



NAZARBAYEV
UNIVERSITY

School of Engineering and Digital Sciences

Bachelor of Engineering in
Mechanical and Aerospace Engineering

**Design of an Unmanned Surface Vessel (USV)
with a Deployable Remotely Operated
Vehicle (ROV)**

by

Dias Serkibayev
Nurislam Tleuliyev
Saniya Shamshayeva
Temirlan Ramazanov

Academic Supervisors: Prof. Konstantinos Kostas,
Prof. Essam Shehab
Industrial Supervisor: Prof. Basman elHadidi
Spring 2026

Declaration

We hereby declare that this report entitled “Design of an Unmanned Surface Vessel (USV) with a Deployable Remotely Operated Vehicle (ROV)” is the result of our own project work except for quotations and citations that have been duly acknowledged. We also declare that it has not been previously or concurrently submitted for any other degree at Nazarbayev University.

Names: Dias Serkibayev

Nurislam Tleuliyev

Saniya Shamshayeva

Temirlan Ramazanov

Date: April 30, 2026

Acknowledgements

We would like to acknowledge the extensive support and guidance of our project supervisors throughout the whole duration of this project – Professor Konstantinos Kostas, Professor Basman elHadidi, and Professor Essam Shehab.

We are also grateful to our academic staff and classmates for their advice, suggestions and feedback throughout the project. Their recommendations helped us improve the quality of our work and refine our ideas.

Lastly, we would like to thank our families and friends for their constant support and motivation.

Abstract

Underwater oil and gas pipes require regular inspections which are conventionally performed manually by divers and manned vessels. The development of Remotely Operated Vehicles (ROVs) enables the performance of these inspections without the need of crew members diving in the water. These vehicles reduce the risks to divers posed by harsh sea conditions, and the dangers associated with highly explosive fuels. However, such risks are not fully eliminated since surface vessels deploying ROVs are still manned, and their crews are therefore exposed to relevant hazards. Hence, there is an increasing demand for Unmanned Surface Vehicles (USVs) aiming at conducting complex marine operations, such as deployment of ROVs, while being safer and cheaper to operate.

The objective of this project is to design such a USV that can carry, deploy, and recover an underwater ROV. The design procedure begins with the identification of relevant requirements and the selection of a preliminary hull geometry, which is then used for the initial weight, hydrostatics, and hydrodynamics calculations. By applying the spiral design process, used in the design of marine vessels, the vessel's design is iteratively modified and optimized to meet safety regulations and industry requirements, while being constantly reevaluated for its stability and structural integrity.

The achieved USV design meets the key requirements for payload capacity, stability, and endurance, and fully complies with classification rules.

Table of contents

Declaration.....	ii
Acknowledgements.....	iii
Abstract.....	iv
Table of contents.....	v
Chapter 1: Introduction.....	1
1.1 Background.....	1
1.2 Project Scope.....	1
1.3 Aim and Objectives.....	2
1.4 Report Structure.....	3
Chapter 2: Related Previous Work.....	4
2.1 Unmanned Surface Vessels (USVs).....	4
2.2 Remotely Operated Vehicle (ROV) Launch and Recovery Systems (LARS).....	5
2.3 Twin-Hull Configuration.....	6
2.4 Relevant Naval Architecture Principles.....	7
2.4.1 Hydrostatics and Stability.....	7
2.4.2 Hydrodynamics and Resistance.....	9
2.5 Propulsion System.....	9
2.5.1 Waterjet.....	10
2.5.2 Transverse Thrusters.....	10
2.5.3 Podded Propulsors.....	11
2.5.4 Comparison of Propulsion System Options.....	12
Chapter 3: Design and Development.....	14
3.1 Material selection.....	14
3.2 Spiral optimization.....	15
3.3 Industrial Requirements.....	16
3.4 Limitations and constraints.....	17
3.5 Engineering Ethics and Standards.....	17
3.45.1 Engineering Ethics.....	17
3.5.2 RINA Rules for Classification.....	18
Chapter 4: Hull Design: Geometrical and Structural Analysis.....	19
4.1 Hull geometry selection.....	19
4.2 Geometry description.....	20
4.3 Hydrostatics.....	21
4.4 Hydrodynamics.....	25
4.5 Structural Analysis.....	26
4.6 Compliance with RINA Rules for Classification.....	29
4.6.1 Bending Moments.....	30
4.6.2 Normal Stresses for Yielding Check.....	31
4.6.3 Section Moduli.....	31

4.6.4 Still Water and Wave Pressures.....	32
4.6.5 Minimum thickness of vessel structure parts.....	33
Chapter 5: Results and Discussion.....	35
5.1 Hull geometry selection.....	35
5.2 Intact and Damaged stability.....	36
5.3 Resistance and Propulsion.....	41
5.4 Component Selection.....	43
5.5 Structural Analysis.....	46
Chapter 6: Conclusions and Future Work.....	49
6.1 Conclusion.....	49
6.2 Future Work.....	49
References.....	50
Appendix A. Drawings.....	52
Appendix B. Scripts and input files.....	55
Appendix C. RINA Rules for Classification.....	69

Chapter 1: Introduction

1.1 Background

Underwater oil and gas pipelines require regular inspections to prevent leaks and to maintain safe operation. Typically, the inspection is conducted using devices deployed and retrieved from manned surface vessels, thus exposing its crew to dangers of such an environment. These inspections may require long-duration offshore stay, which increases crew fatigue and operational cost, while personnel may face additional occupational hazards such as accidents and maritime disasters, piracy, poor treatment for cardiovascular diseases, high-stress from fatigue and long-term isolation, and the risk of infectious diseases [1]. In addition, between 2014 and 2023, 444 marine casualties occurred in EU waters, resulting in 650 fatalities and 7604 injuries, with human elements contributing to 80.1% of all incidents [2]. These data highlight the significant dangers of offshore operations and the critical role of human error in most accidents. To reduce human risk and lower operational costs, the combination of Unmanned Surface Vehicles (USVs) with ROVs provides an optimal alternative that enables unmanned inspection tasks thus eliminating onboard crew in hazardous environments [3].

In recent years, the usage of USVs has increased in military, civilian, and commercial fields. This type of vessel operates on the water surface without onboard personnel and can function autonomously or remotely, carrying inspection equipment and deploying Remotely Operated Vehicle (ROV) safely [4; 3]. Compared to manned vessels, USVs provide autonomy, improved safety, and reduce human risk and operational costs [5]. The removal of onboard crews in conjunction with autonomous operation can significantly reduce the likelihood of accidents in general and in particular the ones attributed to human errors [6].

1.2 Project Scope

This report presents the design of a USV that is capable of transporting, deploying, and retrieving an underwater ROV. The work includes the development of the vessel geometry, starting from the overall hull form and continuing with the arrangement of the main structural and functional parts. Since the vessel must carry a heavy payload and remain stable during operation, the hull design also includes weight distribution, hydrostatic balance, and stability considerations.

The project does not include the control system design and electrical system design. Detailed cost estimation, including budgeting and procurement analysis, physical manufacturing, assembly, and prototype testing of the vessel are also outside the scope of this work.

1.3 Aim and Objectives

The aim of this project is to design a USV that can carry, deploy, and recover an underwater ROV. The vessel must have adequate deck area and structural support to integrate the ROV deployment-and-retrieval platform (see Fig. 1) and carry a total payload of up to 5 tons. It must also satisfy hydrostatic characteristics that guarantee sufficient displacement, stability, and safe operating conditions with sea waves reaching a height of up to 3-4 meters. With a maximum operational velocity of 6 knots, the USV must operate entirely on electric power and achieve a one-week operational autonomy. These requirements make resistance minimization, stability, and propulsion efficiency main factors guiding the design process.

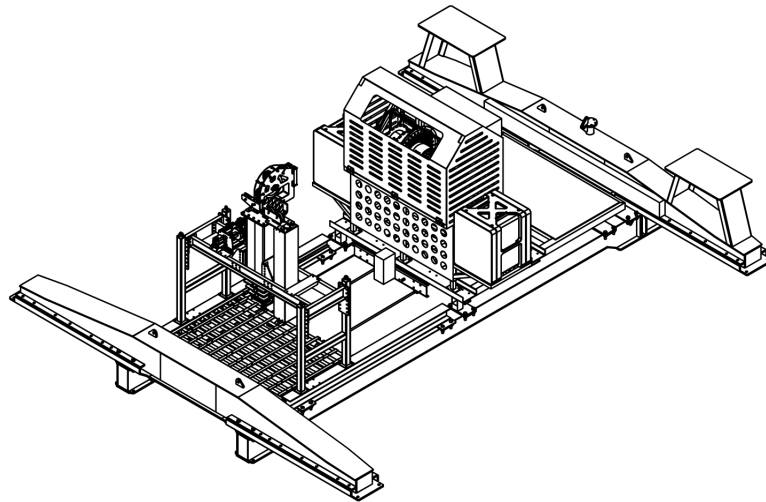


Fig. 1. ROV Launch and Recovery System (LARS) to be carried by the vessel.

To achieve these requirements, a systematic design procedure employed in naval architecture is followed. It constitutes a spiral design methodology that iteratively optimizes the vessel design through cycles of analyses and successive refinements [7]. Key design processes are hull geometry generation, material selection, general layout, hydrostatic and hydrodynamic calculations, structural analysis, power estimation, and propulsion system identification. Open-source simulation software packages such as ArchimedesMB, Michlet, Fastships were used in this work to generate and refine hull designs, as well as evaluate stability, hydrostatic, and hydrodynamic characteristics throughout the design process. The final

complete vessel design should have an optimized balance between operational requirements, stability, endurance, and redundancy.

1.4 Report Structure

The report first reviews relevant literature on USVs, ROV launch and recovery systems, twin-hull configurations, naval architecture principles, propulsion options, and RINA classification rules. It then describes the methodology used for the vessel design, including the spiral optimization, the hydrostatic and hydrodynamic principles, and classification compliance checks. The results section presents the final hull geometry, stability assessment, resistance and propulsion results, component selection, and structural integrity verification. Finally, the conclusion summarizes the design and procedure, and provides recommendations for future work.

Chapter 2: Related Previous Work

2.1 Unmanned Surface Vessels (USVs)

USVs (Fig. 2) have become integral to the field of modern marine operations for their improved effectiveness in performing hazardous, repetitive, and high-volume data-oriented tasks when compared to traditional manned vessels. Recent studies emphasize the rapid technological advancements of USVs which establish them as essential tools in various maritime applications. Specifically, USVs have revolutionized the areas of environmental and territorial surveillance, protection of offshore assets, and Arctic operations, by providing a safer and more cost-effective alternative to manned platforms as indicated in [8]. Furthermore, USVs support sustained data-collection operations in industrial, scientific, and security-oriented activities by prolonged deployment without exposing any personnel to risk. This is enabled by their increased autonomy and the incorporation of advanced sensor packages that facilitate real-time monitoring of complex maritime environments. Therefore, such advancements affirm that USVs may serve as reliable platforms with the capability of performing offshore inspection and surveillance missions which have been traditionally considered the prerogative of crewed vessels.



Fig. 2. Existing Unmanned Surface Vehicle (USV).

As mentioned above, USVs are suitable for offshore inspection because of their developments in autonomy, sensing, communication, and energy management. A set of technological advancements, such as autonomous navigation, obstacle avoidance, real-time sensor integration, and strong communication connections, significantly expanded the operating range of USVs [8]. Improved autonomy allows free maneuvering in dynamic

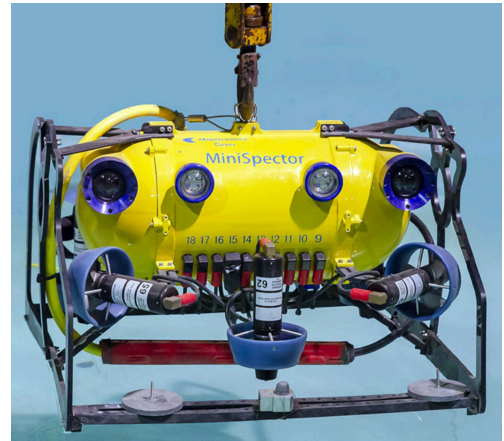
offshore conditions, where built-in sonar, LiDAR, cameras and environmental sensors, allow more detailed structural inspection and seabed mapping. The duration of the inspection missions is extended with the development of better battery systems, hybrid motors, and energy-efficient hulls, which allows USVs to perform continuous surveillance of offshore platforms or support inspection-class ROV operations at sea. Thus, the degree of USVs' technological advancements offers a solid basis in their implementation in marine operations as main inspection platforms which would be able to perform both surface and underwater monitoring activities with high level of reliability and endurance.

