[1]Sandeep Yadav
Sunil Kumar[2*]
Manoj Goyal[3]
Prabhkiran Kaur[4]

Optimizing bionic Dual-Arm Underwater Robot Manipulator Performance: A Whales Optimization Algorithm Approach to PID Tuning

Journal of Electrical Systems

Abstract: - This paper focuses on optimizing a Proportional-Integral-Derivative (PID) controller for a bionic dual-arm with single link underwater robot manipulator. The aim is to control the system's output and minimize the integral of absolute errors (IAE) for enhanced stability and robustness. Employing the Whales Optimization Algorithm Approach (WOA) for PID tuning, the results demonstrate superior performance compared to traditional methods like PSO, GA. The efficacy of the WOA-PID controller is evaluated by comparing its performance with alternative controllers such as Particle Swarm Optimization-PID (PSO-PID) and Ant Colony Optimization-PID (ACO-PID) using the Simulink/MATLAB platform to simulate the robot manipulator model and test all the controllers. The simulation outcomes unequivocally demonstrate the superior efficiency of the WOA-PID controller for trajectory tracking, reducing time to achieve steady state, and minimized Integral of Time-weighted Absolute Error (ITAE) when compared to the other controllers.

Keywords: Underwater Robot Manipulator, bionic dual arm, Tuning, WOA-PID, PSO-PID, ACO-PID

Introduction

A robot manipulator, which is a multi-output and multi-input (MOMI) system characterized by high nonlinearity and coupling, is specifically designed to automate different manufacturing processes to replicating human actions in a defined area. The need for efficient controllers to improve the executions of industrial robotics, especially in rapid and accurate product inspection, presents a significant challenge [1]. Despite the progress in control theory, traditional PID algorithms continue to be extensively employed in controlling robot manipulators [2-5]. Many of these controllers, aiming to enhance performance, have been developed based on linear or linearized models. In some instances, a nonlinear PID controller strategy has been suggested, by embedding nonlinear terms into the proportional and derivative coefficients of the PID controller or by integrating gravity compensation techniques [6-7]. To tackle oscillations in the end-effectors of a single-link flexible joint robot manipulator, a state feedback control approach is employed to accurately follow the position trajectory [8]. Numerous studies have presented alternative underwater robot control algorithms for a single-arm URMS and certified them using computer simulations [8-11]. Despite the limited number of experimental trials conducted in actual underwater environments, there have been even fewer studies converging on the control system for bionic dual-arms underwater Robot Manipulation Systems (URMS). The challenges by proposing and recommending digitally Resolved Acceleration Control (RAC) solutions for both single-arm URMS [6-7] and dual-arm URMS. The efficacy of the suggested RAC approach can be achieved by using computer simulations [12]. Experiments employing a real dual-arm URMS demonstrate the efficacy of the RAC approach. This paper is divided into three sections. This study commenced by describing the mechanical arrangement of a bionic dual-arm manipulator. Subsequently, a concise overview of the unique control strategy for a dual-arm Underwater Vehicle Manipulation System (UVMS), which was initially introduced in our previous study, is presented. Consequently, an underwater robot prepared with anew developed bionic dual-arm URMS was utilized to conduct an underwater experiment to evaluate the efficacy of the suggested control approach. For 6-DOF industrial robot, tracking control employs an adaptive sliding mode technique. The controller incorporates sliding mode, adaptive, and robust controls with real-time updates of the unknown parameters can significantly improves dynamic trajectory tracking [13] .

This research utilizes the Whale Optimization Algorithm (WOA), an innovative optimization method introduced by Mirjalili in 2016[14], to enhance the performance of a PID controller in guiding the trajectory of a 2-degree-of-freedom (2-DOF) underwater robot manipulator. The effectiveness of this approach is related with controllers

[1 ,2,3]Sant Longowal Institute of Engineering & Technology, Longowal -148106, India

IKGPTU, Amritsar-143001, India

sandeepyadav.sliet@gmail.com, * sunil_thappa@yahoo.com,

manojsliet@yahoo.com, prabhkiran@yahoo.co.in

obtained from Particle Swarm Optimization (PSO) and Ant Colony Optimization-PID (ACO-PID). The paper is organized into seven sections. The second section provides a concise introduction to PID control, followed by an examination of the evolutionary algorithms utilized in the study in the third section. The fourth section discusses the dynamics of the 2-DOF robot. The fifth section of the paper presents simulation outcomes for all implemented controllers, delving into discussions on the management of the robot manipulator. Sections 6 and 7 address robustness testing and deliberations on control, respectively. The paper culminates in the final section, proposing avenues for future research endeavours.

Modelling and Simulation

Modelling and simulation depend on one another, but they can be carried out independently. To cross, three accesses must be carried out and coordinated: in this view, a model that embodies the conceptualization of the present study has been established, a simulation has been applied to this model, and it has been ensured that the model and simulation are connected. In the present study modelling and simulation of a newly designed two arms underwater robot, a manipulator has been performed to study the working of a human-like robot in underwater conditions. Figure 1 (a and b) shows the line diagram and 3D model of the underwater two arms manipulator with a single link each that has been considered in the current work. To look like the human underwater robot manipulator has been designed such that the left and right arms are attached at the forward corners of the base. Forward kinematics has been utilized to determine the location and alignment of the end effectors on both the left and right arms of an underwater robotic manipulator. The frames assigned for the transformation are shown in Figure 1. Frame {0} is considered at the CG of the base, frame {L1} is taken at the joint of the base and left arm, and frame {R1} is taken at the joint of the base and right arm. Frame {LT} and {RT} are considered at the end effectors of the left arm and right arm. The 3D-Solid-Work model of the designed two-arm underwater robotic manipulator is shown in Figure 2. Simulation has been performed in the SIMULINK by importing the .xml of the 3D model made in Solid-Work.

