Journal of *Marine Science and Engineering*

MDPI

Technical Note

A Low-Cost, Open-Source, Multi-Purpose Autonomous Surface Vehicle

Thomaz Augusto Kras Benatti †, Emerson Martins de Andrade *,†, Maicon Rodrigo Correa, Felipe da Silva Lopes, João Paulo Machado dos Santos Bernardino, Joel Sena Sales, Jr. and Antonio Carlos Fernandes

Ocean Engineering Program, Waves and Currents Laboratory (LOC), Alberto Luiz Coimbra Institute for Graduate Studies and Research in Engineering (COPPE), Federal University of Rio de Janeiro, Rio de Janeiro 21941-450, Brazil; thomazbenatti@oceanica.ufrj.br (T.A.K.B.); maiconcorrea@oceanica.ufrj.br (M.R.C.); felipelopes@oceanica.ufrj.br (F.d.S.L.); joaopaulomachado@poli.ufrj.br (J.P.M.d.S.B.); joel@oceanica.ufrj.br (J.S.S.J.); acfernandes@oceanica.ufrj.br (A.C.F.)

* Correspondence: mrson@oceanica.ufrj.br

† These authors contributed equally to this work.

Abstract

Autonomous surface vehicles (ASVs) have played a crucial role in various areas, including oceanographic research, environmental monitoring, and asset inspection. However, the high cost and proprietary nature of many platforms limit accessibility. Thus, this work introduces a low-cost, fully open-source ASV platform designed to support a wide range of applications, from academic research to community-driven monitoring projects, bridging the existing gap between low-cost prototyping and naval architecture-based ASV development. Featuring a modular 2 m hull design, the vehicle integrates off-the-shelf components and open-source software to ensure affordability, flexibility, and ease of replication. Field tests were conducted on Ilha do Fundão (Fundão Island), located within the campus of the Federal University of Rio de Janeiro (UFRJ), Brazil. All design files and code are released on GitHub (version 1.0.0) to encourage adoption and collaborative improvement.

Keywords: autonomous surface vehicle; unmanned surface vehicle; open-source

check for updates

Academic Editors: Simone Mancini and Weicheng Cui

Received: 25 October 2025
Revised: 17 November 2025
Accepted: 11 December 2025
Published: 16 December 2025

Citation: Benatti, T.A.K.; de Andrade, E.M.; Correa, M.R.; Lopes, F.d.S.; Bernardino, J.P.M.d.S.; Sales, J.S., Jr.; Fernandes, A.C. A Low-Cost, Open-Source, Multi-Purpose Autonomous Surface Vehicle. *J. Mar. Sci. Eng.* **2025**, *13*, 2380. https://doi.org/10.3390/jmse13122380

Copyright: © 2025 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

Autonomous surface vehicles (ASVs) have played a crucial role in various areas, including oceanographic research, environmental monitoring, and asset inspection. However, most unmanned surface vehicles (USVs) and autonomous surface vehicles (ASVs) remain expensive systems primarily developed for military purposes [1]. Also, scientific research and industrial applications have adopted these platforms in the last decades [2]. Furthermore, the increasing use of such systems illustrates a worldwide movement toward integrating automation and artificial intelligence (AI) [3] in maritime operations, with the objective of improving efficiency, enhancing safety, and reducing costs [4].

In contrast, recent projects have focused on creating smaller, affordable ASV/USV platforms costing less than USD 5000 [5]. These advancements, while helping to democratize ASV developments, also show that few studies have addressed the naval architecture aspects of such projects. Most of the low-cost projects are based on kayak hulls, tube-based catamarans, 3D-printed hulls, commercial RC boats, or model hulls, which may not consider, for instance, hydrodynamic performance or stability issues. Therefore, this research

provides a comprehensive treatment of naval architecture, encompassing hydrostatics, stability, and resistance analysis.

Thus, this work introduces a low-cost, fully open-source ASV platform (called DH200) designed to support a wide range of applications, from academic research to community-driven monitoring projects. From the naval architecture point of view, we tried to offer all the main aspects of the design, which facilitates future design improvements and clarifies the limits of the project.

The ASV features a modular 2 m hull design, integrating off-the-shelf components and a Python-based open-source software to ensure affordability, flexibility, and ease of replication. All design files and code are released on GitHub (https://github.com/mrsonandrade/dh200_asv, accessed on 22 October 2025) to encourage adoption and collaborative improvement. The folders located in the project repository are describe in Appendix A.

The rest of this paper is structured as follows. The next section (Section 2) presents related works. Section 3 discusses the design process from a naval architecture perspective. Section 4 presents the developed software architecture. The main results about the field tests are shown in Section 5. Conclusions are given in Section 6.

2. Related Works

To understand the state-of-the-art in low-cost and open-source ASVs, we describe some related works below.

The work from Gogendeau et al. [5] is a low-cost, reproducible ASV designed for multi-modal surveys, encompassing acoustic tracking, bathymetric mapping, and photogrammetric surveys. Their ASV is built on a standard paddleboard hull for ease of deployment and transportability. Its components were selected to be low-cost, readily available, and easy to assemble. The cost ranges from approximately USD 2500 to USD 11,000, depending on sensor configuration, which makes it significantly more affordable than many commercial alternatives.

Campos et al. [6,7] developed two projects: (1) SENSE and (2) Nautilus, prioritizing open and modular architectures suitable for replication and adaptation. The SENSE ASV is designed as a modular ASV, providing an open-source hardware and software architecture that is easy to replicate for both research institutes and industry. It is a multi-purpose platform capable of acquiring multi-domain data using sensors such as LiDAR, a stereoscopic camera, and a multibeam echosounder. The software architecture is ROS-based (Robot Operating System). The Nautilus ASV also features a modular hardware and software architecture, utilizing the Multilayer Universal Software Architecture (MUSA). A publicly accessible repository is provided, showcasing a practical implementation of the task, skill, and action layers of the MUSA architecture, focused on a positioning algorithm for maritime applications.

In the work from Moulton et al. [8], the AFRL (Autonomous Field Robotics Lab) Jetyak, based on the WHOI Jetyak design, was developed to emphasize modularity and performance. The design builds upon the commercial Mokai Es-Kape boat. It expands capabilities to include long-term deployments and maintains the ability to support manned and unmanned operating modes. The AFRL Jetyak provides a modular design with publicly available documentation and software. The project includes schematics, PCB controller code, diagrams, pictures, and configurations on their open-source Jetyak tutorial page. The platform uses ROS as a middleware framework for software development and standardized data collection.

Carlson et al. [9] developed the ARCAB to be an affordable and portable ASV designed for coastal ocean monitoring, particularly in hazardous environments. It is released under

an Open-Source License (CC BY 4.0) with the hardware design files (CAD files) and software files available in a source file repository. While the design files are open-source, the prototype originally used a National Instruments myRIO-1900, which requires a proprietary LabVIEW license for control. However, the designers suggested that future versions could integrate open-source autopilot systems such as Pixhawk to address this constraint.

The Catabot from Jeong et al. [10] is a modular, research-oriented ASV designed specifically for reliable in-water sensing and environmental monitoring, optimized to minimize interference on sensor readings. The design and parts for Catabot are explicitly stated as open-source on their lab's git repository. It uses the low-level open-source autopilot hardware, Pixhawk, configured with ArduPilot.

In Mattos et al. [11], an open-source system focused purely on affordability leverages the MOOS-IvP software suite. This project implemented an ASV using the MOOS-IvP autonomy framework. The system uses low-cost embedded hardware, including an Arduino Uno R3 Microcontroller and a low-cost Inertial Measurement Unit (IMU), to reduce the vehicle cost. The objective was a basic platform for surveillance and environmental monitoring.

Thus, in contrast to the abovementioned projects, which have focused on creating affordable ASV/USV platforms without (or with a brief) assessment of the naval architecture aspects of such projects (Table 1), this work introduces a fully open-source ASV platform trying to to offer all the main aspects of the design (from the naval architecture point of view), which facilitates future design improvements and clarify the limits of the project. Therefore, important aspects such as the design process, hull compartmentation, deck arrangements, structure, stability, hydrostatics, and hydrodynamics assessment are covered in this work.

Table 1. Summary of related works.

Research	Purpose	Hull	Naval Architecture (Design, Stability, Strength, etc.)	Final Cost
Gogendeau et al. [5]	Acoustic tracking, bathymetric and photogrammetric surveys	Standard paddleboard	NA	USD 2500
Campos et al. [6,7]	Offshore inspection	Two polyethylene pontoons	NA	NA
Moulton et al. [8]	Environmental monitoring	Kayak	NA	USD 5400
Carlson et al. [9]	Coastal ocean monitoring	Fiberglass catamaran hull	Briefly discussed	USD 3400
Jeong et al. [10]	Environmental monitoring	Inflatable catamaran pontoons	Briefly discussed	NA
Mattos et al. [11]	Water quality and environmental monitoring	PVC pipes catamaran	NA	NA
Present work	Multi-purpose	PVC plates monohull with moonpool and wetbay	Design process, hull compartmentation, deck arrangements, structural, stability, hydrostatics and hydrodynamics assessment	USD 1900 (Table 2)

3. Vessel Design

From the perspective of naval architecture, design methods are generally divided into two approaches: (1) conventional and (2) unconventional vessels. Due to its peculiarities and a very sparse database of similar vessels, the latter is more suitable for a multi-purpose autonomous surface vessel design. In this case, based on Lamb [12] and Cross [13,14], the strategy adopted was to organize all requirements imposed to the vessel and relate them to the functional parts that compose the idealized vessel, always ensuring design cycles (Evans Spiral [15]) focus on the integration of all systems to allow for its operation in the most efficient and safe way. Dividing the multi-purpose vessel similarly to other authors in the specialized literature, such as [1], the main requirements and functional parts are presented in Figure 1.