2.2 Remotely Operated Vehicle (ROV) Launch and Recovery Systems (LARS)

The Launch and Recovery System (LARS) is a crucial component of the platform, as it enables safe deployment and retrieval of the ROV. For this project, the Launch and Recovery System, designed for MiniSpector ROV shown in Fig. 3 , will be used. Such systems face significant operational challenges in rough seas due to the relative motion between the ROV and the USV during deployment and retrieval leading to potential collisions and damage [9]. However, these risks can be minimized through careful selection of appropriate LARS.



(a)



(b)

Fig. 3. (a) Exemplary Moonpool Launch and Recovery System (LARS) and (b) existing Remotely Operated Vehicle (ROV).

LARS in practice are generally divided into 4 types: side launch, stern launch, open-deck launch, and moonpool launch. A moonpool refers to a vertical opening built into the hull of the vessel that allows a tethered ROV to be deployed and recovered without being exposed to waves outside the hull. Moonpools are typically located near the middle of the vessel .

Positioning the moonpool near the intersection of the vessel's roll and pitch axes can reduce the rolling and pitching motion that ROV may experience. On the other hand, under certain conditions waves inside the well can enter resonance and produce large oscillations. This can interfere with the ROV launch or recovery operations [10].

2.3 Twin-Hull Configuration

A twin-hull vessel is a highly effective and stable platform configuration for unmanned surface vessels. Compared to conventional monohull vessels, twin-hull vessels exhibit enhanced stability and seakeeping characteristics while facilitating moonpool deployment mechanisms. The findings of the research conducted on catamaran USVs (type of twin-hull vessels) highlight that such twin-hull vessels, in comparison with monohull vessels, experience significantly less rolling motion when operating in waves, reducing the risk of capsizing and improving operational reliability of the USVs in rough sea conditions [11]. This advantage is particularly important for unmanned systems that need to maintain stability without any real-time human intervention.

Moreover, catamarans have significant hydrodynamic and maneuvering advantages. Twin-hull vessels generate lower wave-making resistance, subsequently leading to greater energy efficiency during transits [11]. The static and dynamic stability is also enhanced due to the substantial separation distance between the dual propellers on twin-hull vessels. These attributes make the twin-hull configuration of the USVs well suited for applications that require precise control, such as station-keeping, autonomous navigation, and subsea vehicle deployment.

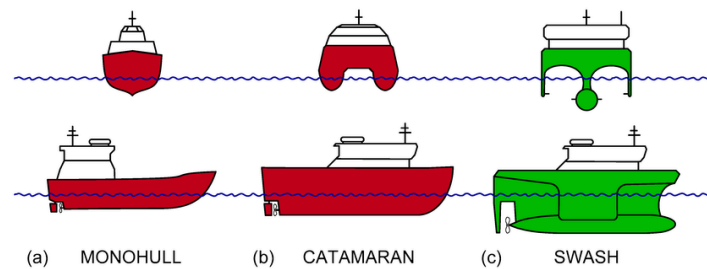


Fig. 4. Comparison of hull configurations: Monohull, Catamaran, and Swash [12].

In addition, catamarans have practical advantages in terms of payload integration. The large deck space of twin-hull vessels allows placing a wide range of mission equipment (such as sensors, communication systems, and solar panels), while maintaining a secure and stable

working platform [11]. The extra space becomes beneficial for the installation of remotely operated vehicle (ROV) launch/recovery systems. Collectively, the high stability, efficient resistance characteristics, and generous deck capacity place the catamaran design as an effective platform for current USVs performing multifaceted offshore operations.

2.4 Relevant Naval Architecture Principles

2.4.1 Hydrostatics and Stability

The first and one of the most important aspects of vessel design is its hydrostatic properties. These properties should be determined before any others and constantly reevaluated during all stages of vessel design. Main principles of hydrostatics are well-researched and studied in pertinent literature.

The simplest and most obvious consideration follows from the equilibrium of two main acting forces: weight and buoyancy. Apart from being equal, their line of action should coincide so that they do not act like a couple. This condition is determined by the following coordinates of the corresponding centers of buoyancy and gravity:

LCB – longitudinal center of buoyancy (m),

LCG – longitudinal center of gravity (m),

TCB – transverse center of buoyancy (m),

TCG – transverse center of gravity (m).

According to *Principles Of Naval Architecture* (The Society of Naval Architects and Marine Engineers, 1988, Vol. 1, Sec. 5), the following quantities are necessary to determine the hydrostatic characteristics and behavior of a vessel [13, pp. 39-41]:

∇ – volume displacement (m^3),

A_{wp} – waterplane area (m^2),

$TP\ cm$ – tons per cm immersion,

$\overline{KB}$ – height of center of buoyancy above baseline (m),

LCF – longitudinal center of flotation (m),

I_T – moment of inertia of waterplane about the longitudinal centerline (m⁴),

$\overline{BM}_T = I_T/\nabla$ - transverse metacentric radius (m),

$\overline{KM}_T = \overline{KB} + \overline{BM}_T$ – transverse metacentric height (m),

I_L – moment of inertia of waterplane about the transverse centerline (m⁴),

$\overline{BM}_L = I_L/\nabla$ – longitudinal metacentric radius (m),

$\overline{KM}_L = \overline{KB} + \overline{BM}_L$ – transverse metacentric height (m).

These quantities need to be calculated for a given vessel geometry for a wide range of drafts.

The volume displacement is defined as the volume of the submerged hull parts, while the waterplane area is the area of the longitudinal section of a vessel's hull cut by the water surface. Having obtained the waterplane area, the center of flotation is defined as its centroid. The vessel's Metacenter M is the point where the buoyancy force vector intersects the vessel's vertical centerline, and the distance $\overline{GM}$ from this point to the vessel's center of gravity G is used as an initial index of stability [13, pp. 18-21]. From the above, the key requirement is to ensure that the longitudinal and transverse metacentric heights $\overline{GM}_T$ and $\overline{GM}_L$ are sufficiently large as described in relevant stability regulations.

Additionally, an important consideration for static stability follows from the center of flotation and the value TP_{cm} that determine the rotating behavior and the increased buoyancy force for each centimeter of additional immersion, respectively. To ensure a zero trim and heel angle during the weight addition, i.e., the vessel's deck remains horizontal, the payload placement zone should be selected such that its center of gravity is directly above the corresponding center of flotation.

Another equally important consideration of the vessel design is its damaged stability. In other words, it is desirable for the vessel to stay afloat even after a fracture and flooding happens. This is achieved by compartmentation of the internal hull space, which limits the flooding only to a concerned compartment and does not lead to sinking or capsizing. The compartmentation depends on where the margin line – the highest permissible waterplane location during damage – is specified.

2.4.2 Hydrodynamics and Resistance

To design a vessel that is efficient, its hydrodynamic properties have to be considered. By reducing various resistance forces which arise when the vessel is in motion, the required propulsion power can be reduced, requiring less energy and extending the offshore operational period.

The service speed of the vessel in this project is relatively low. For such speeds, about 80 to 85 percent of the total resistance is due to frictional resistance [14, pp. 7-15]. The frictional resistance can be estimated using the “*ITTC 1957 model-ship correlation line*” [14, pp. 7-15], which is based on various experimental and research results:

$$C_F = \frac{0.075}{(\log_{10} Re - 2)^2}$$

where Re denotes the Reynolds number and obtained for a given speed v , fluid kinematic viscosity ν , and waterline length L_{wl} as follows:

$$Re = \frac{vL_{wl}}{\nu}.$$

This coefficient is then used to obtain the frictional resistance force for a wetted area S_{wet} (the surface area of the hull that is in contact with the water):

$$R_F = C_F * \frac{1}{2} \rho S_{wet} v^2$$

where: ρ – fluid (water) density.

2.5 Propulsion System

The propulsion system’s selection for an electric USV involves the balancing of hydrodynamic efficiency, maneuverability, mechanical complexity, and compatibility with electric drive systems. Three different propulsion and maneuvering system configurations were evaluated during this work: waterjet propulsion, transverse thrusters, and podded azimuth thrusters.

2.5.1 Waterjet

The working principle of modern waterjet propulsion systems is that water is drawn through an inlet beneath the hull, where a motor-driven impeller accelerates the flow through an internal ducting system and discharges the pressured water through a nozzle at high velocity creating thrust. A typical arrangement of a waterjet system is shown below in Fig. 5. Steering is achieved by a deflector attached to the nozzle that redirects the flow which enables high maneuverability. A reversing bucket redirects the flow forward and applies retarding force to the vessel for deceleration and braking [15].

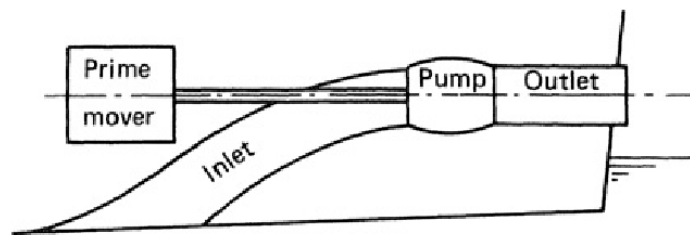


Fig. 5. Typical water jet arrangement [15].

Waterjet propulsion is mainly chosen for high-speed catamarans [16, pp. 4-5] because compared to conventional marine propellers, waterjets enable higher maximum speed, increased maneuverability, and lower noise [17, p. 5]. However, one of the main drawbacks of water jets lies in their decreased propulsive efficiency at lower speeds. To maintain high efficiency, vessels using waterjet propulsion systems must mainly operate at high speeds (e.g., at 33 knots or above for vessels of similar size) [18].

2.5.2 Transverse Thrusters

To achieve high maneuverability in limited spaces, transverse thrusters can be used. A transverse thruster is essentially an impeller inside an appropriately formed tunnel that is oriented transverse to the vessel as shown below in Fig. 6.

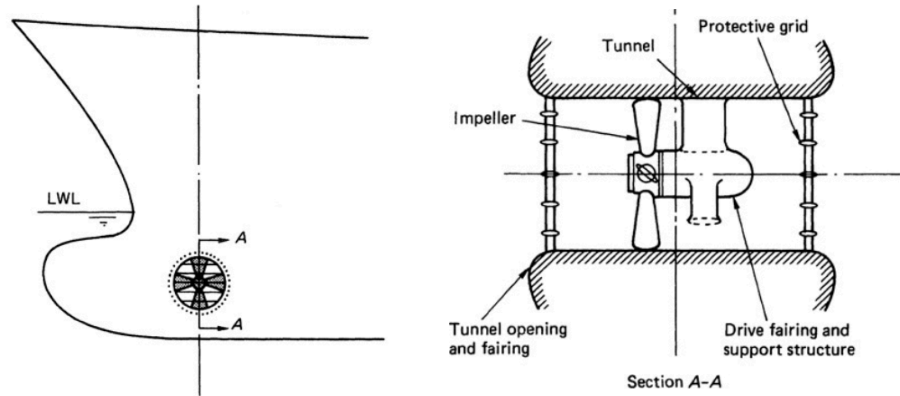


Fig. 6. Transverse thruster unit's general arrangement inside the hull [15].

In such designs, advancing thrust is provided by a separate thruster (propeller, waterjet, etc.) at the stern, and steering is assisted by transverse thrusters typically located at the bow, hence the name bow thrusters [19]; see Fig. 7.

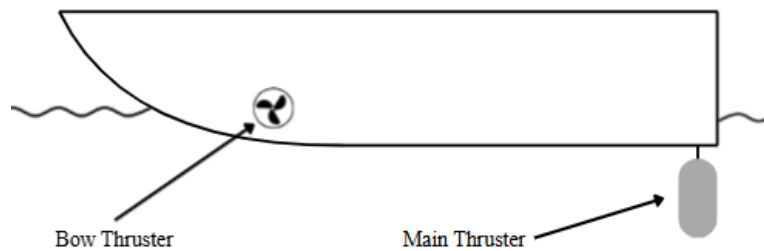


Fig. 7. Illustration of the propulsion layout with bow thrusters.

