Grenze International Journal of Engineering and Technology, Jan Issue

Comparative Study of Metaheuristic Algorithms-based PID-Controller for Triple Link Inverted Pendulum

Disha Sharma[1] and Sheela Tiwari[2]
[1-2]Dr. B. R. Ambedkar National Institute of Technology Jalandhar, Dept. of Instrumentation & Control Engineering
Email: 1998.dishasharma@gmail.com, tiwaris@nitj.ac.in

***Abstract*—The paper focuses on the comparative study of Ant Colony Optimization (ACO), Grey Wolf Optimization (GWO) and Whale Optimization Algorithm (WOA) based PID controllers for stabilizing the linearized model of Triple Link Inverted Pendulum. The model is simulated in a MATLAB environment and the simulation results present a comparison on the basis of settling time, steady-state error, and peak overshoot.**

***Index Terms*— Ant Colony Optimization (ACO), Grey Wolf Optimization (GWO), Whale Optimization Algorithm (WOA), and Triple Link Inverted Pendulum (TLIP).**

I. INTRODUCTION

A classic problem in control theory and dynamics is the problem of the Inverted Pendulum which is used for testing various control algorithms. In the control process, the inverted pendulum can effectively be used for studies of key issues such as the quality of stability, robustness, mobility, and tracking and that is why it is the ideal model for testing various theories in control engineering [1]. The Triple Link Inverted Pendulum is similar in characteristics to the inverted pendulum i.e., it is highly non-linear, multivariable, and continuously tries to move towards the uncontrolled state. It is an open-loop unstable system in which the pendulum length is divided into three small links, hence the name Triple Link Inverted Pendulum (TLIP).

There are multiple solutions currently present to stabilize the Triple Link Inverted Pendulum. The system purported in [2] is considered complicated since instead of including a moving cart it hinges the first link to the ground while increasing the system's controllability by adding horizontal bars to the links. Engineering applications often focus on the use of simpler controller mechanisms [3]. Simple and good performance controller like Linear Quadratic Controller is explained in [4].

This paper focuses on the use of the simplest and easy-to-implement controller in the automation and control industry i.e., the PID controller for stabilizing the Triple Link Inverted Pendulum. Nowadays nature-inspired optimization algorithms like PSO, SDA etc. are increasingly being used to stabilize the TLIP system [5]. Therefore, this work envisages using some nature-inspired algorithms to tune the PID controllers and compare the performances of these controllers with the controller tuned using the conventional manual tuning method. The ACO, GWO, and WOA algorithms are used to find the PID parameters (gains) for the linearized simulated model of the Triple Link Inverted Pendulum. Section 2 discusses the mathematical model of TLIP. Section 3 mentions the algorithms used to tune the PID controller. Section 4 presents the simulation results obtained by conducting the experiment followed by the conclusion in section 5.

Grenze ID: 01.GIJET.9.1.526

II. MATHEMATICAL MODEL

The Lagrange method is used to describe the TLIP model mathematically which is established in [6]. The Lagrange method is considered since it does not require the knowledge of the forces instead it uses the equations of kinetic and potential energy to derive the set of differential equations. The cart TIPS considered here consists of a straight-line rail, a cart moving horizontally on the rail, a lower pendulum hinged on the cart, and a middle and an upper pendulum connected to the lower pendulum. In the same vertical plane with the rail, the lower pendulum link can rotate around the pivot, and the middle and the upper pendulum links can rotate around the linkage in the same vertical plane. This passage is on the stipulation that the turn angle of all the links is clockwise and the force moment is positive. The parameters for the TLIP are listed below,

u is external action(force applied to move the cart)

x is the displacement of the cart

θ_1, θ_2, and θ_3 are the angles of the lower, middle, and upper pendulum links respectively concerning the vertical line

m_0 is the mass of the cart

m_1, m_2, and m_3 are the masses of the lower, middle, and upper pendulum links respectively

f_0 is the friction factor of cart and track

J_1, J_2, and J_3 are the rotary inertia of lower, middle, and upper pendulum links respectively

Ks is the overall system's input conversion gain.