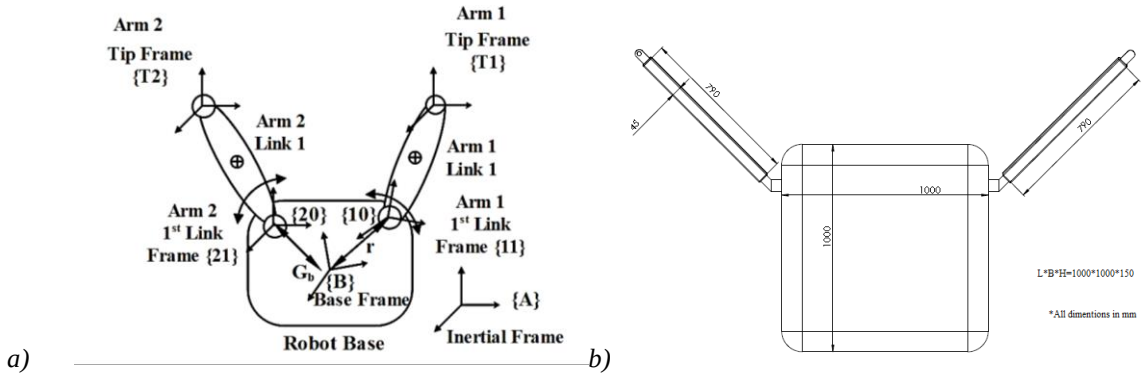


Figure 1. a) Line diagram of two arms underwater robotic manipulator with a single link each,

b) Line diagram with dimensions of a two-arm underwater robotic manipulator with a single link each.

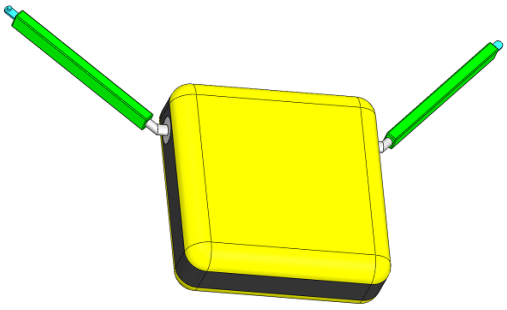


Figure 2. 3D-SolidWork model of two arms underwater robotic manipulator

Hybrid controller

The hybrid controller developed by Kumar et al. [20] has been used for the control of two arms underwater robotic manipulators with a single link each. In this controller, there is a combination of a PID controller and an overwhelming controller. PID controller provides the steady state error and the overwhelming controller handles unwanted disturbance in underwater conditions and overwhelms properly. Hybrid controller used for trajectory control is shown in Figure 3. Equations 1-2 provide the Laplace transfer equation for this PID controller.

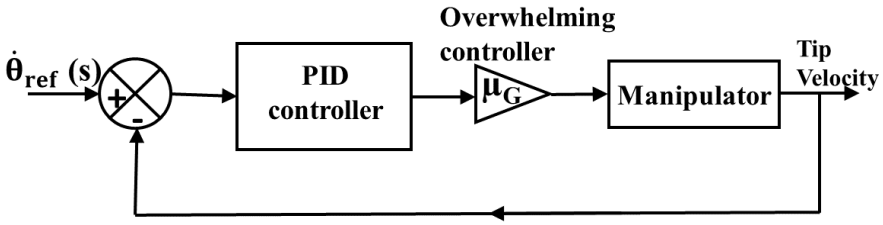


Figure 3. Hybrid Controller Trajectory Control: Visual Representation[20].

$$C(s)pid = K_p + K_i\frac{1}{s} + K_d\frac{N}{1+N\frac{1}{s}} \quad (1)$$

Where: K_p = Proportional gain, K_i = Integral gain, K_d = Derivative gain, N = Filter coefficient

$$ITAE = \int_0^\tau t|e(t)| \quad (2)$$

Where: $ITAE$ = Integral of Time multiplied by

Absolute Error, t = time, $e\,(t)$ = error

The block diagram's representation of the plant function is given in Equation 3.

$$Y(t) = \left(K_p e(t) + K_d\frac{d(e)}{d(t)} + K_i\int_0^t e(t)d(t)\right) \quad (3)$$

On the bases of Eq., (1, 2, and 3) and considering the block diagram in Figure 3 using the signal flow diagram the transfer function representing the relationship between the reference input and the trajectory of the arms' end effector is formulated as Equation 4.

$$\frac{F_{15}(s)}{f_{ref}(s)} = \frac{\mu_{H(K_{gp}s+K_{gI})(R_Rs+K_R)}}{(K_{gd}s^2+K_{gP}s+K_{gi})(I_Rs^3+R_RI_Rs^2+K_RI_Rs)+\mu_{H(K_{gp}s+K_{gI})(R_Rs+K_R)}} \quad (4)$$

Equation 4 can be simplified as:

$$\frac{F_{15}(s)}{f_{ref}(s)} = \frac{\mu_{H(K_{gp}s+K_{gI})(R_Rs+\square_R)}}{\begin{bmatrix} K_{gd}I_Rs^6+(K_{gd}R_RI_R+K_{gp}I_R)s^5+ \\ (K_{gd}K_RI_R+K_{gp}R_RI_R+K_{gi}I_R)s^4+ \\ (K_{gd}K_RI_R+K_{gi}R_RI_R)s^3+ \\ (K_{gi}K_RI_R+\mu_HK_{gp}R_R)s^2+ \\ (\mu_HK_{gp}K_R+\mu_HK_{gi}R_R)s+\mu_HK_{gi}K_R \end{bmatrix}} \quad (5)$$

The signal flow graph and that plant function are constructed to analyze the controller. Mason's gain rule is implemented to obtain the transfer function from the signal flow equation provided in Equation 5. The denominator of Eq. 5 gives the characteristic equation used for the tuning of the PID controller parameters by implementing it in simulation in Simulink.