2.5.3 Podded Propulsors

A podded propulsor is defined as a propulsion unit located outside the vessel's hull and functions as both the steering mechanism and the motor required to drive the propeller [15]. They are derived from the concept of azimuth thrusters, capable of producing thrust in all directions. Podded propulsors are widely used in modern naval engineering because of their increased propulsive efficiency, maneuverability, power efficiency, optimized spatial utilization, and lower maintenance costs [20]. Furthermore, podded azimuth thrusters can have ducted propeller arrangements. Ducted propellers are used to achieve high thrust at lower speeds, but the duct also can negatively affect the thrust at higher speeds [15]. The presence of duct also minimizes the risk of ROV's tether getting damaged during its operations.

However, when two or more azimuth thruster units are used, it is very important to avoid certain azimuth angle configurations in which thrusters can mutually interfere with each other. Such interference may lead to significant forces and moments acting on the shaft system and heavy vibrations may occur [15].

2.5.4 Comparison of Propulsion System Options

Compared to waterjets, propelled propulsors are more efficient for our vessel's requirements. According to study results, propeller propulsion is superior to waterjets up to Froude number of 0.33 [15]. Froude number of the studied vessel was derived using the following equation [15]:

$$F = \frac{V}{\sqrt{gL}}$$

With minimum and maximum speeds of 1 to 6 knots, Froude number ranges between 0.10 and 0.30, respectively. Which means that for the current vessel, propeller propulsion is more efficient than waterjets.

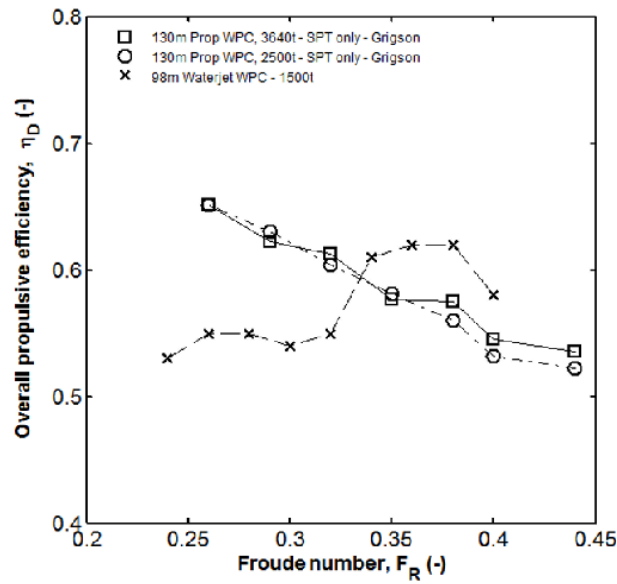


Fig. 8. A typical diagram of propulsive efficiency with respect to the Froude number [16].

In addition, according to the results of the study conducted by Prempraneerach [19], catamaran with azimuth thrusters have a higher yaw rate and better heading change compared to a catamaran with fixed stern propulsor and bow thrusters for steering. Moreover, ships driven by podded propulsors showed better maneuvering compared to

conventionally driven catamaran that had fixed stern propulsor with rudders for steering [15].

To conclude, based on the literature review, podded ducted propellers were selected for the current USV. At low operating speeds, waterjet systems show significantly lower propulsive efficiency compared to conventional propeller-based alternatives. In comparison with shaft-driven propellers combined with transverse thrusters and rudders, podded propulsion systems provide increased maneuverability and better heading control due to their vectored thrust capability. Furthermore, the presence of a duct improves propulsive efficiency in low-speed regimes by reducing tip losses and increasing thrust loading. The duct also serves as a protective barrier between the propellers and ROV's tether, reducing the risk of potential damage to the tether.

Chapter 3: Design and Development

Since this project consists of design and optimization of a twin-hull vessel, there are important procedural aspects associated with naval design lying in the interdependencies of the design stages and subsequent nonlinearity of the design process. Therefore, the general design approach used in this project is to move through the stages along a spiral, constantly revisiting all of them until the result converges to that which fully achieves the aims of the project.

3.1 Material selection

The materials choice for the main components of a marine vessel is an important part of the initial stages of design. The selection may vary from advanced materials, such as composites, to more traditional ones like marine steel. Composites refer to fiber-reinforced composites and molded polymers, while the usual materials are aluminum and steel alloys. However, there is a difference between these materials, depending on what the priority is in the integrated trade-off curve between budget, practicality, feasibility, and the process of manufacturing of the product.

For the initial prototype, in order to examine the most fundamental hypotheses of the project and assess the feasibility of composites, glass-fiber reinforced plastic, and carbon fiber laminates are considered. Given the operational environment of the marine vessel, corrosion and structural loads are important factors. Such composites have a high strength-to-weight ratio and are resistant to corrosion. Nonetheless, the main problem that is linked with these materials is the production process itself. In order to get elements of the desired geometry, molds have to be made, which is known to be a very expensive process, and not very variable because new designs are constantly being created for specific components. Consequently, the labor costs rise, since compared to other industries, the percentage of direct human participation is higher, and the availability of specialists is limited.

Aluminum alloys that are widely used in marine applications, such as 5083-H116 and 5086-H111, are an intermediate material between composites and heavier materials, such as steel. Though light, corrosion-resistant, and relatively durable, aluminum also requires special consideration of its own production processes and modifications. Specifically, aluminum welding needs skilled metalworkers who are used to metalwork and are very

sensitive to heat, as well as special plating. Despite its beneficial performance aspects, aluminum is no easier to manufacture than composites.

Considering the limits of production capacity, steel becomes a more appropriate material in terms of production feasibility and accessibility. This material does not require uncommon production capabilities. When working with steel, well-established rules, regulations, nomenclatures, and alloys already exist, which have proven to be effective for marine systems in practice. A high modulus of elasticity and a relatively high ductility are some of the major factors taken into consideration when choosing material for naval systems [13]. Although steel is heavier, it is financially and logistically affordable.

Specifically, the grades AH36, S355, and ASTM A36 have become commonly used in naval architecture. In terms of structural behavior, AH36 and S355 have a relatively higher yield strength than ASTM A36 and thus can be used in a smaller material thickness with the same expected structural results. This allows both financial savings and a reduction of the total weight of the architecture. For hull plating, the typical thickness is 3-6 mm. However, this is the most frequent range, and this figure can vary both up and down, depending on the structural architecture in question.

Corrosion is one of the most important issues in the operation of this material as the main structural one. However, at the moment, there are a number of corrosion protection technologies available. One of the advantages is to take advantage of polymers, which were already mentioned, in this case, in their use as a coating for pre-prepared steel substrates. Material selection also needs to consider sustainability issues. The use of steel elements is economically feasible and is enabled by a variety of technologies, such as on-site welding and cutting, and re-modification procedures in response to discrepancies or alterations required by structural interventions.

3.2 Spiral optimization

The design process is visually represented by the spiral diagram shown in Fig. 9.

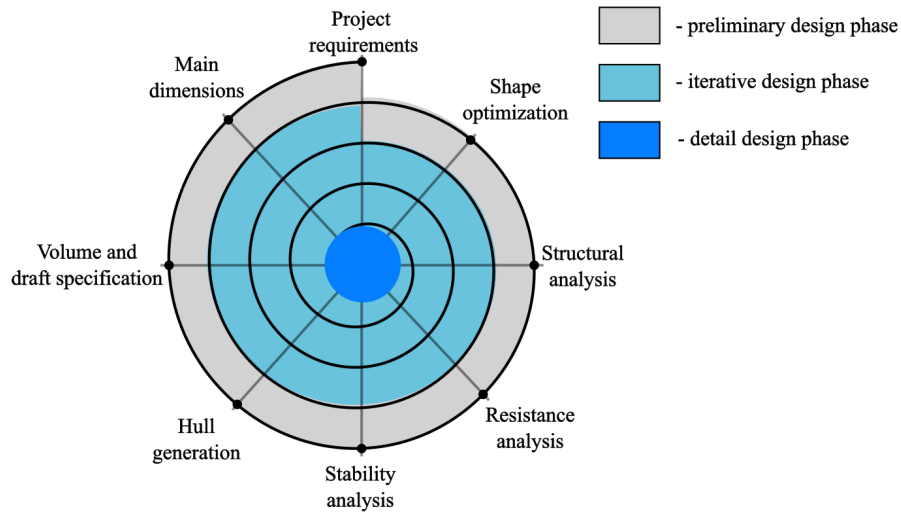


Fig. 9. Spiral design process.

The initial cycle of the process begins with formulating the project requirements and then progresses through the following stages step by step. This comprises the preliminary design phase, which allows us to produce an initial model of the vessel. Since modifications of design and/or parameters at any stage affect many other parameters, the result has to be continuously verified to match the requirements during the subsequent cycles.

When the design fully satisfies the project requirements and possible optimizations are made, the project progresses to the last phase - the detailed design phase. This phase translates the design into fully defined, build-ready specifications. At this stage, all dimensions, structures, powertrain system, materials, and performance parameters are finalized. Drawings, 3D models, and calculations are completed to a level that allows manufacturing, installation, and verification without further interpretation. It is the stage where the vessel's design becomes precise, complete, and ready for production.

3.3 Industrial Requirements

Industrial requirements set by the company involve several functional, operational, and design requirements. It is stated that the vessel is required to:

- Operate in both manned and remote modes with a certain level of autonomy;
- Sustain one week of continuous operation under all-electric propulsion;
- Ensure redundancy within its power and propulsion system;

- Maintain position into current for minimal drag, while thrusters operating at a minimum power;
- Utilize enclosed or ducted propellers to prevent tether damage or any other type of propulsion system without propellers;
- Operate under payload capacity: 5 tons (3 tons for rear and 2 tons for front payload) with maximum speed of 6 knots.

3.4 Limitations and constraints

The project has several design limitations and constraints. The hull geometry is constrained by the existing ROV deployment platform, material availability, manufacturability and structural requirements. The use of steel as the main structural material improves manufacturability and accessibility but increases the total weight of the vessel, which directly affects the displacement, propulsion power and stability of the vessel. The fully electric configuration also constrains the design because the energy storage system occupies significant space and adds considerable weight. Furthermore, the design process is also limited by the use of numerical calculation and simulation tools, which means that final performance still requires experimental validation.

3.5 Engineering Ethics and Standards

3.45.1 Engineering Ethics

This project reflects ethical engineering practice by emphasizing safety, compliance with classification standards, and failure risk assessment. Safety considerations during design phase compel identification of potential failure modes, risk mitigation, and selection of appropriate material and components. Alignment with the classification standards established by RINA reflects ethical commitment to minimize risks and prioritize human safety. Fully electric propulsion systems prevent marine pollution and eliminate exhaust emissions, which aligns with the United Nations Sustainable Development Goals, specifically Climate Action and Life Below Water.

3.5.2 RINA Rules for Classification

RINA is a global, multidisciplinary company contributing to engineering consulting, certification, classification, and inspection services in the energy, marine, transport and infrastructure sectors [21]. For this project, the focus was on the RINA Rules for the classification of ships to ensure full compliance of the vessel design with marine standards. Specifically, the design assessment in the chapter for ships less than 90 m in length was used as a regulatory framework for calculating load, structural sizing, and strength verification. Based on the calculated vertical wave and still water bending moments under both hogging and sagging conditions, the required section moduli at the deck and at the bottom were evaluated to ensure that wave-induced stresses abide by allowable limits. Yielding check was performed by comparing the material yield stress with the calculated stress, verifying the structure integrity under considered load conditions. Structural integrity assessment was further conducted by pressure calculations, which then were used to determine the minimum values for thickness of reinforcement elements to ensure optimal resistance to stresses, and to maintain structural stability.

Chapter 4: Hull Design: Geometrical and Structural Analysis

4.1 Hull geometry selection

As the vessel should house an already existing deployment platform and components, the design process begins from dimensioning the initial hull geometry so that it will match the overall assembly. The dimensions of the hulls, such as its width and height above the water surface, were obtained from the drawings of the platform and components provided by the company.

