

Article

# Cross-Coupled Dynamics and MPA-Optimized Robust MIMO Control for a Compact Unmanned Underwater Vehicle

Ahsan Tanveer <sup>1</sup>  and Sarvat Mushtaq Ahmad <sup>2,3,\*</sup>

<sup>1</sup> Department of Mechanical & Aerospace Engineering, Institute of Avionics & Aeronautics, Air University, Islamabad 44000, Pakistan; ahsantanveer3883@gmail.com

<sup>2</sup> Control & Instrumentation Engineering Department, King Fahd University of Petroleum and Minerals, Dhahran 31261, Saudi Arabia

<sup>3</sup> Interdisciplinary Research Center for Intelligent Manufacturing and Robotics, King Fahd University of Petroleum and Minerals, Dhahran 31261, Saudi Arabia

\* Correspondence: sarvat.ahmad@kfupm.edu.sa

**Abstract:** A compact, 3-degrees-of-freedom (DoF), low-cost, remotely operated unmanned underwater vehicle (UUV), or MicroROV, is custom-designed, developed, instrumented, and interfaced with a PC for real-time data acquisition and control. The nonlinear equations of motion (EoM) are developed for the under-actuated, open-frame, cross-coupled MicroROV utilizing the Newton-Euler approach. The cross-coupling between heave and yaw motion, an important dynamic of a class of compact ROVs that is barely reported, is investigated here. This work is thus motivated towards developing an understanding of the physics of the highly coupled compact ROV and towards developing model-based stabilizing controllers. The linearized EoM aids in developing high-fidelity experimental data-driven transfer function models. The coupled heave-yaw transfer function model is improved to an auto-regressive moving average with exogenous input (ARMAX) model structure. The acquired models facilitate the use of the multi-parameter root-locus (MPRL) technique to design baseline controllers for a cross-coupled multi-input, multi-output (MIMO) MicroROV. The controller gains are further optimized by employing an innovative Marine Predator Algorithm (MPA). The robustness of the designed controllers is gauged using gain and phase margins. In addition, the real-time controllers were deployed on an onboard embedded system utilizing Simulink's automatic C++ code generation capabilities. Finally, pool tests of the MicroROV demonstrate the efficacy of the proposed control strategy.

**Keywords:** open-frame compact ROV; cross-coupled mathematical models; MPA optimization; MIMO ROV control; multi-parameter root locus



**Citation:** Tanveer, A.; Ahmad, S.M. Cross-Coupled Dynamics and MPA-Optimized Robust MIMO Control for a Compact Unmanned Underwater Vehicle. *J. Mar. Sci. Eng.* **2023**, *11*, 1411. <https://doi.org/10.3390/jmse11071411>

Academic Editor: Fausto Pedro García Márquez

Received: 5 June 2023  
Revised: 21 June 2023  
Accepted: 28 June 2023  
Published: 14 July 2023



**Copyright:** © 2023 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (<https://creativecommons.org/licenses/by/4.0/>).

## 1. Introduction

Advancements made in the areas of unmanned underwater vehicles, including autonomous (AUV) and remotely operated (ROV) vehicles, enabling them to conduct a variety of activities, have been quite significant. AUVs and ROVs are improving their abilities to carry out a wide range of missions with accuracy and efficiency. Seafloor mapping, remote environmental research, pollution assessment and monitoring, and marine security are just a few of the common applications of these vehicles [1]. However, this progress is driven primarily by the use of costly, sophisticated sensors, propulsion systems, esoteric structural materials, and embedded electronics. Furthermore, the development or acquisition cost of off-the-shelf vehicles [2–5] dictates their utility. Thus, producing these types of vehicles is hugely expensive and beyond the means of many developing countries. An attempt is made in this work to design and develop a low-cost proof-of-concept demonstrator platform named MicroROV. This will serve as a test bed to develop dynamic models, control strategies, and algorithms for a slightly bigger ROV, which is also under development.

The motivation for this work is to custom design and develop a low-cost ROV for shallow-water applications such as underwater structure inspection and water quality monitoring in rivers and canals. As well as serving as a test bed for gaining insight into vehicle dynamics and evaluating different control strategies. The experience gained will thus feed into ongoing development work on a rugged version of the ROV with a similar vehicle configuration to that of the MicroROV. The compact MicroROV designed and developed has an open frame configuration and is propelled by three thrusters. Such a configuration is inherently complex and under-actuated with cross-coupling amongst its various degrees of freedom and uncertainties in hydrodynamic coefficients, making it a complex modeling and control problem. Nonetheless, such compact ROVs hold promise for the water quality and submerged underwater structure monitoring applications at hand.

The nonlinear dynamics of torpedo- or cigar-shaped flight vehicles are investigated by Martin et al. [6], Petrich et al. [7], Cao et al. [8], and Cozijn et al. [9] in their respective works. An exception to this is the Pluto-Gigas ROV [10], which has pitch-heave cross-coupling. An interesting aspect of symmetric torpedo-shaped vehicles is their decoupled lateral and longitudinal dynamics, akin to aircraft. Thereby rendering subsequent model-based control of this type of vehicle a single-input, single-output (SISO) problem. On the other hand, open-frame UUVs like the one investigated in this work can be configured with an even number of thrusters for actuating the heave behavior so that the lateral and longitudinal dynamics remain decoupled. Some examples of decoupled heave-yaw dynamics in open-frame ROVs are Thetis [11], Garbi [12], Romeo [13], BlueROV2 [14], and Hybrid-ROV [15]. More recent work on an underwater robot named 'Intelligence Ocean-I' [16], although described as an open-frame vehicle, has stabilizing rear wings similar to those of a flight vehicle. The vehicle comprises two counter-rotating magnetically coupled heave thrusters and two thrusters mounted on rear wings for propulsion. Given a pair of thrusters for heave and yaw degrees of freedom, the authors have conveniently assumed that the heave and yaw motion are decoupled.

However, for a compact-class vehicle such as MicroROV with two thrusters for surge motion and just a single thruster for heave, there is necessarily strong heave-yaw dynamic coupling. That is, as the vehicle descends or ascends, the MicroROV also spins rapidly. This occurs due to the absence of an additional counter-rotating thruster to compensate for the torque generated by the single thruster for the heave motion. The presence of an umbilical cord further alters the vehicle's behavior. Although there are modeling similarities between aerial and underwater vehicles, wireless communication frees' the unmanned aerial vehicles (UAVs) of tether encumbrances [17]. On the other hand, micro or mini ROVs have to rely on tethers, while larger ROVs have to resort to complex acoustic and ultrasonic underwater communication systems. Thus, obtaining high-fidelity models representing these complex dynamics is challenging. Detailed mathematical and parametric modeling, including that of the cross-coupled dynamics as well as the influence of the umbilical cord, are the main contributions of this work.

The presence of cross-coupling renders MicroROV a difficult multi-input-multi-output (MIMO) control problem. Classical single-input, single-output (SISO) control strategies using PID have been tried over the years. In the past few years, there has been widespread utilization of intelligent optimization techniques across different domains, aimed at enhancing the effectiveness of classical controllers [18]. Folcher et al. [19], for instance, assessed the performance of a traditional root-locus heave controller. Walker et al. [20] investigated a PID controller for ROV station keeping. Rúa et al. [21] examined a PID controller for ROVs, tuned using heuristic methods. For the most part, feedback controllers are built individually for each body axis, with little consideration given to coupling between degrees of freedom [22]. On the other hand, MIMO control strategies such as sliding mode [23,24], nonlinear backstepping control of an open frame hybrid ROV [15], and  $\mu$ -synthesis for the Marvin modular AUV [25] have been investigated, albeit in simulations. In contrast, this work not only presents classical control design with robustness analysis along with

MPA-tuned MIMO control designs in simulation but also demonstrates the real-time control of the MicroROV in a pool environment, which is another important contribution of this work.

#### *Main Contributions of the Work*

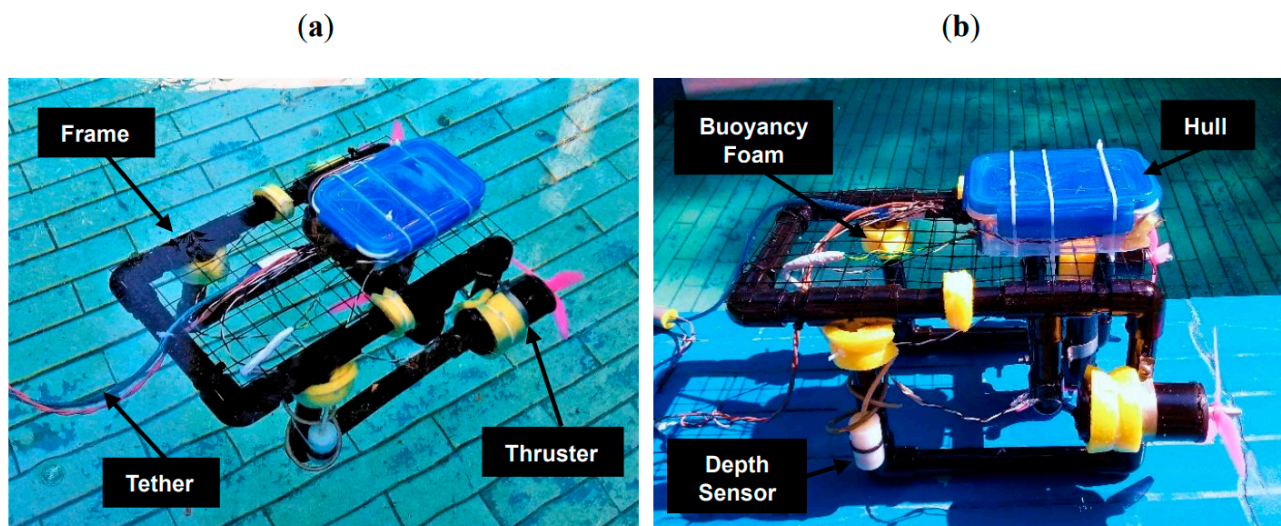
The main contributions of this work are thus:

1. Custom design, development, and demonstration of a compact MicroROV that can achieve precise yaw and heave control despite cross-coupling effects (Section 2);
2. Development of detailed linear EoM for yaw and heave dynamics as well as for normal force to yaw coupled dynamics (Section 3);
3. Explicit identification of the influence of the umbilical cord on the vehicle stiffness parameter, which is generally neglected in the classical Newton-Euler approach to modeling (Section 4);
4. Development of data-driven system identification techniques to arrive at high-fidelity dynamic models (Section 4);
5. Design of a robust MIMO control algorithm employing a novel MPA optimization technique as well as demonstration of real-time implementation of the same on the MicroROV (Section 6).

The following is a breakdown of the article's structure: Section 2 details the experimental setup used in this study. A coupled model for the MicroROV is outlined in Section 3. A data-driven model of the vehicle and its validation are discussed in Section 4. Section 5 investigates the base-line PI controller design. The development of a control algorithm employing a novel MPA optimization technique is the focus of Section 6. The paper concludes with Section 7.

## 2. MicroROV Vehicle Description

It is tough to imagine the development of a ROV and its many subsystems, including the electronic, mechanical, and communication systems, as distinct constructs. As shown in Figure 1, once the design process is finished, each subsystem is tested before integration to produce an operational MicroROV ready for outdoor testing. The interconnection of the ROV's various subsystems is shown in Figure 2.