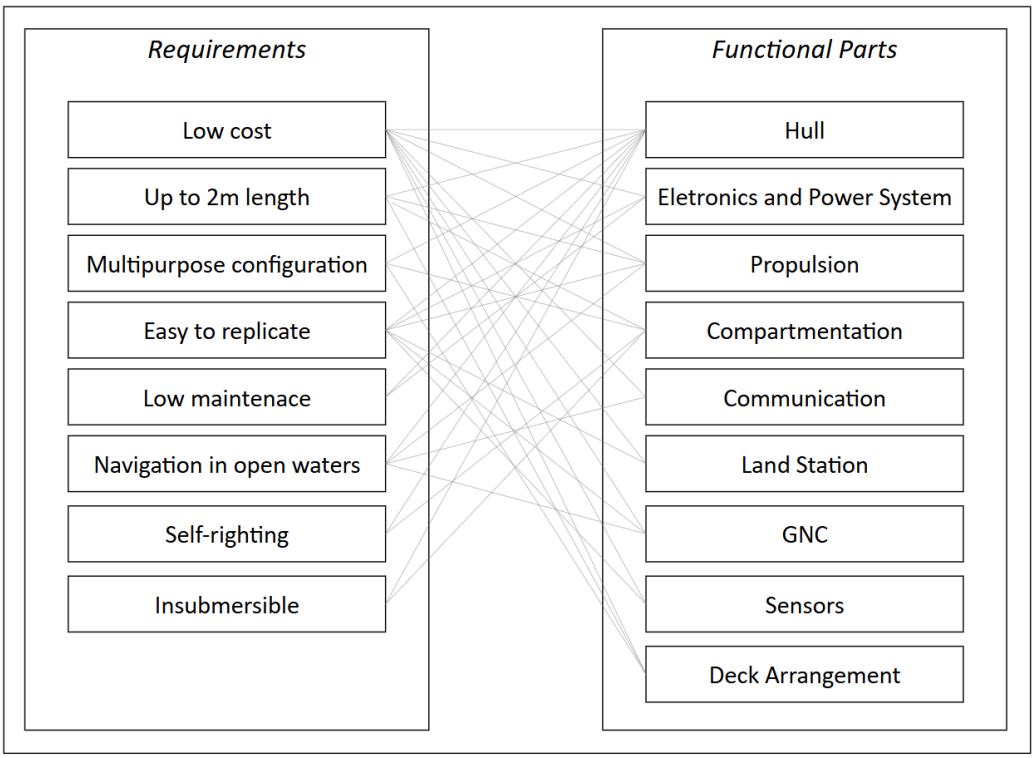


Figure 1. Design methodology mind map.

This method can be challenging to interpret due to the high number of interactions illustrated by the lines connecting requirements and functional parts in Figure 1. This complex interaction exemplifies how many factors must be matched and satisfied simultaneously, and which requirements are more relevant, depending on the quantity of interactions in comparison with the other requirements. Additionally, the interactions between the functional parts themselves must be considered at any time.

In the next sections, the functional parts are presented with more detail. The objective is to discuss how autonomous vehicles can be designed with a more rational approach and show objectively how every part was designed by following the requirements and the interactions that are intrinsic to any engineering design. The final version is first presented in Figure 2.

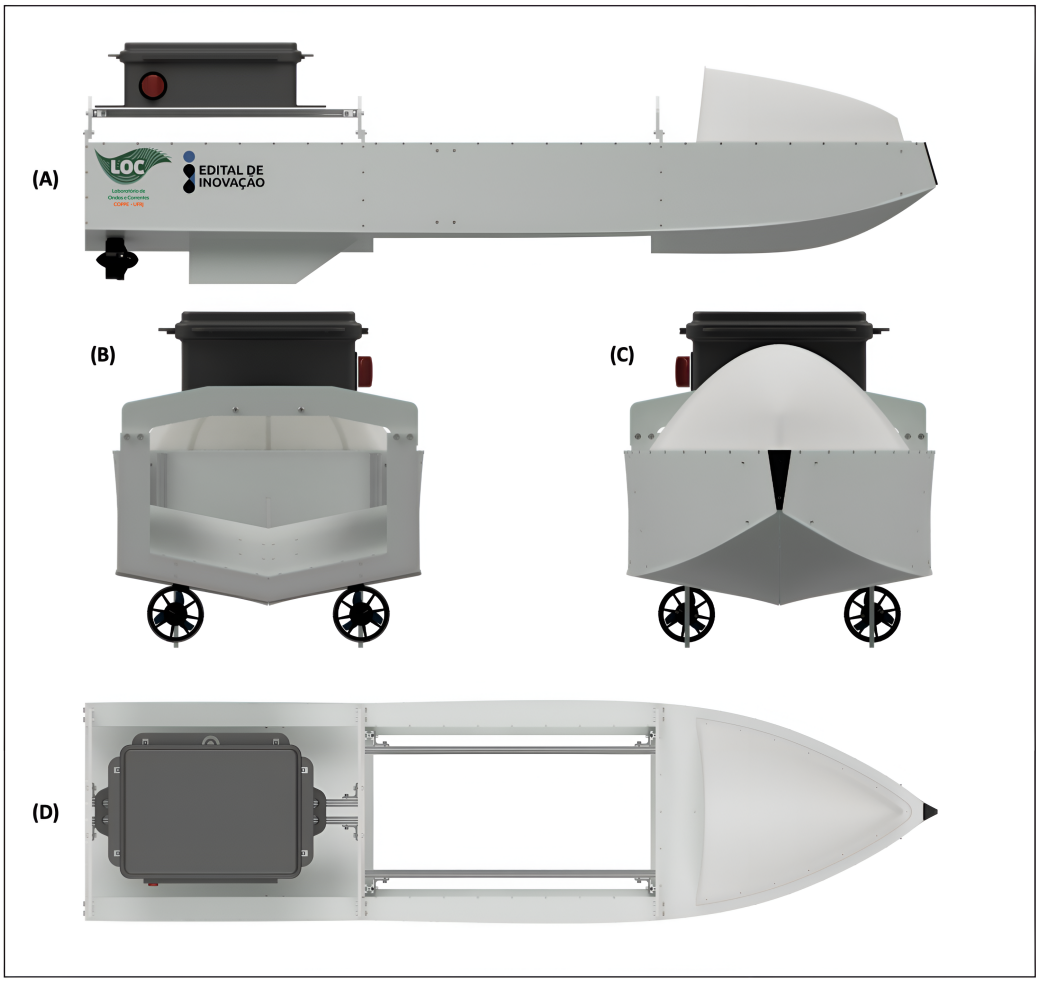


Figure 2. DH200—Fully opened mode: (**A**) side view; (**B**) back view; (**C**) front view; and (**D**) top view.

3.1. Hull, Compartmentation and Deck Arrangement

The hull concept is based on V-shaped hulls that use flat plates, facilitating construction. Due to the multi-purpose requirement, 4 operational profiles were defined to cover the most comprehensive applications, considering the current scenario of unmanned surface vehicles. By adopting a modular concept, each profile has a hull configuration that fits a respective profile application, as presented in Figure 3. Each of these profiles is described below:

1. The fully-opened mode is the main assembly, which can be transformed into other configurations by adding pre-fabricated modules. Although the fully-opened mode does not have a specific purpose, it can be used for task executions. The field tests described in Section 5 were made with this mode.
2. The moonpool mode can carry out tasks such as water quality assessment, ROV (Remotely Operated Vehicle) launch and recovery, and bathymetric surveying. The moonpool is 0.68 m long and 0.40 m wide.
3. For the wetbay mode, externally fixed rails can provide a flexible layout for a launch and recovery system through the stern. This system can be used with AUVs (Autonomous Underwater Vehicles), ROVs, and any other system equipped with sensors.
4. Using the cargo mode, the ASV is capable of transporting other platforms or additional equipment on deck. For cargoes with a low center of gravity, a maximum payload of 50 kg is possible to be reached.

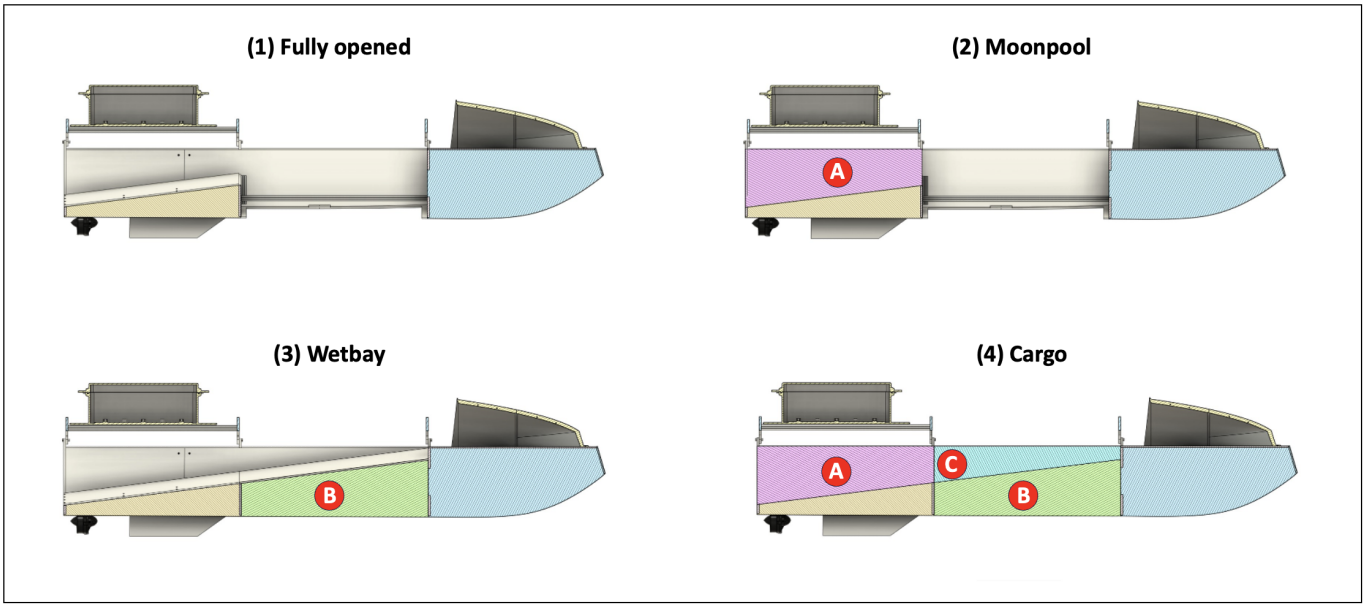


Figure 3. Hull modes: (**1**) fully opened; (**2**) moonpool; (**3**) wetbay; and (**4**) cargo. (A) is the stern module, (B) is the centered bottom module, and (C) is the centered upper module.

The hull can be manufactured with many methods and materials. The construction method adopted is called "one-off", where a temporary frame is first built to assemble the primary structure elements, and then the bottom, side, and deck plates are incorporated. The hull was built using commercial PVC sheets that were CNC-machined and finally attached to the main structure with stainless steel bolts and polymeric sealant. With this method, no surface treatment is required, becoming time- and cost-effective.

Table 2. Items used to build the ASV.