Hull shape design is a complex task which requires extensive knowledge of the naval design procedures and extensive research. Fortunately, the process can be made significantly easier by using software packages which are able to generate hull geometries based on the key parameters such as the hull volume and overall dimensions, while also taking into account the existing research results. One such software package is the *NTNU Fastships*, designed for quick generation of hull geometries for high-speed applications. While the vessel in this project is not designed for high speeds, it should have the lowest possible resistance. Since relatively low resistance values are inherent for high-speed hulls, this software fits the purpose of the project.

The main input parameters are listed below:

L_{wl} – waterline length,

W_{max} – width of the widest hull part,

W_{stern} – width of the hull stern at waterline,

$\alpha_{entrance}$ – opening angle of the vessel's waterline,

$x_{constant\ width}$ – relative length of constant width hull part,

T – draft to the lowest hull point,

CS_{exp} – cross-sectional superellipse exponent (defines roundness of submerged section),

H_{flare} – above-water height of hull flare,

W_{flare} – flare width (protrusion distance),

H_{deck} – above-water height of the deck.

These values are modified in subsequent steps till a set of optimized hull-shape parameters have been achieved. The full circle of hydrostatic, hydrodynamic, and structural analysis needs to follow each parameter change.

4.2 Geometry description

The mesh of the hull generated in the previous step is not readily suitable for further analysis. For that reason, its geometry should first be described in a format that is commonly used in naval engineering software packages - an offset table. Offset tables are constructed by recording the half-breadth (half-width) of the ship hull at different waterlines for several transverse sections. An illustration of this procedure is shown below in Fig. 10:

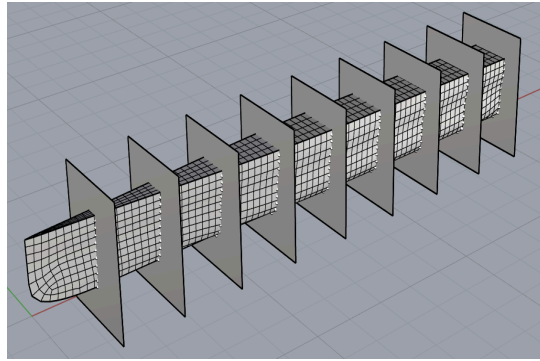


Fig. 10. Example of hull sectioning.

To describe the hull geometry more accurately, a higher number of sections and waterlines is needed, especially in areas of abrupt shape change. Such a procedure is tedious if done manually, and it was therefore automated by developing a Visual Basic script which performs it within the *Rhino*¹ software package. The main advantage of developing a script, apart from saving time, is that it can be applied to any hull size with the position, as well as the number, of sections and waterlines being easily controlled by the user. The table is exported into an easily readable spreadsheet file for further use in subsequent stages.

¹ <https://www.rhino3d.com/>

4.3 Hydrostatics

Hydrostatic characteristics of the hulls is by far the most important factor in designing a safe vessel. The vessel's transverse and longitudinal stability, load carrying capacity, and behavior in harsh sea conditions are all determined by its hydrostatic properties. The aim of conducting the hydrostatic analysis is to ensure that the chosen vessel will be able to:

1. Generate sufficient buoyancy, i.e., equal to the maximum weight at the specified design draft;
2. Stay at a draft higher than specified minimum draft when no payload is present;
3. Remain hydrostatically stable at all times.

Although most hydrostatics calculations are straightforward, they are repetitive and computationally intensive. It is impractical to perform them manually and hence they were done using the open source *ArchimedesMB* software package. *ArchimedesMB* requires hull geometry to be specified in a proprietary format, which is generated by an additional script which was developed in Rhino 3D.

The input information, apart from the geometry, is the draft (denoted as T) range and step, for which the results are going to be calculated and plotted.

The output comprises the following physical quantities:

D – displacement (kilograms of sea water),

VCB – vertical center of buoyancy (m),

LCB – longitudinal center of buoyancy (m),

TCB – transverse center of buoyancy (m),

LCF – longitudinal center of flotation (m),

TCF – transverse center of flotation (m),

MCT – moment to change trim by 1 cm (Nm),

TPI – tonnes per inch immersion (tonnes) – can be converted to $TPcm$,

KMT – transverse metacentric height (m),

KML – longitudinal metacentric height (m),

WL – waterline section width (m), length (m), and area (m²).

The first quantity, the displacement, is the buoyancy force created by the vessel. Its required magnitude is determined by the total weight of the vessel including the installed equipment and payload. The longitudinal and transverse coordinates of the center of buoyancy are used to adjust the vessel's center of gravity so that both centers are on the same vertical line, which results in the vessel's deck being horizontal. Longitudinal and transverse coordinates of the center of flotation represent the pivot point around which the vessel is going to pitch and roll. The most important values are the heights of the metacenter since they indicate the vessel's resistance to capsizing, as mentioned in the previous section.

The values of the quantities above provide sufficient information to assess the vessel's stability if its center of gravity is known as well:

KG – center of gravity height (m).

The main parameter determining the stability are the metacentric heights for transverse and longitudinal perturbations:

$$GMT = KMT - KG,$$

$$GML = KML - KG.$$

According to the technical information about the equipment to be installed on the vessel and the project requirements, the acceptable range for some of these quantities was defined:

Table 1. Constraints placed on hydrostatic quantities.

Quantity	Range
T	$0.9 < T < 1.5 \text{ m}$
GMT	$> 1 \text{ m}$
GML	$> 1 \text{ m}$

The table does not include the quantities for which no constraints have to be imposed.

The results produced by the *ArchimedesMB* software are checked to be inside the specified ranges. In case they do not satisfy the requirements, the design procedure is reverted to the hull geometry selection step, where the key dimensions and parameters are changed and the geometry is generated again.

After the design is verified to fulfill the requirements, the static balancing is performed using a VB script. In essence, it iteratively changes the draft, heel angle, and trim angle to find the balanced position.

Regarding damaged stability, the floodable length curve was developed using Maxsurf software package. The floodable length curve represents the maximum permissible length of a flooded compartment without submerging the reference margin line.

For its computation, the 3D hull model is first imported into Maxsurf Modeller. A design grid is then created by subdividing the hull into many transverse sections. Correspondence to the design reference frame is established, including longitudinal positions (aft, midship, and forward) as well as vertical datums such as the baseline and waterline.

After completing the model, it is transferred to Maxsurf Stability for further analysis. To determine the floodable length, several key input parameters are defined, including maximum and minimum displacement, permeability, margin line position, vertical center of gravity (VCG), longitudinal center of gravity (LCG), and loading conditions. The values of these parameters are provided below in Table 2:

Table 2. Input parameters for floodable length analysis.

Parameter	Value
Max. Displacement	29 t
Min. Displacement	14 t
Permeability	100 %
Margin Line	0.15 m
VCG	1.582 m
LCG	4.763 m
Mass of the vessel	22 t

In order to verify the obtained results, a simplified procedure for obtaining the floodable length curve was used. The hulls were assumed to have constant transverse cross sections, and the permeability coefficient was taken to be equal to 1 (i.e., compartments can be filled completely by water). These assumptions produce results that are stricter than in reality, therefore they are safe to use.

The key idea of finding the maximum floodable length for a given submersion and trim angle lies in treating the flooded compartment as an added weight and calculating the buoyancy force as if the vessel is not damaged:

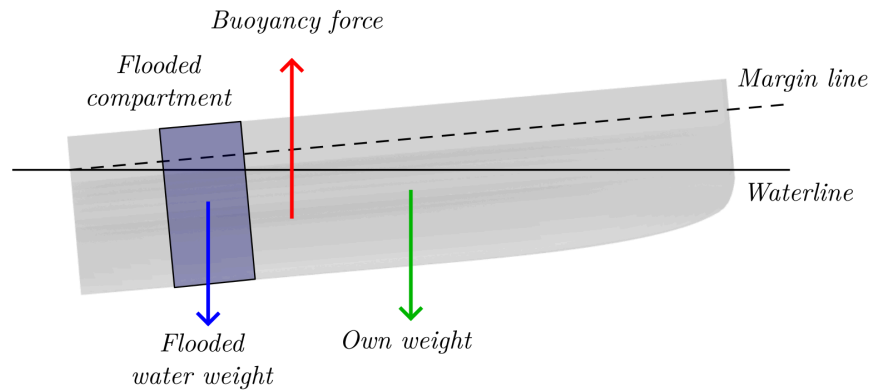


Fig. 11. Forces acting on the damaged hull.

In Fig. 11 above, the vessel is submerged as deep as the margin line allows for a given trim angle. This figure shows the condition for trim by the stern (i.e., the back of the vessel is submerged more than the front). By inspecting the locations of action of three key forces shown (flooded water weight, buoyancy force, ship's weight), it can be seen that apart from the equilibrium of vertical forces, a moment equilibrium equation can be applied as well. Based on these notions, an iterative procedure can be set up for finding the floodable length for different trim angles starting from zero:

- 1) Increasing trim angle;
- 2) Obtaining buoyancy force and its point of action by finding the centroid of submerged hull volumes in *Rhino*;
- 3) Obtaining the necessary flooded water weight from equilibrium condition;
- 4) Verifying that the obtained flooded water weight is positive. If not, equilibrium is not possible and the procedure is finished;
- 5) Defining floodable length as $l_f = \frac{\text{Flooded water weight}}{\text{Transverse section area}}$.

- 6) Obtaining the center of the flooded compartment x_f from moment equilibrium;
- 7) Verifying the following two conditions:
 - a) Obtained center of the flooded compartment is within $(0; L)$ interval. If not, equilibrium is not possible and the procedure is finished.
 - b) Values of $x_f \pm l_f/2$ lie within $(0; L)$ interval. If not, l_f is shortened accordingly so that the flooded compartment bounds do not exceed the physical dimensions of the vessel.

The fact that the deck is flat in the presented design allowed to define the margin line as a straight line. This makes the procedure of varying the trim angle simpler because the increments of this angle happen around a fixed intersection of the waterline and margin line, which is located at the back of the vessel. For examination of the cases where the front of the vessel is submerged more, the increments of the trim angle (now in opposite direction) happen around the intersection at the front.

4.4 Hydrodynamics

The aims of conducting vessel's resistance analysis are:

1. Obtaining the resistance forces for initial power source selection and operating life estimation;
2. Modifying the hull shape in order to minimize resistance forces.

There are several components of resistance force:

- a. Frictional resistance, due to water viscosity;
- b. Air resistance;
- c. Wave-making resistance;
- d. Eddy resistance, due to eddies created by hull appendages, and others.

Manual calculation of the resistance and its components across several hull configurations, parameters, especially during an iterative design, when the values are not fixed, is not practical, so the analysis of hydrodynamic resistance was performed with *MICHLET 9.33*, a piece of software based on Michell's thin-ship theory for wave resistance and viscous resistance of hulls. This software requires an 'in.mlt' text file including an offset table where transverse sections and waterlines define the hull geometry, which is generated by the previously mentioned VB script in *Rhino 3D*.

After defining the input data of the hulls, a specification of the fluid properties, maximum and minimum operational speed, and the number of speed intervals is needed. The most useful feature of *MICHLET* is the computation of the total hydrodynamic resistance and its components, such as:

R_h – hydrostatic resistance on transom stern;

R_f – skin friction resistance;

R_w – total wave resistance;

R_r – residual resistance;

R_t – total resistance.

MICHLET executes simulations for the given speed range, and outputs a geometry summary file, as well as a sequence of files with the total resistance and its components at each speed, combination of which is used to describe hydrodynamics of the vessel and compute resistance graphs. The geometry summary file is used for validation of the imported hull geometry consistency with the hydrostatics analysis mentioned above. In addition, the results are validated using an “*IITC 1957 model-ship correlation line*” frictional resistance estimate for comparison.

The results from hydrodynamics analysis is important for the further development of hull design. Calculation of required power is done with the data extracted from *MICHLET*, and it is a basis for the batteries and motors selection. The breakdown of resistance components helps identify dominant resistance components at the given speed and modify the hull geometry accordingly. For example, if the viscous resistance is high, performance of hull is influenced by the wetted surface rather than wave generation, or if the wave-making resistance is high at moderate speeds, hull slenderness and twin-hull spacing should be taken into consideration.