**Figure 1.** An illustration of the MicroROV being tested in a pool environment; (a) top view, (b) side view.

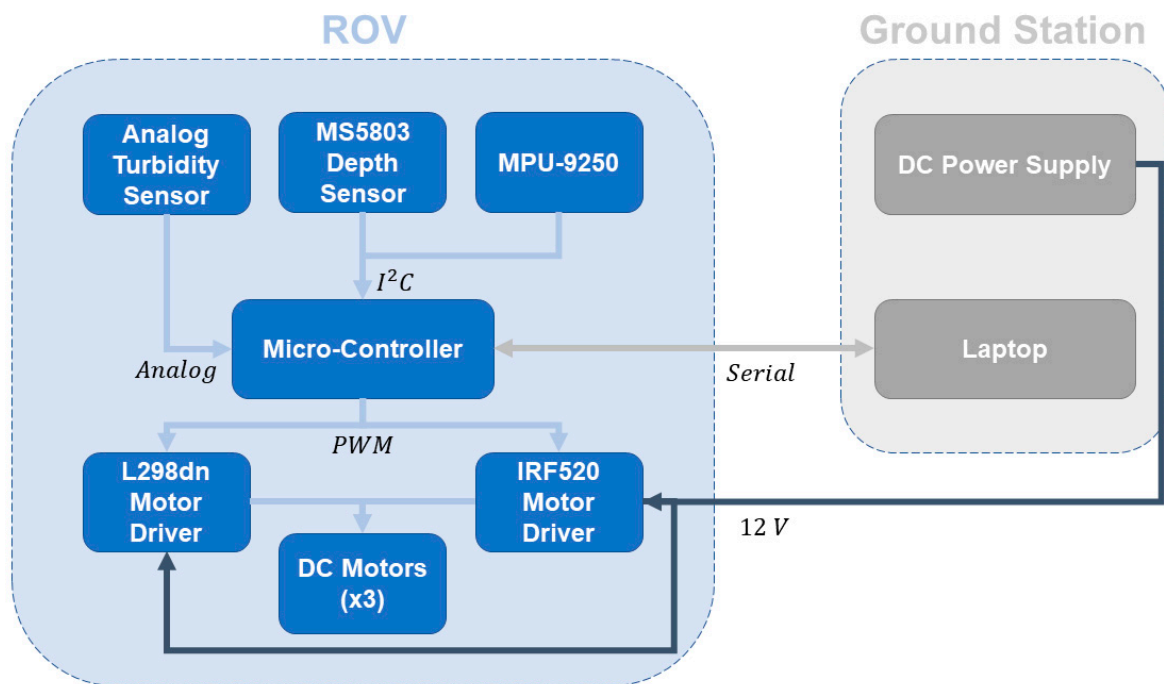


Figure 2. Interacting sub-systems of the MicroROV.

As shown in Figure 2, the electronic subsystem of the MicroROV, which consists of a microcontroller, two motor drivers, a turbidity sensor, and an inertial measurement unit (IMU), is housed in a watertight hull. The ROV’s thrusters, consisting of one downward-facing thruster and two side-mounted thrusters, are one of the mechanical subsystem’s main components. Sinking and surfacing are accomplished by the thruster that faces downward up to a depth of 10–15 m. The vehicle cruise speed of 0.2 to 0.3 m/s roughly describes the operational speed. Through the tether, the ground station-based PC’s power and control signals are delivered to the on-board computer, which runs a C++ program. The only directly controllable DoFs with the current vehicle setup are surge, heave, and yaw. However, because the purpose of developing a MicroROV is to demonstrate its control abilities for inspecting underwater structures, yaw and heave motion will be examined in this article under the influence of heave-yaw coupling. The specification of the vehicle is given in Table 1.

Table 1. MicroROV specifications.

Parameter	Value
Weight of the Hull	0.26 kg
Weight of the ROV	1.68 kg
Weight of Eq. water	0.34 kg
Yaw moment of inertia (Analytical)	0.02 kgm <sup>2</sup>
Yaw moment of inertia (Experimental)	0.019 kgm <sup>2</sup>
Distance between the center of mass and the thruster	0.11 m

### 3. Coupled Math Model of the ROV

Three translational and three rotational motions define the MicroROV’s total six degrees of freedom. Surge, heave, and sway make up the threedegrees of translational motion, whereas yaw, pitch, and roll make up the threedegrees of rotational motion. The motion of the ROV along the x, z, and y axes (comprising the body frame of the vehicle) is denoted by the terms surge, heave, and sway, respectively. The rotational motion of the



ROV about the  $z$ ,  $y$ , and  $x$  axes is represented similarly by the terms yaw, pitch, and roll. The body-fixed frame is coincident with the vehicle's center of gravity, while the earth-fixed world frame lies outside the vehicle. The vehicle coordinate system is shown in Figure 3. The standard 6-DoF notation for an underwater vehicle can be seen in Table 2.

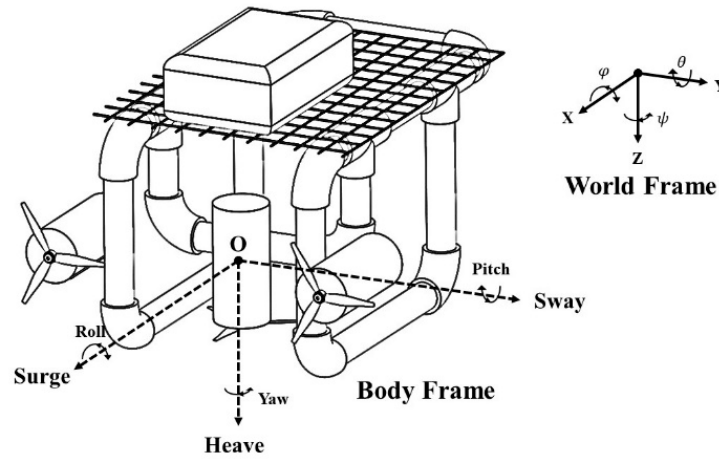


Figure 3. Micro-ROV coordinate system.

Table 2. 6-DOF standard underwater vehicle notation.

DOF	Motions	Forces and Moments	Linear and Angular Velocities	Positions and Euler Angles
1	Surge	$X$	$u$	$x$
2	Sway	$Y$	$v$	$y$
3	Heave	$Z$	$w$	$z$
4	Roll	$L$	$p$	$\varphi$
5	Pitch	$M$	$q$	$\theta$
6	Yaw	$N$	$r$	$\psi$

Since the MicroROV is an under-actuated (i.e., with fewer actuators than degrees of freedom to be controlled) vehicle with heading and heave dynamics to be modeled and controlled, Equations (1) and (2) suffice to model the vehicle's coupled nonlinear dynamics [26].

$$m [\dot{w} - uq + vp + x_G(pr - \dot{q}) + y_G(qr + \dot{p}) - z_G(p^2 + q^2)] = Z \tag{1}$$

$$I_z \dot{r} + (I_y - I_x)pq - I_{xy}(p^2 - q^2) - I_{yz}(pr - \dot{q}) + I_{xz}(qr - \dot{p}) + m[x_G(\dot{v} + ur - wp) - y_G(\dot{u} - vr + wq)] = N \tag{2}$$

Utilizing the condition that the body-fixed frame coincides with the c.g. of the vehicle, Equation (2) is simplified as:

$$I_z \dot{r} + (I_y - I_x)pq - I_{xy}(p^2 - q^2) - I_{yz}(pr - \dot{q}) + I_{xz}(qr - \dot{p}) = N \tag{3}$$

Equations (1) and (3) represent non-linear equations of motion for ROV in heave and yaw. Here  $Z$  is the summation of all the normal forces acting on the vehicle in heave, including thrust and dissipative hydrodynamic damping forces. While  $N$  is the summation of all the external yaw moments acting on the MicroROV about the center of gravity.  $x_G$ ,  $y_G$  and  $z_G$  are rectangular components of the vector linking body-frame to world-frame. Similarly,  $u$ ,  $v$ ,  $w$  expresses translational while  $p$ ,  $q$ ,  $r$  represent angular velocities along  $x$ ,  $y$ ,

and z-axis.  $I_x, I_y, I_z$  represent moment of inertia of the ROV in the  $x, y,$  and  $z$ -plane.  $I_{xy}, I_{yz}, I_{xz}$  refers to the moment of inertia induced in one axis due to motion in the other axis, for example,  $I_{xy}$  represents moment of inertia induced along  $x$ -axis due to motion along the  $y$ -axis.

Rearranging Equations (1) and (3) yields:

$$m\dot{w} = Z + g(p, q, r) \tag{4}$$

$$I_z\dot{r} = N + h(p, q, r) \tag{5}$$

Here  $g(p, q, r)$  and  $h(p, q, r)$  are coupled dynamics affecting heave and yaw, respectively, and are a function of linear and angular velocities. Due to the presence of a single thruster for heave motion with no counter-rotating torque producing complimentary thrust, the ROV spins while it descends. This corresponds to dynamic coupling between heave and yaw motions. The linear mathematical model, i.e., heave and yaw dynamics, of the MicroROV, assuming vehicle symmetry and small perturbations, is as follows [26]:

Heave Dynamics:

$$m\dot{w} + F_d^w = F_t \tag{6}$$

$$m\ddot{z} + b_{heave}^z\dot{z} = F_t \tag{7}$$

Yaw Dynamics:

$$I_z\dot{r} + \tau_d^r = \tau_{rear} \tag{8}$$

$$I_z\ddot{\psi} + b_{yaw}^{\dot{\psi}}\dot{\psi} = \tau_{rear} \tag{9}$$

$$I_z\ddot{\psi} + b_{yaw}^{\dot{\psi}}\dot{\psi} = 2lF_{diff} \tag{10}$$

The authors presented detailed decoupled dynamics in [26], but the MicroROV also has strong coupling between the heave and yaw planes. The cross-coupling effect, if not addressed, inevitably impacts the desired closed-loop performance. This is developed next, wherein Equation (11) depicts the nonlinear coupled heave and yaw dynamics. The coupling of yaw dynamics with heave due to  $\tau_d^w$ —normal velocity ( $w$ ) dependent drag torque is given by Equations (12) and (13).

$$I_z\dot{r} = N + h(p, q, r) \tag{11}$$

$$I_z\dot{r} + \tau_d^r + \tau_d^w = \tau_{rear} \tag{12}$$

$$I_z\ddot{\psi} + b_{yaw}^{\dot{\psi}}\dot{\psi} + b_{heave}^z\dot{z} = \tau_{rear} \tag{13}$$

Rendering Equations (7) and (8) into the Laplace domain yields decoupled transfer functions of heave and yaw dynamics:

$$\frac{z(s)}{F_t(s)} = \frac{1}{ms^2 + b_{heave}^z s} \tag{14}$$

$$\frac{\psi(s)}{F_{diff}(s)} = \frac{2l}{I_z s^2 + b_{yaw}^{\dot{\psi}} s} \tag{15}$$

Similarly, combining the Laplace transforms of Equation (7) and Equation (13) yields a linearized coupled transfer function for yaw output  $\psi$  due to vertical thrust force  $F_t$  input:

$$\frac{\psi(s)}{F_t(s)} = \frac{b_{yaw}^z}{I_z m s^3 + (b_{heave}^z + m b_{yaw}^\psi) s^2 - (b_{yaw}^\psi b_{heave}^z) s} \tag{16}$$

Here,

- $\tau_d^r$ —Yaw rate ( $r$ ) dependent drag torque
- $\tau_d^w$ —Normal ( $z$ -axis) velocity ( $w$ ) dependent drag torque
- $\tau_{rear}$ —Vehicle main propulsive torque
- $F_t$ —Thrust force due to vertical thruster
- $F_{diff}$ —Differential rear thruster force
- $l$ —Moment arm from center of gravity to  $F_{diff}$
- $b_{heave}^z$ —Heave drag force due to normal velocity ( $\dot{z}$ )
- $b_{yaw}^\psi$ —Yaw drag moment due to yaw angular velocity ( $\dot{\psi}$ )
- $b_{yaw}^z$ —Coupled yaw drag moment due to heave velocity ( $\dot{z}$ )

#### 4. High-Fidelity Data-Driven Model

Often, for multi-input, multi-output (MIMO) systems, as demonstrated by Ahmad et al. [27], the cross-coupling amongst various degrees of freedom (DoFs) is too strong to ignore. The nature and strength of cross-coupling can very well be determined from an auto-correlation test of the residuals. An autocorrelation function (ACF) for a specific DoF outside the confidence bounds signifies strong cross-coupling. Furthermore, cross-coupling amongst DoFs of the vehicle affects the resulting vehicle model, as enunciated by Xu et al. [22] while demonstrating the effect of cross-coupling on the parameter estimation of an ROV. Therefore, to address the issue of cross-coupling, it is of utmost significance that analysis of the open-loop responses of the ROV be carried out in order to identify the nature and strength of coupling. In that respect, it is found that for the ROV under investigation, an input in yaw has no effect on the heave response. However, while analyzing real-time heave data, it was observed that the yaw motion of the vehicle is affected to the point that it cannot be neglected. Figure 4 illustrates a schematic highlighting the coupled nature of the vehicle under consideration.