Description	Quantity	Unit Price (USD) *	Total Price (USD)
Flat Allen Screw MA 3 × 20 Stainless Steel 304/A2	216	$0.09	$19.87
Flat Allen Screw MA 4 × 20 Stainless Steel 304/A2	12	$0.09	$1.04
Cylindrical Allen Screw MA 6 × 16 Stainless Steel 304/A2	28	$0.20	$5.60
Hex Nut MA 6 – 1.00 Wrench 10 Stainless Steel 304/A2	16	$0.09	$1.36
Rectangular Nut for Aluminum Profile M6 Base 20	12	$0.38	$4.56
Aluminum Bracket 38 × 38 × 15 mm	8	$0.50	$4.00
Structural Aluminum Profile 20 × 20 V-Slot (1 m)	5	$7.24	$36.20
Gray Waterproof Box Master 43.6 × 32.6 × 14 cm	1	$22.44	$22.44
Rigid PVC Sheet White 1000 × 1000 × 5 mm	6	$76.41	$458.46
Rigid PVC Sheet White 1000 × 1000 × 8 mm	2	$121.08	$242.16
LiPo Battery 14.8 V 5200 mAh 30C 4S	2	$257.00	$514.00
GPS GY-NEO6MV2	1	$36.00	$36.00
XBEE PRO S1	2	$95.92	$191.84
USB to Serial Adapter for XBEE	2	$7.48	$14.96
Raspberry Pi 4B/5	1	$149.32	$149.32
Arduino Mega 2560 R3	1	$30.00	$30.00
Motor 2212 1400 kv + ESC 40A	2	$36.00	$72.00
3D printing PLA filament	1	$18.00	$18.00
Electrical Wires (1 m each)	10	$0.36	$3.60
Battery Switch	1	$31.60	$31.60

Table 2. *Cont.*

Description	Quantity	Unit Price (USD) *	Total Price (USD)
GPS Antenna WT1997 SMA – 5 m	1	$9.95	$9.95
SMA Antenna 3 dBi Wi-Fi 433 MHz	1	$5.03	$5.03
Compass	1	$5.50	$5.50
WiFi Router	1	$14.80	$14.80
High-Strength PVC Adhesive 400 g	2	$4.33	$8.66
-	-	Total	$1900.96

* These unit prices are based on the value we paid for the items in Brazil.

The PVC sheets are more suitable for open threads through the sheet plane with a minimum thickness of 5 mm, which was the minimum thickness adopted for the hull plating. The frame structure was built with 8 mm plates, giving more robustness to the hull, especially to the lift and anchoring points.

With these premises, the main hull section can be assessed to evaluate the integrity of the vessel structure in response to the design loads. The hull beam method is widely used in naval architecture applications, where the hull is represented by a beam with the same sectional area, material, and moment of inertia of the midship section. Through the Euler–Bernoulli formulation, the maximum bending stress is analytically assessed considering the design bend moment (sagging moment). However, this approach must be undertaken carefully, as the Euler–Bernoulli theory does not account for deflections in the transverse beam section, which means it can only be applied when the transverse section is sufficiently stiff.

For such a small vessel, the critical loads arise from transport and handling, particularly when the vessel is subjected to a simply supported condition at both ends or at the midship section, when a single lift can be applied. Table 3 shows the main parameters for the structural assessment, demonstrating how robust the multi-purpose vessel is, due to the construction method and materials chosen. It is worth noting that the fully-open mode has the smallest midship section among all profiles, and it was considered to obtain the results shown in Table 3.

Table 3. Main parameters for structural assessment.

Parameter	Value
Design sagging moment [N·m]	242.3
DAF	2.0
Midship section inertia [m^4]	2.94×10^{-5}
y [m]	0.115
σ_y [MPa]	0.98
Tensile Strength [MPa]	38.00

Although the structure appears to be overpredicted when the tensile strength is compared with σ_y, it is crucial to monitor the structural integrity even for small vessels to ensure the robustness and safety of the vessel design.

The vessel was divided into four compartments. The first one is placed forward of the moonpool to ensure that any damage at the bow cannot lead to a stability issue. Similarly, the double hull along the moonpool section and the aft portion are watertight compartments and were filled with EPS foam to ensure that the insubmersible requirement is fulfilled. The fourth compartment is the electronic box. It is attached to two beams above the deck, allowing the box to adjust the longitudinal position of the CoG depending on the operational mode.

The fundamental aspects regarding hydrostatics and stability are generally expressed with the GZ curve and hydrostatic curves. For details, see Lewis [16]. The GZ curve is a graphic representation of the righting lever (GZ) versus the heel angle. The positive area below the curve can be understood as the work necessary to capsize the vessel, and the negative area as the work to get back upright. So, the self-righting capability is observed when there is no negative area below the GZ curve. Figure 4 shows the GZ curve in blue only for the vessel in the fully opened mode. An A-Frame in the aft and a closed fore castle in the bow can easily turn the vessel into a self-righting boat, represented by the orange curve in Figure 4, which enables it to get back upright after a strong environmental load, like a wave or a boat collision.

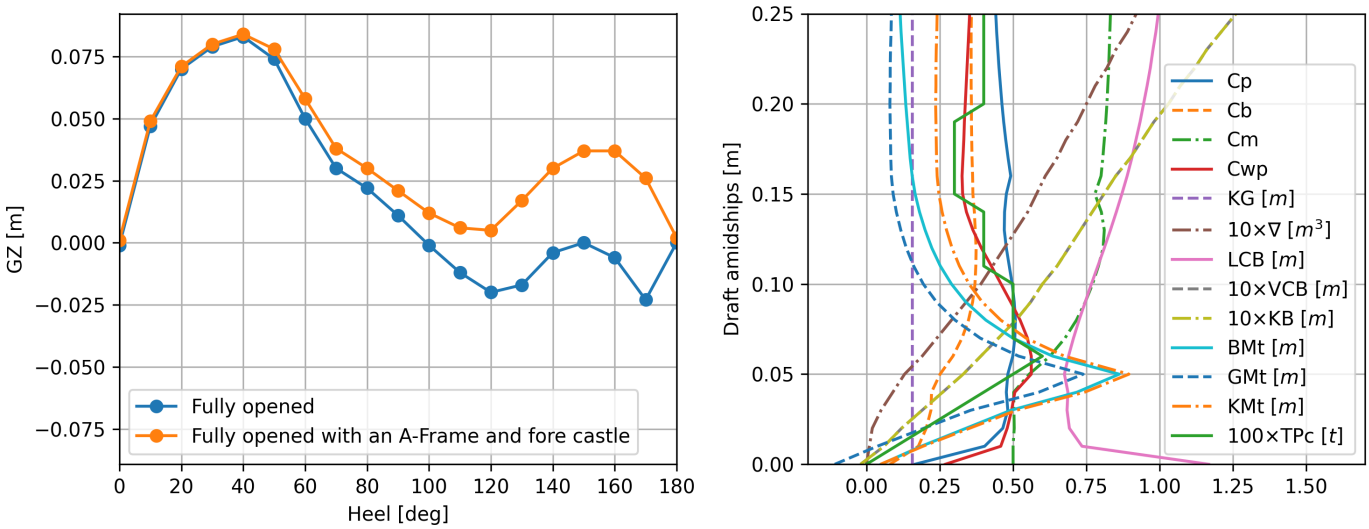


Figure 4. GZ curve and hydrostatics for the fully-opened mode.

The hydrostatic curves express parameters related to geometric aspects (C_p, C_b, C_m, C_{wp}) and boat capacity along the draft (displaced volume ∇, position of center of buoyancy B and metacenter M), which is very useful to assess different payload scenarios in terms of equilibrium. A comprehensive discussion on this topic is presented by Lewis [16]. The hydrostatic curves shown on the right side of Figure 4 were obtained for the fully-opened mode. The geometric coefficients are generally used for design purposes, to compare expected values for each application with the current designed vessel, and will not be extensively discussed in this paper.

The hull resistance at a specific navigating speed is a crucial parameter for the propulsion and power system design. For V-shaped hulls, the Savitsky method [17] offers a simple and easy-to-compute procedure for estimating hull resistance in the planning regime. When a boat is floating with no speed, the buoyancy force is responsible for withstanding the weight. In the planning regime, the hydrodynamic force takes place to support the vessel's weight in a dynamic equilibrium.

Thus, the method proposed by Savitsky is based on geometric characteristics of prismatic planing surfaces and empirical relations to predict the horsepower required, also assessing the trim angle, final draft, and porpoising stability.

Among the operational modes, the cargo mode is the most critical scenario in terms of drag resistance. Hence, this scenario was chosen to estimate the propulsion power needed. The moonpool geometry contributes significantly to the resistance, due to the high vorticity region created by the geometric discontinuity. In this case, for such a peculiar hull, only numerical approaches are capable of predicting the total hull resistance, which was not considered in this work. Table 4 shows the main parameters for hull resistance assessment.

Table 4. Main parameters for hull resistance assessment.

Parameter	Value
Design vessel speed [knots]	3.88
Froud number	1.11
Deadrise angle [deg]	11.00
Trim angle [deg]	4.00
Hull resistance [N]	27.07

To quantify certain hydrodynamic properties of the hull, we computed the loading and responses for the vessel due to surface water waves using potential flow theory under the OrcaWave software [18] (Figure 5). The input parameters we considered are given in Table 5. The output results include the added mass and damping matrices, load RAOs (Response Amplitude Operators), and displacement RAOs. Some examples of these results are shown in Figures 6 and 7, and the full output is given in the *orcawave_results.xlsx* file in the repository of the project.

Table 5. OrcaWave input parameters.

