
Solving the Riccati equation, the solution is matrix P:

$$PA + A^T P + Q - PBR^{-1}B^T P = 0 \tag{17}$$

Control matrix K:

$$K = R^{-1}B^T P \tag{18}$$

Caculate matrix K in Matlab:

$$K = lqr(A, B, Q, R) \tag{19}$$

2. Genetic Algorithms (GA) for LQR

To enhance the performance of the LQR controller, GA is applied to optimize the weighting matrices Q and R in the energy coefficient. This approach enables automatic tuning of control parameters without relying on manual trial-and-error procedures. The GA iteratively evolves a population of candidate solutions by employing selection, crossover, and mutation operations, with the fitness function defined based on the system's tracking accuracy and control effort. By optimizing the LQR gains in this manner, the controller achieves improved stabilization and reduced oscillation for the nonlinear and highly coupled RDPIP system. This method demonstrates robustness and adaptability, particularly suitable for systems with complex dynamics such as RDPIP.

In this study, GA is implemented in an offline manner. The GA configuration includes a population size of N=100, with selection based on linear ranking using a selection pressure coefficient η=0.2. Decimal coding is adopted for chromosome representation, and two-point crossover is used to promote genetic diversity. The crossover and mutation probabilities are set to 0.9 and 0.1, respectively, ensuring a balance between exploration and exploitation during the optimization process. Choose fitness function:

$$J = \sum_{i=0}^{n} ({e_1}^2(i) + {e_2}^2(i) + {e_3}^2(i)) \quad (20)$$

With $e_1 = \phi; e_2 = \theta_1; e_3 = \theta_2$, n as the number of samples during the simulation, value of function J depends on the errors of the arm angle and the two pendulum angles. In this case, with a simulation time of 100 seconds and a sampling interval of 0.01 seconds, we have n = 10001 samples.

As illustrated in Fig. 4, GA is employed to optimize the weighting matrices Q and R of the LQR controller. The process begins with the initialization of a population of candidate solutions, where each individual represents a pair of Q-R matrices encoded as chromosomes.

For each individual in population, fitness function J is evaluated based on the cumulative tracking error of the RDPIP, considering deviations of both pendulums and rotating arm from their reference trajectories. If the current fitness J is lower than the best-known fitness value J min, the algorithm updates J min and stores the corresponding individual as the current best solution.

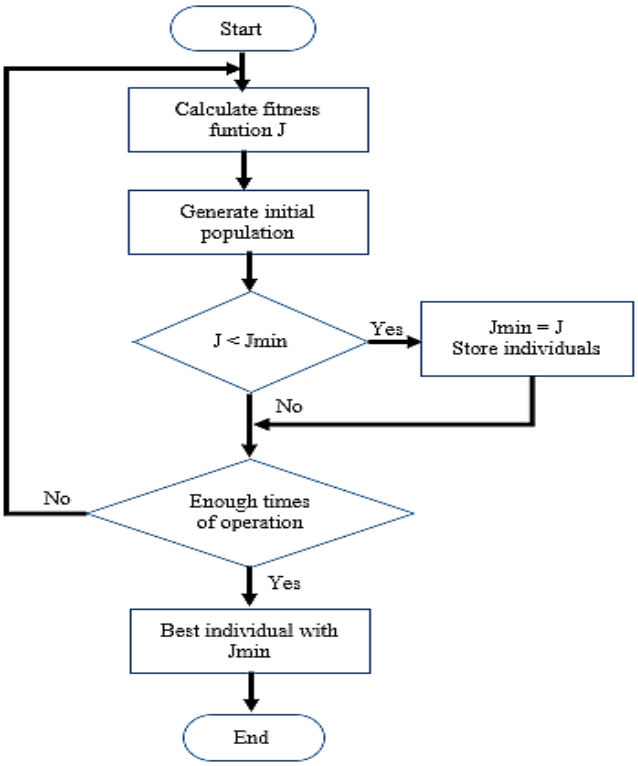

Fig. 4. Flowchart of GA used to optimize LQR weighting matrices Q and R. The fitness function minimizes trajectory error across the arm and two pendulums [17]

The population then undergoes a series of genetic operations, including selection (typically via linear ranking), crossover (e.g., two-point), and mutation, to generate new individuals for the next generation. This process continues iteratively, refining the solution space until a predefined number of generations or convergence criterion is met.

Once the termination condition is satisfied, the individual with the lowest fitness value J min is selected as the optimal solution. This individual defines the final optimized values of Q and R matrices, which are then used in the LQR controller to improve system performance in terms of settling time, stability, and trajectory tracking.