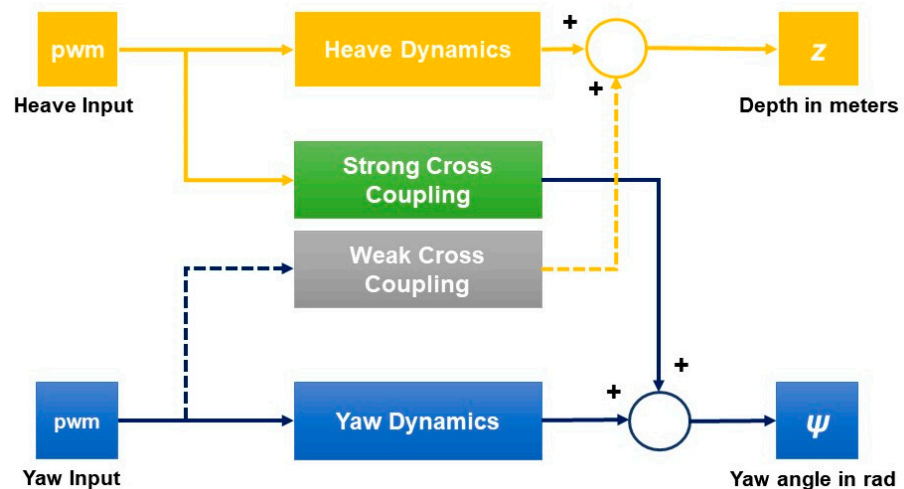


Figure 4. A schematic of ROV cross-coupling.

From Figure 4, it is clear that an input in heave introduces disturbance in yaw. Therefore, yaw is affected by more than one input, making the system multi-input, single-output

(MISO). In designing a control scheme for MISO systems, it is important to account for the disturbance rejection capabilities of the controller. Only a controller with a robust disturbance rejection feature can cater for the cross-coupling effect. However, to design a controller that delivers the desired performance, there must be a detailed dynamic model available for the system that is not only simple but also captures the system’s behavior adequately. The preceding subsection deals with the development of a high-fidelity model capturing the cross-coupling effect prominent in the ROV under consideration.

4.1. System Identification

System identification (SI), a black-box modeling technique to capture system dynamics, is commonly employed in control system engineering. The MATLAB System Identification Toolbox is used in this study to carry out linear model identification of the compact MicroROV. The effect of heave on yaw motion is investigated in the ongoing section.

To excite the modes of MicroROV, a signal rich in eigenfrequencies, such as a multi-step signal, is designed, and the vehicle is energized with this signal. Figure 5 shows the real-time yaw response of the vehicle to the multi-step input in heave. The real-time data obtained from the ROV during experimentation is utilized for model development.

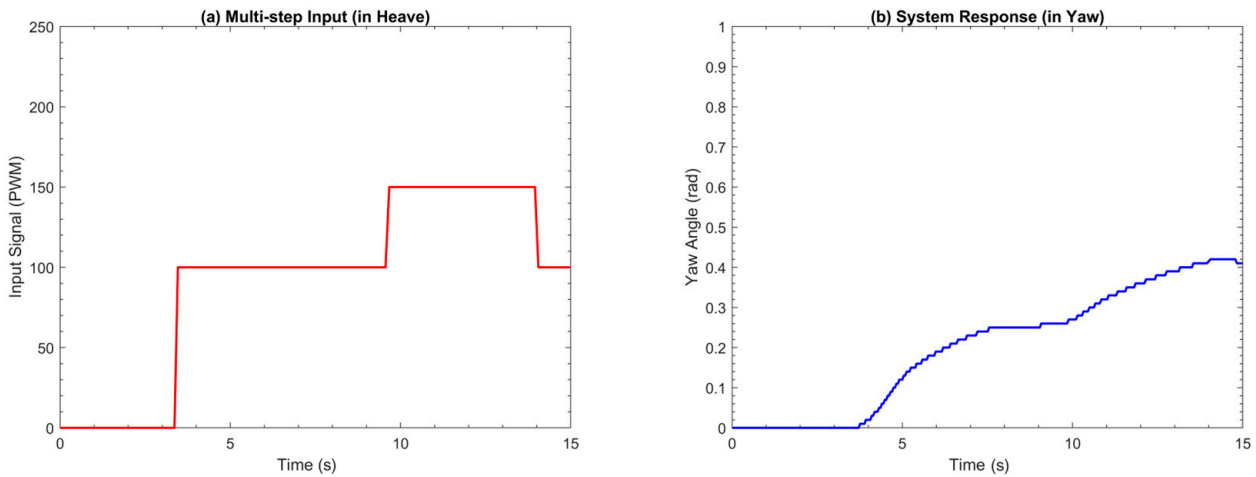


Figure 5. Experimental response showing cross-coupling of yaw and heave; (a) input signal, (b) output signal.

At first, a simple second-order transfer function (TF) model with three poles is investigated. It is found that the response of the selected model architecture fits the experimental data fairly accurately. The resulting 2-DoF coupled transfer function model, relating motion in yaw to an input in heave, is as follows:

$$\frac{\psi(s)}{F_t(s)} = \frac{0.01009}{s^3 + 3.475s^2 + 8.936s + 3.483} \tag{17}$$

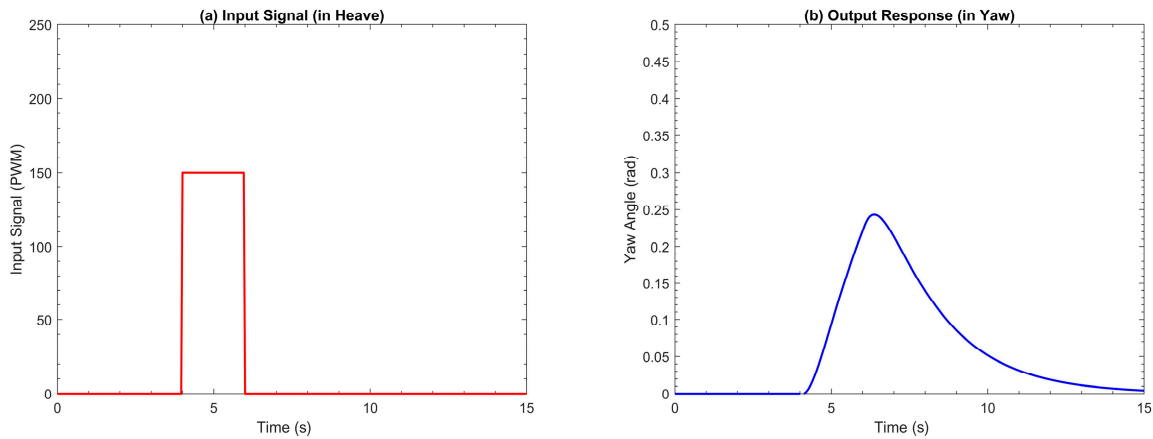
Here  $F_t(s)$  represents thrust force applied by the top thruster. From Equation (17), it is clear that the poles of the system lie at  $-1.5067 \pm 2.2967i$  and  $-0.4616 + 0.0000i$ . The position of the poles signifies that the model and, by extension, the output of the system in yaw is stable. Upon careful examination of Equation (17), it becomes evident that the identified transfer function (TF) model includes an extra term in the denominator polynomial when compared to the Newton-Euler model (Equation (16)). This additional term represents the stiffness factor resulting from the twisting and tangling of the tether caused by the vehicle’s yaw motion. Essentially, the twisted tether behaves like a coiled spring.

If the remotely operated vehicle (ROV) rotates in the same direction as the twisting of the tether, it experiences an assisting effect on its motion. However, if the ROV rotates



against the twisting, the tether exerts a recoiling force, attempting to return the vehicle to its original position despite the input provided. Consequently, this added stiffness term accounts for the dynamics of the tether, which are typically overlooked in the conventional Newton-Euler approach to modeling.

An open-loop response for the model in Equation (17) is shown in Figure 6. As evident, the model is stable exactly as predicted by pole location.



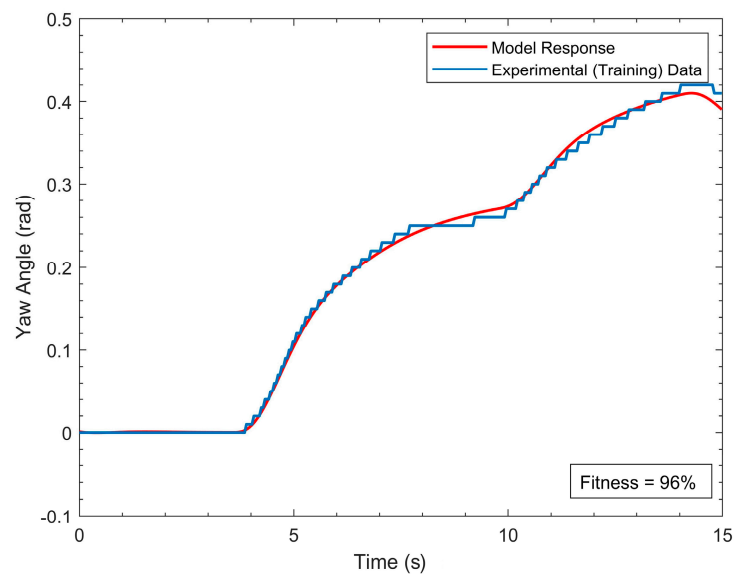
**Figure 6.** Input-output signal for 2-DoF coupled TF model; (a) input signal, (b) output signal.

#### 4.2. Model Validation

The model development process is followed by a model validation procedure. In actuality, model validation plays a significant role in the model-building process as a whole. The goal of model validation is to examine the model’s performance and accuracy. Self-validation, residual testing, and cross-validation are the three distinct model validation tests that are carried out.

##### 4.2.1. Model Self-Validation

Model self-validation is accomplished by benchmarking the model’s response against the training dataset. Figure 7 illustrates the output of a self-validation test. It is evident that the model can accurately reproduce the dynamics of the model from experimental data.



**Figure 7.** TF Model response comparison with training dataset.

#### 4.2.2. Residual Test

In time-domain SI, the residuals  $\epsilon(t)$  are analyzed to determine the best model architecture. Residuals indicate the difference in response between the model and the plant to the same stimulus. Correlation-based model validity tests are employed to verify that

$$e(t) \approx \epsilon(t) \tag{18}$$

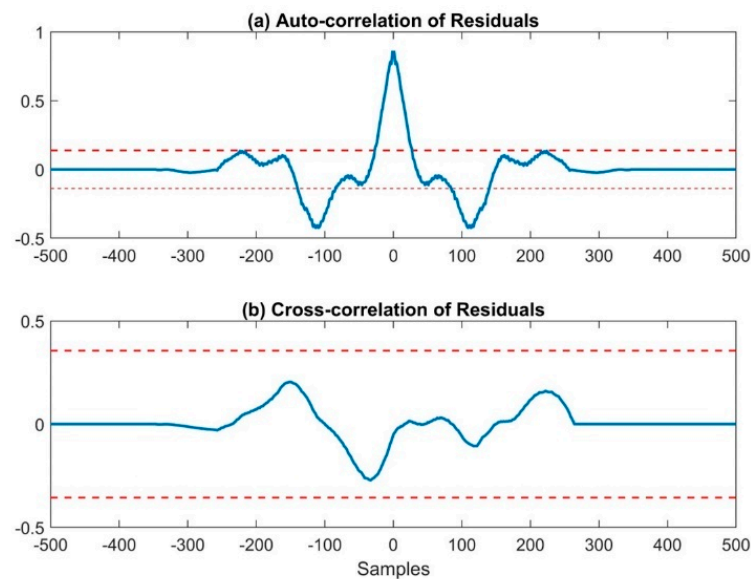
The relation in Equation (18) is satisfied when all correlation functions lie inside the confidence intervals. In other words

$$\phi_{\epsilon\epsilon}(t) = E[\epsilon(t - \tau)\epsilon(t)] \approx \delta(t) \tag{19}$$

$$\phi_{u\epsilon}(t) = E[u(t - \tau)\epsilon(t)] \approx 0 \quad \forall \tau \tag{20}$$

where  $\phi_{\epsilon\epsilon}(t)$  and  $\phi_{u\epsilon}(t)$  are the estimated auto-correlation functions of the residuals and the cross-correlation function between  $u(t)$  and  $\epsilon(t)$ , respectively.  $\delta(t)$  is an impulse function.

The autocorrelation (ACF) of residuals for the TF model is shown in Figure 8 to be outside the 99% confidence interval. This indicates that the noise, disturbance, or coupled dynamics are not well characterized. As a result, it is necessary to estimate a model that integrates all the aforementioned. However, as seen in Figure 8, the cross-correlation function (CCF) is well within the confidence band.



**Figure 8.** Residual test for transfer function model; (a) auto-correlation of residuals, (b) cross-correlation of residuals.