Parameter	Value
Water depth [m]	Infinity
Water density [kg/m^3]	1025
Wave period [s]	0.5, 0.6, 0.7, . . . , 8.0, 8.5, 9.0, . . . , 20.0, 50.0, 100.0
Wave heading [deg]	0, 15, 30, . . . , 345
Draught [m]	0.09, 0.15, 0.30

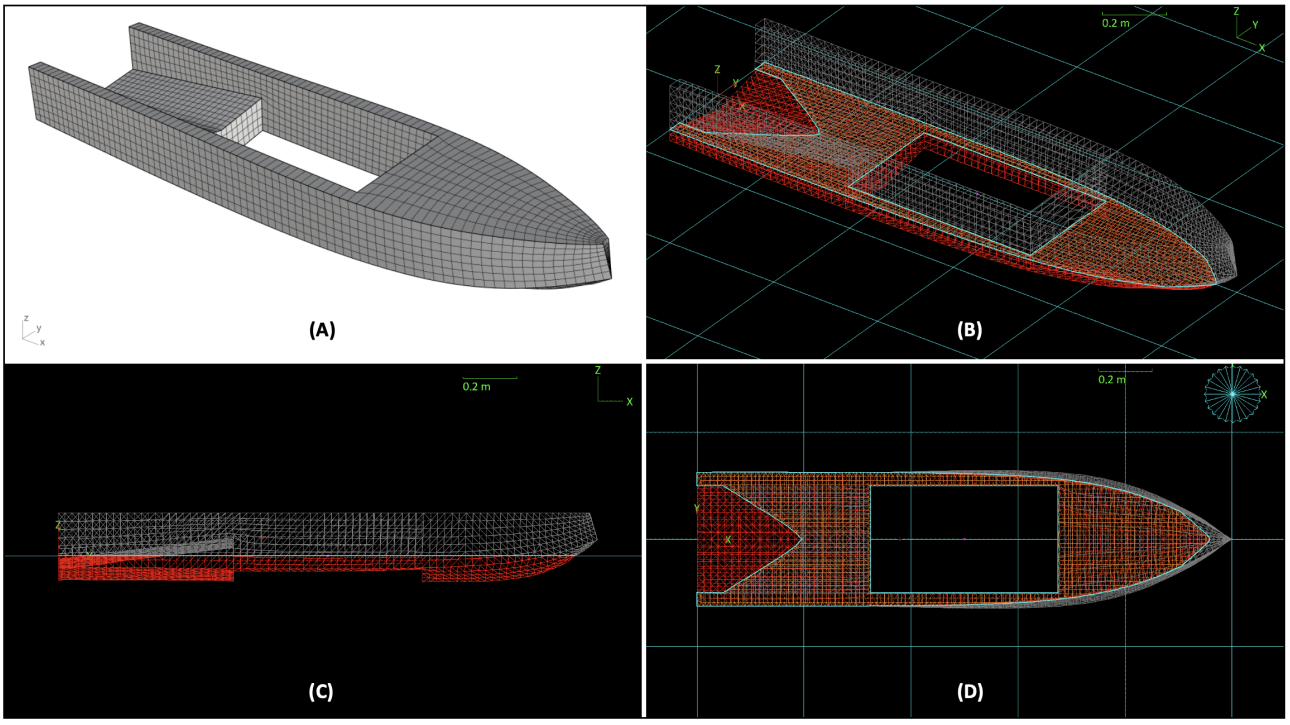


Figure 5. Hull mesh composed of quadrilateral faces (**A**). OrcaWave views (**B**–**D**).

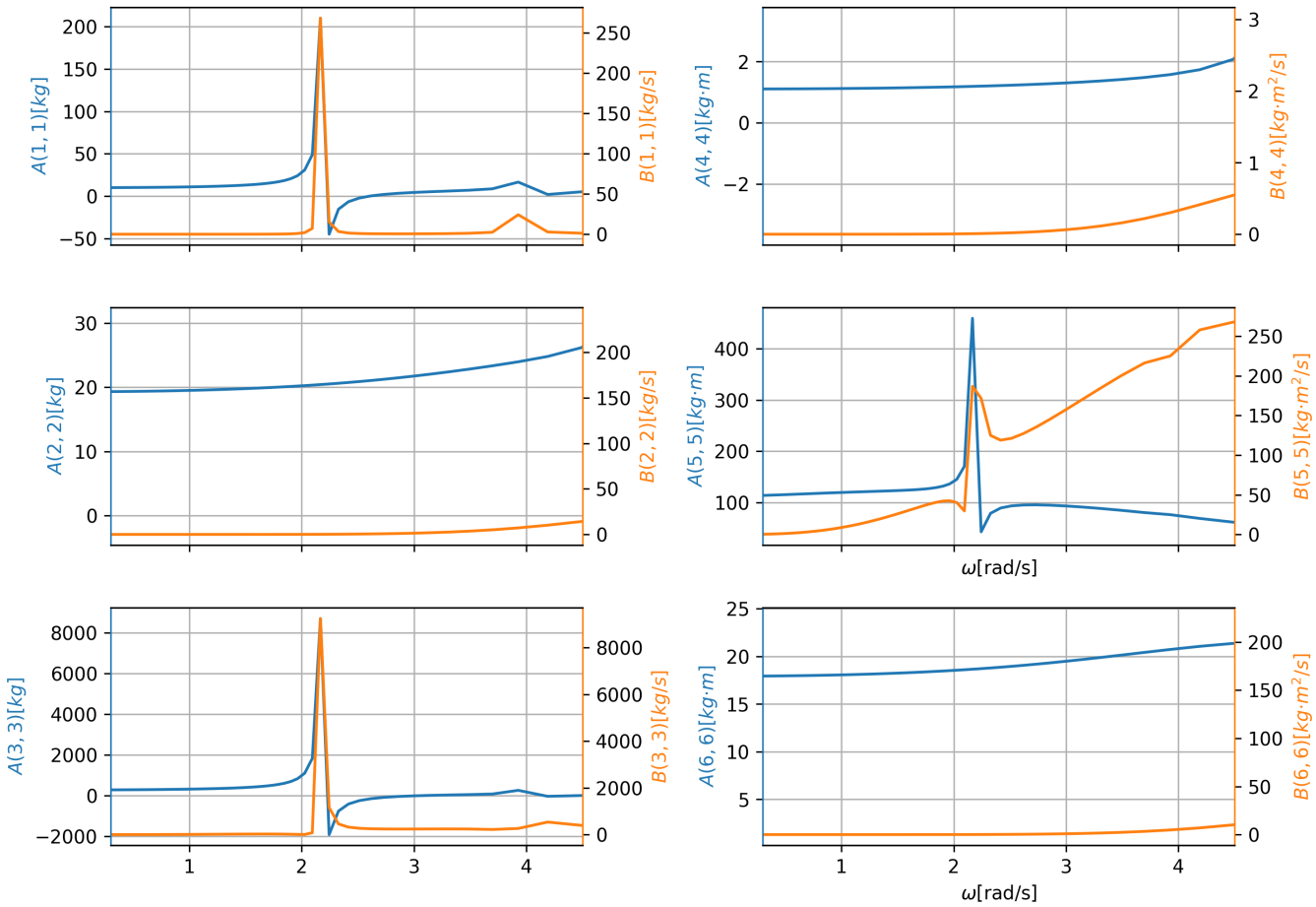


Figure 6. Example of the added mass (A_{ij}) and damping (B_{ij}) coefficients varying with the wave frequency for a 180-degree wave heading.

Concerning the obtained added mass (A_{ij}) and damping (B_{ij}) coefficients, as well as the RAOs, a prominent spike near 2 rad/s can be observed in both the added mass and damping curves. This feature corresponds to the resonance mode of the waves inside the moonpool and is also visible in the RAOs. The resonance frequency depends on the moonpool dimensions, as these determine the wavelength responsible for the resonance effect. Besides size, other factors, such as shape, draft, and vessel speed, also influence the moonpool resonance [19]. It is worth noting that the heave motion coefficients are strongly affected at the resonance frequency due to the excitation of the corresponding mode. Depending on the application of the ASV, these observations are crucial for those interested in placing instruments within the moonpool, since the behavior of the water elevation in that region will depend on the wave frequency. In addition, the RAOs will indicate the regions where the ASV is expected to have amplified responses, which can be important for some instruments and/or for the safety of some operation.

These results can be used to carry out analysis using simulation software such as DynaSOS [20] and Orcaflex [21]. Furthermore, other approaches that allow for studying the maneuvering aspects can be applied based on these frequency-dependent coefficients [22].

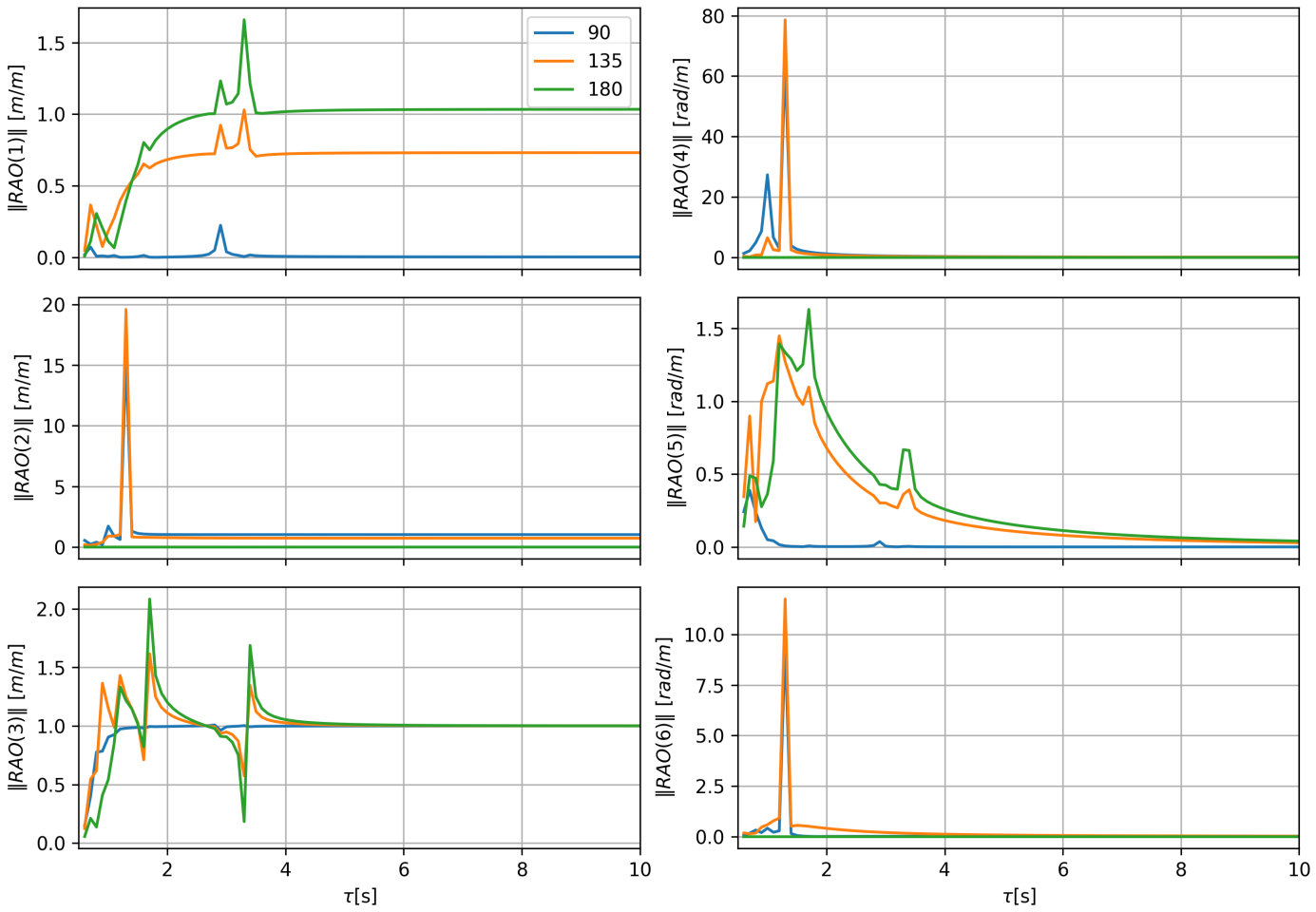


Figure 7. Examples of the displacement RAOs varying with the wave period for three different wave headings (90, 135, and 180 degrees).

