The Efficiency of an Optimized PID Controller Based on Ant Colony Algorithm (ACO-PID) for the Position Control of a Multi-articulated System

Fatima Zahra Baghli [1*], Yassine Lakhal [2], Youssef Ait El Kadi [3,4]

[1,2] Engineering and Applied Physics Team (EAPT), Higher School of Technology, Sultan Moulay Slimane University, Beni Mellal, Morocco

[3] Higher School of Technology, Beni Mellal, Morocco

[4] Microelectronics, Embedded Systems and Telecommunications Team, Faculty of Sciences and Technics, Sultan Moulay Slimane University, Beni Mellal, Morocco

Email: [1] baghli.fatimazahra@gmail.com, [2] lakhal.yassine.doc@gmail.com, [3] youssef.aitelkadi@usms.ma

*Corresponding Author

Abstract—**In this article, a robot manipulator is controlled by the PID controller in a closed loop system with unit feedback. The difficulty of using the controller is parameter tuning, because the tuning parameters still use the trial and error method to find the PID parameter constants, namely Proportional Gain (K_p), Integral Gain (K_i) and Derivative Gain (K_d). In this case the Ant colony Optimization algorithm (ACO) is used to find the best gain parameters of the PID. The Ant algorithm is a method of combinatorial optimization, which utilizes the pattern of ants search for the shortest path from the nest to the place where the food is located, this concept is applied to tuning PID parameters by minimizing the objective function such that the robot manipulator has improved performance characteristics. This work uses the Matlab Simulink environment, First, after obtaining the system model, the ant colony algorithm is used to determine the proper coefficients K_p, K_i, and K_d in order to minimize the trajectory errors of the two joints of the robot manipulator. Then, the parameters will implement in the robot system. According to the results of the computer simulations, the proposed method (ACO-PID) gives a system that has a good performance compared with the classical PID.**

Keywords—***Multi-articulated System; Ant Colony Optimization; Tuning; PID Controller.***

I. INTRODUCTION

Robot manipulators are receiving more and more attention nowadays. This is because they carry out their tasks quickly and with a high degree of precision [1][2].

The robot manipulator is a mechanical system multi-articulated, in which each articulation is driven individually by an electric actuator is the most robot used in industry, this system needs an efficient control strategy based on the dynamic model [3][4][5][6][7].

The control of robots is the spine of robotics. It consists in studying how to make a robot manipulator do what it is desired to do automatically; hence, it includes in designing robot controllers. Typically, these take the form of an equation or an algorithm which is realized via specialized computer programs. Then, controllers form part of the so-called robot control system [8][9][10][11][12].

The Proportional Integral derivative (PID) controllers with simple structure have been in use for many years in industries for process control applications [13][14][15][16][17][18]. This regulator is obtained by the association of three parameters K_p (Proportional Constants), K_i (Integral Constants) and K_d (Derivative Constants) in designing the controllers, the goal is to get the optimal system response according to the desired design specifications [19][20][21][22]. But it is hard to find a proper PID controller [23][24][25]. To solve this problem, various methods have been proposed to tune the parameters of PID including the Ziegler-Nichols method [26] which is the most standard one but it is often difficult to find optimal PID parameters with this method.

In recent years, researchers have used many intelligent methods for tuning PID parameters, such as the Fuzzy method [27][28][29][30][31][32][33][34][35][36], Neural Network [37][38][39][40][41][42], the Ant colony method [43][44], the Genetic Algorithm method and the Particle Swarm Optimization (PSO) method [45][46][47][48].

In this work, Ant colony algorithm is used to determine PID controller parameters on a robot manipulator with two degrees of freedom. Ant colony optimization (ACO) is one of the most recent techniques for approximate optimization and one of the most successful examples of swarm intelligent that was introduced in the early 1990's by the Italian scholar Dorigo. The inspiring source of ant colony optimization is the foraging behavior of real ant colonies. At the core of this behavior is the indirect communication between the ants by means of chemical pheromone trails, and the first ACO algorithm was called the ant system and it was aimed to solve the travelling salesman problem, in which the goal is to find the shortest round-trip to link a series of cities [49][50].

Several research references said that parameter optimization using the smart methods has stable results compared to the classical methods [51][52][53][54].

This paper has two main contributions. First, after system modeling, PID corrector was designed the system using the Matlab / Simulink software. Secondly, for the same system, an ACO-PID corrector has been proposed. It has been established that the latter gives a better result from the point of view of the analysis of the performance (precision and response time).

The article is organized as follows, following the introduction (section I), section II describes the multi-articulated system. Section III describes the design of the PID controller. Section IV presents the ACO-PID design using the ant colony algorithm under MATLAB/Simulink. Section V interprets the numerical results and discussions. Section VI concludes this article.

II. MULTI-ARTICULATED SYSTEM

A. System Description

An industrial robot is defined by ISO as an automatically controlled, reprogrammable, multipurpose manipulator programmable in two or more axes. The field of robotics may be more practically defined as the study, design and use of robot systems for manufacturing (a top-level definition relying on the prior definition of robot).

Typical applications of robots include welding, painting, assembly, pick and place as shown in Fig. 1 (such as packaging, palletizing), product inspection, and testing; all accomplished with high endurance, speed, and precision. Motion control: for some applications, such as simple pick-and-place assembly, the robot needs merely to return repeatedly to a limited number of pre-taught positions. For more sophisticated applications, such as welding and finishing (spray painting), motion must be continuously controlled to follow a path in space, with controlled orientation and velocity.

Fig. 1. Pick and Place Robot manipulator [55]

B. Robot dynamic modeling

The analysis and control strategy design of robots generally includes kinematics analysis and dynamics analysis based on dynamic models. The dynamics model reflects the mathematical relationship between the motion of the robot, the driving torque and load [56].

The dynamical equation of manipulator robot of n solids articulated between us is given by equation (1) the following Lagrange method [57].

$$\tau = M(q)\ddot{q} + C(q,\dot{q}) + G(q) \tag{1}$$

where, τ is array of $(n \times 1)$ of all efforts applied on actuators $M(q)$ is the inertial matrix of $(n \times n)$, $C(q,\dot{q})$ present the array of $(n \times 1)$ of all Coriolis and centrifugal forces, $G(q)$ is the array of $(n \times 1)$ of all gravitational references and $q, \dot{q}, \ddot{q}$ are: position, speed and acceleration of each articulation.

In this work the multi-articulated system retained is a robot manipulator with two degrees of freedom. The model dynamic of the system is defined by (2).

$$\tau = \begin{bmatrix} \tau_1 \\ \tau_2 \end{bmatrix} = M(q)\begin{bmatrix} \ddot{q}_1 \\ \ddot{q}_2 \end{bmatrix} + C(q,\dot{q})\begin{bmatrix} \dot{q}_1 \\ \dot{q}_2 \end{bmatrix} + G(q) \tag{2}$$