Simulink Simulation of Model

Figure 4 illustrates the creation of the Simulink simulation model for the two arms with a single-link robotic manipulator based on the controller outlined above and the solid model made in SOLDWORK. A PID controller with a derivative filter manages each joint on the robotic manipulator. The joint actuator is given a modulus of a sinusoidal reference trajectory to test the efficiency of the controller's settings. The joint torque range for each

joint is controlled by the slider gains. Following ITAE recommendations, the PID controller's control parameters are changed. A pattern search strategy based on ITAE criteria is used to decrease ITAE [15]. After 24 iterations, a pattern search approach successfully optimizes the controller parameter. The research [16-20] covers PID controller parameter optimization. Table 2 and Table 3 display the computed and optimized parameters for the tuning of the PID controller parameters obtained.

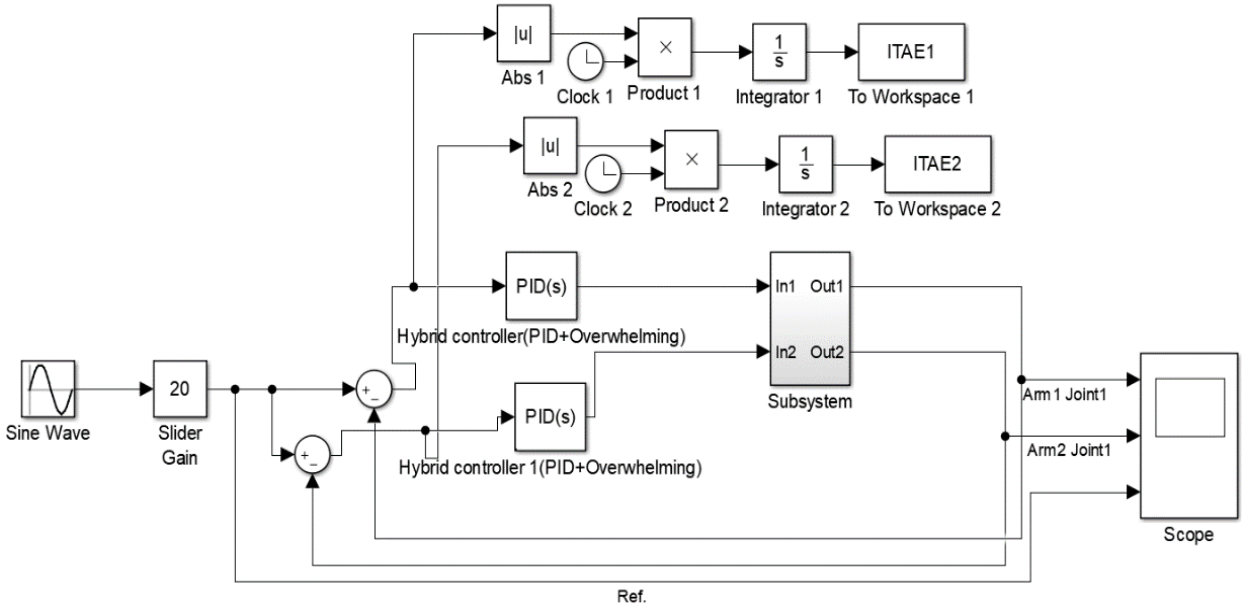


Figure 4. Simulink Model of Two-Arm underwater Robotic Manipulator with Cycloid Input Trajectory

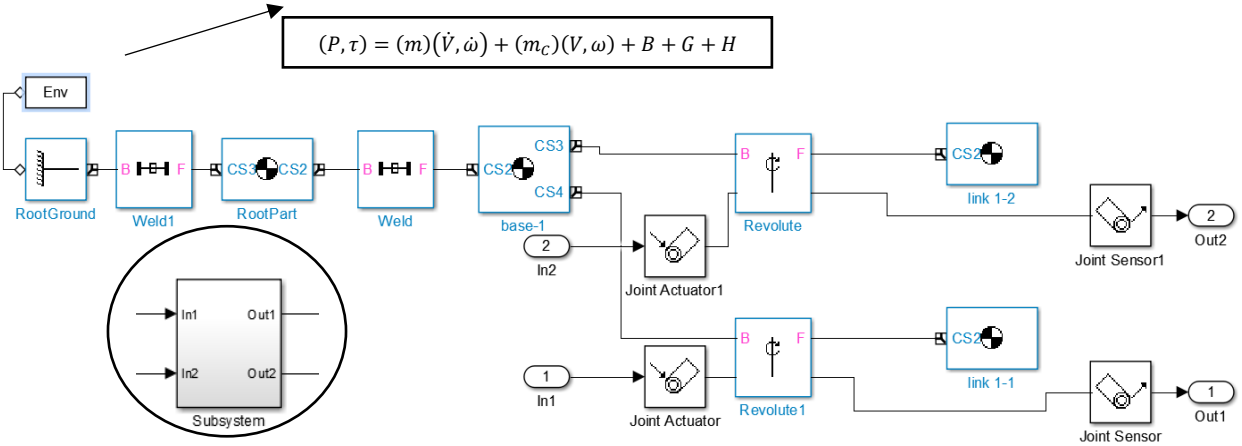


Figure 5. The sub-system of the arms with single links manipulator robot

Figure 5 depicts the creation of a subsystem before adding the input signal to it. Joint actuators have two installed source points, and further two linked output points are connected to the joint sensor by attaching it to the revolute joint. As a result, the subsystem and the slider gain are separated by the PID controller. The ITAE is used to measure the error between input and output, which is used to govern the entire system. This is the Absolute Error times the Integral of Time. For ITAE, the terms' Product, Integrator, Absolute, and Clock have been utilized. The desired output signal of the joint trajectory and end effector trajectory is produced using the scope, which is used for the output signal.

