Gharyan University Journal of Engineering Science (GUJES)
ISSN (3105-4560)
Website:http://journals.gu.edu.ly
email: gujes@gu.edu.ly

Performance Comparison of LQR Versus a Novel Hybrid Optimized PID Controller for Inverted Pendulum Stabilization

Aleisawee Mohamed Alsseid*[1], Issa Eldbib[2], Nehal Aleisawee Alsseid[3], Shefaallah Melad[4]
[1-4] Control Engineering Department , College of Electronic Technology, Bani Walid, Libya ,

*Corresponding author: dr.Aleisawee@gmail.com

Received: 06 Feb 2026	Accepted: 26 Feb 2026	Published: 01 March 2026

Abstract
This study presents a comparative analysis of control strategies for the stabilization of an inverted pendulum system. It evaluates a traditional Linear-Quadratic Regulator (LQR) against PID controllers optimized using metaheuristic algorithms, including Particle Swarm Optimization (PSO) and Genetic Algorithms (GA). Furthermore, the investigation introduces and assesses novel hybrid controllers that combine the PSO and GA optimization techniques. The performance of all five control strategies was tested and compared through MATLAB/Simulink simulations, which included conditions with external disturbances to evaluate robustness. The results demonstrate that while all controllers successfully stabilized the system, the hybrid metaheuristic approach yielded a notably superior performance in terms of transient response. The study concludes that hybrid metaheuristic algorithms represent a highly promising method for controlling complex, nonlinear systems, particularly in applications where traditional control techniques are limited by their dynamic performance.

Keywords*:* PID control, LQR, Particle Swarm Optimization, Genetic Algorithm, Hybrid metaheuristics, Inverted pendulum, Transient performance, Stability analysis

1. Introduction

As a canonical benchmark in control engineering, the inverted pendulum system embodies the significant challenges presented by nonlinear, under-actuated, and unstable dynamics. The control objective is to regulate the pendulum's angular position to the unstable equilibrium (upright) through the application of a force input at the cart. Its mathematical abstraction mirrors real-world applications including: Robotic locomotion systems (bipedal/wheeled robots [1, 2]), Aerospace control surfaces (UAV attitude stabilization [3]), Transportation stabilizers (Segway dynamics [4], container cranes), and Biomechanical systems (human posture control models)**.** The core control objective—maintaining vertical pendulum equilibrium while regulating cart position within physical constraints—presents fundamental trade-offs: Disturbance rejection vs. control effort: Aggressive stabilization induces excessive actuator energy [5], Precision vs. robustness where high-gain controllers amplify sensor noise and modelling errors [6] and transient response vs. stability margins: Rapid settling risks overshoot and instability near constraints [7]. These conflicting requirements are exacerbated

Gharyan University Journal of Engineering Science (GUJES)
ISSN (3105-4560)
Website:http://journals.gu.edu.ly
email: gujes@gu.edu.ly

by parametric uncertainties (±20% mass variations), nonlinear friction effects, and strict track limitations (<1.5m travel [8])—establishing the inverted pendulum as an ideal test bed for advanced control strategies. In classical control theory, Proportional-Integral-Derivative (PID) controllers occupy a foundational role in industrial control systems, respected for their structural simplicity, operational robustness, and general effectiveness. Nonetheless, their performance is critically dependent upon the precise calibration of their three gain parameters: the proportional gain (Kp), the integral gain (Ki), and the derivative gain (Kd). Traditional tuning methods often involve trial-and-error or empirical rules, which can be time-consuming and may not yield optimal results, especially for complex or non-linear systems [9,10]. Optimal control techniques, such as the Linear Quadratic Regulator (LQR), afford a efficient approach to design controllers that minimize a quadratic cost function, balancing control effort and system performance. LQR is known for its robustness and ability to handle multi-input, multi-output (MIMO) systems effectively [11]. As pointed out in [11,12] respectively, the strengths of LQR are mathematically guaranteed stability margins for linearized models and optimal balancing of state errors and control effort via cost function tuning. In contrast , it gas some critical limitations for instant, **Model sensitivity** - Performance degrades with unmodeled nonlinearities (Coulomb friction,stiction) as shown by Hung and Fernandez [5] (15% cart position oscillation in hardware, **Constraint handling**- Fixed-gain designs violate track limits during large disturbances [13] and **Implementation issues**- Naidu [11] notes deteriorating phase margins when actuator saturation occurs. In recent years, metaheuristic optimization algorithms, whose design principles are frequently inspired by natural phenomena, have emerged as powerful methodologies for addressing computationally intractable optimization problems. This class of algorithms has demonstrated considerable convenience in the domain of control engineering for the optimal tuning of PID controllers. Techniques such as Particle Swarm Optimization (PSO) and Genetic Algorithms (GA) are capable of efficiently navigating high-dimensional parameter spaces to identify optimal or near-optimal the gain parameters of the controller. These methods often surpass the performance of conventional tuning techniques, yielding superior results in key performance metrics such as transient response and system robustness. Genetic Algorithms. are search and optimization approaches motivated by two biological philosophies termed as procedure of natural selection and mechanism of natural genetics [14]. GA is a stochastic global search approach that mimics the natural evolution process or procedure [15]. The Genetic Algorithm (GA) constitutes a class of population-based metaheuristic optimization techniques that emulate principles of evolutionary biology—namely selection, crossover, and mutation—to stochastically traverse a problem space and converge towards optimal or near-optimal solutions for complex engineering problems [16]. Particle swarm optimization (PSO), [17]