A model is deemed to reflect the system dynamics well if both ACF and CCF sit within the confidence band [27]. The fact that the ACF of residuals is outside the confidence interval implies that noise, disturbance, or coupled dynamics are not adequately modeled. Hence, it is proposed to use a model for estimation that can take noise and external disturbances into consideration, such as an auto-regressive moving average with exogenous input, or ARMAX model structure [28]. The development of an ARMAX model structure for coupled motion is the topic of the next subsection.

#### 4.3. Coupled ARMAX Model

In contrast to the transfer function (TF) model, the ARMAX model structure take into account the modeling of disturbances or colored noise. The disturbance is modeled as colored noise (mean  $\neq 0$ ), possessing its own dynamics that significantly affect the system

dynamics [29]. The tether, along with water currents and cross-coupling between different degrees of freedom, is considered one of the external disturbances in the problem at hand. Due to the presence of these disturbances and the significant heave-yaw coupling exhibited by the developed ROV, ARMAX models are particularly well-suited for addressing these issues [30,31].

A discrete-time ARMAX model has the following structure and belongs to the class of linear polynomial models:

$$A(z)y(k) = B(z)u(k - n) + C(z)e(k) \tag{21}$$

where  $y(k)$  denotes the system output,  $u(k)$  stands for the system input,  $n$  for the system delay, and  $e(k)$  for disturbance. The following are the system polynomials  $A(z)$ ,  $B(z)$ , and  $C(z)$ :

$$A(z) = 1 + a_1z^{-1} + a_2z^{-2} + \dots + a_kz^{-k} \tag{22}$$

$$B(z) = b_0 + b_1z^{-1} + b_2z^{-2} + \dots + b_{k-1}z^{-(k-1)} \tag{23}$$

$$C(z) = 1 + c_1z^{-1} + c_2z^{-2} + \dots + c_kz^{-k} \tag{24}$$

The order of the aforementioned polynomials relies on the estimated model's order; for a second-order model,  $k = 2$ .

The instrument variable (IV) approach is used to estimate an ARMAX model utilizing the data from Figure 5 as training data. Following several iterations, it was discovered that the ARMAX (3221) model's response closely matches the experimental response. System polynomials obtained from the model are listed below:

$$A(z) = 1 - 2.241z^{-1} + 1.552z^{-2} - 0.3099z^{-3} \tag{25}$$

$$B(z) = -3.201e^{-06}z^{-1} + 6.678e^{-06}z^{-2} \tag{26}$$

$$C(z) = 1 - 1.778z^{-1} + 0.8392z^{-2} \tag{27}$$

The aforementioned system polynomials yield the following transfer function:

$$\frac{\psi(s)}{F_t(s)} = \frac{-7.394e^{-05}s^2 - 0.003232s + 0.0734}{s^3 + 27.06s^2 + 62.66s + 24.94} \tag{28}$$

Equation (28) represents the transfer function for the coupled ARMAX model. The poles of the system lie at  $-24.547$ ,  $-2.0045$ , and  $-0.5068$ . The arrangement of the poles denotes the stability of the model. Figure 9 displays an open-loop response for the ARMAX model. The model behaves as the placement of the poles would suggest.

#### 4.4. ARMAX Model Validation

##### 4.4.1. Model Self-Validation

For model self-validation, the ARMAX model in Equation (28) is benchmarked against the training dataset. Figure 10 shows the results of the self-validation test. It is clear that the model can predict the dynamics of the model from the experimental data with a fair degree of accuracy.

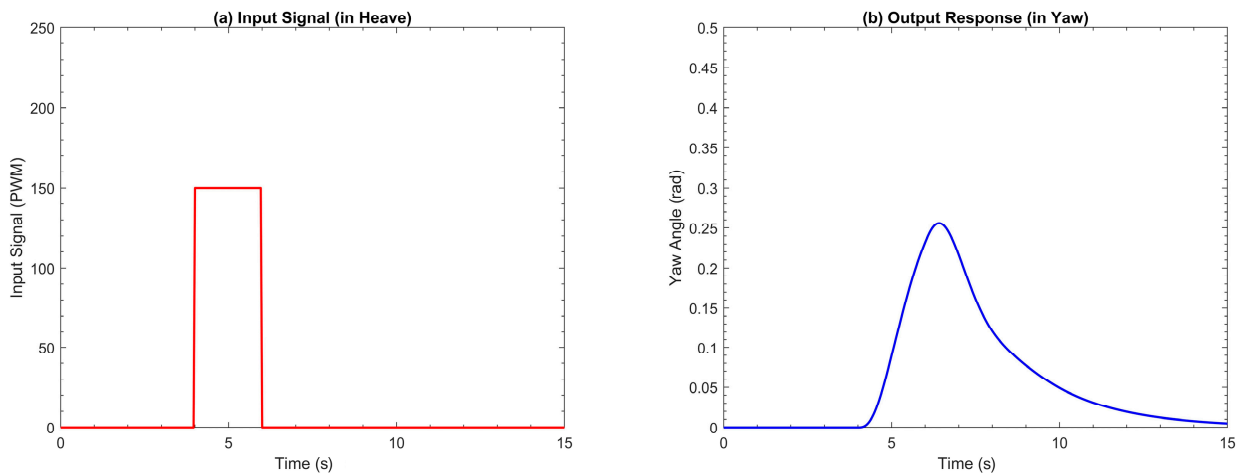


Figure 9. Input-output signal for 2-DoF coupled ARMAX model; (a) input signal, (b) output signal.

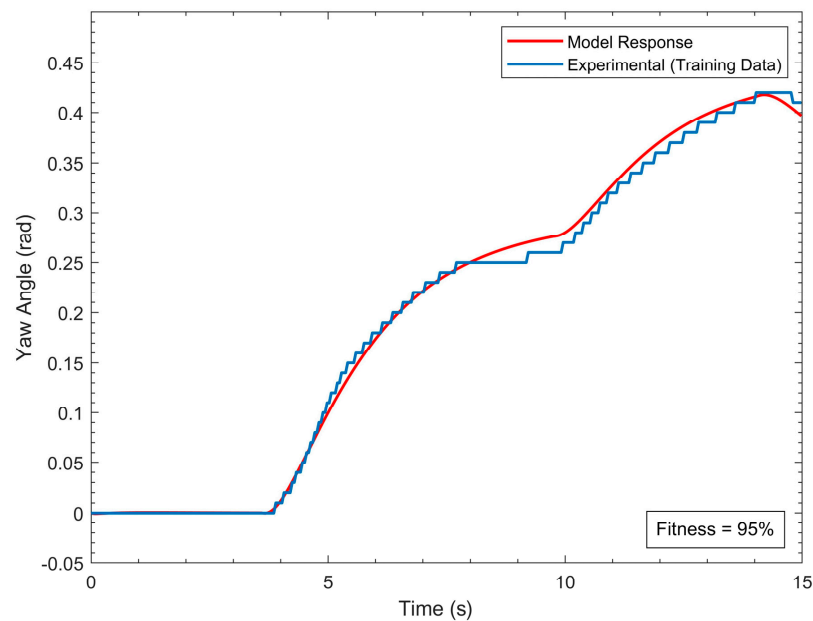


Figure 10. ARMAX Model response comparison with training dataset.

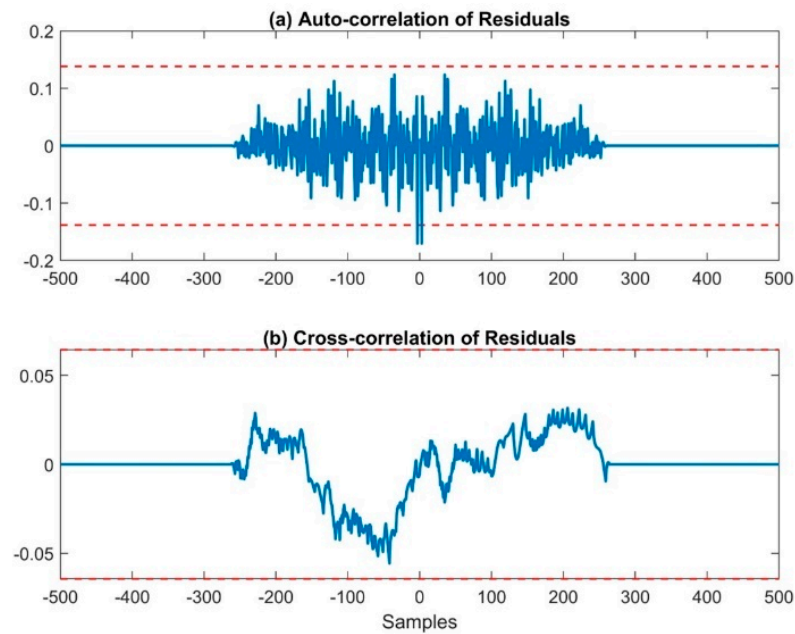
#### 4.4.2. Residual Test

The autocorrelation of residuals is shown in Figure 11 as being well within the 99% confidence band. This indicates that the system’s coupled nature is adequately modeled. Furthermore, the CCF also lies comfortably within the confidence interval. The cross-validation test may now be performed.

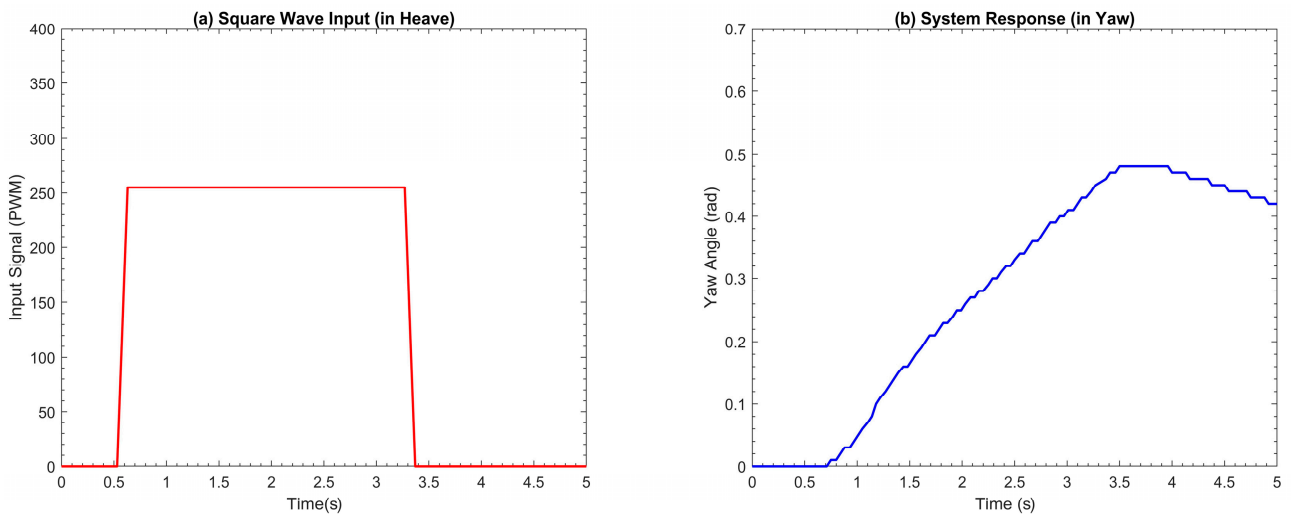
#### 4.4.3. Model Cross-Validation

Cross-validation is a model validation technique used to determine how well a projected model’s results generalize to a previously untested dataset. To test an identified model, cross-validation makes use of various independent datasets. Figure 12 displays the testing data used to validate the ARMAX model.



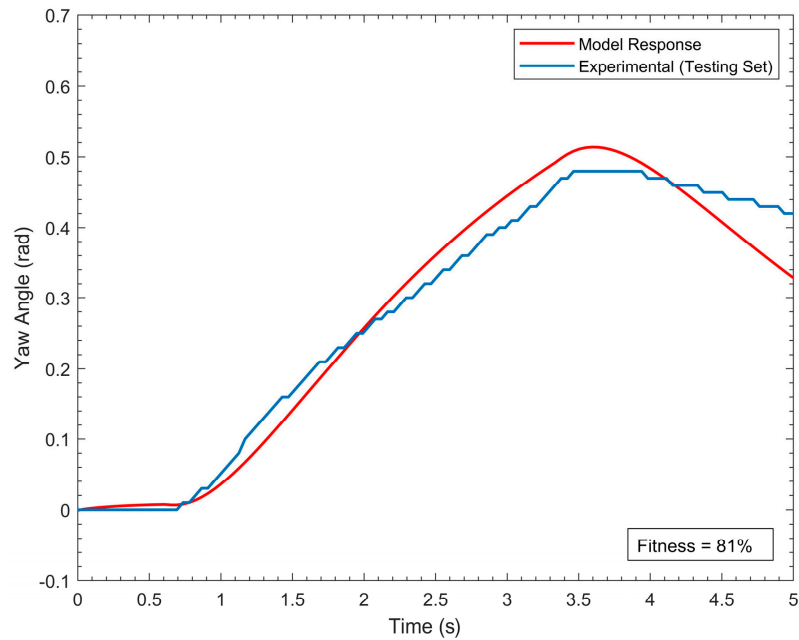


**Figure 11.** Residual test for ARMAX model; (a) auto-correlation of residuals, (b) cross-correlation of residuals.



**Figure 12.** Testing data for ARMAX model cross-validation; (a) input signal, (b) output signal.

The results of the self-validation test are shown in Figure 13. It is evident that the model can estimate the dynamics of the model in reasonable correspondence with the experimental results. As a result, the identified model is suitable for response simulation and control law formulation.