PID Controller

The PID controller operates as a comprehensive closed-loop feedback system, monitoring the difference between the system's measured output and the required set point. Using this error, the controller computes a control signal

to regulate the performance. The PID controller's behavior is governed by a differential Eq. (1-3), which can be expressed as follows:

The PID controller operates using proportional gain, integral gain, and derivative gain, which are expressed as K_p, K_i, and K_d, to achieve its functionality. The combination of these three components forms the basis for adjusting the performance of a process, as illustrated in Figure 6. The expression for the PID controller's operation is obtained by applying the Laplace transform.

Table 1. Factors used in simulation for 2-DOF robot with proposed hybrid controller [21].

S. No.	Factor	Notation	Values
1	Inertia Moment of the Base	IB	10 kgm²
2	PID parameters (Proportional)	K_p	64.432
	(Integral)	K_i	37.9
	(Derivative)	K_d	1.46
3	Damping of robot link	R_R	2.5N/s
4	Stiffness of Robot link	K_R	10^4 N/m
5	robot link Inertia	I_R	$2.13\times10^{-7}m^4$
6	Constant	A	0.1
7	desired joint rotation	D_t	0.5 rad

Evolutionary algorithms

Evolutionary Algorithms, drawing inspiration from biological models, present pioneering probabilistic research tools with immense potential for addressing challenges across various domains. These algorithms are particularly advantageous as they do not inherently assume characteristics of the underlying fitness landscape. This section focuses on optimization methods, namely the Whales Optimizer Algorithm, Ant Colony Optimization, and Particle Swarm Optimization.

Whale Optimization Algorithm (WOA)

The Whale_Optimization_Algorithm (WOA) is an innovative metaheuristic optimization algorithm that emulates the hunting patterns of humpback whales [23]. What sets this approach apart is its unique simulation of random hunting behaviour, where the best search agent is chosen to pursue prey, the hunting strategy involves three sequential steps, integrating a spiral technique to mimic the bubble-net attack behaviour observed in humpback whales as depicted in Figure 7.

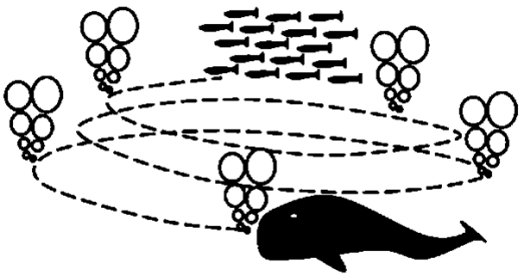


Figure 7: Illustration of the WOA [23].

Encircling prey circumstances where the exact location of the best solution within the search space is unclear, the Whale Optimization Algorithm (WOA) operates on the premise that the current solution is in close proximity to the desired one, akin to how humpback whales encircle prey upon sensing their presence, making it the most

effective search agent. This behaviour, which is regulated by equations that define distance (D) and the revised position (X), can be characterized as follows:

$$D = |CX^*(t) - X(t)| \quad (6)$$

$$X(t+1) = X^*(t) - AD \quad (7)$$

In this context't' signifies the present repetition. 'X_best' denotes the position vector of the best solution discovered up to this point, while 'X_t' represents the position vector being assessed in Eq. (6 and 7). The notation '|' denotes the complete value operation. It's significant to note that 'X_best' should be efficient at each iteration if a superior solution is discovered. 'A_vector' and 'C_vector' are coefficient vectors determined. D_i is defined as the disparity between X_i*(t) and X_i(t), where X_i^*(t) denotes the distance of the i^{th} whales from the target, which signifies the optimal solution obtained thus far. The parameter 'b' determines the configuration of the logarithmic curved, while '1' is a casually generated value within [-1, 1]. Subsequently, we will discuss the exploration and exploitation phases, and the results are shown in Figure 8. The tuned parameters obtained are K_p = 0.7847, K_i = 0.9961, and K_d = 0.3061 for Joint 1 and K_p = 0.4586, K_i =62.42, and K_d = 0.4561 for Joint 2. Which are shared characteristics of both ACO and PSO.

Ant Colony Optimization (ACO)

Ant_Colony_Optimization_(ACO) excels at solving a wide range of optimization issues. An emergent element of ant cooperation is the cooperation of a colony of artificial ants to provide useful solutions. Ant algorithms are adaptable and dependable and may be used to address a wide range of issues because they are based on ant colonies seen in nature. Artificial ants employ a sequence of local motions to decide the shortest route between two sites. Then, utilizing only local data, they apply a stochastic decision procedure to find the optimum option. The following are the three key characteristics of artificial ants: They cooperatively dwell in colonies, they quietly leave pheromones to communicate covertly, and they employ a variety of regional behaviours. The amount of pheromone deposited by an ant reflects the quality of the maneuver that is performed. Therefore, the more pheromone there is, the enhanced the solution is. When an ant discovers an explanation, it expires and is eliminated from the organization. The tuning parameters are estimated using Eq. 5 and the results are shown in Figure 8. The tuned parameters obtained are K_p = 0.002, K_i =321, and K_d = 1.46 for Joint 1 and K_p = 90.58, K_i =65.43, and K_d = 61.78 for Joint 2.

Particle swarm optimization (PSO)

Everyone is mentioned as a particle and signifies a candidate solution in the population-based search process known as the PSO method. Each particle moves through the search space at an adjustable velocity that is dynamically changed based on its own and the other particles' flying histories. Each particle in PSO seeks to become better by adopting the qualities of its successful peers. Additionally, because each particle has a memory, it can recall the best point in the search space that it has ever visited. The position with the best fitness is called P_{best}, and the particle with the best overall performance across the population is known as G_{best}. The resulting PSO algorithm yielded the tuned PID parameter, which is represented in Figure 8 as K_p= 64.432, K_i= 37.9, K_d= 81.15 for Joint 1 and K_p = 53.2, K_i =35.1, and K_d = 43.9 for joint 2.