C. Simulation Program

The selection of control parameters plays a critical role in determining the overall performance and stability of the control system. In practical applications, several metaheuristic optimization techniques have been employed to identify optimal parameters, with PSO [18] and GA [17] being among the most commonly used. In this study, the GA is employed to optimize the controller parameters for all three control strategies under consideration. Specifically, for LQR controller, optimal feedback gain matrix K must be determined. Since K is computed based on weighting matrices Q and R, which directly influence the trade-off between system performance and control effort, GA is utilized to identify the most suitable values for Q and R that minimize cost function and enhance overall system behavior. Optimization process is performed offline with fixed simulation duration and sampling time, allowing the evaluation of control performance through a well-defined fitness function. In addition to the main fitness function J, metrics such as RMSE and settling time are used to assess the quality of control performance. These indicators help quantify oscillation, response speed, and steady-state accuracy. This approach not only improves response quality but also ensures robustness against modeling uncertainties and nonlinearities inherent in RDPIP.

Matrices A and B are calculated based on the parameters presented in Table 3 and Table 4 as follows:

$$A = \begin{bmatrix} 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & -13.1371 & 0.3356 & -0.0007 & 0.1668 & -0.0016 \\ 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & -45.4646 & 67.7307 & -0.1406 & 0.5774 & -0.0055 \\ 0 & 0 & 0 & 0 & 0 & 1 \\ 0 & -33.6328 & 0.8593 & -0.0018 & 49.6723 & -0.4718 \end{bmatrix}$$

$$B = [0 \quad 0.0125 \quad 0 \quad 0.0434 \quad 0 \quad 0.0321]^T \quad (21)$$

Table 3. Parameters of DC Motor

Parameters	Unit	Value
K_b	$V/(rad/sec)$	0.064934
K_t	$V/(rad/sec)$	0.064934
R_m	Ω	6.835271

Table 4. Parameters of RDPIP

Parameters	Pendulum 1	Pendulum 1	Arm
m_i	0.059	0.0475	na
l_{gi}	0.127	0.106	na
J_i	0.0001526	0.0004693	na
L	na	na	0.51
J_0	na	na	0.75
g		9.81	
C_i	1.526×10^{-4}	4.692×10^{-4}	na
C_0	na	na	4.978

The controllability of the system is analyzed as follows:

$$C_t = [B \quad AB \quad A^2B \quad A^3B \quad A^4B \quad A^5B] \tag{22}$$

Substituting matrices, A and B into (21), we obtain:

$$rank(C_t) = 6; det(C_t) = -0.9522 \tag{23}$$

Since the rank of the matrix equals the order of the system (6th order) and the determinant is non-zero, the system is controllable.

Table 5. Initial state variables

Parameters	Value
x_1_init	0.001
x_2_init	0.0012
x_4_init	0.002
x_5_init	0.002
x_7_init	0.002
x_8_init	0.004

The simulations were conducted using MATLAB/Simulink R2022a. The sampling interval was set to 0.01 seconds, with a total simulation time of 10 seconds. The solver used was the variable-step ODE45, and all simulations were executed under identical conditions to ensure consistency in controller performance evaluation.

Fig. 5, Fig. 6, Fig. 7 represent the simulation of the control algorithms using Matlab/Simulink software version 2022a. In the Rotary Inverted Pendulum block, the equations are those outlined in section A of II. Similarly, the blocks for LQR Controller contain the equations presented in section B of II. The output response of the system for each algorithm is observed through the Scope block. All results are presented in section III.

A trial-and-error approach is employed to determine suitable values for the weighting matrices Q and R.

$$Q = \begin{bmatrix} 10^7 & 0 & 0 & 0 & 0 & 0 \\ 0 & 10^7 & 0 & 0 & 0 & 0 \\ 0 & 0 & 10^9 & 0 & 0 & 0 \\ 0 & 0 & 0 & 10^6 & 0 & 0 \\ 0 & 0 & 0 & 0 & 10^7 & 0 \\ 0 & 0 & 0 & 0 & 0 & 10^5 \end{bmatrix}; R = 0.1 \tag{24}$$

The stability of the LQR controller is ensured by solving the Riccati equation at the steady-state operating condition. In this case, the LQR 1 controller stabilizes the RDPIP system when operating around the selected equilibrium point presented in Table 5.

$$K = 10^6[0.01 \quad 0.017 \quad 1.1486 \quad 0.139 \quad -1.3632 \quad -0.1879] \tag{25}$$

III. Results and Discussion

A. Simulation Results of the LQR 1 Controller

The simulation results of the LQR 1 controller using the Q and R matrices selected by the trial-and-error method, as described in (25) and (26), are presented in Fig. 8, Fig. 9, Fig. 10. In this figure, the arm initially oscillates to regulate the two pendulums around the designated operating point. Pendulum 1 reaches its equilibrium state after approximately 2 seconds, while pendulum 2 stabilizes after around 3 seconds. This represents a significant achievement in controlling both pendulums with fast settling times. Once the pendulums are stabilized, the arm converges to 0 rad, with a settling time of approximately 6 seconds.