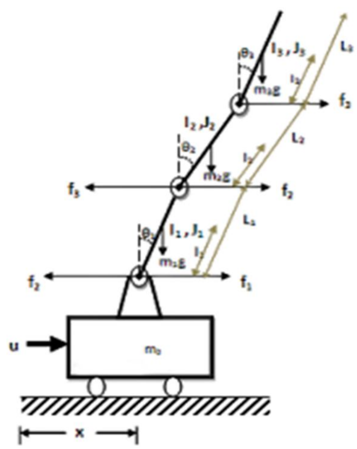

Fig.1 Cart Triple Link Inverted Pendulum Configuration

The state vector for the linearized model about the upright position is given as,

$$X = [x \quad \theta_1 \quad \theta_2 \quad \theta_3 \quad \dot{x} \quad \dot{\theta}_1 \quad \dot{\theta}_2 \quad \dot{\theta}_3]^T$$

The state-space equations are mentioned below,

$$\dot{X} = AX + BU$$

$$\dot{X} = \begin{vmatrix} 0 & I_{3\times4} \\ E^{-1}H & E^{-1}G \end{vmatrix} X + \begin{vmatrix} 0 \\ B^{-1}h_0 \end{vmatrix} U$$

$$Y = CX$$

$$\text{where, } E = \begin{bmatrix} a_0 & a_1 & a_2 & m_3 l_3 \\ a_1 & b_1 & a_2 L_1 & m_3 L_1 l_3 \\ a_2 & a_2 L_1 & b_2 & m_3 L_2 l_3 \\ m_3 l_3 & m_3 L_1 l_3 & m_3 L_2 l_3 & J_3 + m_3 l_3^2 \end{bmatrix}$$

$$H = \begin{bmatrix} 0 & 0 & 0 & 0 \\ 0 & a_1 g & 0 & 0 \\ 0 & 0 & a_2 g & 0 \\ 0 & 0 & 0 & m_3 l_3 g \end{bmatrix} \qquad G = \begin{bmatrix} -f_0 & 0 & 0 & 0 \\ 0 & -f_1 - f_2 & f_2 & 0 \\ 0 & f_2 & -f_2 - f_3 & f_3 \\ 0 & 0 & f_3 & -f_3 \end{bmatrix}$$

The coefficient matrices obtained are as follows,

$$A = \begin{bmatrix} 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \\ 0 & -12.4298 & -2.0824 & 2.2956 & -5.1127 & 0.0075 & 0.0024 & -0.0053 \\ 0 & 67.1071 & 65.2564 & -71.9704 & 14.0176 & 0.0039 & -0.1948 & 0.1659 \\ 0 & 144.5482 & -394.2536 & 272.1049 & 5.2021 & -0.4334 & 1.1287 & -0.7492 \\ 0 & -300.4564 & 512.8310 & -258.9198 & -10.8077 & 0.6476 & -1.3621 & 0.826 \end{bmatrix}$$

$$B = [0 \quad 0 \quad 0 \quad 0 \quad 3.6512 \quad -10.0125 \quad -3.7157 \quad 7.7196]^T$$

$$C = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \end{bmatrix} \qquad D = [\,0\,0\,0\,0\,]^T$$

III. PID CONTROLLER: AN OVERVIEW

A Proportional-Integral-Derivative (PID) controller is a generic control loop feedback mechanism. A PID controller is an amalgamation of three controllers, namely, a Proportional (P) controller, an Integral (I) controller, and a Derivative (D) controller, whose input is an error signal defined by the difference in Process Variable (PV) and Set Point (SP).