These matrices are defined by:

$$M(q) = \begin{bmatrix} M_{11}(q) & M_{12}(q) \\ M_{21}(q) & M_{22}(q) \end{bmatrix}$$

where,

$$M_{11}(q) = I_1 + I_2 + m_1 l_{c1}^2 + m_2 l_{c2}^2 + m_2 l_1^2 + 2m_2 l_1 l_{c2} c_2$$

$$M_{12}(q) = M_{21}(q) = I_2 + m_2 l_{c2}^2 + 2m_2 l_1 l_{c2} c_2$$

$$M_{22}(q) = I_2 + m_2 l_{c2}^2$$

and,

$$C(q) = \begin{bmatrix} C_{11}(q,\dot{q}) & C_{12}(q,\dot{q}) \\ C_{21}(q,\dot{q}) & C_{22}(q,\dot{q}) \end{bmatrix}$$

where,

$$C_{11}(q,\dot{q}) = -m_2 l_1 l_{c2} \dot{q}_2 s_2$$

$$C_{12}(q,\dot{q}) = -m_2 l_1 l_{c2} s_2 (\dot{q}_1 + \dot{q}_2)$$

$$C_{21}(q,\dot{q}) = m_2 l_1 l_{c2} \dot{q}_1 s_2$$

$$C_{22}(q,\dot{q}) = 0$$

Finally, the gravity matrix is given by:

$$G(q) = \begin{bmatrix} G_1(q) \\ G_2(q) \end{bmatrix}$$

where,

$$G_1(q) = (m_1 + m_2)g l_{c1} c_1 + m_2 g l_{c2} c_{12}$$

$$G_2(q) = m_2 g l_{c2} c_{12}$$

where,

$c_1 = cos(q_1); c_2 = cos(q_2); s_1 = sin(q_1); s_2 = sin(q_2)$
$c_{12} = cos(q_1 + q_2); s_{12} = sin(q_1 + q_2);$

By defining the state, the input and the output vectors respectively X and U:

$$X = [X_1, X_2]^T; U = \tau = [\tau_1, \tau_2]^T$$

with,

$$X_1 = [q_1, q_2]^T$$
$$X_2 = [\dot{q}_1, \dot{q}_2]^T$$

The system can be represented by the state-space mathematical model:

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

Finally from the equation (3),

$$\begin{bmatrix} \dot{X}_1 \\ \dot{X}_2 \end{bmatrix} = \begin{bmatrix} X_2 \\ -\Delta \left\{ \begin{bmatrix} -m_2 l_1 l_{c2} \dot{q}_2 s_2 & -m_2 l_1 l_{c2} s_2 (\dot{q}_1 + \dot{q}_2) \\ m_2 l_1 l_{c2} \dot{q}_1 s_2 & 0 \end{bmatrix} + \begin{bmatrix} (m_1 + m_2) g l_{c1} c_1 + m_2 g l_{c2} c_{12} \\ m_2 g l_{c2} c_{12} \end{bmatrix} \right\} \end{bmatrix} + \begin{bmatrix} 0 \\ \Delta \end{bmatrix} \tau$$

where,

$$\Delta = \frac{1}{\det(M)} \begin{bmatrix} I_2 + m_2 l_{c2}^2 & -(I_2 + m_2 l_{c2}^2 + 2m_2 l_1 l_{c2} c_2 m_2 l_1 l_{c2} \dot{q}_1 s_2) \\ -(I_2 + m_2 l_{c2}^2 + 2m_2 l_1 l_{c2} c_2 m_2 l_1 l_{c2} \dot{q}_1 s_2) & I_1 + I_2 + m_1 l_{c1}^2 + m_2 l_{c2}^2 + m_2 l_1^2 + 2m_2 l_1 l_{c2} c_2 \end{bmatrix}$$

The Table I presents the robot manipulator parameters.

TABLE I. THE ROBOT MANIPULATOR PARAMETERS

Symbol	Description
m_1	Mass of link 1
m_2	Mass of link 2
l_1	Length of link 1
l_2	Length of link 2
I_1	Joint inertia 1
I_2	Joint inertia 2
q_1	Position of articulation 1
q_2	Position of articulation 2
g	Gravity
τ_i	The torque applied on actuators for each articulation

III. DESIGN OF PID CONTROLLER

A PID controller is the most widely used controller on an industrial scale, it has a simple structure [58][59][60][61]. This regulator is obtained by the association of the three actions (proportional, integral, derivative) and it essentially fulfills the following three functions [62][63][64][65][66][67]:

• It provides a control signal taking into account the evolution of the output signal in relation to the setpoint.

• It eliminates the static error due to the term integrator.

• It anticipates the variations of the output thanks to the derivative term.

The command structure of the PID controller is given by the Fig. 2. The output $u(t)$ of PID controller is given by (4).

$$u(t) = K_p \varepsilon(t) + K_i \int \varepsilon(t) dt + K_d \frac{d\varepsilon(t)}{dt} \qquad (4)$$

where, K_p, K_i, K_d are respectively the gain proportional, integral and derive and $\varepsilon(t)$ is the difference between the reference signal and the output.

where, the model obtained as (3),

$$\begin{bmatrix} \dot{X}_1 \\ \dot{X}_2 \end{bmatrix} = \underbrace{\begin{bmatrix} X_2 \\ -M^{-1}(X1)\{C(X_1, X_2) + G(X_1)\} \end{bmatrix}}_{A} + \underbrace{\begin{bmatrix} 0 \\ -M^{-1}(X1) \end{bmatrix}}_{B} \tau \qquad (3)$$

and the output given by:

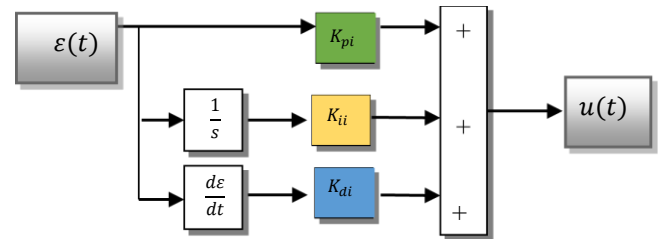$$Y = \underbrace{[I_n : O_{nxn}]}_{C} \begin{bmatrix} X_1 \\ X_2 \end{bmatrix}$$

Fig. 2. Fundamental PID control structure

A. Proportional term

An output value is produced by the proportional term that is proportional to the current error value. By multiplying a constant K_p that can be adjusted by the proportional response, the proportional term reduces the error response. The equation of this parameter is given by equation (5).

$$P_{output} = K_p \varepsilon(t) \qquad (5)$$

where, P_{out} is the proportional part of controller output, K_p is the proportional gain, and $\varepsilon(t)$ is error term.

B. Integral term

Integral term K_i contributes as it proportions to the error duration and also the magnitude of error. It is the sum of the instantaneous error over time that gives the accumulated offset values which are previously corrected. This term given by equation (6) eliminates the steady-state error.

$$I_{output} = K_i \int \varepsilon(t) dt + \frac{1}{T_i} \int \varepsilon(t) dt \qquad (6)$$

where, I_{output} is the integral part of controller output, T_i is reset time or integral time, K_i is the integral gain, and $\varepsilon(t)$ is error term.

C. Derivative term

By determining the error slope over time and multiplying the derivative gain, K_d with this rate of change, the process error derivative can be calculated. The equation of the

derivative gain K_d that dampens the dynamic response and improves the system stability as (7).

$$D_{Output} = K_d \frac{d\varepsilon(t)}{dt} = T_d \frac{d\varepsilon(t)}{dt} \qquad (7)$$

where, D_{Output} is the derivative part of controller output, T_d is the derivative time, K_d is the derivative gain, and $\varepsilon(t)$ is error term.

The independent effect of increasing parameters value in PID control is shown in Table II [41].