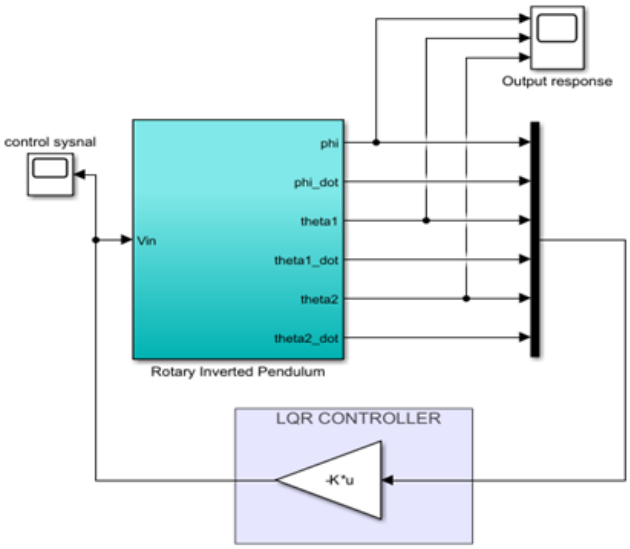

Fig. 5. Simulation for LQR controller for RDPIP

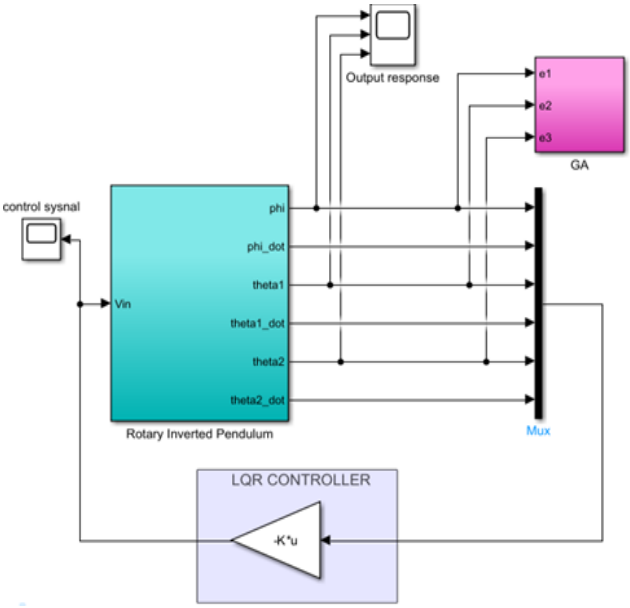

Fig. 6. Simulation using GA to optimize LQR controller for RDPIP

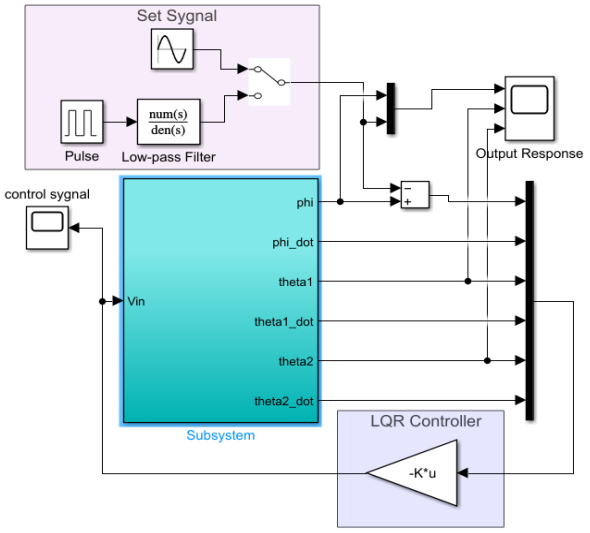

Fig. 7. LQR controller test simulation tracking set signal

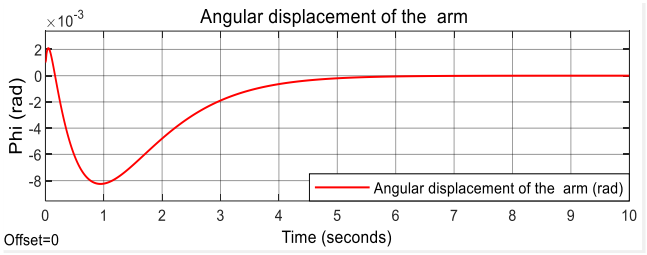

Fig. 8. Angular displacement of the arm

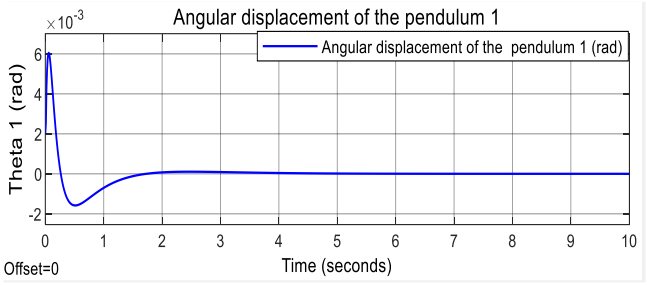

Fig. 9. Angular displacement of the pendulum 1

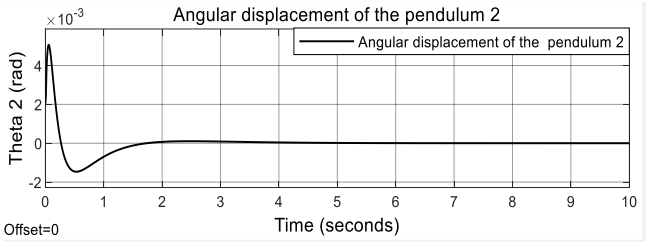

Fig. 10. Angular displacement of the pendulum 2

B. Simulation Results of LQR 2 Controller After Using GA

With LQR 1 controller, controller parameters have not been fully optimized. The result of running the GA program converged after approximately 10 generations. The value of the cost function J is illustrated in Fig. 11, and the optimized LQR 2 controller parameters obtained are as follows:

$$K = 10^5[0.2406 \quad 0.1437 \quad 3.3473 \quad 0.4068 \quad -4.2839 \quad -0.5905] \\ J = 0.4461 \tag{26}$$

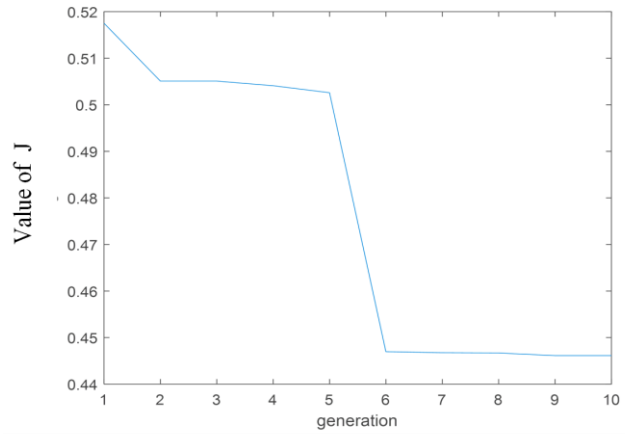

Fig. 11. The change of J over generations

A quantitative comparison of control performance between the conventional LQR (LQR 1) and the GA-optimized LQR (LQR 2) is presented in Table 6 and Fig. 12. The results clearly indicate that LQR 2 significantly improves system response. Specifically, the integral of absolute error (IAE) is reduced from 0.189227 to 0.027819, and the root-mean-square error (RMSE) decreases from 0.034370 to 