3.2. Propulsion, Electronics and Power System

Propulsion systems are generally complex systems that involve many aspects, such as propeller–hull interaction, mechanical efficiency in the transmission system, engine–propeller matching, and the propeller design itself. However, considering the budget limitation, commercial models were considered to compose the propulsion system.

To simplify the propulsion system design, a configuration with two thrusters in parallel and no rudder was chosen. Thus, there are fewer mechanisms to control, and the hull assembly is simpler. The drawback of this choice is the potential loss of control in the event of an engine failure, which implies greater robustness to avoid such a situation.

Due to the lack of specific information about commercial thrusters, a power approach to define the propulsion system is more efficient. Then, based on the hull resistance presented in Section 3.1, the thrust horsepower can be predicted. Considering an open-water efficiency of 50% for ducted propellers as stated in [23], the delivered horsepower can be defined as follows:

$$DHP = \frac{THP}{\eta_0} = \frac{TV_a}{\eta_0} \quad (1)$$

In Equation (1), the thrust (T) is assumed to be equal to the hull resistance, and the speed of advance (V_a) is the vessel speed, both presented in Table 4. So, a maximum power of 108 W is needed to operate the cargo mode with the designed displacement. Thus, the commercial thruster model T100 by Blue Robotics is considered. It already has an electric

engine, propeller, and nozzle, and it is directly attached to the hull bottom. The maximum power of each engine is 130 W, which exceeds the minimum delivered horsepower required. Using this simple power approach, it is possible to make a more reliable power prediction without highly specialized software.

It is important to mention that two skegs were placed aligned with the propellers to protect them in case of any collision with the bottom in shallow water or any floating object.

To provide a starting point for multiple uses, we focused the electrical/electronic system on the essential parts to have a minimal level of remote control and automation.

The idea was to place a Master Switch to turn on/off the system. The power source is composed of two LiPo batteries (Leão, Maringá, Paraná, Brazil) with 14.8 V, each having 5200 mAh of capacity. The software runs on the Raspberry Pi 4B (Raspberry Pi, Pencoed, Wales), which is connected to an Arduino MEGA 2560 (Arduino, Monza, MB, Italy) to allow for the use of analog ports and other sensors that are more integrated into this environment. To remotely control the graphical desktop of the Raspberry Pi OS (version 2025-05-06), the boat has its own WiFi router. The Raspberry connects to the main parts of the system (Figure 8), each one with specific roles:

- Compass: shows the orientation of the vessel;
- GPS: shows the position of the vessel;
- XBee: allows for short-range communication;
- ESC: allows for control of the motor;
- Arduino (version 2.3.6): to be used with sensors.

As the main batteries are 14.8 V, a buck converter is necessary for the Raspberry and the WiFi router, which are 5 V. The motors are powered directly from the batteries (through the switch and the ESCs). Thus, other sensors can be added using the Arduino (which is connected to the Raspberry through the USB) or the Raspberry.

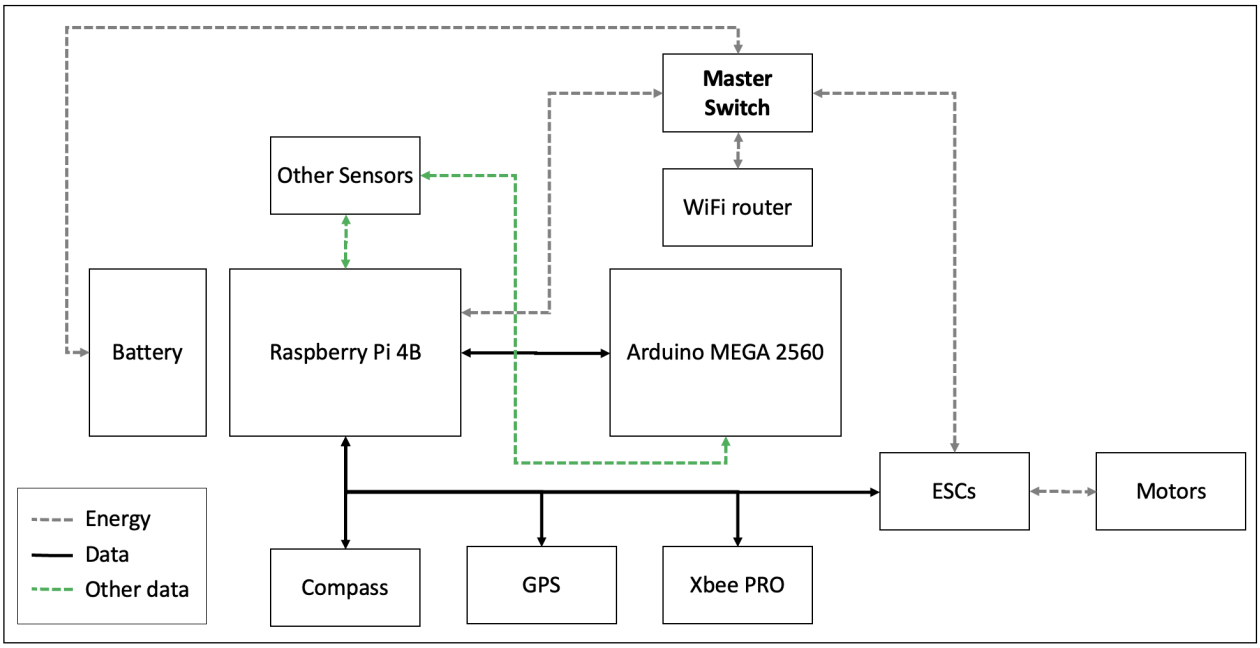


Figure 8. High-level illustration of the connections of the power and electronics system. The dashed grey lines are the energy connections, the solid black lines are the data (and sometimes low-power) connections, while the dashed green lines are new connections that may be needed according to the needs of each user/application.

3.3. Costs

Aligned with recent projects that have focused on creating affordable platforms costing less than USD 5000 [5], we tried as much as possible to select cost-accessible items to lower the barrier for the replication of our platform. At the end, the total cost of the items (without considering any charges related to human labor, hourly wage, or machinery works) is around USD 1900 (Table 2). The most expensive items are the PVC sheets that are used to build the hull. Then, the electronics (batteries, communication, and processing) items follow at the top from the cost perspective. Most of these items are accessible for review in the 3D CAD model stored in the project repository.

4. Software Architecture

The principal idea of the software was to create something accessible, easy to maintain, and using the same language on the ASV and within the GUI (graphical user interface) that is used to control the ASV. Thus, we chose to use Python as a programming language, which allowed us to write all the operating system, inspired by the Python module called *marabunta* [24].

To serve as an earth station and control/monitor the ASV, we are using a Lenovo notebook model ThinkPad (Lenovo, Indaiatuba, São Paulo, Brazil) E14 AMD Ryzen 3 7335U. The notebook has an XBee module connected to it, then it can send and receive data from other XBee in the ASV. Also, the notebook can be used to remotely access the graphical desktop of the Raspberry Pi OS through VNC (Virtual Network Computing) by using the ASV's WiFi router. The notebook system (Windows 11 Home) is running Python version 3.11, the same one as in the Raspberry Pi on the ASV, which runs under the official Raspberry Pi OS, based on the Linux operating system.

4.1. ASV Embedded Code

The Python code running inside the Raspberry Pi that is embedded in the ASV is composed of two main parts (see Figure 9): (A) the Main thread that is infinitely looped and maintains all modules from the ASV alive, and (B) the Robot module that isbased on a Sense-Think-Act decision-making framework [25], with these three phases are infinitely looped, where:

- Sense: the robot gets its own pose/state;
- Think: here, the robot thinks about the objective. For instance, if it is waypoint navigation, it will think about how to achieve it;
- Act: based on the previous phase, the decision is applied.

The Robot module depends on some other high-level modules (Figure 9), which are:

- Communication: in this case, the module is deals with the XBee interface;
- Motion: this module is responsible for the interface with the propulsion system, which in this case is a pair of fixed BLDC electric motors, being treated as a differential system;
- Position: in this case, we are using a GPS to get the ASV position;
- Orientation: this information is given by the HMC5883L compass;
- Sensors: in our case, we are using a temperature sensor, a pH probe, and a conductivity sensor. Then, the module handles sensors in general and each sensor has its own submodule.

The full code is made available in the project repository.