4.5 Structural Analysis

Having a defined hull geometry, weights of the vessel frame with equipment, and hull material’s properties, it was possible to conduct an early structural analysis of the hull. The analysis was done to obtain the longitudinal distribution of weight, buoyancy, loading, and

resultant shear forces and bending moments. Initially, the hull was assumed to be made out of $t = 8 \text{ mm}$ thick steel for initial hull weight estimation, with an additional 20% for reinforcements. After several interactions, the hull structural reinforcement was designed in more detail, and an accurate weight calculation was made.

For the given longitudinal position x_l , counted from the bow, the weight of the hull part corresponding to $x < x_l$ was estimated as follows:

$$dW_{hull} = (m_{reinforced\ hull}/m_{simple\ hull})\rho_{hull}gt * dS_{surf} ,$$

$$W_{hull}(x_l) = (m_{reinforced\ hull}/m_{simple\ hull}) * \rho_{hull}gt \int_{S(x < x_l)} dS_{surf}$$

where:

dS_{surf} – small surface area of the hull which is located at $x < x_l$,

ρ_{hull} – density of hull material,

$m_{simple\ hull}$ – estimated weight of the hull shell, obtained by multiplying its thickness by surface area.

$m_{reinforced\ hull}$ – actual weight of the hull including its structural members.

For the buoyancy force, following expression is used:

$$dF_{buo} = \rho_{water}g(T - z)dS_{XY} ,$$

$$F_{buo}(x_l) = \rho_{water}g \int_{S(x < x_l \& z < T)} (T - z)dS_{XY}$$

where:

dS_{XY} – XY projection of a small dS surface for which $x < x_l$ and $z < T$,

z – corresponding Z-coordinate of the dS surface centroid.

The integrand essentially represents the upwards hydrostatic pressure component, accounting for the fact that the beginning of the Z-axis corresponds to the keel of the vessel.

Since the equipment is housed on a vessel frame which rests on two supports on hulls as shown in Fig. 12, the loading force F_{load} on one hull can be represented by two downward forces F_{mount1} located at $x = x_{mount1}$ and F_{mount2} at $x = x_{mount2}$. These forces are easily calculated given the weight of frame and equipment, and their CG location. As a result, the loading force for the given longitudinal position x_l is given by:

$$F_{load} = 0 \quad \text{for } x_l < x_{mount1},$$

$$F_{mount1} \quad \text{for } x_{mount1} < x_l < x_{mount2},$$

$$F_{mount1} + F_{mount2} \quad \text{for } x_l > x_{mount2}.$$

Later, after the batteries were chosen and placed appropriately, the loading force was also modified by superimposing a uniformly distributed battery weight.

The visual representation of the forces is illustrated in Fig. 13. It should be noted that these are cumulative forces.

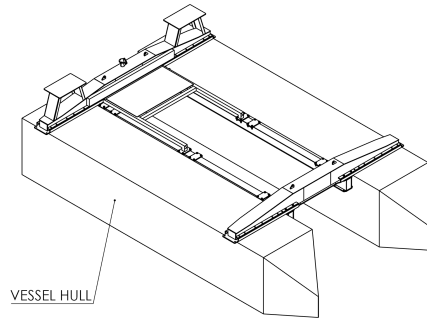


Fig. 12. Vessel Frame and Hulls.

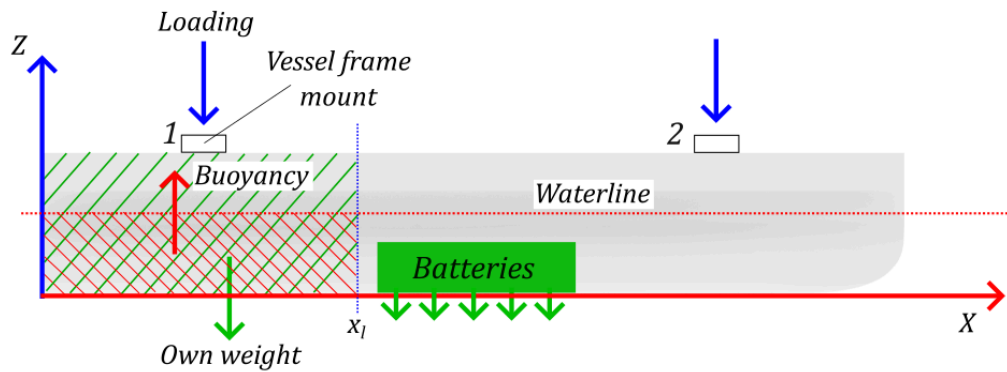


Fig. 13. Forces and loads illustrated.

The resultant of these forces is the shear force inside the hull section. By calculating it for different x_l coordinates, its longitudinal distribution is obtained:

$$SF(x_l) = F_{buo}(x_l) - W_{hull}(x_l) - F_{load}(x_l)$$

It also can be integrated with respect to x_l to produce a bending moment curve:

$$BM(x_l) = \int_0^{x_l} SF(x_l) dx_l$$

By obtaining the transverse cross-section of the hull body for different longitudinal X-coordinates, normal and shear stresses can be evaluated as well. Section properties, such as its area $A_{section}$, centroid height $z_{centroid}$, and moment of inertia I_y are obtained using *Rhino*. The average shear stress is given by:

$$SS(x_l) = \frac{SF(x_l)}{A_{section}}$$

And the maximum normal stress is calculated by modelling the hull as a beam:

$$NS(x_l) = \frac{BM(x_l) * z_{centroid}}{I_y}$$

4.6 Compliance with RINA Rules for Classification

This section describes the main assessment of section modulus and thicknesses of vessel structure parts, and calculations of hull bending moments, pressure and normal stresses following the procedure presented in Part B, Section 8 – “Ships Less than 90 m in Length” of RINA classification rules [21, Pt. B]. Detailed supporting calculations and parameters are provided in Appendix C. The initial parameters for calculations were taken as described in Table 3:

Table 3. Initial parameters used in calculations.

Water density $\rho_w, kg/m^3$	1025
Length L, m	10
Breadth B, m	1.2

Block coefficient C_B	0.652
Moment of inertia I_y, m^4	0.063
Z-centroid N, m	1.410
Area of midship section $A_{section}, m^2$	0.097
Height h, m	2.4
Operational speed $V, knots$	2
Draft T, m	0.848

In RINA Rules, the characteristics of our vessels corresponds to the requirements given in this chapter, as follows:

Table 4. Requirements check with vessel characteristics.

Requirements	Vessel characteristics
$L < 90 m$	$L = 10 m$
$L/B > 5$	$L/B = 8.33$
$B/D < 2.5$	$B/D = 1.42$
$C_B \geq 0.6$	$C_B = 0.65$

The following calculations are performed for the midship section of the hull.

4.6.1 Bending Moments

The vertical wave bending moments are calculated for hogging and sagging conditions, respectively [21, Pt. B, pp. 330-331]:

$$M_{WV,H} = 190F_M nCL^2 BC_B 10^{-3},$$

$$M_{WV,S} = -110F_M nCL^2 B(C_B + 0.7) 10^{-3}$$

The still water bending moments at amidships are obtained for hogging and sagging conditions, respectively [21, Pt. B, p. 330]:

$$M_{SWM,H} = 175n_1 CL^2 B(C_B + 0.7) 10^{-3} - M_{WV,H},$$

$$M_{SWM,S} = 175n_1 CL^2 B(C_B + 0.7) 10^{-3} + M_{WV,S}$$

Table 5. Results of bending moments calculations.

Vertical wave bending moment for hogging $M_{WV,H}$, kNm	13.613
Vertical wave bending moment for sagging $M_{WV,S}$, kNm	– 16.338
Still water bending moment for hogging $M_{SWM,H}$, kNm	15.628
Still water bending moment for sagging $M_{SWM,S}$, kNm	12.903

4.6.2 Normal Stresses for Yielding Check

The calculations for normal stresses of the hull induced by vertical bending moments are calculated using the corresponding formula in [21, Pt. B, p. 341]:

$$\sigma_1 = \frac{M_{SW} + M_{WV}}{Z}$$

As the goal of assessment is to check the compliance of the design with maximum and minimum limitations, the bending moments corresponding to hogging were considered. Moreover, the allowable normal stress was calculated with the next equation [21, Pt. B, p. 341]:

$$\sigma_{1,ALL} = \frac{175}{k} (N/mm^2)$$

where:

k – material factor of the hull steel, which equals to 1.

Table 6. Normal stresses at different points of the hull transverse sections.

	Maximum allowable	Actual
Normal stress at bottom σ_{1b} , N/mm^2	175.000	0.659
Normal stress at deck σ_{1d} , N/mm^2	175.000	0.463

4.6.3 Section Moduli

The equation for the section modulus at any point of a hull transverse midship section is [21, Pt. B, p. 340]:

$$Z_A = \frac{I_Y}{|z-N|}$$

Therefore, the midship section moduli at bottom and deck were obtained, respectively [21, p. 340]:

$$Z_{AB} = \frac{I_Y}{N},$$

$$Z_{AD} = \frac{I_y}{z_D - N}$$

where:

V_D – vertical distance from the centroid to the strength deck
($V_D = z_D - N = 0.99 \text{ m}$).

Minimal required section modulus for ships with block coefficient $C_B \leq 0.8$, the gross section moduli at the midship section is calculated as follows [21, Pt. B, p. 342]:

$$Z_{min} = n_1 CL^2 B (C_B + 0.7) k \cdot 10^{-6}$$

Table 7. Section modulus calculation results.

	Minimum	Actual
Section modulus at bottom Z_{AB}, m^3	0.00017	0.044
Section modulus at deck Z_{AD}, m^3	0.00017	0.063

4.6.4 Still Water and Wave Pressures

Still water and inertial pressures were obtained using Table 6 of [21, Pt. B, p. 333]. According to Fig. 14, it can be seen that the highest-pressure value is at the bottom for still water and at the draft distance for wave conditions.

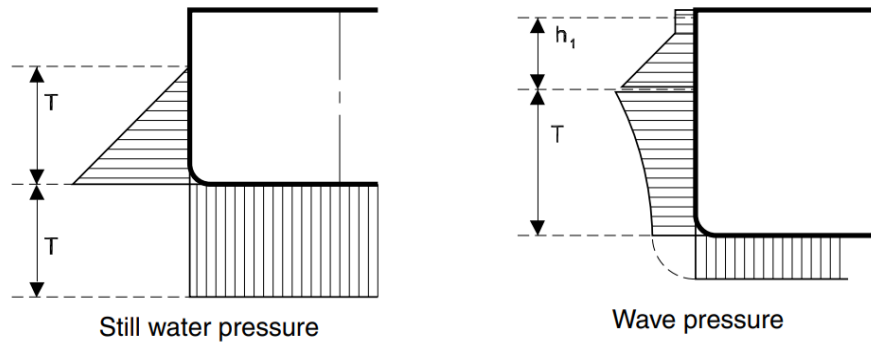


Fig. 14. Pressure value schematic of the hull midship section for still water and wave conditions [21, p. 333].

Therefore, the highest values of still water and wave pressure were computed by [21, Pt. B, p. 333]:

$$p_{sw} = \rho g (T - z)$$

$$p_w = \rho g (T + h_1 - z)$$

Table 8. Pressure calculation for still water and wave conditions.

Still water pressure (at bottom) $p_{sw}, kN/mm^2$	8.484
Wave pressure (at waterline) $p_w, kN/mm^2$	5.202

4.6.5 Minimum thickness of vessel structure parts

Minimum thicknesses of bottom plating and section modulus of ordinary stiffeners for double bottom structures are calculated as given in the Table 4 and Table 5 of [21, Pt. B, pp. 382-385]:

$$t_{bp} = (0.033L + 6.5) \left(\frac{sk}{0.46 + 0.0023L} \right)^{\frac{1}{2}} - 1$$

$$w_{os} = 1.2s\ell^2 p_d Ck$$

$$t_{p_longitudinal} = 0.054Lk^{\frac{1}{2}} + 4.5$$

where:

s – spacing of ordinary stiffeners, which equals to 0.3 m;

ℓ – span of ordinary stiffeners, which equals to 1 m.

Table 9. Minimum value of section modulus for bottom structures.

	Minimum	Actual
Section modulus of stiffeners w_{os}, cm^3	4.680	14.283

Table 10. Minimum values of thickness and section modulus for side structures.