0.010544 radians. In addition, the oscillation amplitude of the arm is attenuated, narrowing from a range of −0.09 to 0.028 (LQR 1) to −0.058 to 0.039 (LQR 2). The settling time is shortened by approximately 80%, from 5.1679 seconds to 1.0444 seconds, and the rise time is reduced from 0.0291 to 0.0128 seconds. These findings demonstrate the superior performance of the GA-tuned controller in terms of speed, accuracy, and stability.

Table 6. Performance metrics of the rotary arm under LQR 1 and LQR 2 controllers

Angular displacement of the arm	LQR 1	LQR 2
IAE	0.189227	0.027819
RMSE	0.034370	0.010544
Oscillation amplitude	-0.09÷0.028	-0.058÷0.039
Settling time	5.1679	1.0444
Rise time	0.0291	0.0128

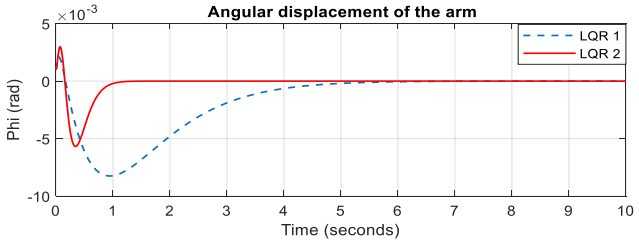

Fig. 12. Angular displacement of the arm

Table 7 and Fig. 13 present the performance metrics for the angular displacement of pendulum 1 under both LQR 1 and LQR 2 control strategies. In contrast to the improvement observed in the rotary arm, the GA-optimized controller (LQR 2) demonstrates a mixed effect on pendulum 1. Although the settling time is reduced from 1.5184 seconds to 1.2225 seconds and the rise time remains nearly unchanged, the error metrics - namely IAE and RMSE - increase slightly under LQR 2, from 0.024246 to 0.033698 and from 0.008010 to 0.013985, respectively. Moreover, the oscillation amplitude expands from [−0.017, 0.067] (LQR 1) to [−0.071, 0.109] (LQR 2), indicating stronger overshoot or more aggressive control action.