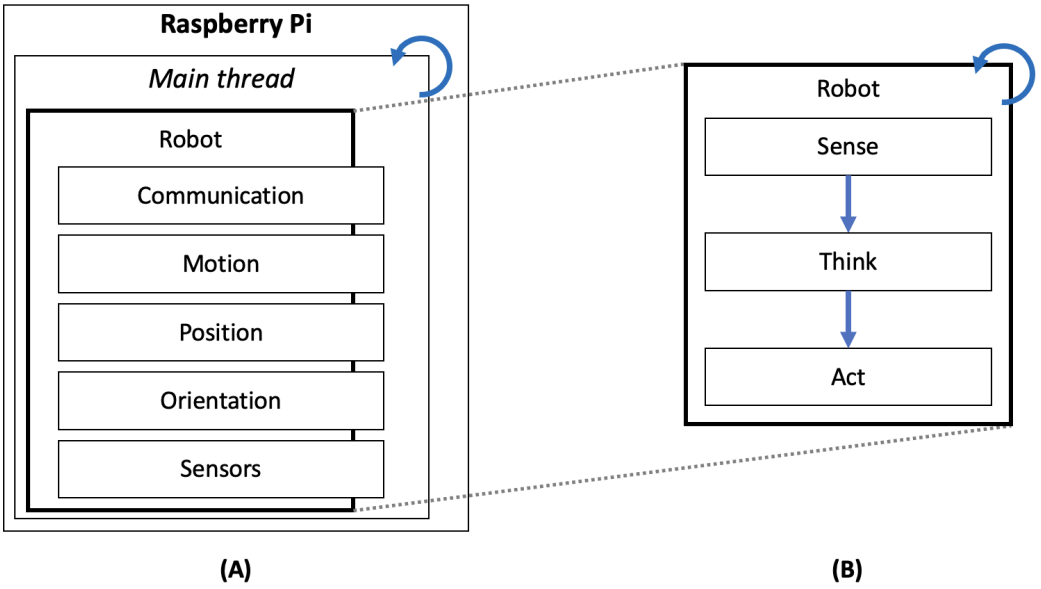


Figure 9. High-level illustration of the architecture of the Python code embedded on the ASV. The Main thread (**A**) inside the Raspberry is infinitely looped to keep the modules alive. The loop (**B**) maintains the robot module working.

4.2. Graphical User Interface

As mentioned before, all the GUI is based on Python; more specifically, it is a PyQt5-based GUI, where this last is a package that allows developers to build cross-platform GUI applications using Python [26].

A screenshot of our GUI, called SPIRE (System for Planning, Integration, and REmote-control), is shown in Figure 10. On the left, we have a dropdown menu where the ASV can be selected (in this case, the number of the selected ASV is 15), and there is also the "ALL" option, where a given command will be sent to all connected ASVs. This allows for the creation of individual or collective missions.

In this current version, the buttons allow for the manual control of the ASV, pin waypoints, send the pinned waypoints to the ASVs, start/stop the mission (based on the previous pinned waypoints), and set the power of the motors from 0 to 100%. Also, the right part of the GUI has the map, which is controlled by buttons allowing the user to switch between satellite or default view, control the zoom, and move the map.

Finally, the bottom part of the GUI mirrors the output from the Python shell, so whatever is printed appears inside the GUI. This helps to debug code when needed, to follow the states of the ASVs, and also be aware of the data sent and received.

4.3. Mission Planning and Navigation

The mission planning is composed of a waypoint navigation system, which is one of the most used approaches among surface vessels [2]. Future versions of the software are planned to include more navigation options, such as swarm navigation of a group of ASVs [24].

The waypoint is based on the work by Fossen [27], where the main idea is to select a set of waypoints that the vessel has to achieve to complete the mission. To calculate the distances and orientation between two GPS-coordinate points, the Haversine formulation will be used [28]. To go from one point to another, a target controller from PySwarming [29] is used

$$b = \frac{T - r}{||T - r||} \tag{2}$$

where T is the target position and $r = (x, y)$ is the ASV position [30]. With the route defined, we have that a waypoint $T_i = (x_i, y_i)$ is associated with a radius R. The waypoint is considered reached when the Euclidean distance between the ASV position r and T_i satisfies:

$$\sqrt{(x - x_i)^2 + (y - y_i)^2} \leq R. \tag{3}$$

Once this condition is met, the navigation system transitions to waypoint T_{i+1}.

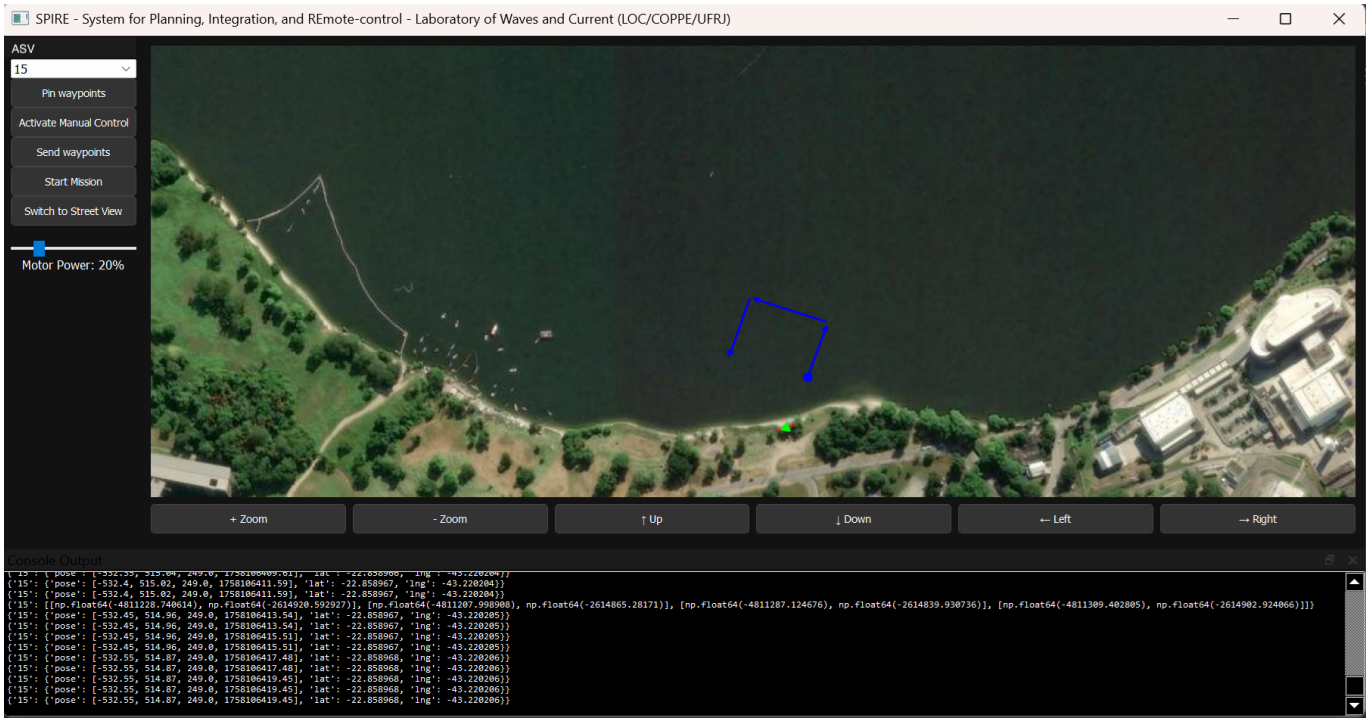


Figure 10. User interface screenshot. The green triangle represents the vessel's position, and the blue arrows indicate the chosen waypoints.

5. Field Tests

To consolidate the proposed solutions, field tests were conducted on Ilha do Fundão (Fundão Island), located within the campus of the Federal University of Rio de Janeiro (UFRJ), Brazil. For this test, the ASV (Figure 11) was equipped with three sensors for water quality assessment:

- Temperature: a DS18B20 digital temperature sensor that uses a 1-Wire interface.
- Conductivity: an EC/TDS transmitter RS485 with output range of 0–44,000 uS/cm:
- pH: a PH4502C pH sensor module designed for measuring the acidity or alkalinity of a solution.

However, despite the presence of the sensors, the objective was to test the ASV waypoint navigation and the overall behavior of the system, and consequently, the sensors were not calibrated, except for the temperature one, which does not need to be. Thus, the sensor measurements presented here (Figures 12 and 13) may not represent reality and are shown only to give a glimpse of the capabilities of the developed platform. Some variations, noticed in the pH measurements, were observed between 400 and 500 s, which may have occurred due to the wind direction influence on the heel angle and the water surface inside the moonpool of the vessel during the last navigation phase, since the pH probe was placed at the rear part of the moonpool.

Therefore, a four-waypoint test with $R = 10$ m was conducted (for details, see Section 4.3). On the day of the test, there was a light breeze, which caused the curved

path between the 2nd and 3rd waypoints (crosswind). Up to the second waypoint, the ASV was in an upwind condition, while the downwind condition was observed after the 3rd waypoint. After the fourth waypoint was achieved, the remote control mode was used to bring the ASV back to land.

To assess and compare the ASV performance during the waypoint navigation, some methods were considered, each one defined by Jekel et al. [31] as follows:

- Partial Curve Mapping (PCM) method: matches the area of a subset between the two curves.
- Discrete Fréchet distance (DF): the shortest distance in-between two curves, where you are allowed to vary the speed at which you travel along each curve independently (walking dog problem).
- Dynamic Time Warping (DTW): a non-metric distance between two time-series curves that has been proven useful for a variety of applications.
- Mean Absolute Error (MAE): average magnitude of errors between desired and experimental values.
- Mean Squared Error (MSE): how far a set of experimental values is from the desired values, calculated by taking the average of the squared differences between experimental and desired outcomes.

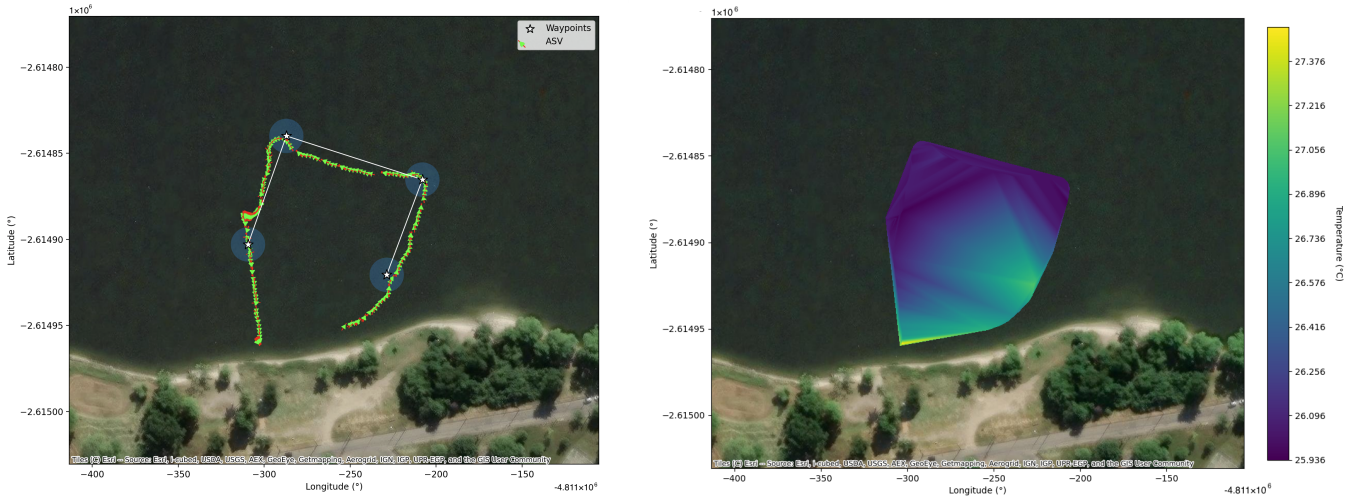
The result of each method described above is presented in Table 6. Based on the obtained values, the Discrete Fréchet distance shows that the obtained value is close to those expected from the accuracy of low-cost GPS modules. Furthermore, the MAE result indicates that the phase shift between the paired points from the two paths may have negatively influenced the value, which also applies to the obtained MSE.