TABLE II. RESULTS EFFECT OF INDEPENDENT P, I, AND D TUNING

	Rise time	Overshoot	Steady State Error
$K_p\uparrow$	Decrease	Increase	Decrease
$K_i\uparrow$	Small Decrease	Increase	Large Decrease
$K_d\uparrow$	Small Decrease	Decrease	Minor Change

The mathematical description of the PID controller applied to the control input of the manipulator arm τ is described by equation (8).

$$\tau = K_p\varepsilon + K_i \int \varepsilon(t)dt + K_d\dot{\varepsilon} \qquad (8)$$

According to equation (2) we can have (9).

$$\ddot{q} = M(q)^{-1}[-C(q,\dot{q}) - G(q)] + \tau' \qquad (9)$$

where,

$$\tau' = M(q)^{-1}\tau$$

and

$$\tau' = \begin{bmatrix} \tau_1' \\ \tau_2' \end{bmatrix}$$

The inputs of the torques applied to the arm are given by :

$$\begin{bmatrix} \tau_1 \\ \tau_2 \end{bmatrix} = M(q) \begin{bmatrix} \tau_1' \\ \tau_2' \end{bmatrix}$$

And the error signals are given by :

$$\varepsilon_1 = (q_1^* - q_1)$$
$$\varepsilon_1 = (q_2^* - q_2)$$

So, the general equation of the manipulator arm by introducing parameters PID controller would be equation (10).

$$\begin{bmatrix} \ddot{q}_1 \\ \ddot{q}_2 \end{bmatrix} = M(q)^{-1}[-C(q,\dot{q}) - G(q)] + \tau' \qquad (10)$$

with,

$$\tau' = \begin{bmatrix} \tau_1' \\ \tau_2' \end{bmatrix} = \begin{bmatrix} K_{p1}(q_1^* - q_1) + K_{i1} \int \varepsilon(q_1)dt + K_{d1}\dot{\varepsilon}_1 \\ K_{p2}(q_2^* - q_2) + K_{i2} \int \varepsilon(q_2)dt + K_{d2}\dot{\varepsilon}_2 \end{bmatrix}$$

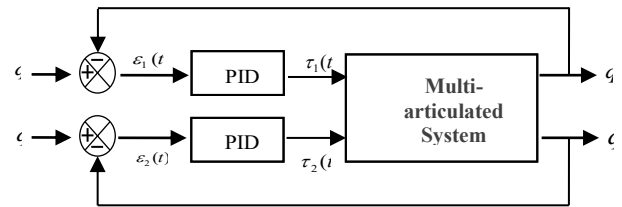The Fig. 3 shows the multi-articulated system classical PID Control

Fig. 3. Multi-articulated system classical PID Control

D. PID controller tuning methods

a) Classical methods:

Some of these classical tuning approaches, such as Ziegler-Nichols tuning method, Cohen-Coon tuning method, and Gain and Phase method are centered on making some assumptions about system model and necessary output [68] [69] [70].

b) Optimization Techniques:

These methods entail the use of data modeling and cost function optimization techniques to adjust PID parameters. Among these are the evolutionary algorithms, swarm intelligence algorithms, and fuzzy logic algorithms. [71][72][73][74][75].

IV.   ANT COLONY ALGORITHM

The ant colony optimization (ACO) algorithm is a meta-heuristic approach, it was first proposed by Marco Dorigo in 1992 in his Ph. D thesis. The principle of this method is based on the behavior of ants in real world, which are known to be able to find the shortest path from their nest to a food source. Ants accomplish this by depositing a substance called a pheromone as they move. This chemical trail can be detected by other ants, which are probabilistically more likely to follow a path rich in pheromone. This trail information can be utilized to adapt to sudden unexpected changes to the terrain, such as when an obstruction blocks a previously used part of the path [76][77][78][78][79][80][81]. The Fig. 4 shows the behavior of ant.
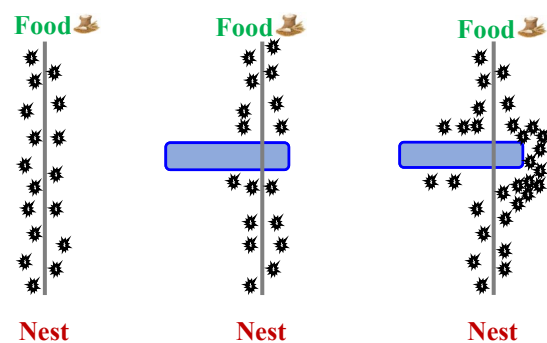

Fig. 4. The behavior of ant between Nest and Food

After each move, an ant leaves a pheromone trail on the connecting path to be collected by other ants to compute the transition probabilities. Starting from the initial session i, an explorer ant m chooses probabilistically session j to observe next using the transition rule, as shown in equation (11).

$$P_m(i,j) = \begin{cases} \dfrac{\left[\tau_{(i,j)}\right]^\alpha \cdot \left[\eta_{(i,j)}\right]^\beta}{\sum_{k \in S_m(i)} \left[\tau_{(i,j)}\right]^\alpha \cdot \left[\eta_{(i,j)}\right]^\beta} & if \ j \in S_m(i) \\ 0 & otherwise \end{cases}$$

(11)

where, $\tau(i,j)$ is the intensity measure of the pheromone deposited by each ant on the path (i,j). The intensity changes during the run of the program. α is the intensity control parameter. $\eta(i,j)$ is the visibility measure of the quality of the path (i,j). This visibility, which remains constant during the run of the program. β is the visibility control parameter. And $S_m(i)$ is the set of sessions that remain to be observed by ant m positioned at session i.

Equation (11) shows that the quality of the path (i,j) is proportional to its shortness and to the highest amount of pheromone deposited on it (i.e., the selection probability is proportional to path quality).

Ants change the pheromone level on the paths between sessions using the following updating rule in equation (12).

$$\tau_{(i,j)} \leftarrow \rho \cdot \tau_{(i,j)} + \Delta\tau_{(i,j)}$$

(12)

where, ρ is the trail evaporation parameter and $\Delta\tau(i,j)$ is the pheromone level.

In this work, we apply the ant colony algorithm in the aim of tuning the optimum solution of the PID controllers (K_p, K_i, and K_d) by minimizin the objective function such that the multi-articulated system has improved performance characteristics.

V. DESIGN ACO-PID FOR À MULTI-ARTICULATED SYSTEM

To achieve good performance, a PID tuning approach based on ant colony optimization is created. This method optimizes the PID parameters for the position control of the robot.

The ant colony optimization (ACO) algorithm is a distinct method inspired by insect swarm behavior and was designed for combinatorial issues at first. ACO is a stochastic based metaheuristic method for solving combinatorial optimization issues that employs artificial ants. ACO's goal is to find shorter routes from their nests to food sources.

Choosing the optimization criteria that are used to assess fitness is the first stage in applying the optimization process.

The most frequently utilized PID controller indexes for measuring transient response performance are:

- IAE : Integral of the absolute value of error in equation (13).

$$IAE = \int_0^\infty |\varepsilon_i(t)| \, dt$$

(13)

- ISE : Integral of the square value of error in equation (14).

$$ISE = \int_0^\infty \varepsilon_i(t)^2 \, dt$$

(14)

- ITAE : Integral of the time weighted absolute value of the error can be seen in (15).

$$ITAE = \int_0^\infty t|\varepsilon_i(t)| \, dt$$

(15)

- ITSE : Integral of the time weighted square of the error as shown in (16).

$$ITSE = \int_0^\infty t\varepsilon_i(t)^2 \, dt$$

(16)

In this paper, we present the optimization of the PID controller using the Ant colony algorithm for the position control of the robot. The goal now is to find the proper coefficients K_p, K_i, and K_d in order to minimize the trajectory errors of the two joints of the robot manipulator. The control strategy is based on the dynamically response of the robotic system. The block structure is shown in Fig. 5.
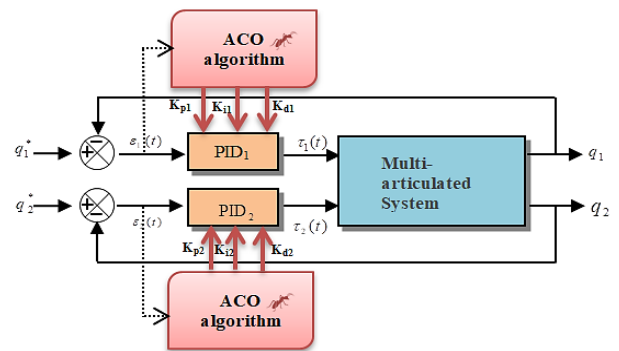