Dynamic Model of Underwater Robot

The underwater manipulator vehicle system's overall dynamic model is provided by [22]

$$(P, \tau) = (m)\left(\dot{V}, \dot{\omega}\right) + (m_C)(V, \omega) + B + G + H \quad (10)$$

Where, ((P, τ) are external forces and torques, (m) are matrices for mass and additional mass inertia, (m_c) are matrices for centripetal and Coriolis forces, G: The gravity force matrix; B: The buoyancy force; and H: The hydrostatic pressure.

The hydrostatic force is caused by the water pressure exerted on the UWR, while the hydrodynamic forces are composed of three categories: propulsion or thruster forces, environmental factors (such as ocean currents, waves, and wind), and radiation-induced forces and moments, which include additional mass. The current work does not take environmental influences into account. Robots move at a depth of 10 meters, thus surface waves have no impact on them. The combined effects of drag, friction, fluid acceleration, and buoyancy can be used to calculate

the net hydrodynamic force acting on a body. Assumption considered are that the fluid is stationary, unbounded, and irrotational. The body's volume centre, also known as the center of buoyancy, is affected by the buoyancy force. This force is represented as

$$f_b = \begin{pmatrix} 0 \\ 0 \\ -\rho g v \end{pmatrix}$$

If the centre of gravity of the body and the centre of buoyancy do not align, where ρ represents fluid density, υ denotes body volume, and g represents gravitational acceleration, then there is a buoyancy torque given as

$$\tau_b = r_b \times f_b$$

The concepts of gravity and hydrostatic force are described with respect to the Earth's stationary reference frame

$$f_g = \begin{pmatrix} 0 \\ 0 \\ G \end{pmatrix} \text{ And } \quad f_{\mathrm{h}} = \begin{pmatrix} 0 \\ 0 \\ \rho g A_s \end{pmatrix}$$

Where, G=mg and as are identify the areas of the arm subjected to water pressure. In the same way, gravity torque is

$$\tau_g = r_g \times f_g$$

Simulation Result

In this segment, we're focusing on implementing a PID controller for trajectory tracking control of a robot manipulator, leveraging optimization techniques outlined earlier. The objective is to determine the optimal parameters (K_p, K_i, and K_d) that result in the most favorable performance metrics, including rise time, overshoot, settling time, and peak response. Our primary focus is on utilizing a novel optimization algorithm (WOA) to fine-tune the PID parameters and showcasing its effectiveness compared to two established algorithms (ACO and PSO)[24]. The optimization algorithm starts with randomly initialized parameters and aims to minimize the Integral of Time-weighted Absolute Error (ITAE) criterion, as defined in Eq. (16), iteratively till achieving the optimum set of (K_p, K_i, and K_d) parameters. Initially, we adjust the PID controller utilizing three Algorithms while experimenting with different agent or population sizes: 10, 20, 30, 50, 100, and, 200. Following this, determine the optimal population size for each algorithm based on its performance.

$e = q_{ref} - q$, is the discrepancy between the robot's actual position and its reference position. Where q_ref, is the reference position q is the actual position of underwater robot manipulator. In this investigation, MATLAB/Simulink is employed for conducting simulation experiments. Figures 4 and Figure 5 showcase the Simulink diagrams representing the control strategy implemented for a two-joint robot manipulator. The simulation configuration encompasses a robot manipulator system, two controllers, a specified trajectory for tracking, and optimization methodologies. The figures depict the intended and achieved positions of the initial and subsequent joints of the robot manipulator, managed through distinct optimization algorithms. Initial conditions are set as q1 (0) = 0.5 deg, q2 (0) = 1 deg, q̇1 (0) = 0, and q̇2 (0) = 0. Simulations are conducted for a time range of 10 seconds with a reference signal magnitude of 10 degrees. In Table 2, it's evident that using 200 agents for PSO optimization yields the lowest objective function (7.94), resulting in the minimum rise time for both joints. Table 3 reveals that ACO optimization performs weakly with a value of 6.464 for only 20 agents. However, GWO shows improvements over PSO in settling times and cost functions for both joints. Meanwhile, WOA achieves the minimum cost function (5.24) with 10 agents, although 30 agents seem to provide the best results overall (6.97). Controller parameter values derived from the optimization process are listed in Table 4. WOA exhibits superior efficiency in computing time compared to GWO and PSO, converging faster to the optimum, as shown in Table 6. Figure 7 and Figure 8 depict the oscillatory nature of the first joint's response across all various strategies for optimization and diverse population sizes. The relative overshoot varies, with ACO-PID and WOA-PID showing significant overshoots. The second joint's response is almost aperiodic, with minor overshoots across all algorithms. Furthermore, the first joint's response converges more quickly towards the final value with ACO and WOA optimizations, while PSO-P ID takes longer. Similarly, the second joint exhibits faster convergence with WOA optimization.

Table 2 Evaluation results of the response characteristics of the 1st and 2nd joints controlled using PSO-PID.

	Nbre_agents	Rise time	Settling-time	Overshoot	Peak	ITAE
Joint 1	10	0.2854	1.5005	2.0262	10.259	5.491
	20	0.3607	0.6160	1.8318	10.199	6.712
	30	0.2548	0.4352	1.8573	10.424	5.501
	50	0.378	0.6601	0.915	10.19	6.123
	10^2	0.136	0.4323	6.804	10.86	2.895
	2×10^2	**0.185**	**1.2986**	**2.601**	**10.31**	**3.876**
Joint 2	10	0.430	3.294	2.899	10.43	6.784
	20	0.3151	2.914	3.1923	10.443	3.786
	30	0.4703	3.454	3.1235	10.63	6.495
	50	0.5410	4.430	3.87	10.344	7.868
	10^2	0.386	3.340	3.162	10.63	6.011
	2×10^2	0.3145	2.6854	3.231	10.443	4.601
	Joint 1 and Joint 2					
Nbre_Agents	10	20	30	50	10^2	2×10^2
Objective fn.	12.596	10.891	11.991	13.546	9.01	**8.012**