Table 6. A comparison between the desired path and the experimental one.

PCM (m^2)	DF (m)	DTW (Non-Metric)	MAE (m)	MSE (m^2)
6.0	8.6	1110.2	15.2	370.2



Figure 11. The ASV during the field test.



(**a**) (**b**)

Figure 12. (**a**) The four waypoints (white stars) with the four circles where we consider the waypoint to have been achieved. The green triangles with a red line are the ASV positions. The motion started on the right and followed a counterclockwise direction. The last part was carried out using the remote control mode. (**b**) The temperature field reconstructed with the measurements from the test.

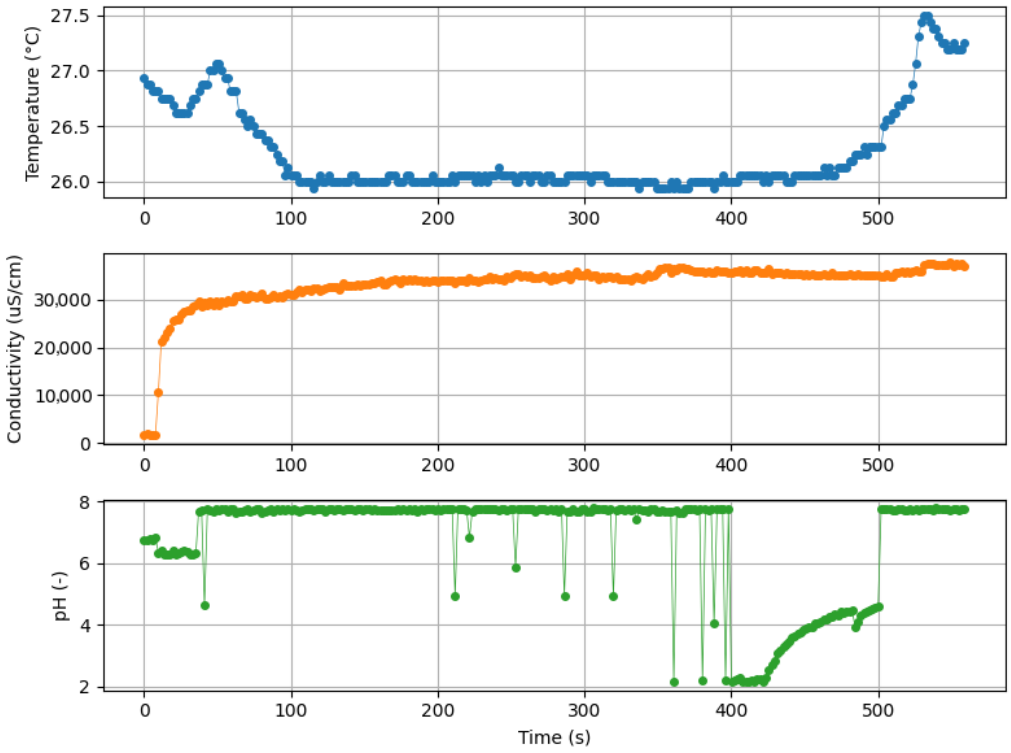


Figure 13. Temperature, conductivity, and pH measurements over time, recorded during the field test. Despite the presence of the sensors, they were not calibrated, except for the temperature, which does not need to be. Thus, the sensor measurements presented here may not represent reality.

Proven Capabilities and Open Questions

This subsection aims to briefly discuss the capabilities we achieved with the proposed system, as well as the areas that remain conceptual, i.e., those that require future research.

Regarding the ASV-Ground-Station system, the field test was sufficient to demonstrate its workability under a real scenario. Thus, the proposed communication, GUI, operational system, and hardware of the ASV worked as expected. Then, this functional complex

integration achieved the goal by being field-proven, with only remaining improvements in new directions.

On the other hand, it will be important to extensively assess some approaches used to develop the design, even those well known in the literature, in future specific research. One example of this is the moonpool presence, which, as discussed throughout the work, introduces nonlinearities that impact the resistance of the hull during navigation. Another point is, for instance, to explore the limits of the present communication system by imposing packet loss conditions and other critical circumstances to map the operational window of the system. Thus, some important future points to be covered can be listed:

- Hull: Investigate each different operational mode and carry out numerical and experimental tests to compare. These comparisons can be made using reduced-scale experiments and CFD for the case of resistance and seakeeping, and numerical tools such as FEM to study the structural nuances of the model. Also, the stability of the hull can be investigated experimentally, being compared to the presented analysis shown here.
- Electronics: Various conditions of operation can be used to test the communication system and also additional links, such as a Starlink system or similar. Concerning the power system, a more meticulous management of the power would be greatly appreciated, since it is strongly related to the endurance of the ASV. In addition, a study covering the need for redundant systems can be explored, improving the robustness of the system.
- Propulsion: Some aspects, such as the propellers' geometry, combined with the motor's performance, can be extensively investigated to achieve a desired objective, such as more speed, or efficiency in some operational regime, or even more thrust to tug another object. Failures of the components can also be studied, the behavior of the ASV under these failures can be mapped, and contingency plans can be developed.

6. Conclusions and Future Work

This work described the design, construction, and testing of a low-cost, open-source, and multi-purpose autonomous surface vehicle (ASV). The project demonstrates that it is possible to develop a robust and modular ASV platform with a total cost of approximately USD 1900, significantly below the average price of most commercial or research-grade vehicles.

Unlike previous low-cost ASV projects, which often employ repurposed hulls or neglect hydrodynamic considerations, this research provides a complete naval architecture treatment, with a rational design approach that considers hydrostatics, stability, and resistance analysis. This systematic approach not only improves performance and safety but also enhances understanding of design trade-offs, contributing to both academic and practical domains.

Its modular hull allows for reconfiguration for diverse missions such as environmental monitoring, oceanographic sampling, or object deployment. The electronics and software architecture were designed with accessibility and replicability in mind. The system integrates off-the-shelf components such as the Raspberry Pi and Arduino, combined with an entirely Python-based open-source control software that includes a graphical user interface (SPIRE) for mission planning and real-time monitoring. Field tests conducted at Ilha do Fundão confirmed the reliability of the waypoint navigation system and demonstrated the platform's potential for environmental data acquisition. All design files, source code, and documentation have been released on GitHub, allowing researchers and hobbyists to replicate, modify, and improve the system freely.

For future work, several directions are intended to improve the ASV, which cover from hardware to software implementation, such as:

- Integration of computer vision or LiDAR systems to enable obstacle detection and real-time path adjustment.
- Implementation of coordinated multi-ASV missions [32], enabling collaborative dynamic monitoring [33], transportation of objects [34], structure assembly [35,36], and area coverage [24].
- Use of onboard AI algorithms for adaptive navigation, sensor calibration, and environmental feature recognition.
- Exploration of hybrid or renewable energy sources (e.g., solar panels and wind) to increase operational autonomy.
- Conducting model basin experiments and CFD simulations to refine hull performance and maneuverability.
- Encouraging open-source collaboration to extend the range of supported sensors, improve mission planning tools, and enhance data visualization modules.

In summary, the proposed ASV provides a scalable, transparent, and collaborative framework that lowers the barrier to entry for ASV research and education. It represents a step toward democratizing marine robotics and fostering innovation through open science and shared engineering design.

Author Contributions: Conceptualization, E.M.d.A. and T.A.K.B.; methodology, T.A.K.B. and E.M.d.A.; software, E.M.d.A.; validation, E.M.d.A., T.A.K.B., M.R.C. and F.d.S.L.; formal analysis, T.A.K.B. and E.M.d.A.; investigation, T.A.K.B., J.P.M.d.S.B. and E.M.d.A.; resources, J.S.S.J. and A.C.F.; data curation, T.A.K.B., E.M.d.A., M.R.C., F.d.S.L. and J.P.M.d.S.B.; writing—original draft preparation, E.M.d.A., T.A.K.B., J.S.S.J. and A.C.F.; writing—review and editing, T.A.K.B., E.M.d.A., J.S.S.J. and A.C.F.; visualization, E.M.d.A.; supervision, J.S.S.J. and A.C.F.; project administration, E.M.d.A. and T.A.K.B.; funding acquisition, J.S.S.J. and A.C.F. All authors have read and agreed to the published version of the manuscript.

Funding: The authors would like to thank the Programa de Recursos Humanos da Agência Nacional do Petróleo, Gás Natural e Biocombustíveis (PRH18-ANP) for their support, supported with resources from investment by oil companies qualified in the P, D&I Clause of ANP Resolution no. 50/2015. This work was supported by the National Council for Scientific and Technological Development (CNPq), the Foundation for Research Support of the State of Rio de Janeiro (FAPERJ) through the "Jovem Cientista do Nosso Estado", and the Instituto Reditus through the "Edital de Inovação".

Data Availability Statement: The project repository can be found in: https://github.com/mrsonandrade/dh200_asv (accessed on 22 October 2025).

Acknowledgments: The authors gratefully acknowledge the support of the Laboratory of Waves and Current (LOC/COPPE/UFRJ) at Federal University of Rio de Janeiro for providing the research facilities and resources necessary to conduct this research.

Conflicts of Interest: The authors declare no conflicts of interest.

Abbreviations

The following abbreviations and symbols are used in this manuscript:

ASV	Autonomous Surface Vessel
USV	Unmanned Surface Vessel
DAF	Dynamic Amplification Factor
DHP	Delivered horsepower
GNC	Guidance, navigation, and control
GUI	Graphical User Interface

LOC	Laboratory of Waves and Current
ROV	Remotely Operated Vehicle
T	Thrust
THP	Thrust horsepower
UFRJ	Federal University of Rio de Janeiro
V_a	Speed of advance
σ_y	Normal Stress
y	Maximum height from the neutral line
η_o	Open-water efficiency

Appendix A. Project Directory Structure

This appendix aims to briefly describe the folders located in the project repository, which can be very useful for those interested in the replication of the ASV presented in this work.