	Minimum	Actual
Thickness of plating in case of having longitudinal reinforcements $t_{p_longitudinal}, mm$	7.060	8.000
Section modulus of longitudinal stiffeners w_l, cm^3	20.000	21.177

Minimum thickness of deck plating and minimum section modulus of ordinary stiffeners for deck structures were calculated using Table 7 of [21, Pt. B, p. 388]:

$$t_d = s(1.4L^{\frac{1}{2}} - 1.1)(\frac{F_{DS}}{k})^{\frac{1}{2}}$$

$$w_{os} = 0.75C_1C_2s\ell^2(p_0 + p_1)k$$

Table 11. Minimum values of thickness and section modulus for deck structures.

	Minimum	Actual
Deck thickness t_d , mm	7.520	8.000
Section modulus of deck stiffeners w_{os} , cm^3	5.470	14.283

Minimum thickness of watertight bulkheads were calculated using Table 12 of [21, Pt. B, p. 392]:

$$t_{bh_g} = 3.8s(hk)^{\frac{1}{2}}$$

$$t_{bh_c} = 4.35s(hk)^{\frac{1}{2}}$$

Table 12. Minimum values of thickness for watertight bulkheads.

	Minimum	Actual
General bulkhead thickness t_{bh_g} , mm	5.240	6.000
Collision bulkhead thickness t_{bh_c} , mm	5.990	6.000

Overall, the actual values of design parameters were selected with consideration of the requirements specified in RINA Classification Rules.

Chapter 5: Results and Discussion

5.1 Hull geometry selection

The first generation of the hull geometry with rough estimates of its parameters used as an input yielded the following shape shown in Fig. 15.



Fig. 15. Initial hull geometry.

As can be seen, this hull has a long, sharp bow which minimizes its resistance. However, its deck (upper surface) was not suitable for installation of the provided vessel frame and equipment. It also failed to provide sufficient buoyancy force. Nevertheless, it was used for testing the initial spiral design and optimization loop and for improving the developed Visual Basic scripts².

After several design iterations aimed at getting more suitable deck shape, the following geometry of the vessel was achieved:

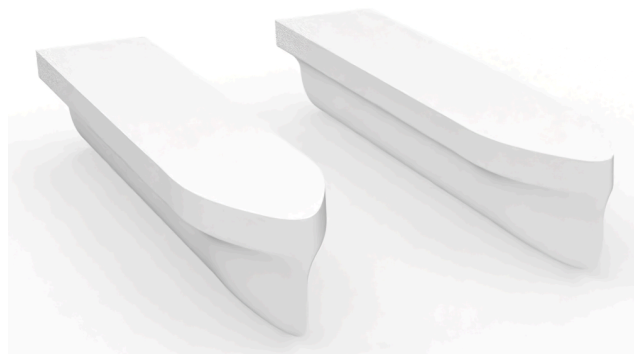


Fig. 16. Intermediate hull geometries.

² See Appendix A

This geometry did not provide sufficient displacement after the components and other loads were added throughout the design process, so it was further modified to accommodate larger displacements. In addition, a slight narrowing of the aft parts was added to improve the resistance characteristics and to shift the center of buoyancy. The final design is shown below in Fig. 17:

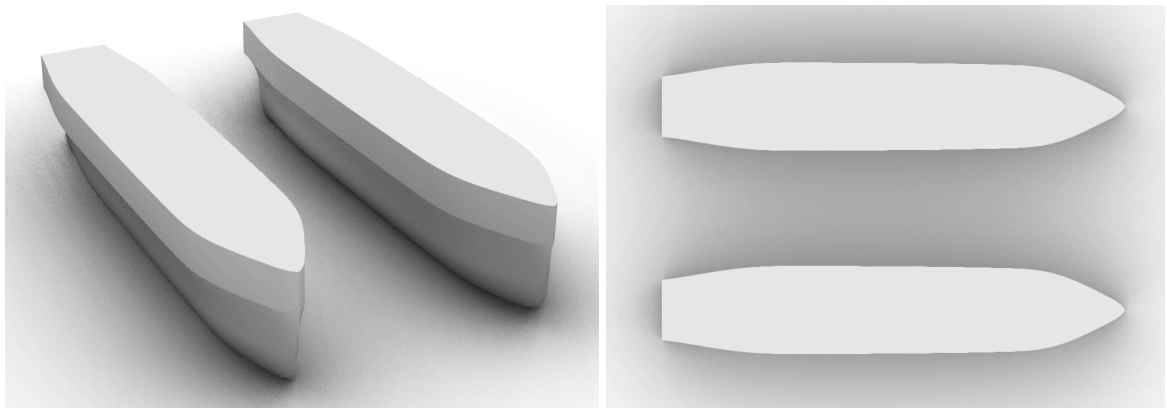


Fig. 17. Current vessel design: Perspective (left) and Top (right) views.

The overall dimensions of a single hull are:

Table 13. Hull dimensions.

Length (m)	Width (m)	Height (m)
10.00	1.91	2.40

5.2 Intact and Damaged stability

The hydrostatic analysis for the above vessel geometry was performed for different drafts. The results are plotted in Fig. 18:

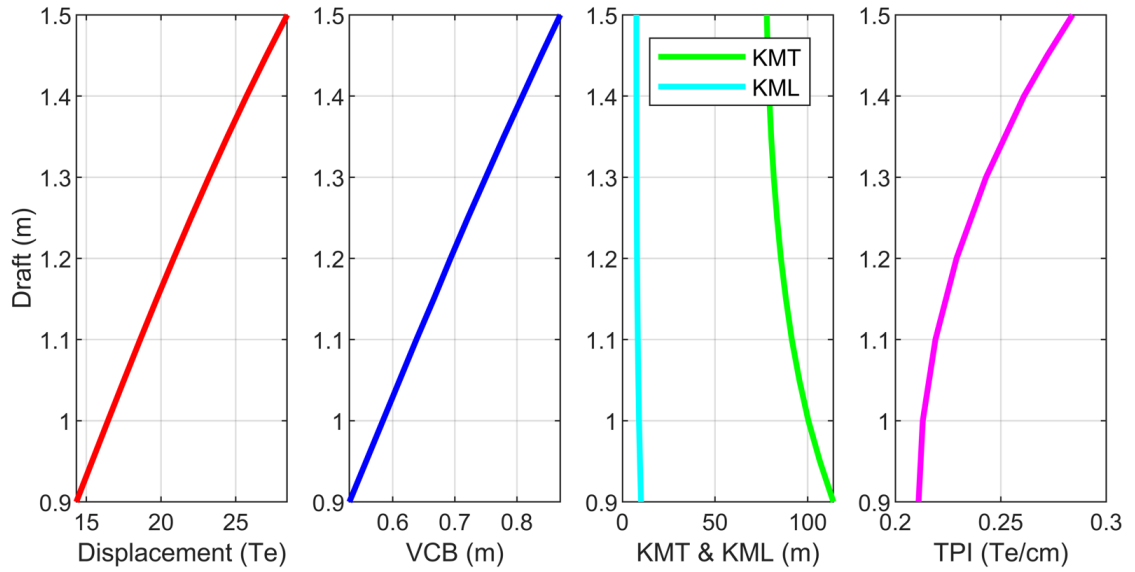


Fig. 18. Hydrostatic analysis results.

The relevant values are shown in Table 14:

Table 14. Hydrostatic analysis results.

Draught (m)	Displacement (Ton equiv.)	VCB (m)	LCB (m)	TCB (m)	GMT (m)	GML (m)
0.900	14.328	0.530	4.718	0.000	112.291	8.443
1.000	16.445	0.584	4.730	0.000	98.795	7.375
1.100	18.602	0.638	4.738	0.000	89.847	6.701
1.200	20.842	0.693	4.744	0.000	83.959	6.293
1.300	23.203	0.750	4.748	0.000	80.102	6.095
1.400	25.723	0.809	4.749	0.000	77.682	5.964
1.500	28.445	0.870	4.750	0.000	76.329	5.952

The key values for assessing the vessel's stability – its metacentric heights *GMT* and *GML* – are sufficiently high at all drafts, ensuring that the stability requirements are met.

Regarding the volume displacement, the results show that the vessel provides $V_{min} = 14.3 \text{ tons}$ at minimum draft, and $V_{max} = 28.5 \text{ tons}$ at maximum draft. The maximum displacement is higher than the maximum possible weight of the vessel as it will be shown later. In addition, at maximum draft there is an additional 0.9-meter headroom to

the deck, and hulls flare outward which will provide progressively more displacement if the draft increases.

A layout of the key components of the vessel are shown on Fig. 19:

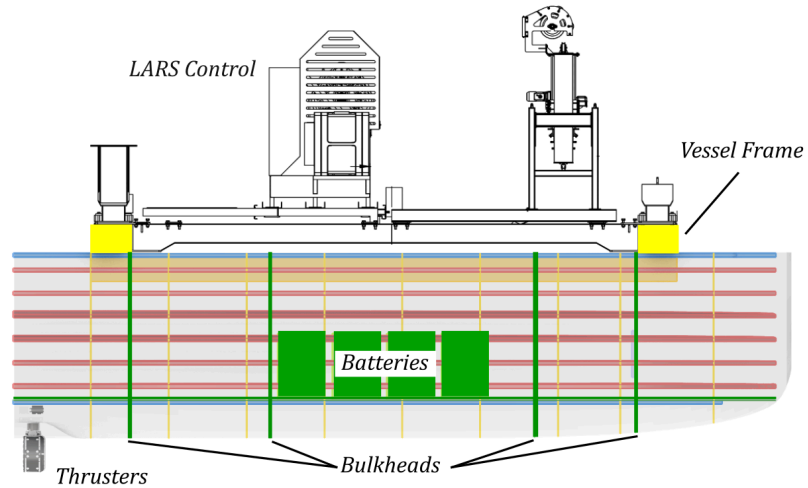


Fig. 19. Component layout.

For the layout above, the masses and locations of centers of gravity of the hulls and major components is summarized in the table below:

Table 15. Masses and centers of gravity.

Item	Mass (kg)	CG x-coordinate (m)	CG y-coordinate (m)	CG z-coordinate (m)
Hulls combined	13756	4.734	0.000	1.408
Batteries combined	3840	4.750	0.000	0.930
Motors combined	200	0.230	0.000	0.000
Vessel Frame	1665	4.762	0.000	2.400
LARS Control	2065	5.419	0	3.442
Payload (optional)	5000	-	-	-
Total	21526	4.763	0.000	1.582
with Payload:	26526	-	-	-

It should be noted that the positions of the components above were not obtained right away. Relatively heavy components that affect the overall center of gravity, such as the Vessel

Frame, LARS Control, and Batteries, were shifted multiple times to make the centers of gravity and buoyancy match. As the hulls were modified to include the structural reinforcements, these adjustments were made iteratively to account for these changes as well.

The obtained center of gravity of the vessel is used for calculating the equilibrium position:

Table 16. Static balancing results.

Draft (m)	Heel angle (deg)	Trim angle (deg)
1.309	0.00	-0.24

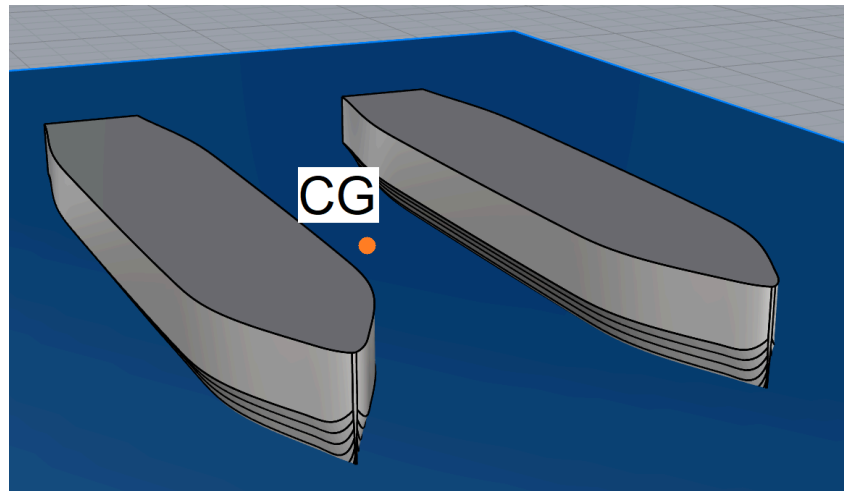


Fig. 20. Water surface for balanced condition.