*the bold indicated the finest values

Table 3 Assessment results of the response characteristics of the 1st and 2nd joints controlled using ACO-PID

	Nbre_agents	Rise_time	Settling_time	Overshoot	Highest	ITAE
Joint 1	10	0.079	0.489	15.230	11.45	2.564
	20	**0.149**	**0.449**	**7.851**	**10.828**	**3.157**
	30	0.0792	0.454	13.152	11.423	1.999
	50	0.1749	0.728	3.512	10.401	3.456
	10^2	0.0899	0.3989	8.657	10.796	2.289
	2×10^2	0.1898	0.4562	2.213	10.199	3.222
Joint 2	10	0.4699	2.212	1.898	10.355	7.123
	20	**0.343**	**0.579**	**0.4099**	**10.109**	**3.307**
	30	0.2899	0.621	1.498	10.199	5.679
	50	0.451	3.655	2.669	10.286	6.893
	10^2	0.555	2.253	2.041	10.31	6.768
	2×10^2	0.4326	4.1356	4.4989	10.506	6.654
	Joint 1 and Joint 2					
Nbre_Agents	10	20	30	50	10^2	2×10^2
Objective_fn.	8.99	**6.464**	6.825	13.412	9.081	**9.541**

*the bold showed the finest values

Table 4 Assessment results of the response characteristics of the 1st and 2nd joints controlled using WOA-PID

	Nbre agents	Rise time	Settling time	Overshoot	Peak	ITAE
Joint 1	10	0.099	0.510	20.963	12.321	2.031
	20	0.256	0.40	0.563	10.09	3.921
	30	**0.2103**	**0.2756**	**1.2365**	**10.20**	**3.678**
	50	0.1234	0.556	17.45	11.81	2.496
	10^2	0.2012	0.4562	3.053	10.43	3.768
	2×10^2	0.0801	0.469	20.21	12.31	2.23
Joint 2	10	0.350	0.601	0.403	10.79	3.21
	20	0.496	3.399	2.698	10.43	7.39
	30	**0.325**	**0.6136**	**0.3987**	**10.090**	**3.299**
	50	0.415	0.736	0.632	10.31	4.098
	10^2	0.367	0.6021	1.296	10.23	5.01
	2×10^2	0.519	0.856	1.136	10.51	6.989
	Joint 1 and Joint 2					
Nbre_agents	10	20	30	50	10^2	2×10^2
Objective_fn	5.125	11.562	**6.98**	6.171	8.610	9.312

*the bold implied the finest values

Table 5 PSO-PID, ACO-PID, and WOA-PID parameters

Algorithm/PID Parameters	K_p		K_i		K_d	
	Joint 1	Joint 2	Joint 1	Joint 2	Joint 1	Joint 2
PSO	4×10^3	3×10^3	9×10^2	9×10^2	463	450
ACO	4×10^3	3×10^3	6.4×10^2	2.1×10^2	348	458
WOA	4×10^3	3×10^3	3.51×10^2	2.1×10^2	454	450

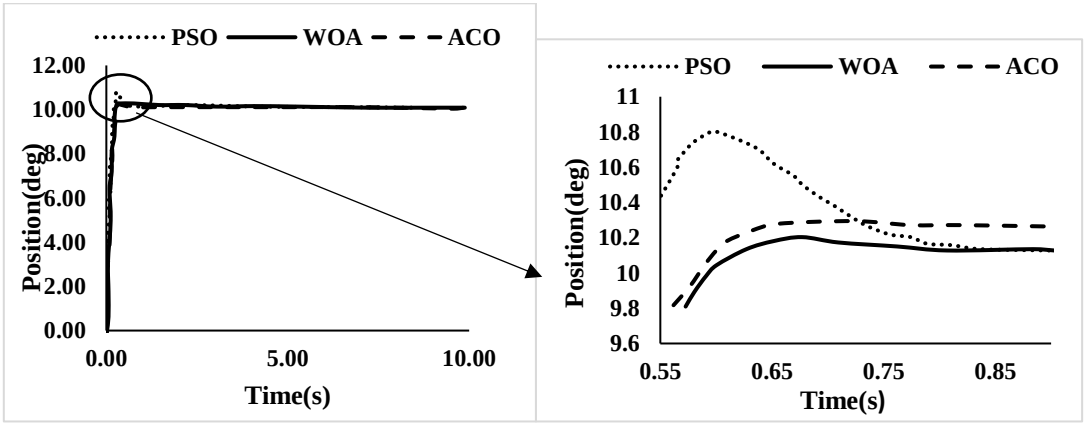
Table 6 Convergence time of 3- processes

	PSO	ACO	WOA
Convergence_time/ Nbre_Agents	$09min58s/200$	$2.0236s/20$	$1.412s/30$
	53.456s/20	1.901s/20	$0.8992s/20$

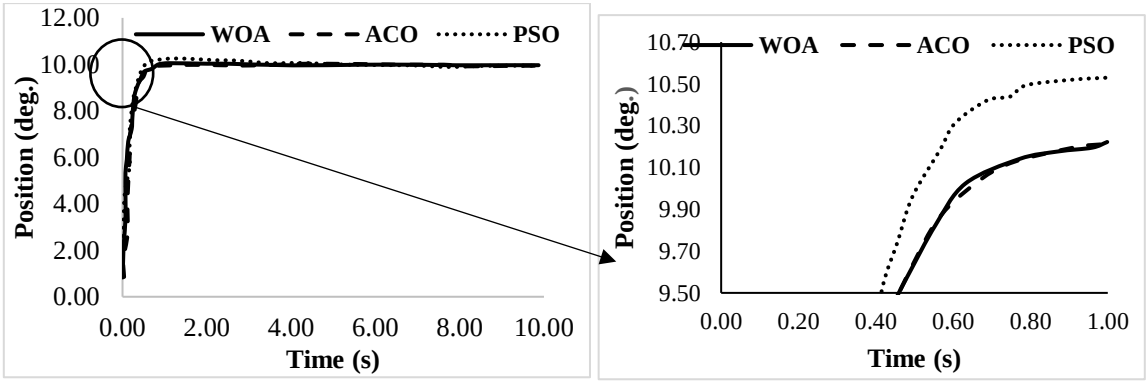
Manipulator Robot Control

To better assess the effectiveness of The WOA algorithm was implemented to enhance the trajectory tracking of an underwater robot manipulator, in conjunction with PID control. Three different algorithms were employed for tuning purposes: PSO with 200 agents, ACO with 20 agents, and WOA with 30 agents. As depicted in Figure 8, both the WOA-PID and ACO-PID controllers outperform the PSO-PID controller, showcasing their superior