The repository is structured by folders, each one with the following content:

- loc_mind: The "operational system" that runs inside the Raspberry Pi is here, as well the "graphical user interface" (GUI) that runs in the land station. All the codes are written in Python. Here, we call "LOC MIND" the "Modular Intelligent Navigation Device (MIND)" created at the "Laboratório de Ondas e Correntes (LOC)".
- rhino_files: A 3D CAD model for those who are familiar with the Rhinoceros software (version 7). This model contains the fully-opened hull configuration.
- fusion_files: A 3D CAD model for those who are familiar with the Fusion software (version 2602.1.25). This model contains the fully-opened hull configuration with all details of screws, PVC sheets, electronic box, and so on.
- cnc_files: CNC (Computer Numerical Control) router files for those interested in replicate the hull of the ASV. The files (CRV3D format) are already prepared to be used in CNC machines.
- hydrostatics: this folder contains the hydrostatics table and the stability table, which are crucial information for those planning to explore the capacities of the ASV.
- hydrodynamics: as illustrated at the beginning of the paper, this folder contains tables of added mass and damping, load RAOs (Response Amplitude Operators), and displacement RAOs.
- docs: the main steps to replicate the ASV hull and electronics are in this folder, which contains PDF files and spreadsheets.
- media: this folder contains some pictures and videos of the built ASV.

References

1. Liu, Z.; Zhang, Y.; Yu, X.; Yuan, C. Unmanned surface vehicles: An overview of developments and challenges. *Annu. Rev. Control* **2016**, *41*, 71–93. [CrossRef]
2. de Andrade, E.M.; Sales, J.S.; Fernandes, A.C. Operative Unmanned Surface Vessels (USVs): A Review of Market-Ready Solutions. *Automation* **2025**, *6*, 17. [CrossRef]
3. Gao, K.; Gao, M.; Zhou, M.; Ma, Z. Artificial intelligence algorithms in unmanned surface vessel task assignment and path planning: A survey. *Swarm Evol. Comput.* **2024**, *86*, 101505. [CrossRef]
4. Patterson, R.G.; Lawson, E.; Udyawer, V.; Brassington, G.B.; Groom, R.A.; Campbell, H.A. Uncrewed Surface Vessel Technological Diffusion Depends on Cross-Sectoral Investment in Open-Ocean Archetypes: A Systematic Review of USV Applications and Drivers. *Front. Mar. Sci.* **2022**, *8*, 736984. [CrossRef]
5. Gogendeau, P.; Bonhommeau, S.; Fourati, H.; Julien, M.; Contini, M.; Chevrier, T.; Nieblas, A.E.; Bernard, S. An autonomous surface vehicle for acoustic tracking, bathymetric and photogrammetric surveys. *Ocean Eng.* **2025**, *331*, 121201. [CrossRef]

6. Campos, D.F.; Matos, A.; Pinto, A.M. Modular Multi-Domain Aware Autonomous Surface Vehicle for Inspection. *IEEE Access* **2022**, *10*, 113355–113375. [CrossRef]
7. Campos, D.F.; Gonçalves, E.P.; Campos, H.J.; Pereira, M.I.; Pinto, A.M. Nautilus: An autonomous surface vehicle with a multilayer software architecture for offshore inspection. *J. Field Robot.* **2024**, *41*, 966–990. [CrossRef]
8. Moulton, J.; Karapetyan, N.; Bukhsbaum, S.; McKinney, C.; Malebary, S.; Sophocleous, G.; Li, A.Q.; Rekleitis, I. An Autonomous Surface Vehicle for Long Term Operations. In Proceedings of the OCEANS 2018 MTS/IEEE Charleston, Charleston, SC, USA, 22–25 October 2018; pp. 1–10. [CrossRef]
9. Carlson, D.F.; Fürsterling, A.; Vesterled, L.; Skovby, M.; Pedersen, S.S.; Melvad, C.; Rysgaard, S. An affordable and portable autonomous surface vehicle with obstacle avoidance for coastal ocean monitoring. *HardwareX* **2019**, *5*, e00059. [CrossRef]
10. Jeong, M.; Roznere, M.; Lensgraf, S.; Sniffen, A.; Balkcom, D.; Li, A.Q. Catabot: Autonomous Surface Vehicle with an Optimized Design for Environmental Monitoring. In Proceedings of the Global Oceans 2020: Singapore–U.S. Gulf Coast, Virtual, 5–30 October 2020; pp. 1–9. [CrossRef]
11. Mattos, D.I.; Soares dos Santos, D.; Nascimento, C.L. Development of a low-cost autonomous surface vehicle using MOOS-IvP. In Proceedings of the 2016 Annual IEEE Systems Conference (SysCon), Orlando, FL, USA, 18–21 April 2016; pp. 1–6. [CrossRef]
12. Lamb, T. A Ship Design Procedure. *Mar. Technol.* **1969**, *6*, 57–78. [CrossRef]
13. Cross, N. *Developments in Design Methodology*; John Wiley & Sons: Chichester, UK, 1984.
14. Cross, N. *Engineering Design Methods*; John Wiley & Sons: Chichester, UK, 1991.
15. Evans, J.H. Basic Design Concept of Naval Engineering. *Nav. Eng.* **1959**, *71*, 671–678.
16. Lewis, E.V. (Ed.). *Principles of Naval Architecture, Volume I: Stability and Strength*; The Society of Naval Architects and Marine Engineers: Jersey City, NJ, USA, 1988.
17. Savitsky, D. Hydrodynamic Design of Planing Hulls. *Mar. Technol.* **1964**, *1*, 71–95. [CrossRef]
18. Orcina Ltd. *OrcaWave*, Version 11.x; Wave Diffraction/Potential-Flow Solver Software; Orcina Ltd: Ulverston, UK, 2025.
19. Yang, S.H.; Yang, Y.J.; Lee, S.B.; Do, J.; Kwon, S.H. Study on Moonpool Resonance Effect on Motion of Modern Compact Drillship. *J. Ocean Eng. Technol.* **2013**, *27*, 53–60. [CrossRef]
20. floatdyn. *DYNASOS: Computer Program for Dynamic Simulation of Offshore Structures*; Floatdyn: Copenhagen, Denmark, 2019.
21. Orcina Ltd. OrcaFlex. Available online: https://www.orcina.com/SoftwareProducts/OrcaFlex/ (accessed on 20 September 2025).
22. Fossen, T.I. Maneuvering coefficient estimation from frequency-dependent added mass and damping: A power-based approach. *Ocean Eng.* **2025**, *341*, 122494. [CrossRef]
23. Carlton, J. *Marine Propellers and Propulsion*, 2nd ed.; Butterworth-Heinemann: Oxford, UK, 2012.
24. Chamanbaz, M.; Mateo, D.; Zoss, B.M.; Tokić, G.; Wilhelm, E.; Bouffanais, R.; Yue, D.K.P. Swarm-Enabling Technology for Multi-Robot Systems. *Front. Robot. AI* **2017**, *4*, 12. [CrossRef]
25. Ren, Z.; Kim, J.I. The Role of AI in On-Site Construction Robotics: A State-of-the-Art Review Using the Sense–Think–Act Framework. *Buildings* **2025**, *15*, 2374. [CrossRef]
26. Willman, J. Overview of PyQt5. In *Modern PyQt*; Apress: Berkeley, CA, USA, 2020; pp. 1–42. [CrossRef]
27. Fossen, T.I. *Guidance and Control of Ocean Vehicles*; John Wiley & Sons: New York, NY, USA, 1994; p. 494.
28. Brummelen, G.V. *Heavenly Mathematics: The Forgotten Art of Spherical Trigonometry*; Princeton University Press: Princeton, NJ, USA; Oxford, UK, 2013; p. 162.
29. de Andrade, E.M.; Fernandes, A.C.; Junior, J.S.S. PySwarming: A research toolkit for Swarm Robotics. *J. Open Source Softw.* **2023**, *8*, 5647. [CrossRef]
30. De Andrade, E.M.; Junior, J.S.S.; Fernandes, A.C. A review of straightforward distributed behaviors in swarm robotics. In Proceedings of the 2023 Latin American Robotics Symposium (LARS), 2023 Brazilian Symposium on Robotics (SBR), and 2023 Workshop on Robotics in Education (WRE), Salvador, Brazil, 15–18 November 2023; pp. 113–118. [CrossRef]
31. Jekel, C.F.; Venter, G.; Venter, M.P.; Stander, N.; Haftka, R.T. Similarity measures for identifying material parameters from hysteresis loops using inverse analysis. *Int. J. Mater. Form.* **2019**, *12*, 355–378. [CrossRef]
32. de Andrade, E.M.; Junior, J.S.; Fernandes, A.C. A Data-driven Approach to Simulate Collective Behaviors. In Proceedings of the 2023 Latin American Robotics Symposium (LARS), 2023 Brazilian Symposium on Robotics (SBR), and 2023 Workshop on Robotics in Education (WRE), Salvador, Brazil, 15–18 November 2023; pp. 125–128. [CrossRef]
33. Zoss, B.M.; Mateo, D.; Kuan, Y.K.; Tokić, G.; Chamanbaz, M.; Goh, L.; Vallegra, F.; Bouffanais, R.; Yue, D.K.P. Distributed system of autonomous buoys for scalable deployment and monitoring of large waterbodies. *Auton. Robot.* **2018**, *42*, 1669–1689. [CrossRef]
34. de Andrade, E.M.; Fernandes, A.C.; Junior, J.S.S. Decentralized floating object transportation using a swarm of autonomous surface vehicles. *J. Ocean Eng. Mar. Energy* **2024**, *10*, 509–521. [CrossRef]

35. O'Hara, I.; Paulos, J.; Davey, J.; Eckenstein, N.; Doshi, N.; Tosun, T.; Greco, J.; Seo, J.; Turpin, M.; Kumar, V.; et al. Self-assembly of a swarm of autonomous boats into floating structures. In Proceedings of the 2014 IEEE International Conference on Robotics and Automation (ICRA), Hong Kong, China, 31 May–7 June 2014; pp. 1234–1240. [CrossRef]
36. Saldana, D.; Gabrich, B.; Whitzer, M.; Prorok, A.; Campos, M.F.M.; Yim, M.; Kumar, V. A decentralized algorithm for assembling structures with modular robots. In Proceedings of the 2017 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS), Vancouver, BC, Canada, 24–28 September 2017; pp. 2736–2743. [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.