**Figure 13.** ARMAX Model evaluation against a distinct testing dataset.

Similarly, authors have developed high-fidelity data-driven models for yaw [26] and heave [32] using the same approach as above:

$$\frac{\psi(s)}{F_{diff}(s)} = \frac{6.653e^{-5}s + 0.01068}{s^2 + 1.563s + 0.3512} \tag{29}$$

$$\frac{z(s)}{F_t(s)} = \frac{0.001124s + 0.0007853}{s^2 + 0.4956s + 0.1273} \tag{30}$$

Here  $F_{diff}$  represents differential rear thruster force. It is worth mentioning that the coupled model obtained in Equation (28) is not utilized in closed-loop control design; instead, cross-coupling is countered using the intrinsic robustness of the independent yaw and heave controllers. Control law development is covered in the section that follows.

### 5. Baseline Controller Design

The primary objective of controller design is to enable a system to achieve the desired performance. To accomplish this, it is crucial to establish well-defined performance criterion prior to the controller design process. The effectiveness and performance of the resulting controller are evaluated based on its ability to meet predefined performance criteria. When considering the specific vehicle in question, both time-domain and frequency-domain analysis approaches are employed. To ensure a cautious approach, conservative performance requirements are established, taking into careful consideration the actuation capabilities of the ROV. The following performance requirements are defined:

1. Stability: The closed-loop system must be stable in yaw and heave;
2. Peak Time: The time taken for the response to reach its peak value should not exceed 3 s for yaw and 5 s for heave;
3. Overshoot: The maximum allowable overshoot should not exceed 20% for yaw and 25% for heave;
4. Settling Time: The settling time, defined as the time required for the response to reach and stay within a specified tolerance band, should be less than 5 s for yaw and 10 s for heave;
5. Gain and Phase Margin: While there are no universally accepted standards regarding the minimum requirements for gain and phase margins specifically for underwa-

ter vehicles, some researchers have proposed guidelines. Safonov et al. [33] and Lehtomaki et al. [34] suggest that ROVs should possess the following characteristics:

- a. 60° of phase margin in at least one DoF
- b. Infinite gain margin

The MicroROV has two modes of operation that must be simultaneously controlled: rotation in the x-y plane (yaw) and translation along the z-axis (heave). As previously indicated, both DoFs exhibit substantial coupling. The motion in yaw has a negligible impact on heave and may be disregarded. Heave, however, has a direct impact on the vehicle’s rotational velocities. Therefore, it is imperative that the yaw controller be built to reject disturbances brought on by the coupled heave motion. A schematic of the ROV’s control system is shown in Figure 14.

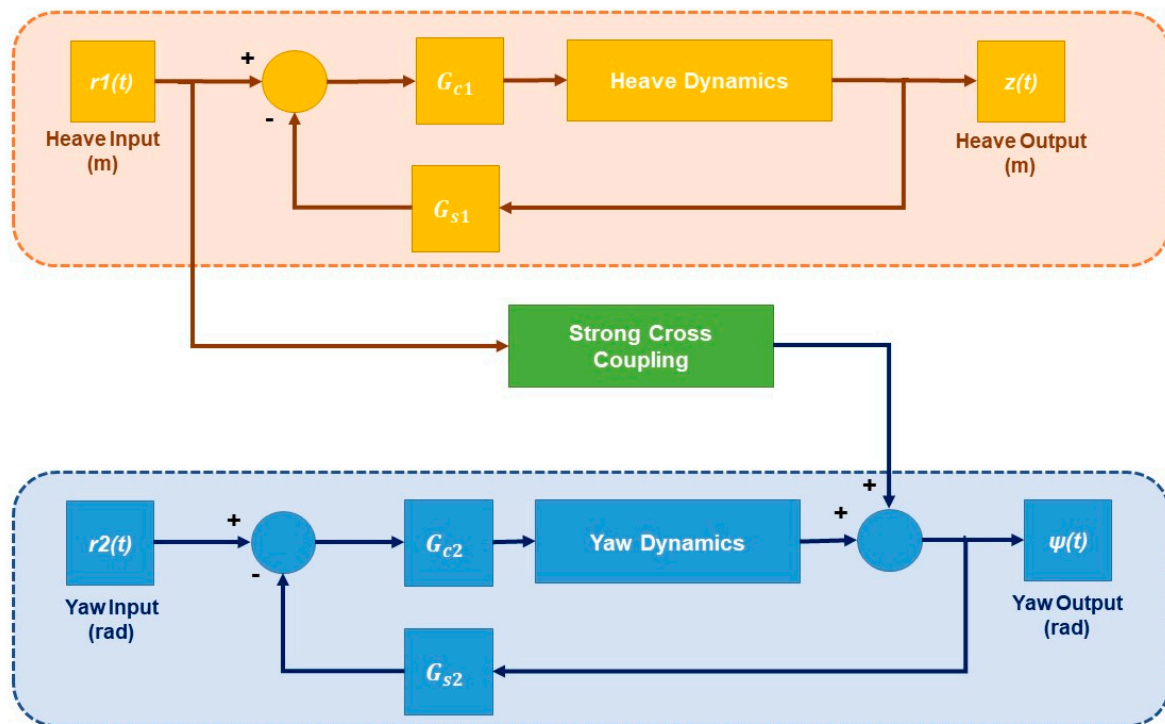


Figure 14. ROV closed-loop control schematic.

The transfer functions for the depth sensor and gyroscope are  $G_{s1}$  and  $G_{s2}$ , respectively, where  $G_{c1}$  and  $G_{c2}$  stand for the heave and yaw controllers. Since both sensors have sampling frequencies that are substantially higher than the vehicle’s bandwidth,  $G_{s1}$  and  $G_{s2}$  may be safely considered equal to unity for the sake of simplicity.

Independent development of a yaw and a heave controller employing the root locus method is carried out for the sake of MIMO closed-loop control design. To address the cross-coupling phenomenon, the resultant controllers are concurrently implemented on the vehicle in real-time. The root locus approach is typically employed for control law design due to its simplicity and usability. The next sub-section briefly explain a novel root-locus-based design of yaw and heave controllers.

*Multi-Parameter Root-Locus PI (MPRL-PI) Control*

The purpose of controller design is to attain the required performance. A controller’s capability to match the pre-established performance criteria serves as a measure of its efficiency and performance. The precise PID control architecture is defined once the requirements are specified. The determination of controller gains is all that is required in the design of a PID controller. Although there are other options, root-locus offers the most straightforward and user-friendly approach for controller tuning. However, the

traditional root-locus method is only capable of handling one variable parameter at a time. MPRL, on the other hand, enables simultaneous tuning of all the gains. The technique has been investigated by authors [26,32] for the compact ROV, albeit for the single-input, single-output case.

The closed-loop heave transfer function is as follows in accordance with the performance criteria:

$$G_{cl}^h(s) = \frac{a_1 k_p^h s^2 + (b_1 k_p^h + a_1 k_i^h) s + b_1 k_i^h}{s^3 + (a_1 k_i^h + c_1 + d_1 k_p^h) s^2 + (e_1 + b_1 k_p^h + b_1 k_i^h) s} \tag{31}$$

Here  $k_p^h$  and  $k_i^h$  are heave closed loop ( $G_{cl}^h$ ) controller gains while  $a_1 = 0.0006$ ,  $b_1 = 0.0008$ ,  $c_1 = 0.45$ ,  $d_1 = 0.006$  and  $e_1 = 0.144$ .

Figure 15 plots Equation (31). The figure shows  $k_p^h$  and  $k_i^h$  of 550 and 120, respectively. Upon plotting Equation (31) with the predicted gains of the model output, we get a response with a 3.8 s peak time, an overshoot of almost 26 percent, and a 10 s settling time.

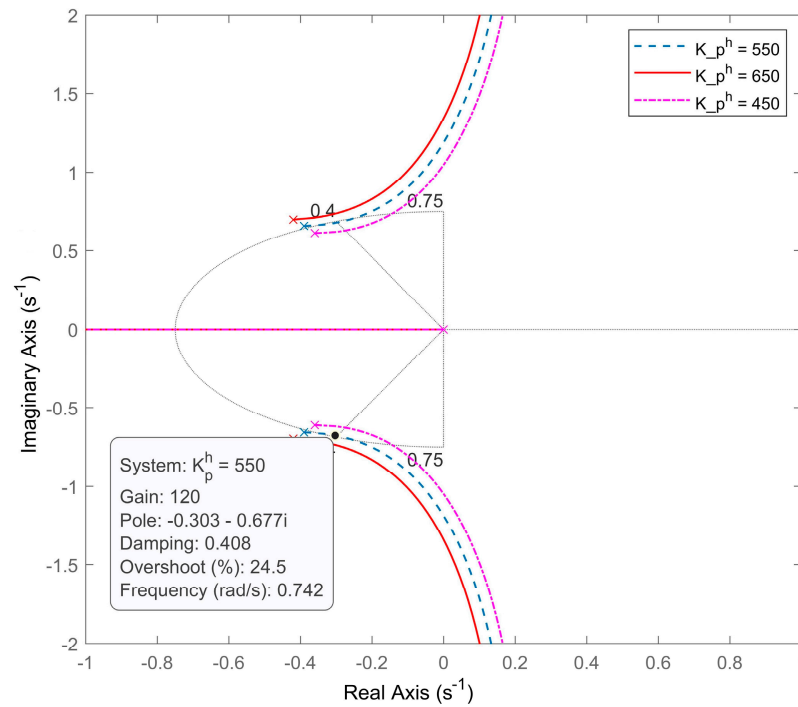


Figure 15. MPRL contour plot for heave closed-loop.

Similarly, the closed-loop transfer function for yaw is as follows:

$$G_{cl}^y(s) = \frac{a_2 (k_p^y + k_i^y)}{s^3 + b_2 s^2 + (c_2 + a_2 k_p^y) s + a_2 k_i^y} \tag{32}$$

Here  $k_p^y$  and  $k_i^y$  are yaw closed loop ( $G_{cl}^y$ ) controller gains while  $a_2 = 0.01394$ ,  $b_2 = 2.08$  and  $c_2 = 0.4681$ .

Figure 16 plots Equation (32). The figure shows  $k_p^y$  and  $k_i^y$  of 230 and 90, respectively. Upon plotting Equation (32) with the predicted gains, the output of the model exhibits a peak time of three seconds, an overshoot of almost 22 percent, and a 4.3-s settling time.



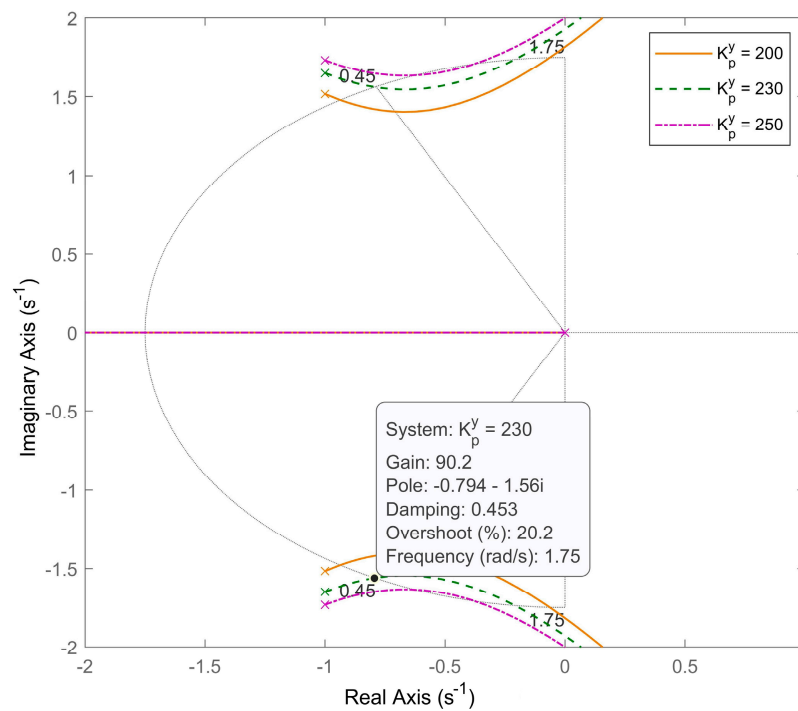


Figure 16. MPRL contour plot for yaw closed-loop.