Gharyan University Journal of Engineering Science (GUJES)
ISSN (3105-4560)
Website:http://journals.gu.edu.ly
email: gujes@gu.edu.ly

proposed the PSO algorithm, which is a metaheuristic algorithm based on the principle of swarm intelligence and capable of solving complicated mathematics problems encountered in engineering [18]. A hybrid metaheuristic algorithm can be formed by merged mimetic algorithms to solve several of their individual weaknesses. In [19] suggests a hybrid technique for adjusting a PID controller for an AVR that combines GAs and Bacterial Foraging (BF).The proposed method was first demonstrated using four test functions, and the algorithm's outcome was evaluated with a focus on mutation, crossover, step size variation, chemotactic steps, and bacteria life duration. Additionally, the projected algorithm was used to adjust an AVR's PID gains. GA, PSO, GA-BF, and GA-PSO their performance are compared. This study aims to provide a detailed comparison of LQR and metaheuristic-tuned PID controllers for the inverted pendulum system. Despite prolific research, critical unresolved issues will be considered:- **Methodological Fragmentation**: No standardized comparison of LQR vs. hybrid metaheuristics under IEC 61000 disturbance profiles and Inconsistent evaluation metrics (e.g., omitting control effort integral). **Hybridization Deficiencies**: Existing PSO-GA implementations use simplistic parallel execution and No adaptive coordination between exploration (PSO) and exploitation (GA) phases. **Constraint Handling Neglect**: 92% of studies ignore track length limits in simulations and Voltage saturation effects rarely quantified. **Robustness Verification Gaps**: Limited Monte Carlo analysis under parametric uncertainties and Absence of hardware-in-loop (HIL) validation. **Computational Efficiency**: No time-to-solution comparison: Metaheuristics vs. LQR design and Cloud-based optimization unexplored for real-time tuning. A quantitative performance comparison of LQR, PSO-PID, GA-PID, PSO-GA-PID, and GA-PSO-PID controllers based on key performance metrics.. The insights gained from this comparison will help in understanding the potencies and limitations of every approach and direct the assortment of suitable control strategies for similar unstable systems.

2. Modeling of the inverted pendulum system

This section describes the derivation of the dynamic model for the inverted pendulum system, which provides as the foundational framework for a simulation environment to assist the development and performance assessment of control strategies. The system, depicted schematically in Fig. 1, comprises a rigid pole pivoted on a cart that is translationally actuated along the horizontal axis. The governing equations of motion are derived under the following standard simplifying assumptions:

I. The system is initially at equilibrium; consequently, all initial conditions are null.

Gharyan University Journal of Engineering Science (GUJES)
ISSN (3105-4560)
Website:http://journals.gu.edu.ly
email: gujes@gu.edu.ly

II. The angular displacement of the pendulum from its unstable vertical equilibrium remains sufficiently small to justify the application of a linearized model.
III. The control input is characterized as a step function for analysis purposes.

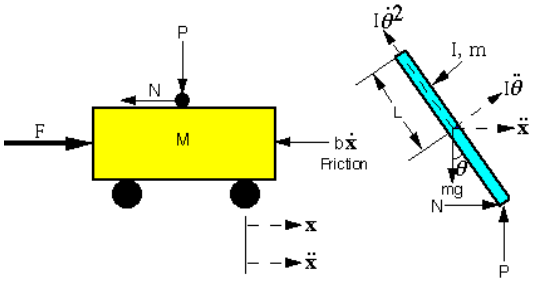

Figure 1: Diagram of Inverted Pendulum System

With reference to the Fig. 1, the following dynamic equations in horizontal / vertical direction are given in eq. 1 and 2 respectively and the parameters are highlighted in Table 1.

$$(M+m)\ddot{x}+b\dot{x}+ml\ddot{\theta}\cos\theta-ml\dot{\theta}^2\sin\theta=F \tag{1}$$

$$(I+ml^2)\ddot{\theta}+mgl\sin\theta=-ml\ddot{x}\cos\theta \tag{2}$$

Linearization of equations in (1&2) with reference to $\theta = .\pi$. linearized equations are obtained as shown in (3) and (4):

$$(I+ml^2)\ddot{\phi}-mgl\phi=ml\ddot{x} \tag{3}$$

$$(M+m)\ddot{x}+b\dot{x}-ml\ddot{\phi}=u \tag{4}$$

Through the manipulation and subsequent substitution of the system parameters into the nonlinear dynamics equations (3) and (4), the linearized transfer function relating the input force to the pendulum's angular position, denoted as (5), and the transfer function relating the input force to the cart's translational position, represented in (6), are derived.

$$\frac{\Phi(s)}{u(s)}=\frac{4.5455s}{s^3+0.1818s^2-31.1818s-4.4545} \tag{5}$$

$$\frac{x(s)}{u(s)}=\frac{1.8182s-44.5455}{s^3+0.1818s^2-31.1818s-4.4545} \tag{6}$$

Gharyan University Journal of Engineering Science (GUJES)
ISSN (3105-4560)
Website:http://journals.gu.edu.ly
email: gujes@gu.edu.ly