Table 7. Performance metrics of angular displacement of the pendulum 1

Angular displacement of the arm	LQR 1	LQR 2
IAE	0.024246	0.033698
RMSE	0.008010	0.013985
Oscillation amplitude	-0.017÷0.067	-0.071÷0.109
Settling time	1.5184	1.2225
Rise time	0.0032	0.003

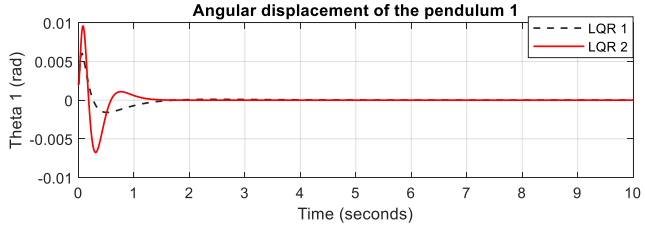

Fig. 13. Angular displacement of the pendulum 1

Table 8 and Fig. 14 present the performance metrics for the angular displacement of pendulum 2 under both LQR configurations. The GA-optimized LQR (LQR 2) shows a clear improvement in most performance indicators compared to the manually tuned version (LQR 1). Notably, the integral of absolute error (IAE) is significantly reduced from 0.022783 to 0.007281, and the RMSE drops from 0.029085 to 0.011777 radians. These reductions reflect a more accurate tracking performance of LQR 2. Additionally, the rise time improves substantially, decreasing from 0.0393 seconds (LQR 1) to 0.0077 seconds (LQR 2), which indicates a much faster initial response.

Table 8. Performance metrics of angular displacement of the pendulum 2

Angular displacement of the arm	LQR 1	LQR 2
IAE	0.022783	0.007281
RMSE	0.029085	0.011777
Oscillation amplitude	-0.016÷0.059	-0.06÷0.09
Settling time	1.55	1.2609
Rise time	0.0393	0.0077

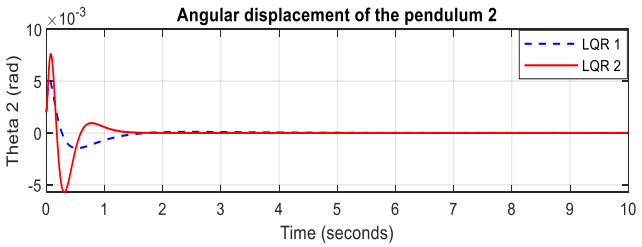

Fig. 14. Angular displacement of the pendulum 2

The results in Table 6, Table 7, Table 8 confirm that the GA-optimized LQR controller significantly improves tracking accuracy and response speed for the RDPIP system. Notable reductions in IAE, RMSE, and settling time were achieved for both the rotary arm and pendulum 2, while pendulum 1 showed improved dynamic response with minor trade-offs in error metrics. These outcomes demonstrate the effectiveness of GA in tuning LQR parameters, offering a robust and adaptive control strategy for nonlinear SIMO systems. The proposed method provides a practical and scalable solution for advanced pendulum control applications.

C. LQR Setpoint Tracking Control

To evaluate whether LQR controller 2 is capable of maintaining the balance of both pendulums at designated operating point, a reference tracking scenario is implemented by varying the desired arm angle as illustrated in Fig. 7. In this study, the reference signal is defined as a sine wave with an amplitude of $\pi/2$ and a frequency of 0.005π, time stop is 1000 seconds representing a smooth and continuous disturbance to test the robustness of controller.

To evaluate the tracking capability of the proposed GA-optimized LQR controller, the RDPIP system was subjected to reference trajectories with different profiles: a continuous sinusoidal signal and a discrete pulse signal. Fig. 15, Fig. 16, Fig. 17 illustrate the system response under the sinusoidal reference, where the rotary arm tracks a sine wave of amplitude $\pm \pi/2$ rad. As shown in Fig. 15, the arm output closely follows the reference curve with minimal phase delay and negligible steady-state error. The pendulums remain stable around the upright position during the entire trajectory, as seen in Fig. 16 and Fig. 17, where their angular displacements stay within ±0.01 rad with quick damping.