After evaluating the controller’s performance against the predetermined criteria, it is evident that the MPRL-PI controller successfully fulfills the peak time and settling time requirements. However, it falls short of meeting the specified criterion for percentage overshoot. Furthermore, it is worth mentioning that the gains obtained via traditional PID tuning methods do not meet the criteria for optimality, as Ho et al. [35] showed. Instead, they tend to reduce parametric robustness. For precisely these reasons, a relatively new marine predator algorithm (MPA) is employed to optimize the controller’s gains.

### 6. Marine Predator Algorithm

The marine predator algorithm (MPA), a newly devised optimization method, is centered around the idea that both predators and their prey must regularly update their positions to track their respective sources of food [36]. Brownian and Lévy arbitrary motions are common scavenging tactics that predators use when interacting with their prey in the marine ecosystem. Predators utilize the Brownian approach when there is a large concentration of prey in the hunting region and the Lévy method when there is a low concentration. Environmental issues, including eddy formation and the effects of fish aggregating devices (FADs), are among the factors that change the behavior of marine predators. The effects of FADs are seen as entrapping in search space, while the FADs are considered local optima. The population member, i.e., predator or prey, who is in the strongest contrast with other members is selected as the top member and expressed through the elite matrix, in accordance with the survival of the fittest theory [37].

The MPA road map for PI controller tuning is shown in Figure 17. Setup parameters for MPA-based optimization of the heave and yaw controller are shown in Table 3.

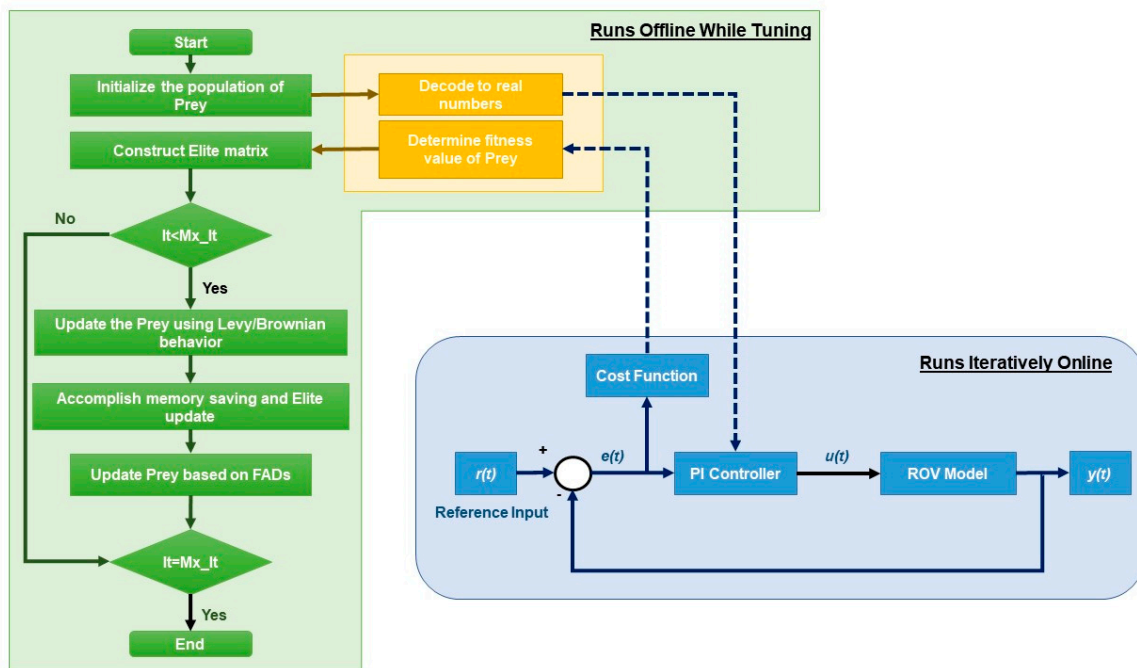


Figure 17. A simplified workflow for MPA-PI optimization problems.

Table 3. MPA problem setup parameters for PI controller optimization.

Parameter	Value (for Heave)	Value (for Yaw)
Search space bounds	[20, 400]	[50, 300]
Number of agents/prey	50	50
Maximum iterations	50	50
Dimensions/variables	2	2
Cost function	Integral Time Absolute Error (ITAE)	ITAE

The optimization problem is executed ten times using the parameters listed in Table 3. Heave and yaw controllers are determined from the optimization problem to have proportional gains of 400 and 270 and integral gains of 104 and 68, respectively. Plotting the system’s response with the gains from MPA optimization demonstrates that it has superior performance metrics than the root-locus-tuned controller. Table 4 offers a direct performance comparison between both controllers.

Table 4. Time-domain performance comparison of the designed controllers.

	Heave			Yaw		
	Desired Performance	Experimental		Desired Performance	Experimental	
		Root-Locus	MPA		Root-Locus	MPA
$T_p$ (in second)	5	3.8	4.4	3	3	1.18
$T_s$ (in second)	10	10.5	11.2	5	4.3	4.3
% OS	25	26	14.8	20	22%	18%

Here  $T_p$  represents peak time,  $T_s$  settling time, and % OS percentage overshoot. As depicted, the MPA-PI controller exhibits a peak time of 4.4 sec and a percentage overshoot

of 14.8% in the heave motion, both of which are significantly lower than the specified requirements of a 5 sec peak time and 25% overshoot. Similar observations can be made for the yaw motion. However, it is worth noting that the settling time in the heave motion is slightly longer than anticipated. This can be attributed to the influence of buoyancy effects experienced by the vehicle in heave, which restricts the ROV from settling rapidly. Furthermore, peak time and settling time for MPA are slightly higher than for MPRL; however, the overshoot characteristics of the latter are considerably higher. Nonetheless, Table 4 makes it clear that MPA-PI performs better in simulations holistically, which indicates that it will arguably perform better in real-time. However, it is important to determine the frequency-domain performance characteristics of the controllers before they can be implemented on the experimental hardware.

6.1. Robustness Analysis

It is challenging to represent a system’s real-world dynamics with absolute precision and accuracy. This is due to the fact that in the real world, systems are susceptible to external disturbances. Since ROVs function in a variable underwater environment, robustness is crucial for ensuring smooth operation. Robustness is typically ascertained by gain and phase margins. Although there is no established minimum gain or phase margin requirement for underwater vehicles, Lehtomaki et al. [34] and Safonov et al. [33] have indicated that ROVs must have (a) a 60° phase margin in at least one DoF and (b) an infinite gain margin.

Figure 18 shows a frequency response comparison for heave dynamics, root-locus-tuned PI, and MPA-PI, whereas the response for yaw is shown in Figure 19. Furthermore, Table 5 offers a quantitative performance comparison of both controllers in the frequency domain.

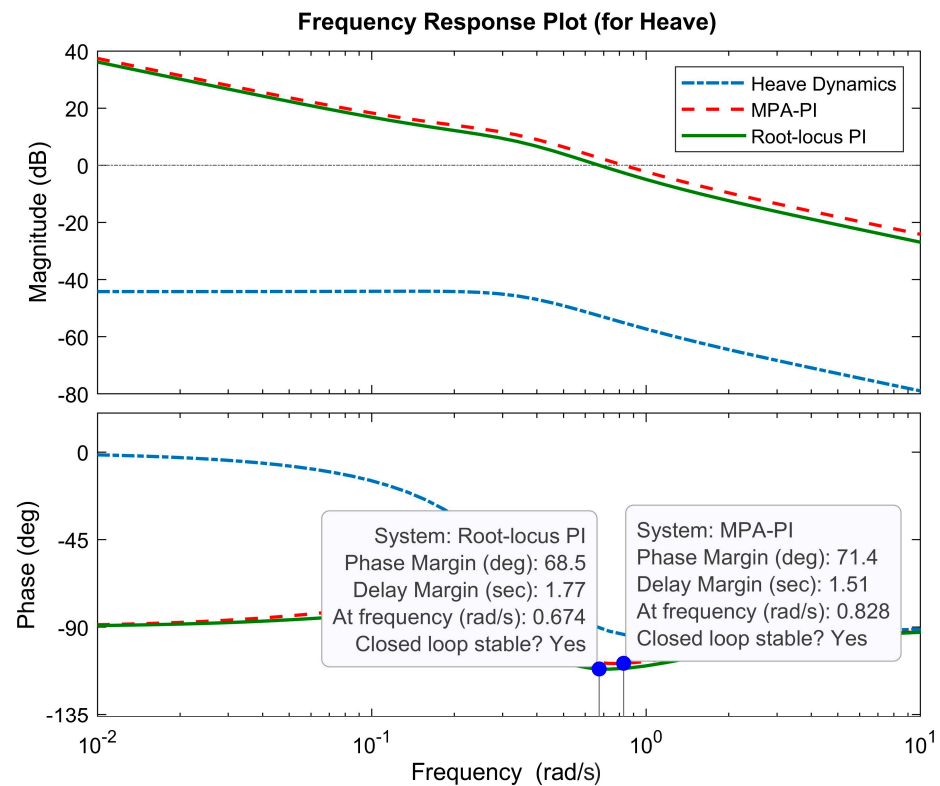


Figure 18. Frequency response comparison for heave.

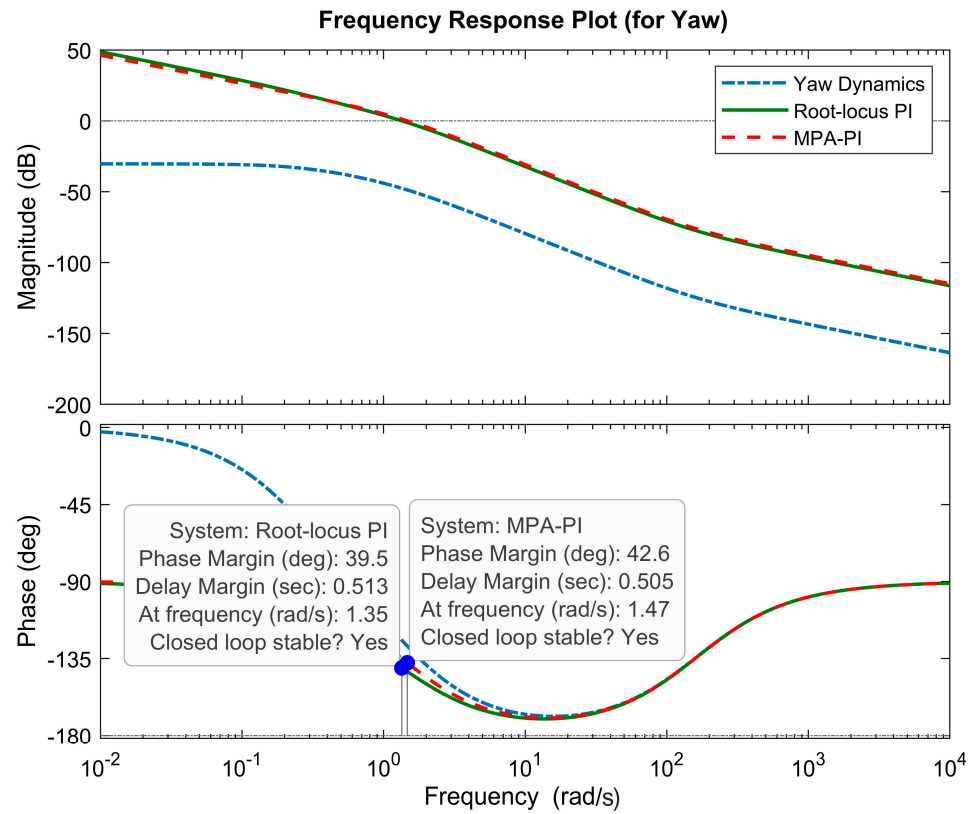


Figure 19. Frequency response comparison for yaw.

Table 5. Frequency-domain performance comparison of the designed controllers.