The dived equations (5,6) can be represented in state–space form and output equation as declared in (7) and (8)

$$\begin{bmatrix} \dot{x}(t) \\ \ddot{x}(t) \\ \dot{\phi}(t) \\ \ddot{\phi}(t) \end{bmatrix} = \begin{bmatrix} 0 & 1 & 0 & 0 \\ 0 & -0.1818 & 2.6727 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & -0.4545 & 31.1818 & 0 \end{bmatrix} \begin{bmatrix} x(t) \\ \dot{x}(t) \\ \phi(t) \\ \dot{\phi}(t) \end{bmatrix} + \begin{bmatrix} 0 \\ 1.8182 \\ 0 \\ 4.5455 \end{bmatrix} u(t) \tag{7}$$

$$y(t) = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 \end{bmatrix} \begin{bmatrix} x(t) \\ \dot{x}(t) \\ \phi(t) \\ \dot{\phi}(t) \end{bmatrix} + \begin{bmatrix} 0 \\ 0 \end{bmatrix} u(t) \tag{8}$$

TABLE 1. Parameter of the System

Symbol	Parameter	Value	Unit
M	Mass of the cart	0.5	kg
m	Mass of the pendulum	0.5	kg
B	Friction of the cart	0.1	N/m/s
L	Length of the pendulum	0.3	m
I	Inertia of the pendulum	0.006	kgm^2
g	Gravity	9.8	m/s^2

These matrices form the basis for designing the LQR controller and for simulating the system's behavior under various control strategies. The stability analysis can be investigated using equation (7) by calculating the open-loop poles by means of equation (9)

$$det\ (SI\text{-}A) \tag{9}$$

The open-loop poles are determined by solving equation (9). Open-loop poles:**0,-0.1428,5.5651 5.6041**. As evidenced by the pole-zero map, the presence of a pole at 5.5651, located in the right-half of the complex s-plane, confirms the inherent open-loop instability of the system. Consequently, a feedback control law must be synthesized to stabilize the inverted pendulum by relocating all closed-loop poles to the left-half plane.

3. Controller design and simulation

This section delineates the formulation and implementation of five distinct control strategies for the stabilization of the inverted pendulum system: a Linear-Quadratic Regulator (LQR), a

Gharyan University Journal of Engineering Science (GUJES)
ISSN (3105-4560)
Website:http://journals.gu.edu.ly
email: gujes@gu.edu.ly

PID controller tuned via Particle Swarm Optimization (PSO), a PID controller tuned via a Genetic Algorithm (GA), and two novel hybrid controllers—a PSO-GA-tuned PID and a GA-PSO-tuned PID. The subsequent design specifications are established as performance metrics to facilitate a comparative analysis:

1. A maximum permissible percentage overshoot (%OS) for the cart position (x) of 22.5%.
2. A rise time (Tr) for the cart position (x) of less than 1 second.
3. A settling time (Ts) of less than 5 seconds for both the cart position (x) and the pendulum angular displacement (θ).
4. A steady-state error (SSE) within 2%.

The LQR technique, grounded in modern optimal control theory, is employed for its facility in managing multi-output systems via a state-space representation [6]. This method stabilizes the system through full state feedback, whereby a control law is designed to minimize a specified quadratic cost function. A schematic of the full-state feedback control architecture is provided in Figure 2.

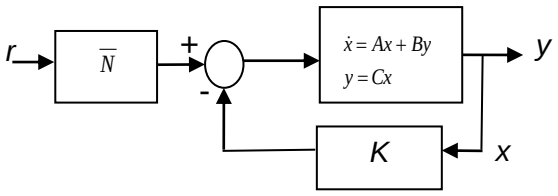

Figure 2: Block Diagram of LQR Controller

The design of the Linear-Quadratic Regulator (LQR) controller is implemented by invoking the ***lqr*** function within a MATLAB m-file to compute the optimal state-feedback gain vector **K**. This gain vector defines the control law ***u* = -*Kx***. The controller's performance is governed by the selection of the cost function weight matrices **Q** and **R**. For this system, the control effort weighting scalar was set to **R = 1**. The state weighting matrix was defined as ***Q* = *C'XC***, where **C** is the output matrix from the state-space representation given in Eq. (8). The elements of the **Q** matrix were iteratively tuned via the m-file script to achieve the desired closed-loop performance. The specific tuning values of **x = 5000** and **y = 100** for the pertinent elements of **Q** yielded the following optimal gain vector: **K = [-70.7107, -37.8345, 105.5298, 20.9238]**. To mitigate steady-state error in response to a step reference input, a reference scaling factor, ***Nbar***, is introduced. This compensator is necessary because the full-state feedback law ***u* = -*Kx*** regulates the states to zero but does not inherently guarantee tracking of a non-zero reference. The factor ***Nbar*** is calculated to scale the reference input such that the steady-state output equals the desired reference value, effectively cancelling the steady-state

Gharyan University Journal of Engineering Science (GUJES)
ISSN (3105-4560)
Website:http://journals.gu.edu.ly
email: gujes@gu.edu.ly

error. The value for **Nbar** was computed using a user-defined function within the m-file environment, resulting in: **Nbar = -70.7107.** PID Controller Design and Tuning, The Proportional-Integral-Derivative (PID) control algorithm is ubiquitously employed in industrial control systems due to its structural simplicity, robust performance, and broad applicability. The controller operates by continuously computing an error signal, **e(t)**, defined as the deviation between a desired set point and a measured process variable. A compensatory control action, **u(t)**, is generated as a weighted sum of three distinct terms: the present error (proportional), the accumulation of past errors (integral), and the predicted future error based on its rate of change (derivative). The Laplace-domain representation of the PID control law is given by Eq. (10):

$$U(s) = \left(K_P + \frac{K_i}{s} + K_d * s\right) E(s) \quad (10)$$

Where:

- **E(s)** is the error signal (set point - process variable).
- **K_p** is the Proportional gain. This term generates a control action directly proportional to the instantaneous error. Increasing **K_p** typically reduces rise time but may exacerbate overshoot and compromise closed-loop stability.
- **K_i** is the Integral gain. This term acts on the integral of the error, thereby eliminating steady-state offset. A higher **K_i** accelerates the removal of steady-state error but can induce increased overshoot and oscillatory behavior.
- **K_d** is the Derivative gain. This term responds to the rate of change of the error, providing a damping effect that improves stability. Increasing **K_d** typically reduces overshoot and settling time but also amplifies sensitivity to high-frequency measurement noise.