K_p = proportional gain , K_i = integral gain , K_d = derivative gain, e(t) = error = SP – PV

The controller output is given as, $u(t) = K_p e(t) + K_i \int_0^t e(t)dt + K_d \frac{de(t)}{dt}$

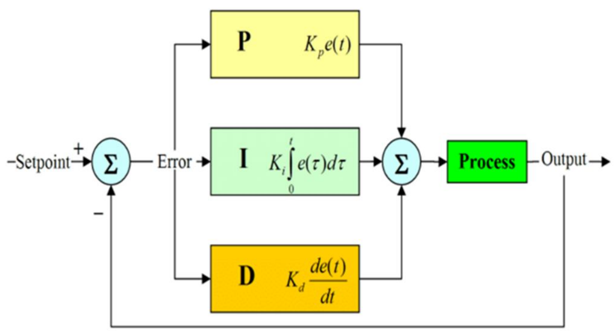

Fig. 2 Block diagram of a basic PID controller

The gain parameters i.e., K_p, K_i, and K_d are adjusted to tune the PID controller. The various ways to tune the controller include conventional methods like manual tuning, Ziegler-Nichols method, Cohen-Coon method, etc. and nature-inspired algorithms like ACO, WOA, and PSO can be used to tune PID controllers. Some of the tuning methods that are proposed in the work are listed below,

A. Manual Tuning

Manual tuning is the most basic tuning method. It is simple and gives reasonable performance for simple process loops. The steps to be followed during manual tuning are mentioned below,

- Set K_i and K_d to zero and make the PID controller a proportional (P) controller.
- Increase K_p till the output loop oscillates.
- Set the value of K_p at approximately half of that value for ' Quarter Amplitude Decay' response.
- Increase K_i to remove the offset error.
- Lastly, increase K_d to get faster system response.

B. Ant Colony Optimization

ACO uses the concept of ants reaching out in search of food along various paths. As they move along the path, they leave a trail of pheromone which changes its concentration according to the ant population on the way.

ACO works on this principle and finds the best solution until the number of iterations are finished or end criterion is fulfilled. The steps for ACO are discussed below,

- Define the ACO parameters as shown in Table I .
- Initialize the pheromone trail, set the lower bound and upper bound values for parameters, and generate nodes.

TABLE I. ACO PARAMETERS

No. of iterations	50
Alpha	0.8
Beta	0.5
Evaporation rate	0.7
No. of nodes	1000
No. of ants	30

- Place an ant on the node. The probability of choosing the successive node is given by, $P_{ij}^{A}(t) = \frac{[\tau_{ij}(t)]^{\alpha}[\eta_{ij}]^{\beta}}{\sum_{i,j\in T^{A}}[\tau_{ij}(t)]^{\alpha}[\eta_{ij}]^{\beta}}$, if $i,j \in T^{A}$ where η_{ij} represents heuristic function, T^{A} is path effectuated by the ant.
- Compute the cost value associated with the objective function.
  The objective function selected here is Integral Time Absolute Error (ITAE) $= \int_{0}^{\infty} t \mid e(t)dt$.
- Quantity of pheromone for each path can be defined as, $\Delta\tau_{ij}^{A} = \begin{cases} \frac{L^{min}}{L^{A}}, & \text{if } i,j \in T^{A} \\ 0, & \text{else} \end{cases}$ where L^{min} is the best solution found by the ants in the current iteration.
- The rate of change of pheromone is done to avoid the unlimited increase in pheromone trails also it helps to forget the bad choices and select the best path. The pheromone evaporation is given by, $\tau_{ij}(t) = \rho\tau_{ij}(t-1) + \sum_{A=1}^{\mathrm{NA}} \Delta\tau_{ij}^{A}(t)$ where NA= no. of ants and ρ= evaporation rate (0< $\rho \leq 1$).
- Display the optimum values of the optimization parameters.
- Update the pheromone matrix.
- Repeat the steps until all the iterations are completed or the termination criterion is met.