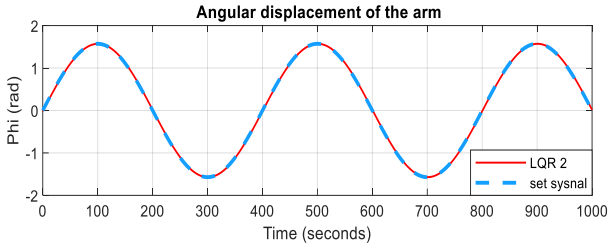

Fig. 15. Angular displacement of the arm and set sine signal

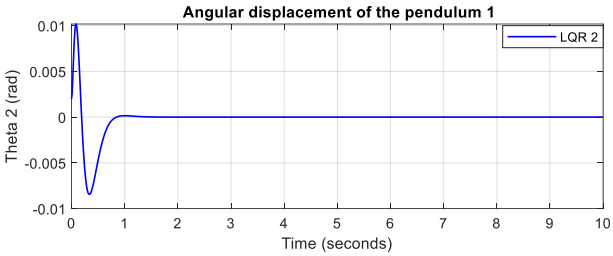

Fig. 16. Angular displacement of the pendulum 1

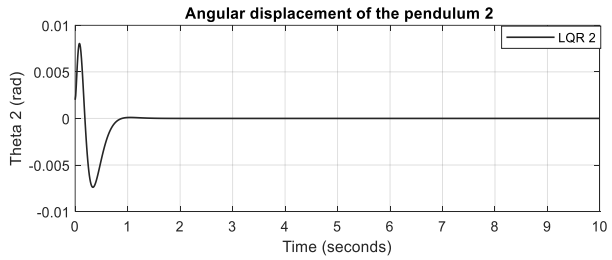

Fig. 17. Angular displacement of the pendulum 2

The reference signal is a square wave with an amplitude of 0.1 rad, a period of 50 seconds, and a duty cycle of 50%, which is passed through a filter to smooth out the sharp transitions:

Under the pulse signal reference shown in Fig. 18, Fig. 19, Fig. 20, the controller again demonstrates robust performance. In Fig. 18, the arm successfully tracks the square wave reference with sharp transitions and minimal overshoot. Despite the abrupt changes in desired angle, the system maintains good precision, and the response remains consistent across cycles. Fig. 19 and Fig. 20 reveal that both pendulums exhibit small, bounded oscillations with magnitudes below ±0.01 rad. The low amplitude and consistent damping after each transition highlight the controller's ability to maintain pendulum stability during dynamic reference shifts.

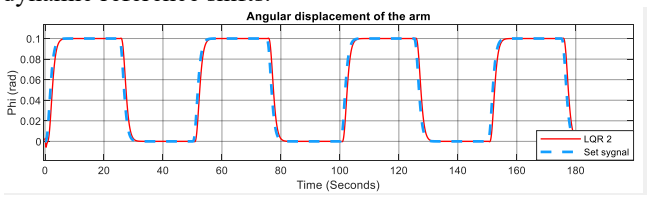

Fig. 18. Angular displacement of the arm and set pulse signal

The results shown in Fig. 15, Fig. 16, Fig. 17 and Fig. 18, Fig. 19, Fig. 20 demonstrate that the LQR-GA controller not

only ensures stabilization at a fixed point but also provides reliable tracking for time-varying references. Its robustness against abrupt disturbances and continuous variation validates the method's suitability for real-time control of nonlinear and under-actuated systems.

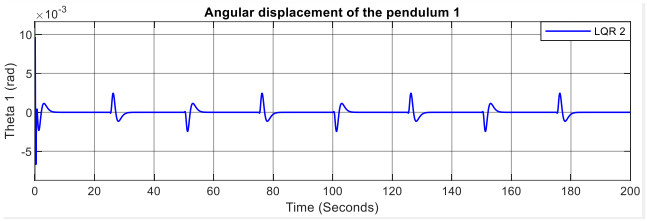

Fig. 19. Angular displacement of the pendulum 1

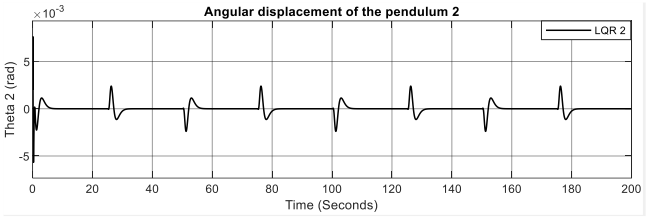

Fig. 20. Angular displacement of the pendulum 1

IV. CONCLUSION

This paper has presented a hybrid LQR-GA control strategy for the stabilization and trajectory tracking of RDPIP. The system was linearized around the TOP-TOP equilibrium, and a state-space model was used for LQR controller design. To enhance performance, GA was applied to optimize the weighting matrices Q and R. The novelty of this work lies in integrating GA-based optimization into LQR tuning for a nonlinear SIMO system, which has been rarely addressed in the context of RDPIP control. Simulation results confirmed that the optimized controller significantly improved system behavior. For the rotary arm, the settling time was reduced from 5.17 s (LQR 1) to 1.04 s (LQR 2), and RMSE decreased from 0.0344 to 0.0105 rad. Similar improvements were observed for the pendulums, particularly in steady-state error and response time. The LQR-GA controller also demonstrated robust tracking of both sinusoidal and pulse reference signals, maintaining pendulum stability under time-varying conditions. These findings demonstrate the effectiveness of GA-based tuning for complex nonlinear SIMO systems and provide a solid foundation for future experimental implementation and extension to higher-order inverted pendulum models.