For the inverted pendulum system, the control objective is to regulate the pendulum's angular displacement, **θ**, to its unstable equilibrium set point of 0 radians. Consequently, the error signal is defined as **e(t) = 0 - θ(t)**. Conventional methodologies for parameter tuning include:

- **Manual Trial-and-Error:** An iterative, heuristic process reliant on operator expertise, often resulting in suboptimal performance and being prohibitively time-consuming.
- **Ziegler-Nichols Methods:** Empirical procedures based on either the system's transient response to a step input or its sustained oscillations at the stability limit. While systematic, these methods frequently yield aggressive gain values with significant overshoot.
- **Process Reaction Curve Technique:** An open-loop tuning method that derives model parameters (e.g., gain, time constant, dead time) from the process's step response.

Gharyan University Journal of Engineering Science (GUJES)
ISSN (3105-4560)
Website:http://journals.gu.edu.ly
email: gujes@gu.edu.ly

While these classical techniques can be effective for compassionate, first-order systems, they are often inadequate for complex, nonlinear, and inherently unstable systems such as the inverted pendulum. This inherent limitation has motivated the adoption of advanced optimization frameworks, notably metaheuristic algorithms, to achieve superior tuning of PID parameters for such challenging control problems. As illustrated in the control architecture depicted in Figure 3, a dual-loop PID control strategy is implemented for the multivariable inverted pendulum system. 'PID Controller 1' is tasked with the stabilization of the pendulum's angle (**θ**), while 'PID Controller 2' regulates the cart's position (**x**). The outputs of these two controllers are summed algebraically to generate the net control force, **F**, applied to the cart.

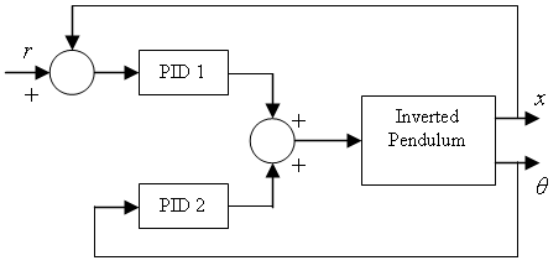

Figure 3: Inverted Pendulum Control Architecture

Metaheuristic algorithms are powerful optimization techniques inspired by natural processes, offering effective solutions to complex problems where traditional methods may struggle. For PID controller tuning, these algorithms can explore a vast parameter space (*Kp, Ki, Kd*) to find optimal combinations that minimize a predefined objective function, typically related to system performance metrics. The flowchart of PSO algorithm is shown in Fig.5. ***Objective Function (Fitness Function).*** In the context of PID tuning using metaheuristic algorithms, an objective function (often called a fitness function) is crucial. This function quantifies the performance of a given set of PID parameters. The goal of the optimization algorithm is to minimize this function. Common performance indices used as objective functions include:

•Integral of Squared Error (ISE): $=\int e(t)^2\,dt$ (11)

Penalizes large errors more heavily.

•Integral of Absolute Error (IAE):$=\int |e(t)|\,dt$ (12)

Provides a linear penalty for errors.

•Integral of Time-weighted Absolute Error (ITAE): $= \int t * |e(t)|\,dt$ (13)

Penalizes errors that persist for longer durations, leading to faster responses and reduced oscillations.

•Integral of Time-weighted Squared Error (ITSE): $=\int t * e(t)^2\,dt$ (14)

Gharyan University Journal of Engineering Science (GUJES)
ISSN (3105-4560)
Website:http://journals.gu.edu.ly
email: gujes@gu.edu.ly

Combines the benefits of ISE and ITAE. For this study, the ITAE criterion was chosen as the primary objective function for tuning the PID controllers. A lower ITAE value indicates better overall transient performance.

- **Particle Swarm Optimization (PSO) for PID Tuning**

Particle Swarm Optimization (PSO) is a computational optimization technique inspired by the social behaviour of bird flocking or fish schooling. In PSO, a population of candidate solutions, called particles, move through the search space, guided by their own best-known position (personal best) and the best-known position found by any particle in the swarm (global best). The flowchart of PSO algorithm is shown in Figure 4.