Fig. 5. Multi-articulated ACO-PID control

To solve the problem of the design of the PID controller with the ACO algorithm, this may be described as a network problem. The graphical representation of the optimized problem is shown in Fig. 6.
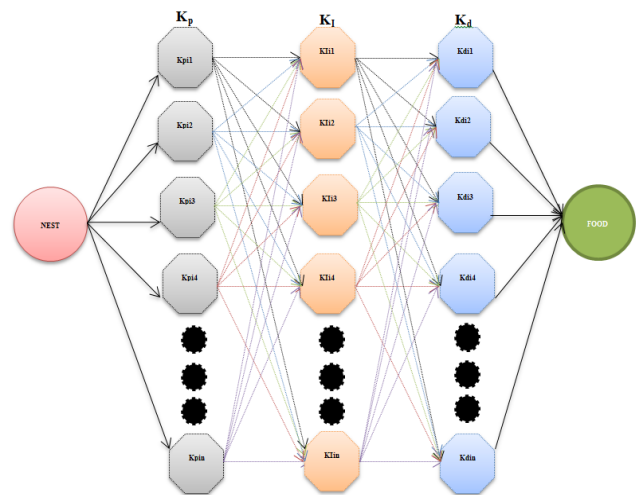

Fig. 6. Graphical representation of ACO for PID$_i$ (i=1,2) tuning process

For each parameter of tunable parameter of the PID$_{i(i=1,2)}$. The limiting values are defined by:

$$K_{Pi}^{min} \leq K_{Pi} \leq K_{Pi}^{max}$$

$$K_{Ii}^{min} \leq K_{Ii} \leq K_{Ii}^{max}$$

$$K_{di}^{min} \leq K_{di} \leq K_{di}^{max}$$

The steps of PID optimization with ACO are given in the algorithm of ACO presented below:

Begin
Step1: Initialize the algorithm parameters. Initialize randomly a potential solution of the parameters of PIDi
$$(K_{Pi}, K_{Ii}, K_{di})$$
Step 2: Run the process model and evaluate the cost function
Step 3: choose the successive node with probability by using Equation (9).
Step 4: Calculate the optimum values of $K_{Pi}, K_{Ii} et K_{di}$
Step 5: Use pheromone evaporation to avoid unlimited increase of pheromone trails and allow the forgetfulness of bad choices.
Step 6: Globally update the pheromone according to the optimum solution. Iterate from step 2 until the maximum of iteration is reached.
Step 7: choose the path of the maximum pheromone and optimum values of $K_{Pi}, K_{Ii} et K_{di}$.
End.

VI.　NUMERICAL RESULTS AND ANALYSIS

In this paper design and implementation of Single Input and Single Output (SISO) control based on the proposed Optimized PID Controller (ACO-PID) and the classical PID model were tested to a reference signal (Fig. 7). This simulation applied to two degrees of freedom robot arm was implemented in Matlab/Simulink. Trajectory performance and position error are compared in these controllers.
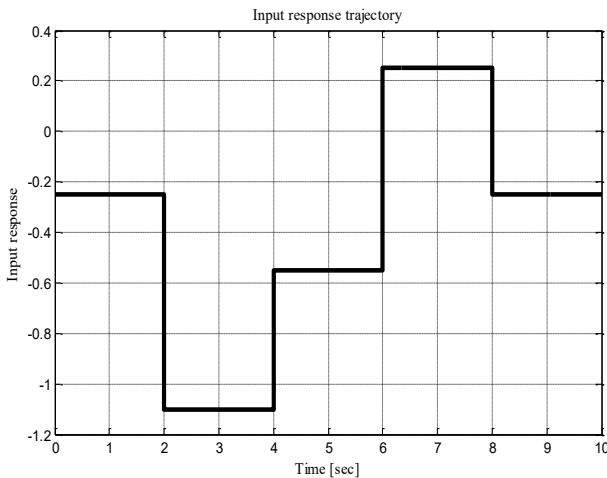

Fig. 7. The reference signal

The flowchart for ant colony is as shown in Fig. 8. In this work, we have used the following parameters values for the ACO for PID which is step in the Table III.

TABLE III. ANT COLONY PARAMETERS

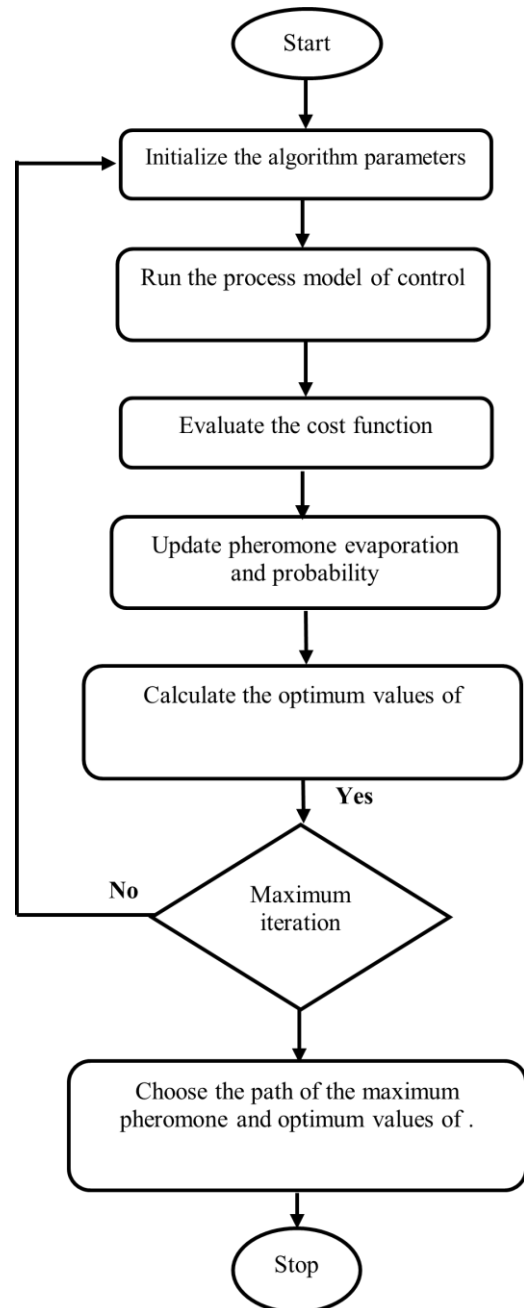| Parameter	value
Number of ant	10
Number of maximum cycle	100
Parameter of evaporation	0.5
Relative important parameter of trail intensity	1
Relative important parameter of visibility	5
Initial pheromone trails	0.1

Fig. 8. Graphical The flowchart for ant colony

The Table IV displays the various PID parameter values tuned using the ACO technique.