ACKNOWLEDGEMENT

We want to give thanks to PhD. Van-Dong-Hai Nguyen (lecturer of HCMUTE) due to his supervision for us to complete this research.

REFERENCES

[1] M. W. Spong, "The swing up control problem for the Acrobot," *IEEE Control Systems Magazine*, vol. 15, no. 1, pp. 49-55, 1995, https://doi.org/10.1109/37.341864.

[2] S. Awtar *et al*., "Inverted pendulum systems: rotary and arm-driven-a mechatronic system design case study," *Mechatronics*, vol. 12, no. 2, pp. 357-370, 2002, https://doi.org/10.1016/S0957-4158(01)00075-7.

[3] K. J. Åström and K. Furuta, "Swinging up a pendulum by energy control," *Automatica*, vol. 36, no. 2, pp. 287–295, 2000, https://doi.org/10.1016/S0005-1098(99)00140-5.

[4] K. Ogata, "Modern control engineering," *Prentice hall*, 2010, https://thuvienso.tnut.edu.vn/bitstream/123456789/1094/1/PED%20304.3%20Katsuhiko%20Ogata%20_%20Modern%20Control%20Engineering%205th%20Edition.pdf.

[5] Y. Fujita, M. Izutsu and S. Hatakeyama, "Swing-up and stabilization control of twin furuta pendulums by energy control," *IECON 2014 - 40th Annual Conference of the IEEE Industrial Electronics Society*, pp. 2878-2883, 2014, https://doi.org/10.1109/IECON.2014.7048917.

[6] I. B. Chawla, "Real-time stabilization control of a rotary double inverted pendulum system," *Arabian Journal for Science and Engineering*, vol. 46, pp. 8239-8251, 2021, https://doi.org/10.1007/s13369-020-05161-7.

[7] D. Zhang, S. Cong, Z. Li and Z. Qin, "The study of swing up and balance control for rotary parallel inverted-pendulum," *Proceedings of the 4th World Congress on Intelligent Control and Automation (Cat. No.02EX527)*, vol. 3, pp. 2370-2374, 2002, https://doi.org/10.1109/WCICA.2002.1021515.

[8] A. Rai, B. Bhushan and B. Jaint, "Stabilization and Performance Analysis of Double Link-Rotary Inverted Pendulum Using LQR-I Controller," *2025 3rd IEEE International Conference on Industrial Electronics: Developments & Applications (ICIDeA)*, pp. 1-5, 2025, https://doi.org/10.1109/ICIDeA64800.2025.10962818.

[9] R. Eini and S. Abdelwahed, "Indirect Adaptive fuzzy Controller Design for a Rotational Inverted Pendulum," *2018 Annual American Control Conference (ACC)*, pp. 1677-1682, 2018, https://doi.org/10.23919/ACC.2018.8431796.

[10] K. C. Desouza, G. S. Dawson, D. Chenok, "Designing, developing, and deploying artificial intelligence systems: Lessons from and for the public sector," *Business Horizons*, vol. 63, no. 2, pp. 205-213, 2020, https://doi.org/10.1016/j.bushor.2019.11.004.

[11] M. Vo *et al*, "Linear Control Schemes for Rotary Double Inverted Pendulum," *Robotica & Managmenet*, vol. 28, no. 1, pp. 59-67, 2023, https://doi.org/10.24193/rm.2023.1.8.

[12] A. K. Patra, A. Nanda, A. Gantayet, S. K. Kar, A. K. Mishra and D. K. Subudhi, "Balancing and Optimal Control of Inverted Pendulum Using Adaptive Controller," *2025 International Conference in Advances in Power, Signal, and Information Technology (APSIT)*, pp. 1-5, 2025, https://doi.org/10.1109/APSIT63993.2025.11086128.

[13] F. L. Lewis, D. L. Vrabie, V. L. Syrmos, "Optimal control," *John Wiley & Sons*, 1986, https://doi.org/10.1002/9781118122631.

[14] J. J. Grefenstette, "Optimization of Control Parameters for Genetic Algorithms," *IEEE Transactions on Systems, Man, and Cybernetics*, vol. 16, no. 1, pp. 122-128, 1986, https://doi.org/10.1109/TSMC.1986.289288.