	Heave			Yaw		
	Desired Performance	Experimental		Desired Performance	Experimental	
		Root-Locus	MPA		Root-Locus	MPA
GM	$\infty$	$\infty$	$\infty$	$\infty$	$\infty$	$\infty$
PM	60° (in one DoF at least)	68.5	71.4	60° (in one DoF at least)	39.5	42.6

Here *GM* represents gain margin, and *PM* phase margin.

From the analysis of frequency response plots and Table 5, it is clear that the gain margin and phase margin for MPA-PI are higher than those for the root-locus-tuned PI controller. Moreover, both controllers comfortably fulfill the desired performance criteria for gain and phase margin. However, it should be noted that in the yaw direction, the phase margin falls slightly below the required value. Nevertheless, the criterion of having a minimum 60-degree phase margin in at least one degree of freedom is met. Furthermore, it is demonstrated in the authors’ previous works that the control effort demanded by the MPRL-PI is higher than that of the MPA-PI controller [26]. As a result, it can be said that the vehicle with MPA-PI is not only optimal but also more robust to withstand variations in parameters and external disturbances. Hence, the controller is implemented in real-time to further assess its effectiveness in a practical setting.

### 6.2. Real-Time Experimentation

Real-time implementation of control algorithms is far from trivial. That is why rigorous time and frequency domain analysis is conducted to ensure that the designed controller meets the required performance specifications. Once confidence is developed in the designed controllers, the resulting gains are hard-coded using C++ code onto the ATmega



328 embedded microcontroller. Real-time data is transmitted to the ground station via the umbilical cord and communication link.

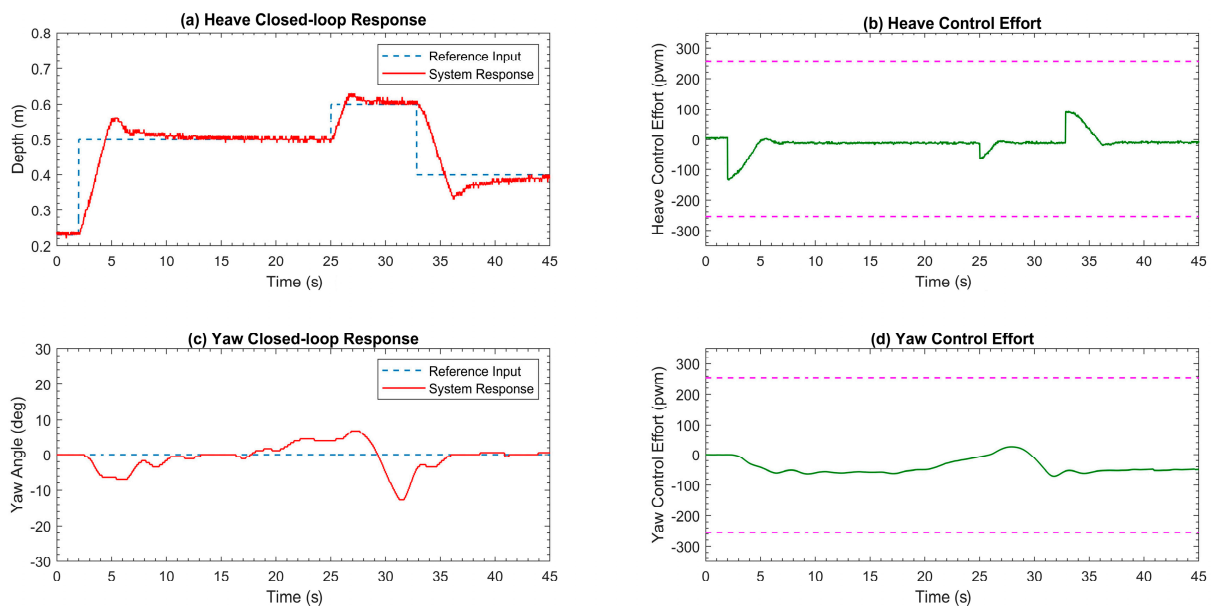
Implementation of designed controllers into a real system allows design engineers to develop a deeper understanding of their system’s behavior. The substandard performance of the controller in real-time signifies the probability that the system has not been modeled correctly. While evaluating the performance of the designed controllers during coupling experiments, two distinct scenarios are analyzed:

6.2.1. Scenario I

The designed controller is put into practice in experimental Scenario I in the manner described below:

1. The ROV’s response is recorded in heave while multiple step input commands are supplied to continuously alter its depth;
2. The ability of the heave controller to track reference input is evaluated from the recorded response (servo problem);
3. The yaw controller is engaged while the input signal is being provided in heave;
4. The goal of the yaw controller is to preserve zero reference despite the disturbance signal that is produced by motion in heave (regulator problem).

It can be seen that both controllers respond as expected to the reference input, as illustrated in Figure 20.

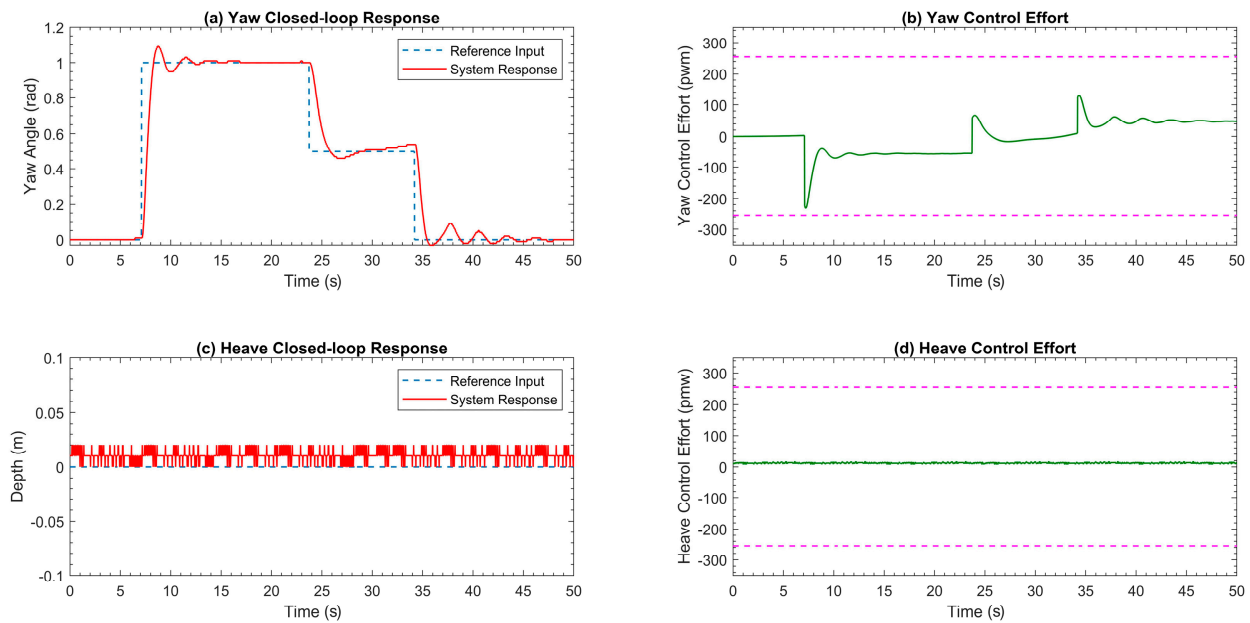


**Figure 20.** Closed-loop response of the coupled MIMO system with heave input; (a) heave closed-loop response, (b) heave control effort, (c) yaw closed-loop response, (d) yaw control effort.

The heave response of the vehicle, as shown in Figure 20, demonstrates effective tracking of input, as indicated by the simulations (Figure 20a). Furthermore, the control effort demanded by the heave controller is within the actuator limits (Figure 20b). In regards to the yaw response, the vehicle appears to experience torque whenever an input signal is provided in heave. The yaw controller, nevertheless, mitigates the induced torque and retains the reference position specified by the input signal (Figure 20c). The control effort demanded by the yaw controller is under the actuator limit as well (Figure 20d). Furthermore, the performance metrics observed closely align with the predictions made during the simulation study.

### 6.2.2. Scenario II

In contrast to Scenario I, the yaw angle of the ROV is continually adjusted, but the heave remains fixed. The ability of the yaw controller to track reference input is examined in the recorded response. In this instance, the goal of the heave controller is to maintain zero reference notwithstanding any disturbance generated by yaw motion. Figure 21 depicts the ROV’s yaw and heave response when both controls are engaged simultaneously. The yaw response of the vehicle exhibits effective input tracking with similar performance metrics, as anticipated by simulations (Figure 21a). The control effort also stays within the saturation limits of the actuators (Figure 21b). Regarding the heave response, there is no indication of any yaw-induced motion in the heave (Figure 21c). The data from the depth sensor (displayed in Figure 21c) is, however, quite noisy and can be smoothed by using a digital low-pass filter. In sum, the developed 2-DoF controller mitigates the influence of motion caused by one DoF in another, demonstrating active reference tracking in both DoFs with little control effort.



**Figure 21.** Closed-loop response of the coupled MIMO system with yaw input; (a) yaw closed-loop response, (b) yaw control effort, (c) heave closed-loop response, (d) heave control effort.

### 6.3. Discussion

As demonstrated in Sections 6.1 and 6.2, MPA-PI not only fulfills but outperforms conventionally tuned PI controllers in almost all of the performance criteria. This superiority of the proposed strategy, not only for ROVs but for other engineering systems as well, was also demonstrated in a 2021 study conducted by Sobhy et al. [38]. In their study, the authors compared the MPA-PID controller with both Grey Wolf Optimization and Artificial Bee Colony (ABC) optimization algorithms for load frequency control of a PV array. The authors’ previous work encompasses a similar study that focuses on comparing one of the proposed controllers with a Simulated Annealing (SA) optimization in order to draw meaningful comparisons [39]. The comparisons carried out in all of the aforementioned studies further support the findings and highlight the effectiveness of the proposed strategy. Furthermore, to the best of the authors’ knowledge, there is no existing work in the open literature that has implemented MPA in the context of a MIMO ROV study. This absence of prior research further emphasizes the novelty and valuable contribution of this article in proposing and evaluating the MPA for ROV control.

#### 6.4. Limitations of MPA Optimization

Researchers from various domains have taken note of the MPA's robustness and impressive performance, as it has demonstrated considerable efficacy in addressing a wide range of optimization problems [40]. However, similar to other optimization methods, the MPA is not without its limitations and drawbacks, which can hinder its search performance and effectiveness when applied to real-world optimization problems.

1. The first limitation of the marine predators algorithm (MPA) is the lack of diversity among its candidate solutions in the population [41]. The algorithm's operators are primarily designed to address optimization problems with flat search spaces, which can result in limited diversity among the solutions within the MPA population.
2. The MPA suffers from another limitation that hampers its performance, which is the lack of flexibility in its parameters [42]. This limitation restricts the ability of the search agents to efficiently explore large search spaces.
3. Lastly, another concern that arises when dealing with optimization problems of high complexity is that the original MPA is specifically designed to handle continuous optimization problems with a single objective [43]. As a result, it may encounter challenges when confronted with more intricate optimization problems involving multiple objectives or discrete variables [44].

#### 7. Conclusions

In this work, the design and development of an unmanned underwater vehicle called a MicroROV is presented. A detailed mathematical framework is developed for complex MIMO MicroROV control utilizing Newton-Euler approaches. The cross-coupling effect between heave and yaw degrees of freedom is also successfully modeled. Additionally, the model takes into account the independent heave and yaw degrees of freedom while also considering the influence of the umbilical cord. The linear system identification technique is then employed to develop high-fidelity data-driven dynamic models meant for model-based closed-loop control design. A multi-parameter root locus technique is employed for the baseline controllers design for all the 3-DoFs, which are compared with MPA-tuned PI controllers. Time and frequency domain analyses are carried out, which demonstrate the robustness of both control strategies. In particular, the frequency response analysis indicates that both controllers have an infinite gain margin with sizable phase margins of  $68.5^\circ$  and  $71.4^\circ$  in heave and  $39.5^\circ$  and  $42.8^\circ$  in yaw for root-locus Pi and MPA-PI, respectively. Hence, it is apparent that the cross-coupling effect is mitigated by the robustly designed controllers. Finally, a real-time controller implementation on an onboard embedded system and pool testing demonstrate the efficacy of the proposed control strategies. The proposed underactuated ROV design renders the vehicle compact by foregoing the need for an additional heave thruster along with the associated motor driver, electronics, and power requirement modules. However, at the cost of more involved modeling and control design. The experience gained will thus feed into an on-going work on an improved and more rugged ROV with a similar vehicle configuration.