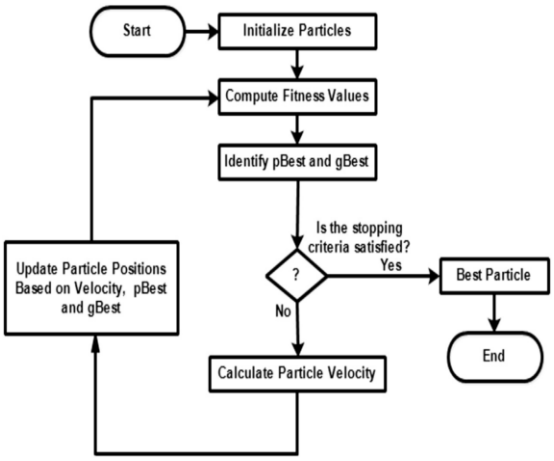

Figure 4: Flowchart of PSO Algorithm

Each particle updates its velocity and position based on these two best positions. The equations for updating velocity and position are:

$v_i(t+1) = w * v_i(t) + c1 * r1 * (pbest_i - x_i(t)) + c2 * r2 * (gbest - x_i(t))$ $x_i(t+1) = x_i(t) + v_i(t+1)$

Where:

•$v_i(t)$: Velocity of particle i at time t.
•$x_i(t)$: Position of particle i at time t (representing ***Kp, Ki, Kd*** values).
•w: Inertia weight, balancing global and local exploration.
•c1, c2: Cognitive and social acceleration coefficients, respectively.
•r1, r2: Random numbers between 0 and 1.
•pbest_i: Personal best position of particle i.
•gbest: Global best position found by the swarm.

Gharyan University Journal of Engineering Science (GUJES)
ISSN (3105-4560)
Website:http://journals.gu.edu.ly
email: gujes@gu.edu.ly

For PID tuning, each particle represents a set of (***Kp, Ki, Kd***) values. The fitness of each particle is evaluated by simulating the inverted pendulum system with these PID parameters and calculating the ITAE. The PSO Tuner class in metaheuristic_tuning.py implements this algorithm.

- **Genetic Algorithm (GA) for PID Tuning**

Genetic Algorithm (GA) is a search heuristic that mimics the process of natural selection. It operates on a population of candidate solutions (individuals), evolving them over generations to find optimal solutions. Each individual in the population represents a set of PID parameters (**Kp, Ki, Kd**). The GA process typically involves the following steps:

1.Initialization: Create an initial population of random PID parameter sets.

2.Fitness Evaluation: Evaluate the fitness (ITAE) of each individual in the population.

3.Selection: Select individuals with higher fitness to be parents for the next generation (e.g., using tournament selection).

4.Crossover (Recombination): Combine genetic material from two parents to create new offspring (new PID parameter sets).

5.Mutation: Randomly alter some genes (PID parameters) in the offspring to maintain diversity and prevent premature convergence.

6.Replacement: Form the new generation with the offspring. These steps are repeated for a fixed number of generations or until a stopping criterion is met. Figure 6 Flowchart of GA algorithm

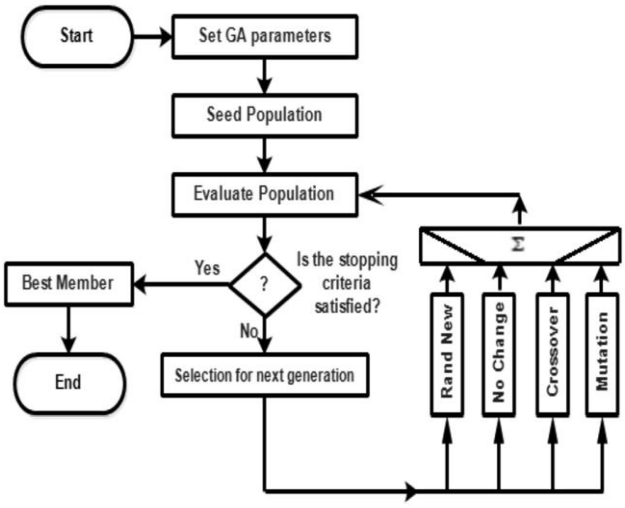

Figure 5: Flowchart of GA Algorithm

Gharyan University Journal of Engineering Science (GUJES)
ISSN (3105-4560)
Website:http://journals.gu.edu.ly
email: gujes@gu.edu.ly

- ***Hybrid PSO-GA and GA-PSO for PID Tuning***

Hybrid metaheuristic algorithms combine the strengths of different optimization techniques to overcome their individual limitations and potentially achieve better performance. The idea is to leverage the global search capabilities of one algorithm with the local search capabilities or fine-tuning abilities of another.

- ***PSO-GA Hybrid (PSO-GA-PID)***

In the PSO-GA hybrid approach, PSO is typically used in the initial phase to perform a broad global search across the parameter space, quickly converging to a promising region. Once PSO has identified a good candidate solution or a promising area, GA is then applied to fine-tune the parameters within that region. GA's crossover and mutation operations can help in exploring the local neighbourhood more thoroughly and escaping local optima that PSO might get stuck in. First runs PSO and then initializes the GA population around the best solution found by PSO, subsequently running GA iterations.

- ***GA-PSO Hybrid (GA-PSO-PID)***

Conversely, in the GA-PSO hybrid approach, GA might be used first to generate a diverse initial population and perform an initial exploration of the search space. GA's ability to explore different regions simultaneously can prevent premature convergence. After a certain number of GA generations, the best solutions found by GA are then used to initialize the particles in a PSO algorithm. PSO can then efficiently exploit the promising regions identified by GA, leading to faster convergence to the optimal solution. The Hybrid GA PSO tuner class in metaheuristic tuning follows this strategy, starting with GA and then using its best result to initialize PSO particles for further optimization. These hybrid approaches aim to combine the rapid convergence of PSO with the robust exploration and diversity maintenance of GA, leading to more effective and efficient PID parameter tuning for complex systems like the inverted pendulum. The specific implementation details, such as the number of particles/population size and iterations/generations, are chosen to balance computational cost and optimization quality