TABLE IV. RESULTS OF TUNING OF PID CONTROLLER BY ANT COLONY ALGORITHM

	Parameter band	Optimal gain (PID₁)	Optimal gain (PID₂)
K_p	[0.01 500]	220	150
K_i	[0.01 500]	50	10
K_d	[0.01 500]	350	100

❖ **The trajectory performances:**

The Fig. 9, Fig. 10, Fig. 11, and Fig. 12 are show tracking performance for first and second link with PID and ACO-PID for the reference trajectory. By comparing the input signal with PID and ACO-PID: For the first link (joint) controlled by

PID, the output does not coincide with the reference (Fig. 9) but by the ACO-PID they coincident as shown in Fig. 10, the overshoot PID's higher than ACO-PID.
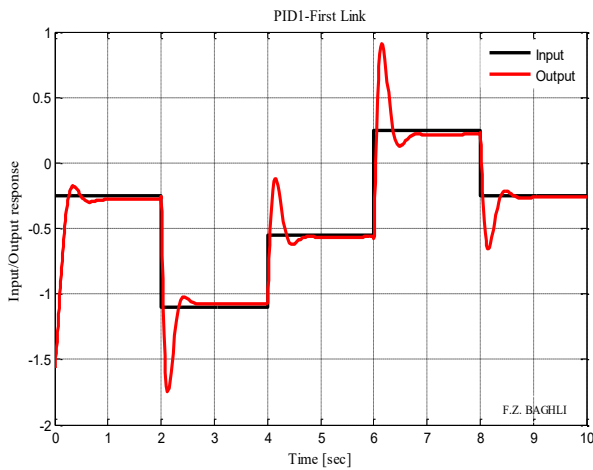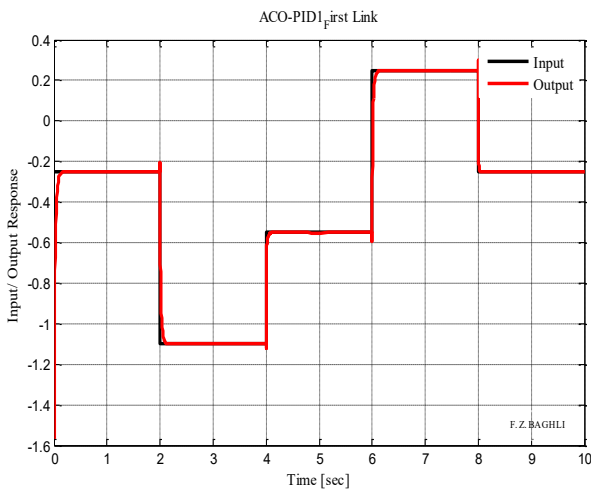

Fig. 9. PID₁ First link trajectory

Fig. 10. ACO-PID₁ First link trajectory

For the second link controlled by PID, the output does not attaint the reference signal (Fig. 11) but by the ACO-PID they coincident as shown in Fig. 12.
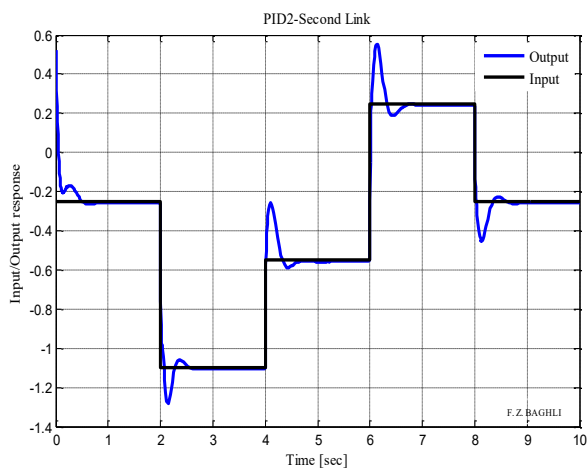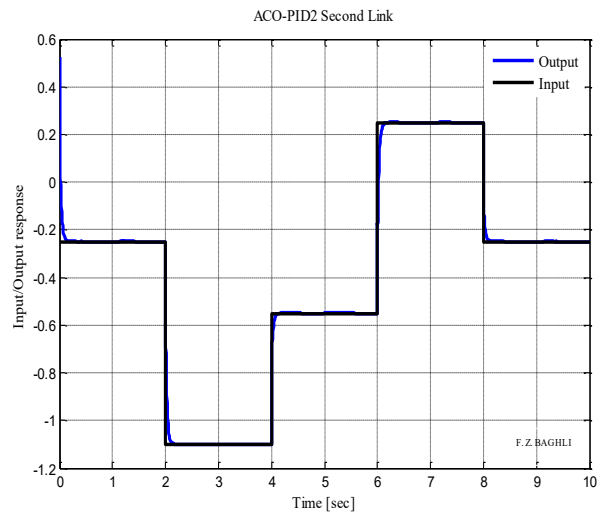

Fig. 11. PID₂ second link trajectory

Fig. 12. ACO-PID₂ second link trajectory

❖ **Error computation compare:**

The Fig. 13, Fig. 14, Fig. 15 and Fig. 16 are shown error performance, by comparing position error for the first and second link. By comparing position error for the first and second link; PID's error is higher than ACO-PID.
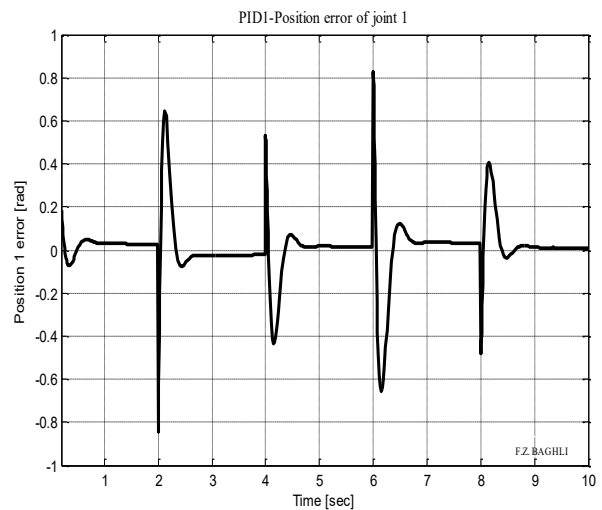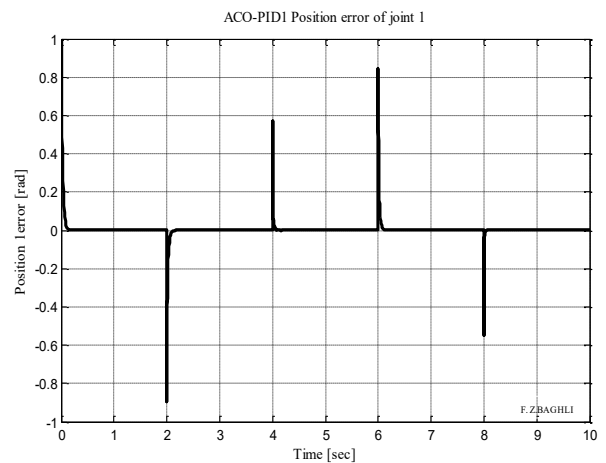

Fig. 13. PID₁ for the first link position error
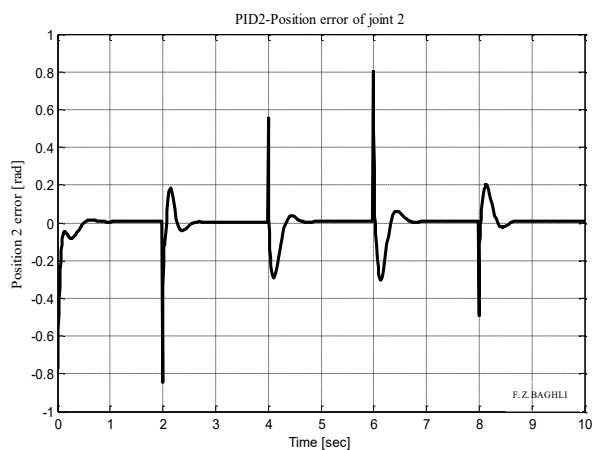

Fig. 14. ACO-PID₁ for the first link position error

Fig. 15. PID₂ for the second link position error

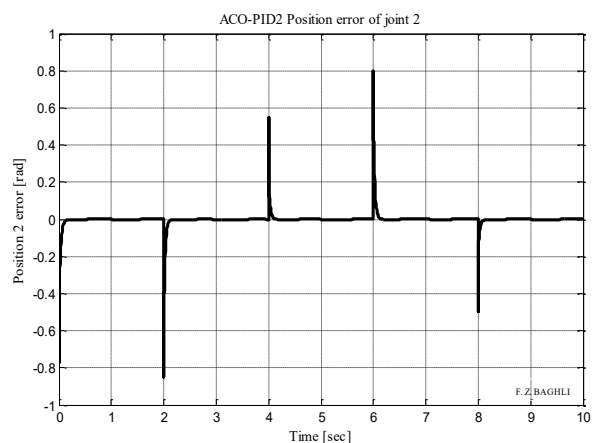
Fig. 16. PID₂ for the second link position error

According to the results of the computer simulations, the ACO- PID controller better than the traditional PID.