**Author Contributions:** Conceptualization, A.T. and S.M.A.; methodology, A.T.; software, A.T.; validation, A.T. and S.M.A.; formal analysis, A.T.; investigation, A.T.; resources, S.M.A.; data curation, A.T.; writing—original draft preparation, A.T.; writing—review and editing, S.M.A.; visualization, A.T.; supervision, S.M.A.; project administration, S.M.A. All authors have read and agreed to the published version of the manuscript.

**Funding:** This research received no external funding.

**Institutional Review Board Statement:** Not applicable.

**Informed Consent Statement:** Not applicable.

**Data Availability Statement:** Data will be made available on request.

**Conflicts of Interest:** The authors declare no conflict of interest.

**Acknowledgments:** The authors would like to thank the King Fahd University of Petroleum and Minerals, and the deanship of research oversight and coordination for their assistance and support.

## References

1. Konoplin, A.Y.; Borovik, A.I.; Mikhailov, D.N.; Vaulin, Y.V.; Scherbatyuk, A.F.; Boreiko, A.A.; Babaev, R.A.; Bolovin, D.A.; Tregubenko, D.I. Application of Autonomous Underwater Vehicles for Research of Ecosystems in the Southern Ocean. In *Antarctic Peninsula Region of the Southern Ocean: Oceanography and Ecology*; Springer: Cham, Switzerland, 2021; pp. 421–432.
2. Soriano, T.; Pham, H.A.; Gies, V. Experimental Investigation of Relative Localization Estimation in a Coordinated Formation Control of Low-Cost Underwater Drones. *Sensors* **2023**, *23*, 3028. [[CrossRef](#)] [[PubMed](#)]
3. Liu, T.; Wan, L.; Liang, X.W. A monocular vision measurement algorithm based on the underwater robot. *Appl. Mech. Mater.* **2014**, *532*, 165–169. [[CrossRef](#)]
4. Wang, W.; Clark, C.M. Modeling and simulation of the VideoRay Pro III underwater vehicle. In Proceedings of the OCEANS 2006-Asia Pacific, Singapore, 16–19 May 2006; pp. 1–7.
5. Azmi, N.H.H. Deep Trekker Inc. UiTM Institutional Repositories 2022. Available online: <https://ir.uitm.edu.my/id/eprint/58139/> (accessed on 27 June 2023).
6. Martin, S.C.; Whitcomb, L.L. Nonlinear model-based tracking control of underwater vehicles with three degree-of-freedom fully coupled dynamical plant models: Theory and experimental evaluation. *IEEE Trans. Control Syst. Technol.* **2017**, *26*, 404–414. [[CrossRef](#)]
7. Petrich, J.; Stilwell, D.J. Robust control for an autonomous underwater vehicle that suppresses pitch and yaw coupling. *Ocean Eng.* **2011**, *38*, 197–204. [[CrossRef](#)]
8. Cao, Y.; Li, B.; Li, Q.; Stokes, A.A.; Ingram, D.M.; Kiprakis, A. A nonlinear model predictive controller for remotely operated underwater vehicles with disturbance rejection. *IEEE Access* **2020**, *8*, 158622–158634. [[CrossRef](#)]
9. Cozijn, H.; van der Schaaf, H.; de Kruif, B.; Ypma, E. Design of an underwater vehicle for use in basin experiments, development of marin’s modular auv. *IFAC-Pap.* **2019**, *52*, 21–26. [[CrossRef](#)]
10. Savaresi, S.M.; Previdi, F.; Dester, A.; Bittanti, S.; Ruggeri, A. Modeling, identification, and analysis of limit-cycling pitch and heave dynamics in an ROV. *IEEE J. Ocean. Eng.* **2004**, *29*, 407–417. [[CrossRef](#)]
11. Lygouras, J.N.; Lalakos, K.A.; Tsalides, P.G. THETIS: An underwater remotely operated vehicle for water pollution measurements. *Microprocess. Microsyst.* **1998**, *22*, 227–237. [[CrossRef](#)]
12. Amat, J.; Monferrer, A.; Batlle, J.; Cufi, X. GARBI: A low-cost underwater vehicle. *Microprocess. Microsyst.* **1999**, *23*, 61–67. [[CrossRef](#)]
13. Caccia, M.; Indiveri, G.; Veruggio, G. Modeling and identification of open-frame variable configuration unmanned underwater vehicles. *IEEE J. Ocean. Eng.* **2000**, *25*, 227–240. [[CrossRef](#)]
14. Centelles, D.; Soriano, A.; Marin, R.; Sanz, P.J. Wireless HROV control with compressed visual feedback using acoustic and RF links. *J. Intell. Robot. Syst.* **2020**, *99*, 713–728. [[CrossRef](#)]
15. Ferreira, C.Z.; Cardoso, R.; Meza, M.E.M.; Ávila, J.P.J. Controlling tracking trajectory of a robotic vehicle for inspection of underwater structures. *Ocean Eng.* **2018**, *149*, 373–382. [[CrossRef](#)]
16. Kong, F.; Guo, Y.; Lyu, W. Dynamics modeling and motion control of a new unmanned underwater vehicle. *IEEE Access* **2020**, *8*, 30119–30126. [[CrossRef](#)]
17. Wang, C.-N.; Yang, F.-C.; Vo, N.T.; Nguyen, V.T.T. Wireless communications for data security: Efficiency assessment of cybersecurity industry—A promising application for UAVs. *Drones* **2022**, *6*, 363. [[CrossRef](#)]
18. Wang, C.-N.; Yang, F.-C.; Nguyen, V.T.T.; Vo, N.T. CFD analysis and optimum design for a centrifugal pump using an effectively artificial intelligent algorithm. *Micromachines* **2022**, *13*, 1208. [[CrossRef](#)] [[PubMed](#)]
19. Folcher, J.-P.; Rendas, M.-J. Identification and control of the Phantom 500 body motion. In Proceedings of the MTS/IEEE Oceans 2001. An Ocean Odyssey. Conference Proceedings (IEEE Cat. No. 01CH37295), Honolulu, HI, USA, 5–8 November 2001; Volume 1, pp. 529–535.
20. Walker, K.L.; Stokes, A.A.; Kiprakis, A.; Giorgio-Serchi, F. Investigating PID control for station keeping ROVs. In Proceedings of the UKRAS20 Conference: “Robots into the Real World”, Lincoln, UK, 17 April 2020; pp. 51–53.
21. Rúa, S.; Vásquez, R.E. Development of a low-level control system for the ROV Visor3. *Int. J. Navig. Obs.* **2016**, *2016*, 8029124. [[CrossRef](#)]
22. Xu, S.J.; Han, D.F.; Ma, Q.W. Experimental Study on the Cross-Coupling Hydrodynamic Coefficients of ROV. In Proceedings of the Twenty-Fifth International Ocean and Polar Engineering Conference, Kona, HI, USA, 21–26 June 2015.
23. Huang, B.; Yang, Q. Double-loop sliding mode controller with a novel switching term for the trajectory tracking of work-class ROVs. *Ocean Eng.* **2019**, *178*, 80–94. [[CrossRef](#)]
24. Farhan, M.; Bhatti, A.I.; Kamal, W.; Yousafzai, I. Sliding modebased MIMO control of autonomous underwater vehicle. In Proceedings of the 2017 11th Asian Control Conference (ASCC), Gold Coast, Australia, 17–20 December 2017; pp. 2899–2904.
25. de Kruif, B.J.; Cozijn, H.; van der Schaaf, H.; Ypma, E. Control design for a multi-regime 6-DOF underwater vehicle; development of MARIN’s modular AUV. *IFAC-Pap.* **2019**, *52*, 230–235. [[CrossRef](#)]

26. Tanveer, A.; Ahmad, S.M. High fidelity modelling and GA optimized control of yaw dynamics of a custom built remotely operated unmanned underwater vehicle. *Ocean Eng.* **2022**, *266*, 112836. [[CrossRef](#)]
27. Ahmad, S.M.; Chipperfield, A.; Tokhi, M. Dynamic modelling and open-loop control of a twin rotor multi-input multi-output system. *Proc. Inst. Mech. Eng. Part I J. Syst. Control Eng.* **2002**, *216*, 477–496. [[CrossRef](#)]
28. Baillie, R.T. Predictions from ARMAX models. *J. Econom.* **1980**, *12*, 365–374. [[CrossRef](#)]
29. Zohedi, F.N.; Aras, M.S.M.; Kasdirin, H.A. Comprehensive study of current trend of the remotely operated vehicle for underwater systems. *TELKOMNIKA (Telecommun. Comput. Electron. Control)* **2022**, *20*, 437–446. [[CrossRef](#)]
30. Ahmad, S.M.; Sutton, R. Dynamic modelling of a remotely operated vehicle. *IFAC Proc. Vol.* **2003**, *36*, 43–48. [[CrossRef](#)]
31. Goheen, K.; Jefferys, E. The application of alternative modelling techniques to ROV dynamics. In Proceedings of the IEEE International Conference on Robotics and Automation, Cincinnati, OH, USA, 13–18 May 1990; pp. 1302–1309.
32. Tanveer, A.; Ahmad, S.M. Heave Modeling and Control of a Micro-ROV. In Proceedings of the 2021 International Conference on Robotics and Automation in Industry (ICRAI), Rawalpindi, Pakistan, 26–27 October 2021; pp. 1–5.
33. Safonov, M.; Athans, M. Gain and phase margin for multiloop LQG regulators. *IEEE Trans. Autom. Control* **1977**, *22*, 173–179. [[CrossRef](#)]
34. Lehtomaki, N.; Sandell, N.; Athans, M. Robustness results in linear-quadratic Gaussian based multivariable control designs. *IEEE Trans. Autom. Control* **1981**, *26*, 75–93. [[CrossRef](#)]
35. Ho, M.-T.; Datta, A.; Keel, L.; Bhattacharyya, S. Robust and Optimal PID Controller Design. *IFAC Proc. Vol.* **1997**, *30*, 55–60. [[CrossRef](#)]
36. Faramarzi, A.; Heidarinejad, M.; Mirjalili, S.; Gandomi, A.H. Marine Predators Algorithm: A nature-inspired metaheuristic. *Expert Syst. Appl.* **2020**, *152*, 113377. [[CrossRef](#)]
37. Rai, R.; Dhal, K.G.; Das, A.; Ray, S. An inclusive survey on marine predators algorithm: Variants and applications. *Arch. Comput. Methods Eng.* **2023**, *30*, 3133–3172. [[CrossRef](#)]
38. Sobhy, M.A.; Abdelaziz, A.Y.; Hasanien, H.M.; Ezzat, M. Marine predators algorithm for load frequency control of modern interconnected power systems including renewable energy sources and energy storage units. *Ain Shams Eng. J.* **2021**, *12*, 3843–3857. [[CrossRef](#)]
39. Tanveer, A.; Ahmad, S.M. Genetic-Algorithm-Based Proportional Integral Controller (GAPI) for ROV Steering Control. *Eng. Proc.* **2023**, *32*, 4.
40. Al-Betar, M.A.; Awadallah, M.A.; Makhadmeh, S.N.; Alyasseri, Z.A.A.; Al-Naymat, G.; Mirjalili, S. Marine Predators Algorithm: A Review. *Arch. Comput. Methods Eng.* **2023**, *30*, 3405–3435. [[CrossRef](#)] [[PubMed](#)]
41. Eid, A.; Kamel, S.; Abualigah, L. Marine predators algorithm for optimal allocation of active and reactive power resources in distribution networks. *Neural Comput. Appl.* **2021**, *33*, 14327–14355. [[CrossRef](#)]
42. Fan, Q.; Huang, H.; Chen, Q.; Yao, L.; Yang, K.; Huang, D. A modified self-adaptive marine predators algorithm: Framework and engineering applications. *Eng. Comput.* **2021**, *38*, 3269–3294. [[CrossRef](#)]
43. Abdel-Basset, M.; Mohamed, R.; Mirjalili, S.; Chakraborty, R.K.; Ryan, M. An efficient marine predators algorithm for solving multi-objective optimization problems: Analysis and validations. *IEEE Access* **2021**, *9*, 42817–42844. [[CrossRef](#)]
44. Abd Elminaam, D.S.; Nabil, A.; Ibraheem, S.A.; Houssein, E.H. An efficient marine predators algorithm for feature selection. *IEEE Access* **2021**, *9*, 60136–60153. [[CrossRef](#)]

**Disclaimer/Publisher’s Note:** The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.