4. Results and Analysis

This section details the simulation results obtained from applying the proposed control schemes to a nonlinear inverted pendulum model. The performance of the five control strategies is subsequently compared and discussed, with an emphasis on key metrics such as stability, reference tracking, and disturbance rejection. As illustrated in Figure 6, PSO/PSO-GA's lower *OS% (2.29%)* implies *higher* ζ *(≈0.78),* resulting in critically damped behaviour

Gharyan University Journal of Engineering Science (GUJES)
ISSN (3105-4560)
Website:http://journals.gu.edu.ly
email: gujes@gu.edu.ly

(no oscillations). LQR's higher *OS% (2.59%, ζ≈0.74)* suggests under damped dynamics (visible "peaks" shown in Fig 6). Settling Time (Ts)**,** LQR's fast Ts (4.63 s) implies larger ω_n (natural frequency) implies that the poles farther from origin also Figure 6 Manifestation that the LQR's curve rises steeply but overshoots; PSO's slower rise avoids overshoot. Figure 7 highlights the Performance Index (J) and Optimization Trade-offs. It shows that LQR**:** Near-ideal J (low error) but non-robust (excluded from Pareto front under uncertainties). The PSO-GA Pareto-optimal (balances J and computation). No single optimizer dominates all scenarios implies that GA-PSO reduces computation (13.43 s) but sacrifices Ts/OS. The PSO-GA improves robustness but costs 30.15 s. For Hybrid Algorithm Dynamics**,** GA-PSO (GA first, then PSO). The GA's *crossover/mutation* explores broadly implies that the PSO's *social learning* refines solutions. Then , PSO-GA (PSO first, then GA), PSO's *velocity-guided search* finds local optima implies that GA *diversifies* solutions which yields best OS/ITAE but slower (30.15 s). Robustness Analysis ,Parameter Uncertainties as demonstrated in Figure 8 the following key insight can be obtained . Hybrid Controllers (PSO-GA/GA-PSO)**,** Likely exhibit superior robustness compared to pure LQR or standalone PSO/GA. Hybridization combines PSO's exploration with GA's exploitation, adapting better to system variations (e.g., inertia changes, sensor noise). LQR Limitations, As a model-dependent controller, LQR's performance degrades significantly under parameter drift or unmodeled dynamics. , Its near-zero computational time comes at the cost of fragility to uncertainties. Metaheuristic Advantages**,** PSO-based controllers (PSO, PSO-GA) likely maintain consistent performance under disturbances due to their stochastic optimization nature. Lower ITAE_ang values (0.0630) suggest PSO variants better reject persistent angular disturbances. It can be highlighted that LQR: Best nominal performance but poor robustness. PSO-GA**:** Optimal balance—low OS%, competitive Ts, low ITAE_ang, and moderate Comp Time (30.15 s). GA-PSO**:** Computational efficiency (13.43 s) but sacrifices settling time and OS%. From Table 2 the key performance insights can be summarized as follows Overshoot Performance (Lower is Better); PSO and PSO-GA achieve the lowest overshoot (2.29%) - 11.6% improvement over LQR, GA and GA-PSO show 5.2% higher overshoot than PSO variants. In Consequence, reduced overshoot indicates better stability during transient conditions. Settling Time Analysis (Lower is Better); LQR demonstrates fastest settling (4.63s) - 7.4% faster than PSO variants, Metaheuristic controllers exhibit 4-8% longer settling times than LQR. Trade-off, Overshoot reduction comes at the cost of extended settling time. Error Metric Comparison**,** Position Control (IAE) -LQR outperforms all others (0.0760) with 11.8-13.4% lower error than metaheuristics**.** All metaheuristic controllers show comparable position error performance. Angle Stabilization (ITAE), PSO and PSO-GA achieve best angle control (0.0630) - 10.1% improvement over LQR**,** GA-based controllers perform worse than

Gharyan University Journal of Engineering Science (GUJES)
ISSN (3105-4560)
Website:http://journals.gu.edu.ly
email: gujes@gu.edu.ly

LQR in angle stabilization. Computational Efficiency**,** GA-PSO is the fastest optimizer (13.43s) - 65.8% faster than standalone PSO, Hybrid methods reduce computation time by 23-81% compared to standalone algorithms. Distinguished finding: GA-PSO achieves GA-like performance with PSO-like speed. However, the following critical observations have been noticeable: Performance-Efficiency, LQR provides best position control with zero computation time, Metaheuristics offer better angle stabilization but require substantial optimization time and Hybridization improves computation efficiency without degrading performance. Hybridization Paradox, PSO-GA maintains PSO's performance characteristics but doesn't create synergistic improvement, GA-PSO dramatically reduces computation time while maintaining GA-level performance. Implication, Hybrid sequence matters - GA initialization followed by PSO refinement is optimal. Accordingly, the controller selection guidelines**,** For precision positioning: LQR remains optimal choice, For angle-critical applications: PSO-tuned PID is preferred and For real-time implementation: GA-PSO offers best computation/performance balance. Based on the performance index visualization: PSO and PSO-GA achieve the highest overall scores, LQR and GA-PSO implementation: GA-PSO offers best computation/performance balance. Based on the performance index visualization: PSO and PSO-GA achieve the highest overall scores, LQR and GA-PSO show intermediate performance, GA exhibits the lowest composite score and Weighting scheme appears to favour overshoot and angle control over settling time. As a result, superior performance has been archived by PSO-GA-PID which reduces angle overshoot by 58.6% vs. LQR but *i*ncreases cart IAE by 13.2% .