CONCLUSION

In this work a multi-articulated system with two degree of freedom was controlled using two types of controls strategies, a control based on classical control PID, and optimized PID using Ant colony algorithm, this last one is used to determine the proper coefficients K_p, K_i, and K_d in order to minimize the trajectory errors of the two joints of the robot manipulator.

From the results of research and discussion, it can be concluded that the control process with a control parameter tuning system PID with Ant colony algorithm present maximum control structure of our control model and give more and more efficiency for the robot model with more position stability and good dynamical performances so industrials would take into account the efficiency of the developing control model for the futures robot design considerations such to solving the inverse kinematics problem and analysis the workspace of a robotic arm.

REFERENCES

[1] B. AlKhlidi, A. T. Abdulsadda, and A. Al Bakri, "Optimal Robotic Path Planning Using Intelligent Search Algorithms," *Journal of Robotics and Control (JRC)*, vol. 2, no. 6, pp. 519-526, 2021.

[2] M. Marsono, Y. Yoto, A. Suyetno, and R. Nurmalasari, "Design and programming of 5 axis manipulator robot with grblgru open-source software on preparing vocational students' robotic skills," *Journal of Robotics and Control (JRC)*, vol. 2, no. 6, pp. 539–545, 2021.

[3] F. Z. Baghli, L. El bakkali, Y. Lakhal, A. Nasri, B. Gasbaoui, "The efficiency of the inference system knowledge strategy for the position control of a robot manipulator with two degrees of freedom," *International Journal of Research in Engineering and Technology*, vol. 2, no. 7, pp. 138-145, 2013.

[4] F. Z. Baghli, Y. Lakhal, L. El Bakkali, and O. Hamdoun, "Design and simulation of Adaptive Neuro Fuzzy Inference System (ANFIS) controller for a robot manipulator," *2014 Second World Conference on Complex Systems (WCCS)*, pp. 298-303, 2014.

[5] R. Çelikel, "Speed Control of BLDC Using NARMA-L2 Controller in Single Link Manipulator," *Balkan Journal of Electrical and Computer Engineering*, vol. 7, no. 2, pp. 143-148, 2019.

[6] E. H. Kadhim and A. T. Abdulsadda, "Mini Drone Linear and Nonlinear Controller System Design and Analyzing," *Journal of Robotics and Control (JRC)*, vol. 3, no. 2, pp. 212-218, 2022.

[7] W. M. Thet, M. Myint, and E. E. Khin, "Modelling and Control of Quadrotor Control System using MATLAB/Simulink," *International Journal of Science and Engineering Applications*, vol. 7, no. 7, pp. 125–129, 2018.

[8] F. Z. Baghli, L. El Bakkali, Y. Lakhal, A. Nasri, and B. Gasbaoui, "Arm Manipulator Position Control Based on Multi-Input Multi-Output PID Strategy," *Journal of Automation, Mobile Robotics and Intelligent Systems*, vol. 8 no. 2, pp. 36-39, 2014.

[9] A. Abhinash and A. K. Jha, "Decision of Robotics Trajectory based on Curve Fitting Techniques," *International Journal of Scientific Engineerring and Research (IJSER)*, vol. 2, no. 2, pp. 35-38, 2014.

[10] A. Kharola, S. Raiwani, and S. Kharola, "Soft-Computing-Based Real-Time Control of Two-Wheel Mobile Robot (TWMR)," in *Handbook of Research on Modeling, Analysis, and Control of Complex Systems*, pp. 434–449, 2021.

[11] H. Chhabra, V. Mohan, A. Rani, and V. Singh, "Robust nonlinear fractional order fuzzy PD plus fuzzy I controller applied to robotic manipulator," *Neural Computing and Applications*, vol. 32, no. 7, pp. 2055–2079, 2020.

[12] R. R. Ardeshiri, H. N. Kashani, and A. Reza-Ahrabi, "Design and simulation of self-Tuning fractional order fuzzy PID controller for robotic manipulator," *International Journal of Automation and Control*, vol. 13, no. 5, pp. 595–618, 2019.

[13] A. L. Shuraiji and S. W. Shneen, "Fuzzy Logic Control and PID Controller for Brushless Permanent Magnetic Direct Current Motor: A Comparative Study," *Journal of Robotics and Control (JRC)*, vol. 3, no. 6, pp. 762-768, 2022.

[14] A. L. Shuraiji and S. W. Shneen, "Fuzzy Logic Control and PID Controller for Brushless Permanent Magnetic Direct Current Motor: A Comparative Study," *Journal of Robotics and Control (JRC)*, vol. 3, no. 6, pp. 762-768, 2022.

[15] M. Maaruf, M. S. Mahmoud, and A. Ma'arif, "A Survey of Control Methods for Quadrotor UAV," *International Journal of Robotics and Control Systems*, vol. 2, no. 4, pp. 652-665, 2022.

[16] M. Maaruf, M. S. Mahmoud, and A. Ma'arif, "A Survey of Control Methods for Quadrotor UAV," *International Journal of Robotics and Control Systems*, vol. 2, no. 4, pp. 652-665, 2022.

[17] A. O. Amole, O. E. Olabode, D. O. Akinyele, and S. G. Akinjobi, "Optimal Temperature Control Scheme for Milk Pasteurization Process Using Different Tuning Techniques for a Proportional Integral Derivative Controller," *Iranian Journal of Electrical and Electronic Engineering*, vol. 2170, pp. 2170-2170, 2022.

[18] D. Maneetham and P. Sutyasadi, "Controlling a knee CPM machine using PID and iterative learning control algorithm," *TELKOMNIKA (Telecommunication Computing Electronics and Control)*, vol. 18, no. 2, pp. 1047-1053, 2022.

[19] E. W. Suseno and A. Ma'arif, "Tuning of PID Controller Parameters with Genetic Algorithm Method on DC Motor," *International Journal of Robotics and Control Systems*, vol. 1, no. 1, pp. 41-53, 2021.

[20] S.K. Mallempati, G. Satheesh, and S. Peddakotla, "Design of optimal PI controller for torque ripple minimization of SVPWM-DTC of BLDC motor," *International Journal of Power Electronics and Drive Systems*, vol. 14, no. 1, pp. 283-293, 2023.

[21] M. A. Shamseldin, "Adaptive Controller with PID, FOPID, and NPID Compensators for Tracking Control of Electric – Wind Vehicle," *Journal of Robotics and Control (JRC)*, vol. 3, no. 5, pp. 546-565, 2022.