The obtained equilibrium draft is within the required range. The heel angle is non-existent since the vessel was designed to be symmetrical. As for the trim, the angle is insignificant, meaning that the deck is almost horizontal.

For a final comparison, the positions of centers of gravity and buoyancy for the equilibrium draft are shown together:

Table 17. Centers of gravity and buoyancy of the vessel.

	X-coordinate (m)	Y-coordinate (m)
CG	4.763	0.000
CB	4.748	0.000

It can be seen that their key positions, longitudinal (x) and transverse (y) ones, are nearly identical. Therefore, the matching of CG and CB was achieved.

Damage stability analysis was performed on the current design to determine the floodable length curve of the vessel. The results are shown below in Fig. 21. Based on the maximum displacement curves, locations of the watertight bulkheads were determined. Bulkheads were placed in a way that ensures that the vessel stays afloat even if one compartment is fully flooded. The longitudinal positions of the bulkheads coincide with the beginnings of flooding line triangles (shown in pink, Fig. 21).

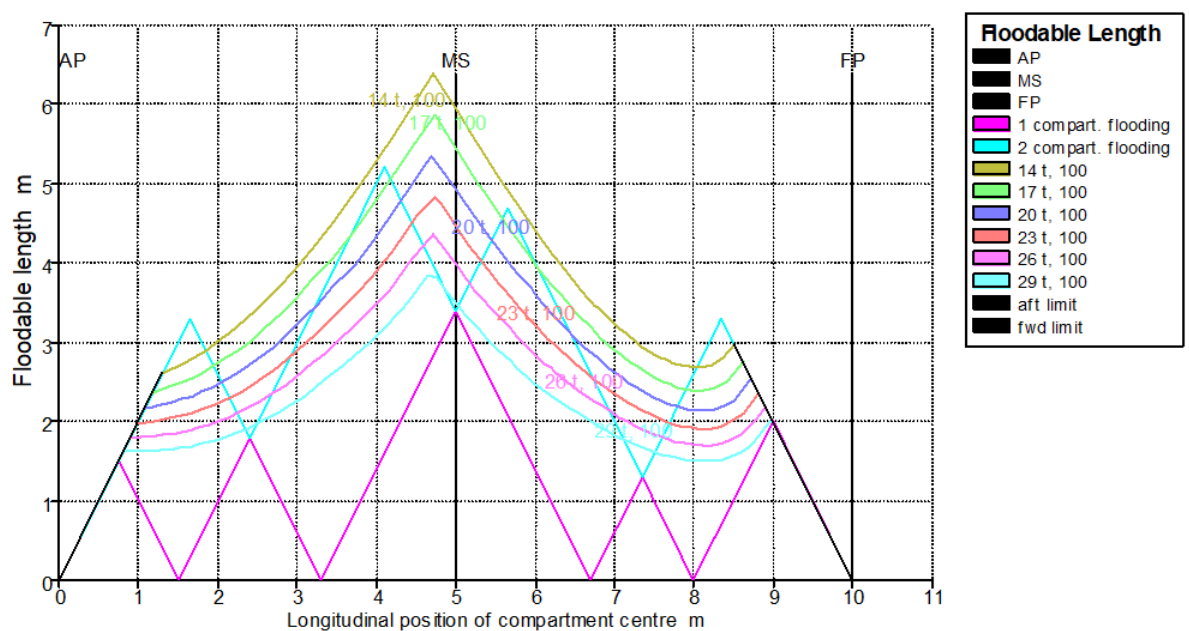


Fig. 21. Floodable length curves for different displacement and compartment flooding lines.

A floodable length curve obtained manually using a simplified procedure is also shown below:

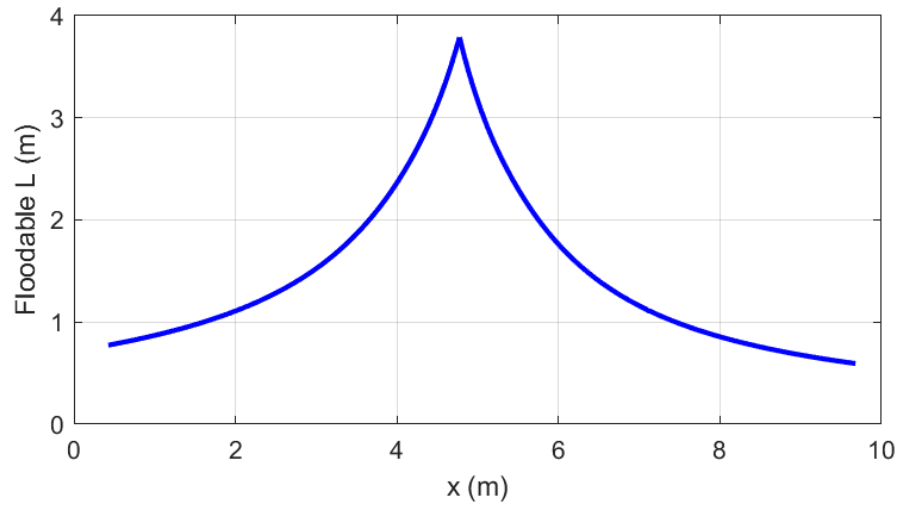


Fig. 22. Manually obtained floodable length curve.

It can be seen that the curve above is similar to the curve obtained from the software.

5.3 Resistance and Propulsion

The hydrodynamic resistance analysis of the vessel was performed in *MICHLET* for the design draft of 1.3 meters:

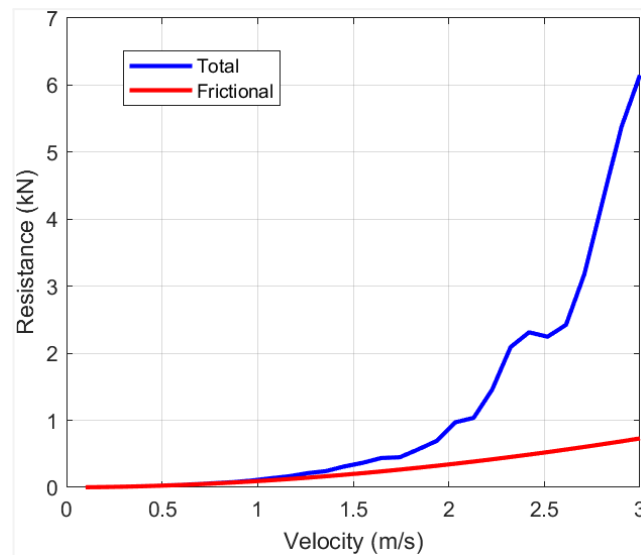


Fig. 23. Hydrodynamic resistance analysis results.

An excerpt of these results is also shown in Table 18.

Table 18. Resistance analysis results.

$v \text{ (m/s)}$	$R_F \text{ (N)}$	$R_{total} \text{ (N)}$
1	105.5	134.8
3	729.7	6148.5

To validate the output results, the values needed for estimating the frictional resistance using “*ITTC 1957 model-ship correlation line*” were obtained first. For the given design draft, the wetted surface area of one hull was calculated to be:

$$S_{wet} = 30.46 \text{ m}^2$$

The operating velocity is specified as 2 knots, or $v = 1 \text{ m/s}$, and the waterline length for the design used is $L_{wl} = 10 \text{ m}$. Water kinematic viscosity for given conditions was taken to be $\nu = 10^{-6} \text{ m}^2/\text{s}$, and its density to be $\rho = 1025 \text{ kg/m}^3$.

The resultant Reynolds number is:

$$Re = \frac{1 \text{ m/s} * 10 \text{ m}}{10^{-6} \text{ m}^2/\text{s}} = 10^7,$$

and the frictional resistance coefficient is:

$$C_F = \frac{0.075}{(\log_{10} 10^7 - 2)^2} = 0.003.$$

Thus, the frictional resistance for one hull is:

$$R_{F1} = 0.003 * \frac{1}{2} * 1025 \text{ kg/m}^3 * 30.46 \text{ m}^2 * (1 \text{ m/s})^2 = 46.8 \text{ N}$$

For the vessel (consisting of two hulls), the frictional resistance is then:

$$R_F = 2R_{F1} = 93.7 \text{ N}$$

Similarly, for the maximum speed of $v = 3 \text{ m/s}$:

$$R_F = 843.0 \text{ N}$$

The obtained frictional resistance values are close to those provided by *MICHLET*.

Taking the total resistance value at $v = 3 \text{ m/s}$ from *MICHLET*, the theoretical demand for propulsion power can be estimated:

$$P_{vmax} = R_{total} v_{max} = 6148.5 \text{ N} * 3 \text{ m/s} = 18\,445 \text{ W}.$$

5.4 Component Selection

The power and energy calculations were defined with the sequential methodology including hydrodynamic analysis, component specifications, and requirements of the project. The resistance value from numerical simulations in *MICHLET* corresponded to the 0.135 kW of power for operating speed. Assuming a motor efficiency of 90% [23], the effective power (EP) was calculated. According to the two-weeks endurance requirement of the project, energy demand for propulsion was calculated for that period of time. Then, to provide long-term battery use, 80% depth of discharge limit percentage was taken into account, then the value was rounded to overcome uncertainties and unexpected fluctuations. Calculations are shown below:

$$EP = \frac{0.135}{0.9} = 0.15 \text{ kW}$$

$$\text{Energy Demand} = 0.96 \text{ kW} * 14 \text{ days} * 24 \text{ hours} = 50.4 \text{ kWh}$$

$$\text{DoD \& losses: } \frac{50.4}{0.80} = 63 \text{ kWh} \approx 65 \text{ kWh}$$

While the value of 65 kWh represents the theoretical propulsion energy requirement at design speed, final energy of 350 kWh was taken as an engineering design value, which significantly differs from the calculated value to compensate for the increased power consumption from rapid rise of resistance with the speed increase. Additionally, it includes conversion inefficiencies of the system, operational safety margins, and 20-50% emergency reserve. After calculation of the energy requirement, the battery was selected to be a modular lithium-ion marine battery. Lithium-ion batteries are widely used in marine applications due to their stable cycle life, high energy density, and relatively low mass. For the configuration of 350 kWh, the 64 modules of marine certified Corvus Orca Energy (Fig. 24) multi-module systems with 5.6 kWh per module were chosen, so the total weight of batteries will be 3840 kg. Corvus Orca ESS provides a flexible arrangement of modules that is especially useful for twin hull vessels, has a battery management system, and maintains a strong safety quality.

The propulsion motors were chosen to have more than 18 kW for both hulls for redundancy and stability considerations. For the purpose of providing required motor efficiency of more than 90%, low-speed hydrodynamic loading torque, and compatibility with the batteries, the permanent-magnet steerable marine POD 25.0 (Fig. 25) motors were chosen. The motors each are rated for 25 kW power, have minimal maintenance, high reliability, and are controlled remotely.

In addition, the battery modules were distributed across both hulls, which addresses the redundancy requirement and allows the USV to continue its function in case of power or propulsion system failure in one hull. Such operation is possible since the selected motors are steerable, meaning that a single motor can control both position and orientation of the vessel.

According to the specifications of the selected motor, its nominal voltage is 96V [23]. Each module of the selected battery system provides 50V [22]. To match these voltages, the battery pack has to be configured as two parallel module groups connected in series. This results in each hull having a 2S18P configuration battery system.



Fig. 24. Corvus Orca Energy [22].



Fig. 25. POD 25.0 motor [23].

The vessel's onboard electronic and powertrain components generate heat during operation, especially during continuous use. It is crucial to lower the internal temperature to ensure that reliability, efficiency and lifespan of the equipment are not negatively affected by the increased heat. Therefore, a cooling system is required to maintain stable conditions inside the vessel. To achieve proper cooling of the components, a marine-rated and compact BlueCool S-Series 230 V forced-air cooling system was considered.