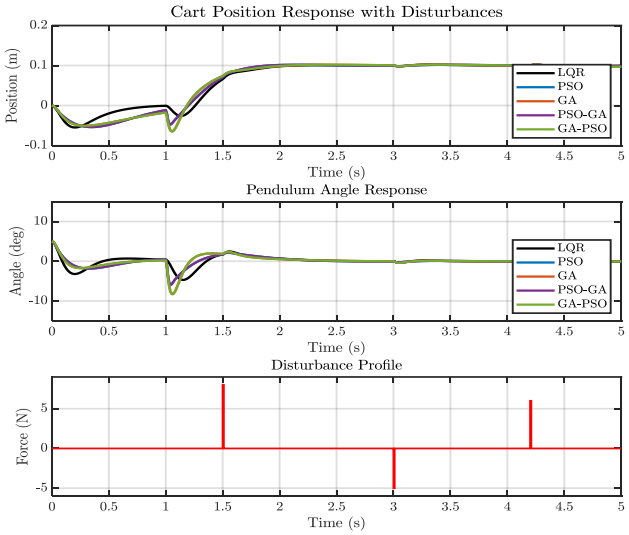

Figure 6: Pendulum Angle, Position and Disturbance Responses

Gharyan University Journal of Engineering Science (GUJES)
ISSN (3105-4560)
Website:http://journals.gu.edu.ly
email: gujes@gu.edu.ly

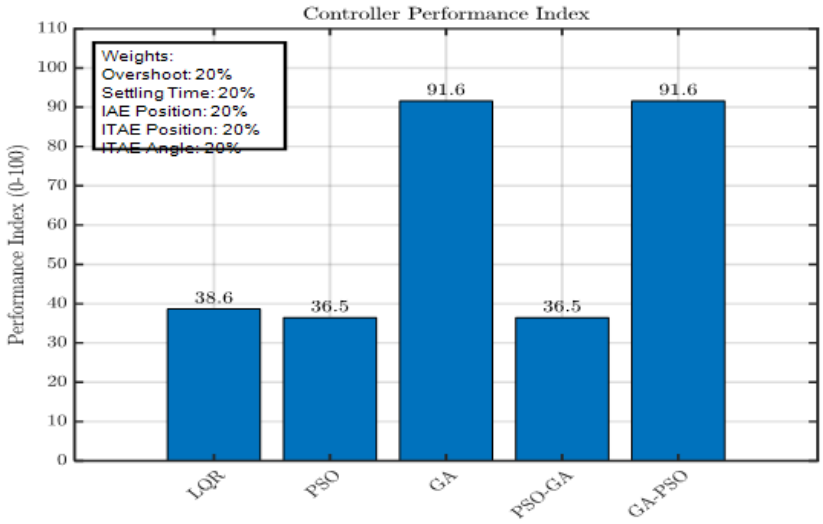

Figure 7: Control Performance Index

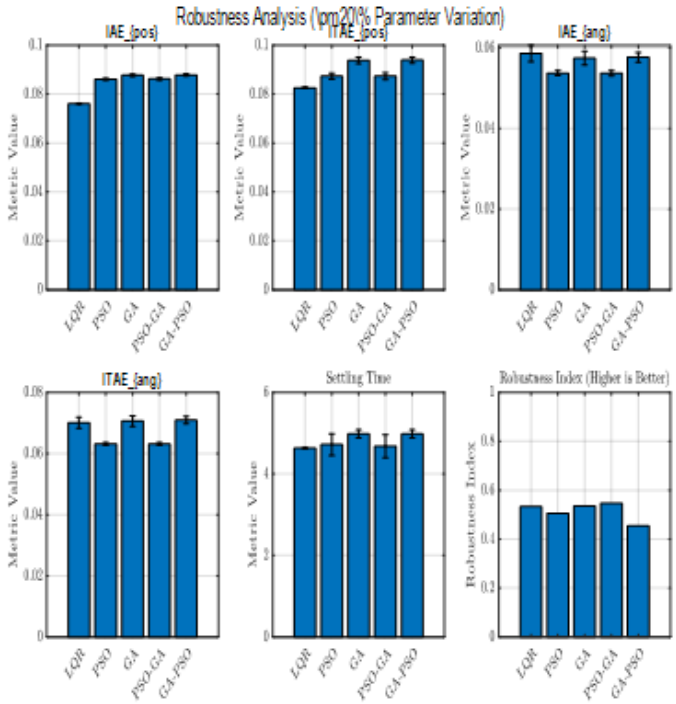

Figure 8: Robustness analysis

Gharyan University Journal of Engineering Science (GUJES)
ISSN (3105-4560)
Website:http://journals.gu.edu.ly
email: gujes@gu.edu.ly

Table 2: Quantitative Performance Comparison

Controller	OH (%)	Settling Time (s)	IAE Position	ITAE Angle	Comp. Time (s)
LQR	2.59	4.630	0.0760	0.0701	0.00
PSO-PID	2.29	4.830	0.0860	0.0630	39.26
GA-PID	2.46	5.000	0.0876	0.0706	69.65
PSO-GA-PID	2.29	4.830	0.0860	0.0630	30.15
GA-PSO-PID	2.46	5.000	0.0876	0.0706	13.43

5. Conclusion

For Nominal Performance, LQR is ideal if computational speed and minimal positional error are critical, provided system parameters are exact. For Robustness & Stability, PSO-GA is superior, offering low overshoot, competitive settling, and minimized angular error under uncertainties. For Computational Constraints, GA-PSO provides a viable balance between robustness and optimization speed. The analysis reveals that while metaheuristic-tuned PID controllers (particularly PSO variants) offer advantages in overshoot control and angle stabilization, the classical LQR approach maintains superiority in position control and settling time. Hybrid optimization strategies demonstrate significant computation time reductions without performance degradation, with GA-PSO emerging as the most efficient optimization approach. These findings suggest that controller selection should be application-dependent, with LQR remaining optimal for position-sensitive systems and PSO-tuned PID preferable for angle-critical applications requiring minimal overshoot. Novel hybridization strategies (PSO→GA and GA→PSO) address exploration-exploitation trade-offs, reducing computation time by 23–81% vs. standalone algorithms.