performance. Notably, the behaviour of the second joint closely resembles that of both WOA-PID and ACO-PID controllers, while significant performance enhancements are observed in the first joint of the robot. Significantly, the WOA algorithm enhances the settling time of the first joint in comparison to ACO tuning, reducing it from 0.45 seconds to 0.27 seconds, while also minimizing overshoot from 7.85% to 1.4%. As a result, the system meets the settling time criteria and exhibits accelerated responses with the WOA-PID controller. Figures 9A and 9B showcase the convergence rate of the error through the WOA algorithm, affirming its competitive edge. Furthermore, Figure 9C demonstrates the WOA algorithm's ability to reach the optimum with reduced iterations. In Figure 10, the cost function (ITAE) is presented for both scenarios [25].

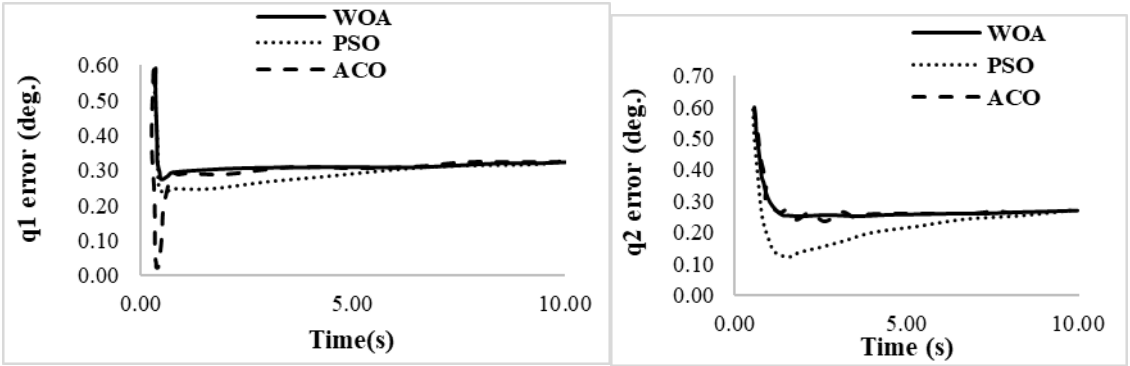


(a) **Joint 1**

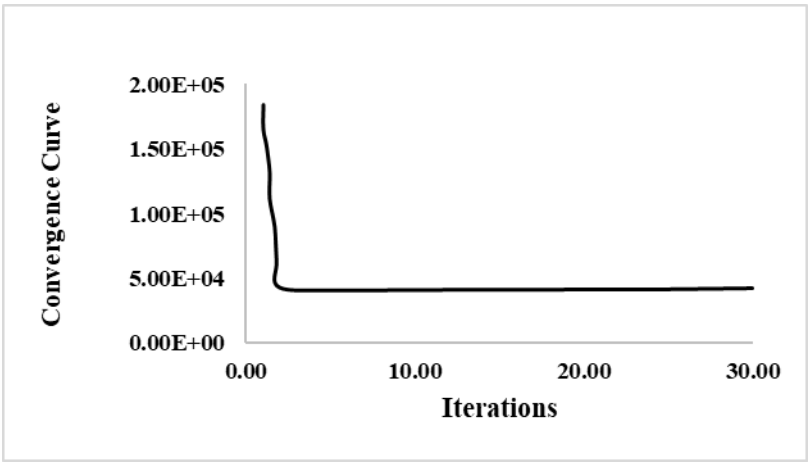


(b) Joint 2

Figure 8 : Optimal Joint Placement under Three Algorithmic Controller for joint 1 and Joint 2.



(a) **(b)**



(c)

Figure 9: a) and (b) are Error Signals for the Initial and Secondary Articulations following Step Consign, (c) Alongside Convergence Curve of the WOA Algorithm,

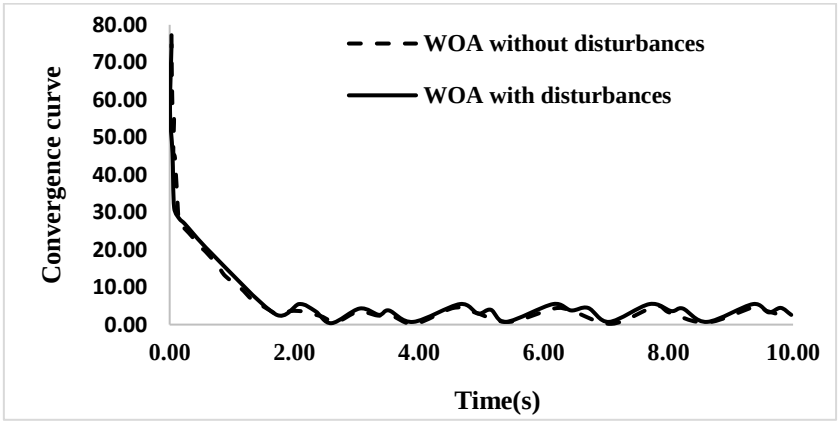


Figure 10: Cost fn. (ITAE) of the WOA algorithm

Conclusion

This paper aims to explore a novel application of the Whale_Optimizer_Algorithm (WOA) in fine-tuning PID parameters for tracking control of two arms with single link each having 2-degree-of-freedom underwater robot manipulator. Results demonstrate the efficacy of the proposed WOA-PID approach in improving settling time, reducing errors, and enhancing convergence time, while also showcasing its robustness in handling disturbances during robot tracking control. As future research directions, it is suggested that modifying the objective function and evaluating the controller's performance on robots with higher degrees of freedom. Additionally, comparative studies with alternative control techniques.