[15] P. Kumar, O. N. Mehrotra, J. Mahto, "Tuning of PID controller of inverted pendulum using genetic algorithm," *IJRET: International Journal of Research in Engineering and Technology*, vol. 1, no. 3, pp. 359-364, 2012, https://doi.org/10.15623/ijret.2012.0103029.

[16] H. D. Khoi, "Using LQR algorithm to control circular two stage parallel inverted pendulum system," *Bulletin of College of Engineering National Ilan University*, 2013, https://lic2.niu.edu.tw/ezfiles/10/1010/img/6/Eng2013n9P101-114.pdf.

[17] C. Wongsathan and C. Sirima, "Application of GA to design LQR controller for an Inverted Pendulum System," *2008 IEEE International Conference on Robotics and Biomimetics*, pp. 951-954, 2009, https://doi.org/10.1109/ROBIO.2009.4913127.

[18] J. Kennedy and R. Eberhart, "Particle swarm optimization," *Proceedings of ICNN'95 - International Conference on Neural Networks*, vol. 4, pp. 1942-1948, 1995, https://doi.org/10.1109/ICNN.1995.488968.

[19] C.-H. Nguyen *et al*, "ANFIS-based LQR Control for Rotary Double Parallel Inverted Pendulum," *Journal of Fuzzy Systems and Control*, vol. 2, no. 2, pp. 109-116, 2024, https://doi.org/10.59247/jfsc.v2i2.214.

[20] T.-B. Dang *et al*, "PID Control for Cart and Pole system: Simulation and Experiment," *Journal of Fuzzy Systems and Control*, vol. 2, no. 1, pp. 29-35, 2024, https://doi.org/10.59247/jfsc.v2i1.165.

[21] N.-C. Tran *et al*, "LQR Control for Experimental Double Rotary Inverted Pendulum," *Journal of Fuzzy Systems and Control*, vol. 2, no. 2, pp. 104-108, 2024, https://doi.org/10.59247/jfsc.v2i2.212.

[22] Fahmizal, Geonoky, and H. Maghfiroh, "Rotary Inverted Pendulum Control with Pole Placement," *Journal of Fuzzy Systems and Control*, vol. 1, no. 3, pp. 90-96, 2023, https://doi.org/10.59247/jfsc.v1i3.152.

[23] I. M. Mehedi, U. M. Al-Saggaf, R. Mansouri, M. Bettayeb, "Stabilization of a double inverted rotary pendulum through fractional order integral control scheme," *International Journal of Advanced Robotic Systems*, vol. 16, no. 4, 2019, https://doi.org/10.1177/1729881419846741.

[24] S. D. Sanjeewa, M. Parnichkun, "Control of rotary double inverted pendulum system using mixed sensitivity H∞ controller," *International Journal of Advanced Robotic Systems*, vol. 16, no. 2, 2019, https://doi.org/10.1177/1729881419833273.

[25] Y.-F. Chen and A.-C. Huang, "Adaptive control of rotary inverted pendulum system with time-varying uncertainties," *Nonlinear Dynamics*, vol. 76, pp. 237-249, 2014, https://doi.org/10.1007/s11071-013-1112-4.

[26] D.-A.-Q. Nguyen *et al*, "Adaptive Evaluation of LQR Control using Particle Swarm Optimization for Pendubot," *Journal of Fuzzy Systems and Control*, vol. 2, no. 2, pp. 99-103, 2024, https://doi.org/10.59247/jfsc.v2i2.203.

[27] T.-T.-D. Le *et al*, "A Comparison of Genetic Algorithms in Optimizing Controllers for Inverted Pendulum," *Robotica & Management*, vol. 28, no. 2, pp. 21-27, 2023, https://doi.org/10.24193/rm.2023.2.4.

[28] B. Panjwani, V. Kumar, J. Yadav, V. Mohan, "Optimum LQR Controller for Inverted Pendulum Using Whale Optimization Algorithm," *Signals, Machines and Automation*, pp. 307-315, 2023, https://doi.org/10.1007/978-981-99-0969-8_31.

[29] S. Karthick, J. Jerome, E. V. Kumar, G. Raaja, "APSO Based Weighting Matrices Selection of LQR Applied to Tracking Control of SIMO System," *Proceedings of 3rd International Conference on Advanced Computing, Networking and Informatics*, pp. 11-20, 2015, https://doi.org/10.1007/978-81-322-2538-6_2.

[30] M.-D. Tran *et al*, "Analysis of Linear and Intelligent Control for Balancing Pendubot: LQR vs. Fuzzy," *Journal of Fuzzy Systems and Control*, vol. 2, no. 2, pp. 93-98, 2025, https://doi.org/10.59247/jfsc.v3i1.272.