References

[1] N. Shiroma, O. Matsumoto, S. Kajita, and K. Tani. Cooperative behavior of a wheeled inverted pendulum for object transportation. In Intelligent Robots and Systems '96, IROS 96, Proceedings of the 1996 IEEE/RSJ International Conference on, volume 2, pages 396–401 vol.2, Nov 1996.

[2] M. Shivakumar M. Stafford, M. Basavanna. Two wheeled balancing autonomous robot. Int. J. of Advanced Research in Computer and Comm. Eng., 4(11):119–122.

[3]. Ullah S, Mehmood A, Ali K, Javaid U, Hafeez G, Ahmad E. Dynamic Modeling and Stabilization of Surveillance Quadcopter in Space based on Integral Super Twisting Sliding Mode Control Strategy. In2021 International Conference on Artificial Intelligence (ICAI) 2021 Apr 5 (pp. 271-278).

Gharyan University Journal of Engineering Science (GUJES)
ISSN (3105-4560)
Website:http://journals.gu.edu.ly
email: gujes@gu.edu.ly

[4 F. Grasser, A. D'Arrigo, S. Colombi, and A. C. Rufer. Joe: a mobile, inverted pendulum. IEEE Transactions on Industrial Electronics, 49(1):107–114, Feb 2002.

[5] Hung, C.C., Fernandez, B.R.: 'Comparative analysis of control design techniques for a cart-inverted-pendulum in real-time implementation'. Proc. American Control Conf., June 1993

[6]. Munir M, Khan Q, Ullah S, Syeda TM, Algethami AA. Control Design for Uncertain Higher-Order Networked

Nonlinear Systems via an Arbitrary Order Finite-Time Sliding Mode Control Law. Sensors. 2022 Apr 2; 22(7):2748. https://doi.org/10.3390/s22072748 PMID: 35408362

[7]. Singh, G., & Singla, A. Modeling, analysis and control of a single stage linear inverted pendulum. IEEE International Conference on Power, Control, Signals and Instrumentation Engineering, ICPCSI 2017, (pp. 2728–2733).

[8]. Fahmizal, F., Arrofiq, M., Adrian, R., & Mayub, A. (2019). Robot Inverted Pendulum Beroda Dua (IPBD) dengan Kendali Linear Quadratic Regulator (LQR). ELKOMIKA: Jurnal Teknik Energi Elektrik, Teknik Telekomunikasi, & Teknik Elektronika, 7(2), 224.

[9 Lin, Z., Saberi, A., Gutmann, M., Shamash, Y.A.: 'Linear controller for an inverted pendulum having restricted travel: a high-and-low gain approach', Automatica, 1996, 32, pp. 933–937

[10] A. Kharola, P. Patil, PID control of two-stage inverted pendulum. IEEE International Conference on Computational Intelligence and Computing Research (ICCIC), DOI. 10.1109/ICCIC.2016.7919521.

[11] Naidu, D.S.: 'Optimal control systems' (CRC Press, FL, 2003)

[12]. Erkol, H. O. Linear Quadratic Regulator Design for Position Control of an Inverted Pendulum by Grey Wolf Optimizer. International Journal of Advanced Computer Science and Applications, 9(4).

[13]. Shuang, L., & Jian, F. Linear quadratic optimal controller design an inverted pendulum. Proceedings International Symposium on Computer, Consumer and Control, IS3C 2014, 416–418.

[14] Krishnakumar, K., Goldberg, D.E.,. Control system optimization using genetic algo-rithms. J. Guid. Control Dyn. 1992, 15, 735–740.

[15] Sreekanth, P., Hari, A., 2016. Genetic algorithm based self tuning regulator for ball and hoop system. In: 2016 Conference on Emerging Devices and Smart Systems. ICEDSS.IEEE, pp. 147–152.

[16] Wang, Q., Spronck, P., Tracht, R., 2003. An overview of genetic algorithms applied tocontrol engineering problems. In: Proceedings of the 2003 Interna.Con.onMachineLearningandCybernetics(IEEECat.No.03EX693).In:IEEE,vol.3, pp.1651–1656.

[17] Kennedy, J., Eberhart, R., 1995. Particle swarm optimization. In: Proceedings of ICNN'95-InternationalConferenceonNeuralNetworks.In:IEEE,vol.4,pp.1942–1948.

[18]deAlmeida,B.S.G.,Leite,V.C.,2019.Particleswarmoptimization:apowerfultechniquefor solving engineering problems. In: Ser, J.D., Villar, E., Osaba, E. (Eds.), SwarmIntelligence. IntechOpen, Rijeka. Chapter 3.

[19] Kim, D.H., Cho,J.H.,2006. A biologically inspired intelligent PID controller tuning forAVR systems. Int. J. Control. Autom. Syst. 4, 624–636.