C. Grey Wolf Optimization

GWO follows the hunting mechanism of grey wolves which follows the leadership hierarchy as Alpha (α), the leader, Beta (β), Delta (δ), and Omega (ω) as the weakest. Alpha takes all the decisions n the group and beta is the 2nd in command to the leader. GWO algorithm involves three hunting steps of grey wolves i.e., tracking, chasing, and hunting (or encircling) the prey. Alpha denotes the best optimal solution followed by Beta and Delta other solutions are considered redundant. A detailed explanation of the mathematical model of GWO is given in [7]. The steps for the GWO algorithm are as follows,

TABLE II. GWO PARAMETERS

No. of search agents	30
No. of iterations	50
No. of dimensions	12

- Define GWO parameters as shown in Table II. Table II GWO Parameters
- Initialize the grey wolf population and start with a set of random solutions.
- Calculate the fitness of each search agent where, X_{α} is the best search agent, X_{β} is the second-best agent, and X_{δ} is the third-best search agent.
- For each search agent update the position of the current agent as,
  $\vec{X}(\text{iter} + 1) = (\vec{X}_1 + \vec{X}_2 + \vec{X}_3)/3$ where, $\vec{X}_1, \vec{X}_2, \vec{X}_3$ represents position vector of prey and $\vec{X}$ grey wolf, respectively. iter is the current iteration.
- Calculate the fitness for all search agents and keep updating X_{α}, X_{β}, and X_{δ}.

- Finally terminate the algorithm when all the iterations are completed or the end criterion is met. Potential solutions tend to leave from prey (If $|\vec{A}| > 1$) or converge towards the prey (If $|\vec{A}| < 1$).
- Display the X_α as the optimal solution for controller parameters.

D. Whale Optimization Algorithm

The new metaheuristic algorithm known as WOA was introduced by Mirjalili et. Al in 2016 [8]. It is based on the special hunting method called the bubble-net feeding method used by humpback whales this algorithm also has three stages similar to GWO i.e., encircling prey followed by bubble-net attacking method (exploitation) and search for prey (exploration). The steps to be followed for implementing WOA are discussed below,

- Set the WOA parameters the same as in Table 2.
- Start with a random set of solutions by initializing the population.
- For every iteration update the position of the search agent concerning the randomly chosen search agent.
- Check for the value of probability to switch between the two bubble-net attacking methods; shrinking encircling prey and spiral updating position.
- When A≥1, choose a random search agent whose solution given by the equation,
  $$\vec{D} = |\vec{C} \cdot \overrightarrow{X_{\text{rand}}} - \vec{X}|$$
  $$\vec{X}(t+1) = \overrightarrow{X_{\text{rand}}} - \vec{A} \cdot \vec{D}$$
  where, $\vec{A}$ *and* $\vec{C}$ are coefficient vectors, $\vec{X}$ is position vector and $\overrightarrow{X_{\text{rand}}}$ is random position vector.
- When A<1, choose the best solution for updating the position of search agents given by the equation,
  $$\vec{D} = |\vec{C} \cdot \vec{X}^*(t) - X(t)|$$
  $$\vec{X}(t+1) = \vec{X}^*(t) - \vec{A} \cdot \vec{D}$$
  where, $\vec{A}$ *and* $\vec{C}$ are coefficient vectors and $\vec{X}$ is position vector.
- Terminate the algorithm when the termination criterion is met or the number of iterations is completed.

IV. Simulation Results

The algorithms are executed in MATALB environment on a Simulink-based Triple Link Inverted Pendulum model controlled using PID controller. The comparative analysis for settling time, steady-state error, and peak overshoot for cart position, lower link angle, middle link angle, and upper link angle is given in Table 3, Table 4, Table 5, and Table 6 respectively.