[22] M. Elouni, B. Rabaoui, H. Hamdi, and N. B. Braiek, "Adaptive PID Takagi-Sugeno Actuator Fault Tolerant Tracking Control for Nonlinear Systems," in *2020 20th International Conference on Sciences and Techniques of Automatic Control and Computer Engineering (STA)*, pp. 119–124, Dec. 2020.

[23] R. P. Borase, D. K. Maghade, S. Y. Sondkar, and S. N. Pawar, "A review of PID control, tuning methods and applications," *International Journal of Dynamics and Control*, vol. 9, no. 2, pp. 818-827, 2021.

[24] H. Abdelfattah, S. A. Kotb, M. Esmail, and M. I. Mosaad, "Adaptive Neuro-Fuzzy Self-Tuned-PID Controller for," *International Journal of Robotics and Control Systems*, vol. 3, no. 1, pp. 1–18, 2023.

[25] P. Dutta and S. K. Nayak, "Grey Wolf Optimizer Based PID Controller for Speed Control of BLDC Motor," *Journal of Electrical Engineering & Technology*, vol. 16, no. 2, pp. 955-961, 2021.

[26] V. V. Patel, "Ziegler-Nichols Tuning Method," *Resonance*, vol. 25, no. 10, pp. 1385-1397, 2020.

[27] Z. Lin, C. Cui, and G. Wu, "Dynamic Modeling and Torque Feedforward based Optimal Fuzzy PD control of a High-Speed Parallel Manipulator," *Journal of Robotics and Control (JRC)*, vol. 2, no. 6, pp. 527-538, 2021.

[28] F. Z. Baghli and L. El Bakkali, "Design and Simulation of Robot Manipulator Position Control System Based on Adaptive Fuzzy PID Controller," *Robotics and Mechatronics*, vol. 37, 2016.

[29] N. Farouk and T. Bingqi, "Application of self-tuning fuzzy PID controller on the AVR system," *2012 IEEE International Conference on Mechatronics and Automation*, pp. 2510-2514, 2012.

[30] A. Dhyani, M. K. Panda, and B. Jha, "Moth-flame optimization-based fuzzy-PID controller for optimal control of active magnetic bearing system," *Iranian Journal of Science and Technology, Transactions of Electrical Engineering*, vol. 42, no. 4, pp. 451-463, 2018.

[31] C. T. Chao, N. Sutarna, J. S. Chiou, and C. J. Wang, "An optimal fuzzy PID controller design based on conventional PID control and nonlinear factors," *Applied Sciences*, vol. 9, no. 6, p. 1224, 2019.

[32] T. Y. Wu, Y. Z. Jiang, Y. Z. Su, and W. C. Yeh, "Using Simplified Swarm Optimization on Multiloop Fuzzy PID Controller Tuning Design for Flow and Temperature Control System," *Applied Sciences*, vol. 10, no. 23, p. 8472, 2020.

[33] H. Zhao, J. Guo, W. Xu, and S. Yan, "Research on trajectory planning of mobile robot based on fuzzy ant colony algorithm," *Computer simulation*, vol. 35, no. 5, pp. 318–321, 2018.

[34] D. Somwanshi, M. Bundele, G. Kumar, and G. Parashar, "Comparison of Fuzzy-PID and PID Controller for Speed Control of DC Motor Using LabVIEW," in *Procedia Computer Science*, vol. 152, pp. 252-260, 2019.

[35] M. A. Eissa, A. Sali, F. A. Ahmad, and R. R. Darwish, "Observer-Based Fault Detection Approach Using Fuzzy Adaptive Poles Placement System with Real-Time Implementation," *IEEE Access*, vol. 9, pp. 83272–83284, 2021.

[36] M. A. Eissa, A. Sali, M. K. Hassan, A. M. Bassiuny, and R. R. Darwish, "Observer-Based Fault Detection with Fuzzy Variable Gains and Its Application to Industrial Servo System," *IEEE Access*, vol. 8, pp. 131224–131238, 2020.

[37] S. B. Joseph, E. G. Dada, A. Abidemi, D. O. Oyewola, and B. M. Khammasall, "Metaheuristic algorithms for PID controller parameters tuning: review, approaches and open problems," *Heliyon*, vol. 8, no. 5, 2022.

[38] C. B. Jabeur and H. Seddik, "Optimized Neural Networks-PID Controller with Wind Rejection Strategy for a Quad-Rotor Chiraz," *Journal of Robotics and Control (JRC)*, vol. 3, no. 1, pp. 62-72, 2022.

[39] I. Chiha, N. Liouane, and P. Borne, "Tuning PID Controller Using Multiobjective Ant Colony Optimization," *Applied Computational Intelligence and Soft Computing*, vol. 2012, 2012.

[40] K. Kinoshita, S. Wakitani, and S. Ohno, "Design of neural network PID controller based on E-FRIT," *Electrical Engineering in Japan*, vol. 205, no. 2, pp. 33–42, 2018.

[41] C. B. Jabeur and H. Seddik, "Optimized Neural Networks-PID Controller with Wind Rejection Strategy for a Quad-Rotor," *Journal of Robotics and Control (JRC)*, vol. 3, no. 1, pp. 62–72, 2022.

[42] S. Gobinath and M. Madheswaran, "Deep Perceptron Neural Network with Fuzzy PID Controller for Speed Control and Stability Analysis of BLDC Motor," *Soft Computing*, vol. 24, no. 13, pp. 10161-10180, 2020.

[43] H. Du, P. Liu, Q. Cui, X. Ma, and H. Wang, "PID Controller Parameter Optimized by Reformative Artificial Bee Colony Algorithm," *Journal of Mathematics*, vol. 2022, pp. 1–16, 2022.

[44] H. Wang, H. Du, Q. Cui, and H. Song, "Artificial bee colony algorithm based PID controller for steel stripe deviation control system," *Science Progress*, vol. 105, no. 1, pp. 1–22, 2022.

[45] E. S. Rahayu, A. Ma'arif, and A. Cakan," Particle Swarm Optimization (PSO) Tuning of PID Control on DC Motor," *International Journal of Robotics and Control Systems*, vol. 2, no. 2, pp. 435-447, 2022.

[46] Z. Qi, Q. Shi, and H. Zhang, "Tuning of digital PID controllers using particle swarm optimization algorithm for a CAN-Based DC motor subject to stochastic delays," *IEEE Transactions on Industrial Electronics*, vol. 67, no. 7, pp. 5637–5646, 2020.

[47] F. Li, Y. Li, C. Yan, C. Ma, C. Liu, and Q. Suo, "Swing Speed Control Strategy of Fuzzy PID Roadheader Based on PSO-BP Algorithm," *IEEE 6th Information Technology and Mechatronics Engineering Conference (ITOEC)*, pp. 827-833, 2022.

[48] T. F. Abaas and A. H. Shabeeb, "Autonomous mobile robot navigation based on PSO algorithm with inertia weight variants for optimal path planning," *IOP Conf. Ser. Mater. Sci. Eng.*, vol. 928, no. 2, 2020.

[49] F. Z. Baghli, L. El bakkali, and Y. Lakhal, "Optimization of Arm Manipulator Trajectory Planning in The Presence of Obstacles by Ant Colony Algorithm," *Procedia Engineering*, vol. 181, pp. 560 – 567, 2017.

[50] D. Xu and L. Fu, "Ant Colony Optimization and its Applications," *2009 International Workshop on Intelligent Systems and Applications*, pp. 1-4, 2009.

[51] A. Ma'arif, H. Nabila, Iswanto, and O. Wahyunggoro, "Application of Intelligent Search Algorithms in Proportional-Integral-Derivative Control of Direct-Current Motor System," *Journal of Physics: Conference Series*, vol. 1373 p. 012039, 2019.