References

[1] Bingul, Z., & Gul, K. (2023). Intelligent-PID with PD Feedforward Trajectory Tracking Control of an Autonomous Underwater Vehicle. Machines, 11(2), 300.

[2] Yadav, S., Kumar, S., & Goyal, M. (2022). PID Tuning and Stability Analysis of Hybrid Controller for Robotic Arm Using ZN, PSO, ACO, and GA.

[3] Yadav, S., Kumar, S., & Goyal, M. (2024). Integration and Optimization of Software to Control Robotic Arms: A Comprehensive Study on Modeling, Hardware Implementation, and PID Tuning. International Journal of Intelligent Systems and Applications in Engineering, 12(15s), 240-248.

[4] Su Y, Muller PC, Zheng C (2010) Global asymptotic saturated PID control for robot manipulator. IEEE Trans Control Syst Technol 8(6):1280–1288

[5] Abdelhedi F, Bouteraa Y, Chemori A, Derbel N (2014) Nonlinear PID and feed forward control of robotic manipulators. In: 15th International conference on sciences and techniques of automatic control and computer engineering (STA), pp 349–354

[6] Bascetta L, Rocco P (2007) Revising the robust control design for rigid robot manipulator. In: International conference on robotics and automation. IEEE, pp 4478–4483
[7] Kasac J, Novakovic B, Majetic D, Brezak D (2006) Global positioning of robot manipulator with mixed revolute and prismatic joints. IEEE Trans Autom Control 51:1035–1040
[8] Su YX, Zheng CH (2017) PID control for global fnite-time regulation of robotic manipulators. Int J Syst Sci 48:547–558
[9] Mendoza M, Zavala-rio A, Santibanez V, Reyes F (2015) A generalized PID-type control scheme with simple tuning for the global regulation of robot manipulators with contrained inputs. Int J Constr 88:1995–2012
[10] Akyuz IH, Yolacan E, Ertunc HM, Bingul Z (2011) PID and state feedback control of single-link fexible joint robot. In: International conference on mechatronics. IEEE, pp 409–414.
[11] McLain, Timothy W., Stephen M. Rock, and Michael J. Lee. "Experiments in the coordinated control of an underwater arm/vehicle system." Underwater Robots (1996): 139-158.
[12] Petillot, Y. R., Antonelli, G., Casalino, G., & Ferreira, F. (2019). Underwater robots: From remotely operated vehicles to intervention-autonomous underwater vehicles. IEEE Robotics & Automation Magazine, 26(2), 94-101.
[13] Wang, Y., Jiang, S., Yan, F., Gu, L., & Chen, B. (2017). A new redundancy resolution for underwater vehicle–manipulator system considering payload. International Journal of Advanced Robotic Systems, 14(5), 1729881417733934.
[14] Mirjalili, S., & Lewis, A. (2016). The whale optimization algorithm. Advances in engineering software, 95, 51-67.
[15] Ghamari, S. M., Mollaee, H., & Khavari, F. (2021). Robust self-tuning regressive adaptive controller design for a DC–DC BUCK converter. Measurement, 174, 109071.
[16] Pan, L., Bao, G., Xu, F., & Zhang, L. (2018). Adaptive robust sliding mode trajectory tracking control for 6 degree-of-freedom industrial assembly robot with disturbances. Assembly Automation, 38(3), 259-267.
[17] Zarrin, A., Azizi, S., & Aliasghary, M. (2020). A novel inverse kinematics scheme for the design and fabrication of a five degree of freedom arm robot. International Journal of Dynamics and Control, 8, 604-614.
[18] Özdemir, M. T., & Öztürk, D. (2017). Comparative performance analysis of optimal PID parameters tuning based on the optics inspired optimization methods for automatic generation control. Energies, 10(12), 2134.
[19] ŞEN, M. A., & KALYONCU, M. (2018). Optimal tuning of PID controller using grey wolf optimizer algorithm for quadruped robot. Balkan Journal of Electrical and Computer Engineering, 6(1), 29-35.
[20] Kumar, S., Rastogi, V., & Gupta, P. (2014). Trajectory Control of Single Arm Underwater Flexible Welding Robot using Bond Graphs. In The Proceedings of the 2014 International conference on bond graph modelling and simulation (ICBGM'2014), Simulation Series (Vol. 46, No. 8, pp. 79-84).
[21] Kumar, S., Kumar, S., & Singh, C. D. (2015). Modeling and simulation of underwater flexible manipulator as Raleigh beam using bond graph. World Academy of Science, Engineering and Technology, International Science Index, 104, 1457-1460.
[22] Kumar, S., Kumar, S., & Singh, C. D. (2015). Modeling and simulation of underwater flexible manipulator as Raleigh beam using bond graph. World Academy of Science, Engineering and Technology, International Science Index, 104, 1457-1460.
[23] Mirjalili S, Lewis A (2016) the whale optimization algorithm. Adv Eng Softw 95:51–67.
[24] Paredes-Sanchez, M. C., Jimenez-Nixon, D. A., & Ordoñez-Avila, J. L. (2022, October). Underwater Robot Design Proposed Method Based on CAD and CFD. In 2022 IEEE Central America and Panama Student Conference (CONESCAPAN) (pp. 1-6). IEEE.
[25] Kumar, S., Rastogi, V., & Gupta, P. (2016). Recent developments in modeling and control of underwater robot manipulator: A review. Indian J Sci Technol, 9, 48.