TABLE III. Comparison of output response of Cart Position

	MANUAL PID	ACO-PID	GWO-PID	WOA-PID
Settling time (sec)	3.28	3.31	3.32	3.29
Peak overshoot	0.035	3.45e-08	4e-08	3.23e-08
Steady-state error	7.67e-05	0	0	0

TABLE IV. Comparison of output response of Lower Link Angle

	MANUAL PID	ACO-PID	GWO-PID	WOA-PID
Settling time (sec)	2.5	4.23	4.49	4.12
Peak overshoot	0.01	2.44e-09	2.77e-09	2.33e-09
Steady-state error	0	0	0	0

TABLE V. Comparison of output response of Middle Link Angle

	MANUAL PID	ACO-PID	GWO-PID	WOA-PID
Settling time (sec)	2.87	4.37	4.17	4.28
Peak overshoot	0.0057	1.83e-09	1.79e-09	1.71e-09
Steady-state error	0	0	0	0

Table VI. Comparison of output response of Upper Link Angle

	MANUAL PID	ACO-PID	GWO-PID	WOA-PID
Settling time (sec)	2.67	4.25	4.46	4.22
Peak overshoot	0.0065	1.62e-09	1.57e-09	1.40e-09
Steady-state error	0	0	0	0

Nature-inspired algorithms are advised because they provide efficient solution to the most complex problems easily as compared to conventional tuning methods. They reduce the computational efforts and time complexity to get the optimal solutions. For TLIP system, the settling time for conventional PID is less as compared but it production a steady-state error for cart position higher peak amplitude than the other metaheuristic based PID controllers makes it a bad choice for stabilization.

V. CONCLUSION

The results clearly depict that ACO, GWO and WOA algorithms are much better in comparison to conventional PID controller in terms of steady-state error and peak overshoot. Out of the three algorithms ACO and WOA show similar results although the computational time for ACO is much higher than WOA. So WOA is preferred among the three given metaheuristic algorithms. Hybrid algorithms along with other nature-inspired algorithms can be implemented to better the results in future.

APPENDIX

m_0 = 2.4 Kg m_1=1.323 Kg m_2=1.389 Kg m_3 = 0.8655 Kg L_1=0.402 m L_2=0.332 m L_3=0.72 m
l_1=0.2449 m l_2=0.193 m l_3=0.3405 J_1=.0119 Kgm^2 J_2=.0069e-3 Kgm^2 J_3=.0291 Kgm^2
f_0=13.611 Nsm^{-1} f_1=.0045 Nsm f2=0.0045 Nsm f_3=0.0045 Nsm Ks = 9.722 NV
g=9.81ms^{-2}

REFERENCES

[1] Gupta, M.K., Bansal, K. and Singh, A.K. (2014) Stabilization of Triple Link Inverted Pendulum System Based on LQR Control Technique. Recent Advances and Innovations in Engineering (ICRAIE), IEEE

[2] Furuta, K., Ochiai, T., and Ono, N., 1984, Attitude control of a triple inverted pendulum.

[3] Desineni Subbaram Naidu, “Optimal Control Systems”, Idaho State University, 2002, p. 129.

[4] S. Sehgal and S. Tiwari, "LQR control for stabilizing triple link inverted pendulum system," 2012 2nd International Conference on Power, Control and Embedded Systems, 2012, pp. 1-5, DOI: 10.1109/ICPCES.2012.6508052.

[5] Masrom, MF & Ghani, Nor Maniha & Tokhi, M.O. (2021). Particle swarm optimization and spiral dynamic algorithm-based interval type-2 fuzzy logic control of triple-link inverted pendulum system: A comparative assessment. Journal of Low Frequency Noise, Vibration and Active Control. 40. 146134841987378. 10.1177/1461348419873780.

[6] Lijuan Zhang, Yaqing Tu, “Research of Car Inverted Pendulum Model Based on Lagrange Equation”, Proceedings of the 6th World Congress on Intelligent Control and Automation, IEEE Press, June 2006,pp. 820-824.

[7] Mirjalili, S., Mirjalili, S. M. and Lewis, A., “Grey wolf optimizer”, Advances in Engineering Software, 69: 46- 61, (2014).

[8] Mirjalili, S., & Lewis, A. “The whale optimization algorithm.” Advances in Engineering Software, 2016, 95, 51-67.