[52] A. J. Abougarair, M. K. I. Aburakhis, and M. M. Edardar, "Adaptive Neural Networks Based Robust Output Feedback Controllers for Nonlinear Systems," *International Journal of Robotics and Control Systems*, vol. 2, no. 1, pp. 37-56, 2022.

[53] S. B. Joseph, E. G. Dada, A. Abidemi, D. O. Oyewola, and B. M. Khammas, "Metaheuristic algorithms for PID controller parameters tuning: review, approaches and open problems," *Heliyon*, vol. 8, no. 5, p. e09399, 2022.

[54] K. Vanchinathan and N. Selvaganesan, "Adaptive Fractional Order PID Controller Tuning for Brushless DC Motor Using Artificial Bee Colony Algorithm," *Results in Control and Optimization*, vol. 4, 2021.

[55] ROBOT KUKA, "Equipment for experimentation - Lucas Nülle", Higher School of Technology, Beni Mellal, Morocco.

[56] S. Dong, Z. Yuan, and F. Zhang, "A simplified Method for Dynamic Equation of Robot in Generalized Coordinate System," *Journal of Physics: Conference Series*, vol. 1345, p. 042077, 2019.

[57] F. Z. Baghli and L. El bakkali, "Artificial Intelligence Application's for a Robot Manipulator with Two Degrees of Freedom Position Control," *International Journal of Mechatronics, Electrical and Computer Technology*, vol. 4, no. 11, pp. 349-368, Apr. 2014.

[58] Y. Zahraoui, M. Akherraz, and A. Ma'arif, "A Comparative Study of Nonlinear Control Schemes for Induction Motor Operation Improvement," *International Journal of Robotics and Control Systems*, vol. 2, no. 1, pp. 1-17, 2022.

[59] D. Handaya and R. Fauziah, "Proportional-integral-derivative and linear quadratic regulator control of direct current motor position using multi-turn based on LabView," *Journal of Robotics and Control (JRC)*, vol. 2, no. 4, pp. 332–336, 2021.

[60] S. J. Hammoodi, K. S. Flayyih, and A. R. Hamad, "Design and Implementation of Speed Control System for DC Motor Based on PID Control and Matlab Simulink," *International Journal of Power Electronics and Drive Systems*, vol. 11, no. 1, pp. 127-134, 2020.

[61] V. Dubey, "Comparative Analysis of PID Tuning Techniques for Blood Glucose Level of Diabetic Patient," *Turkish Journal of Computer and Mathematics Education (TURCOMAT)*, vol. 12, no. 11, pp. 2948-2953, 2021.

[62] Y. Kang, Z. Li, and T. Wang, "Application of PID Control and Improved Ant Colony Algorithm in Path Planning of Substation Inspection Robot," *Mathematical Problems in Engineering*, vol. 2022, 2022.

[63] D. Handaya and R. Fauziah, "Proportional-Integral-Derivative and Linear Quadratic Regulator Control of Direct Current Motor Position using Multi-Turn Based on LabView," *Journal of Robotics and Control (JRC)*, vol. 2, no. 4, pp. 332-336, July 2021.

[64] A. Ma'arif and N. R. Setiawan, "Control of DC Motor Using Integral State Feedback and Comparison with PID: Simulation and Arduino Implementation," *Journal of Robotics and Control (JRC)*, vol. 2, no. 5, pp. 456-461, 2021.

[65] Y. Li, K. H. Ang, and G. C. Chong, "PID control system analysis and design," *IEEE Control Systems Magazine*, vol. 26, no. 1, pp. 32-41, 2006.

[66] R. R. Alla, N. Lekyasri, and K. Rajani, "PID Control Design for Second Order Systems," *International Journal of Engineering and Manufacturing (IJEM)*, vol. 9, no. 4, pp. 45-56, 2019.

[67] E. Bashier and O. Mohammed, "Optimally Tuned Proportional Integral Derivatives (PID) Controllers for Set-Point," *Journal of Engineering and Computer Science (JECS)*, vol. 13, no. 1, pp. 48-53, 2019.

[68] P. Chotikunnan and R. Chotikunnan, "Dual Design PID Controller for Robotic Manipulator Application," *Journal of Robotics and Control (JRC)*, vol. 4, no. 1, pp. 23-34, 2023.

[69] Y. Li, K. H. Ang, and G. C. Y. Chong, "PID Control System Analysis and Design," *IEEE Control Systems*, vol. 26, no. 1, pp. 32–41, 2006.

[70] F. N. Abdullah, G. A. Aziz, and S. W. Shneen, "Simulation Model of Servo Motor by Using Matlab," *Journal of Robotics and Control (JRC)*, vol. 3, no. 2, pp. 176–179, 2022.

[71] M. Elouni, H. Hamdi, B. Rabaoui, and N. B. Braiek, "Adaptive PID Fault-Tolerant Tracking Controller for Takagi Sugeno Fuzzy Systems with Actuator Faults: Application to Single-Link Flexible Joint Robot," *International Journal of Robotics and Control Systems*, vol. 2, no. 3, pp. 523-546, 2022.

[72] R. Kristiyono and W. Wiyono, "Autotuning Fuzzy PID Controller for Speed Control of BLDC Motor," *Journal of Robotics and Control (JRC)*, vol. 2, no. 5, pp. 400-407, 2021.

[73] M. Zadehbagheri, A. Ma'arif, R. Ildarabadi, and M. Ansarifard, "Design of Multivariate PID Controller for Power Networks Using GEA and PSO," *Journal of Robotics and Control (JRC)*, vol. 4, no. 1, pp. 108-117, 2023.

[74] C. B. Jabeur and H. Seddik, "Optimized Neural Networks-PID Controller with Wind Rejection Strategy for a Quad-Rotor," *Journal of Robotics and Control (JRC)*, vol. 3, no. 1, pp. 62-72, 2022.

[75] G. Cao, X. Zhao, C. Ye, S. Yu, B. Li, and C. Jiang, "Fuzzy Adaptive PID Control Method for Multi-Mecanum-Wheeled Mobile Robot," *Journal of Mechanical Science and Technology*, vol. 36, no. 4, pp. 2019-2029, 2022.

[76] M. Tomera, "Ant colony optimization algorithm applied to ship steering control," *Procedia Computer Science*, vol. 35, pp. 83 – 92, 2014.

[77] S. Mahmoud, A. Derouich, A. Iqbal, and N. El Ouanjli, "ANT-colony optimization-direct torque control for a doubly fed induction motor: An experimental validation," *Energy Reports*, vol. 8, pp. 81-98, 2022

[78] A. Reshamwala and D. P. Vinchurkar, "Robot Path Planning using An Ant Colony Optimization Approach: A Survey," *(IJARAI) International Journal of Advanced Research in Artificial Intelligence*, vol. 2, no.3, 2013.

[79] M. O. Okwu, L. K. Tartibu, *Metaheuristic optimization: Nature-inspired algorithms swarm and computational intelligence, theory and applications*, vol 927, Springer Nature, 2021.

[80] D. Qiao, W. Bai, K. Wang, and Y. Wang, "Review on the Improvement and Application of Ant Colony Algorithm," *Bio-inspired Computing: Theories and Applications. Communications in Computer and Information Science*, vol. 1159, 2020.

[81] J. Bai, L. Chen, H. Jin, R. Chen, and H. Mao, "Robot path planning based on random expansion of ant colony optimization," *Recent Advances in Computer Science and Information Engineering. Lecture Notes in Electrical Engineering*, vol. 125, pp. 141–146, 2012.