Assuming that the battery discharge efficiency is 95% and approximate power required per motor is 10 kW, the heat produced from the batteries in each hull can be estimated:

$$Q_{battery} = P_{battery} \left(\frac{1}{\eta} - 1 \right) = 10 \text{ kW} * 0.0526 = 0.526 \text{ kW}$$

Adding a safety margin of 30% to account for ambient temperature variation, ageing, and additional losses, the heat released was estimated to be:

$$Q_{battery} = 1.3 * 0.526 \text{ kW} = 0.683 \text{ kW} = 2333 \text{ BTU/h}$$

$$Q_{total} = 2 * Q_{battery} = 1.366 \text{ kW} = 4660 \text{ BTU/h}$$

Considering the amount of heat load to be dissipated, BlueCool S6 230 V forced-air cooling system was selected that can provide a cooling capacity of 6000 BTU/h, equivalent to 1.8 kW [24]. The given cooling capacity allows to cover the final thermal load of the vessel, which involves additional possible heat from motor controllers and power supply systems. Therefore, it was decided to place one cooling system per hull, as it fully covers cooling needed and improves system redundancy. The system operates with a 230 V supply voltage, making it compatible with a standard onboard AC electrical system.



Fig. 26. BlueCool S-Series 230 V [24].

5.5 Structural Analysis

For reference, the employed coordinate system is shown on Fig. 27.

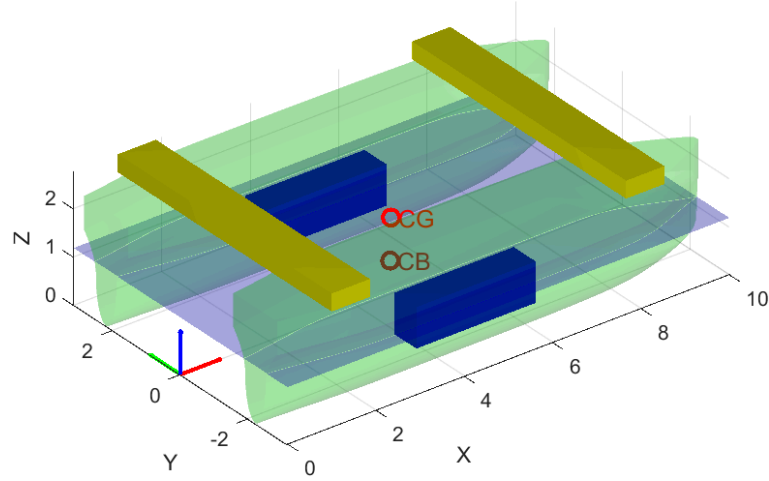


Fig. 27. Vessel coordinate system.

Following the procedure of integrating and calculating the forces acting on hull for different x -coordinates, as described in methodology, their cumulative distribution and the resultant shear force (Fig. 28) were obtained for the design draft (using the same masses and component layout as was shown before):

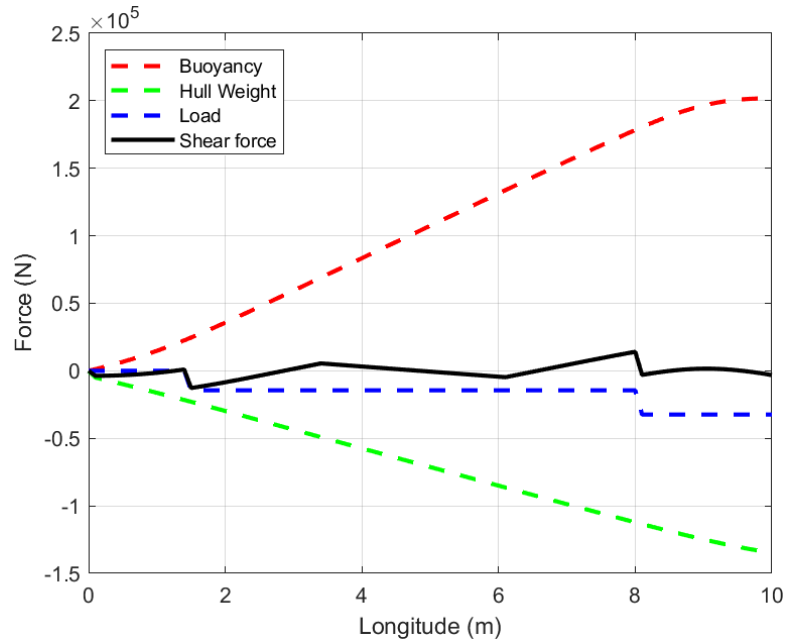


Fig. 28. Cumulative distributions of forces.

Since the cross-sections of the hulls are uniform for most of their length, properties of cross-section were taken to be constant lengthwise. These were obtained in *Rhino*:

$$A_{section} = 0.079 \text{ m}^2,$$

$$z_{centroid} = 1.39 \text{ m},$$

$$I_y = 0.0541 \text{ m}^4$$

The resultant plot of normal stress is drawn in Fig. 29:

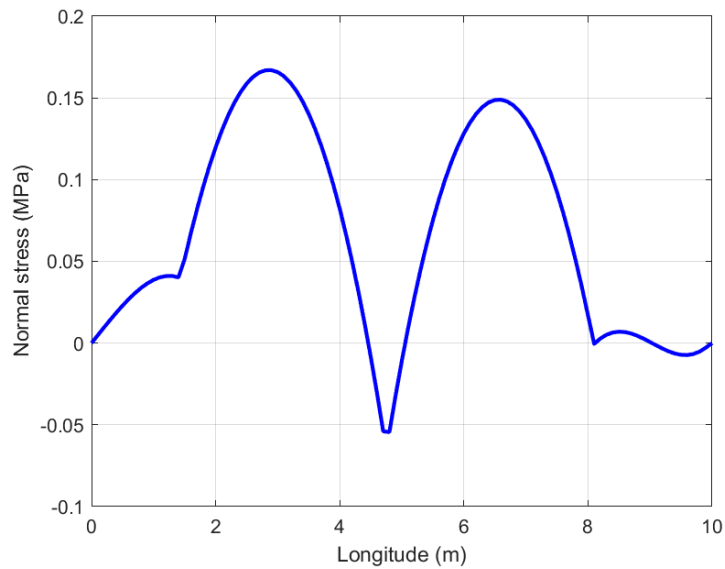


Fig. 29. Normal stress plot.

The resultant stresses are far lower than the yield strength of steel.

In order to verify that the vessel structure is safe, a Finite Element Analysis was carried out. The hull was assumed to have fixed supports in the places where the vessel frame is mounted, and hydrostatic pressure was applied for the equilibrium draft:

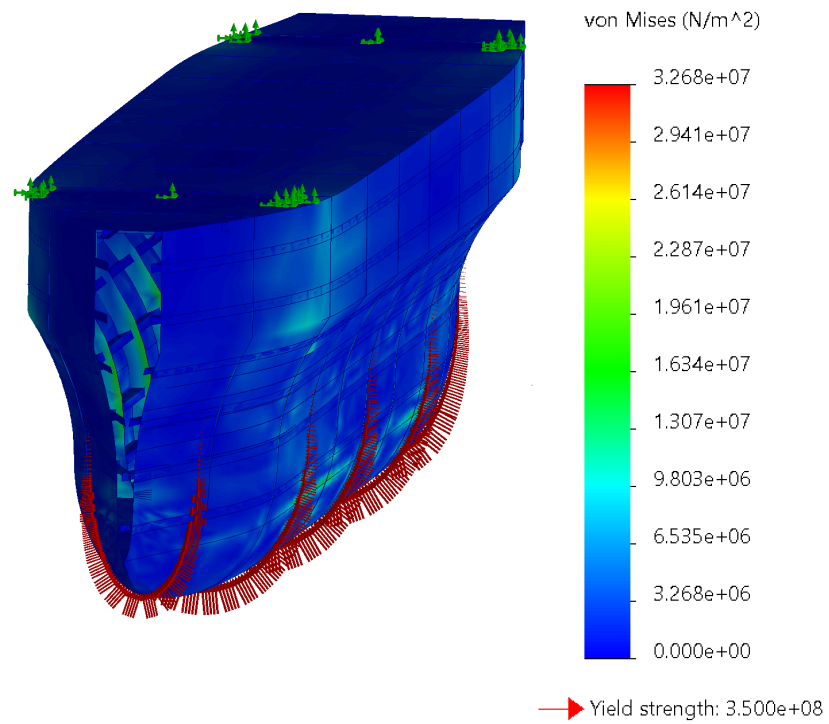


Figure 30. Finite Element Analysis of the hull structure.

It can be seen that the maximum stress is considerably lower than the yield stress, giving a factor of safety of more than 10. Therefore, it can be concluded that the vessel is structurally safe.

Chapter 6: Conclusions and Future Work

6.1 Conclusion

The work done in this project presents not only the design of a vessel, but also the design framework that was followed for achieving the final result and that can be followed for other similar projects. The design, presented and analyzed in this work, successfully addresses the project objectives and industry requirements. Namely, it satisfies the payload capacity, stability, endurance, structural integrity, and certification requirements while remaining safe to operate in the mentioned environment.

In terms of manufacturing, the presented design includes the geometry³ of the hulls, the choice of propulsion and energy system components, key dimensions of the structural members of the hull, and defines the placement of these components, members, and equipment.

This project required researching and studying several new concepts related to naval and marine engineering that are beyond the standard mechanical engineering curriculum. Nevertheless, they all are derived from the basics of mechanics, which provided us with an opportunity to leverage the knowledge obtained through the past years of studies for understanding those concepts effectively.

6.2 Future Work

For future work, it is recommended to design detailed drawings and manufacturing documentation of the vessel. In order to transition from design to fabrication phase, a fully dimensioned CAD model, details of welding, assigned tolerances, and bill of materials are essential. It may be necessary to reposition the access points to the hull interior (i.e. deck and bulkhead hatches) according to the manufacturing methods and equipment to be used. Additionally, a validated control and navigation system for USV should be implemented. This includes implementation of control algorithms within the simulation environment, specifically focus should be on the trajectory tracking and stability. Therefore, the long-term goal is to develop a fully defined vessel with an advanced control framework.

³ CAD files available upon request.

References

- [1] M. Oldenburg, X. Baur, and C. Schlaich, "Occupational Risks and Challenges of Seafaring," *Journal of Occupational Health*, vol. 52, no. 5, pp. 249–256, Jul. 2010, doi: <https://doi.org/10.1539/joh.K10004>
- [2] Webmaster, "Publications - Annual Overview of Marine Casualties and Incidents 2024 - EMSA - European Maritime Safety Agency," Europa.eu, Nov. 20, 2024. Available: <https://www.emsa.europa.eu/publications/reports/item/5352-annual-overview-of-marine-casualties-and-incidents-2024.html>
- [3] C. Zhao, P. Thies, J. Lars, and J. Cowles, "ROV launch and recovery from an unmanned autonomous surface vessel – Hydrodynamic modelling and system integration," *Ocean Engineering*, vol. 232, p. 109019, Jul. 2021, doi: <https://doi.org/10.1016/j.oceaneng.2021.109019>
- [4] Y. Wang, W. Zhou, M. Fei, and H. Yan, "An Unmanned Surface Vehicle for the Launch and Recovery of Autonomous Underwater Vehicles: A Novel Design," *IEEE Robotics & Automation Magazine*, vol. 31, no. 1, pp. 53–61, Jan. 2024, doi: <https://doi.org/10.1109/mra.2023.3348302>
- [5] M. H. M. Ghazali, M. H. A. Satar, and W. Rahiman, "Unmanned surface vehicles: From a hull design perspective," *Ocean Engineering*, vol. 312, p. 118977, Aug. 2024, doi: <https://doi.org/10.1016/j.oceaneng.2024.118977>
- [6] K. Wróbel, J. Montewka, and P. Kujala, "Towards the assessment of potential impact of unmanned vessels on maritime transportation safety," *Reliability Engineering & System Safety*, vol. 165, pp. 155–169, Sep. 2017, doi: <https://doi.org/10.1016/j.ress.2017.03.029>
- [7] C. Vossen, R. Kleppe, and S. R. Hjørungnes, "Ship Design and System Integration," presented at the DMK 2013, Dec. 2013. Available: https://www.researchgate.net/publication/273026917_Ship_Design_and_System_Integration
- [8] S. Kale, "Developments in unmanned surface vehicles (USVs): A Review," *International Conference on Applied Engineering and Natural Sciences*, vol. 1, no. 1, pp. 596–600, Jul. 2023. Available: <https://doi.org/10.59287/icaens.1064>
- [9] O. Landstad, H. S. Halvorsen, H. Øveraas, V. Smines, and T. A. Johansen, "Dynamic positioning of ROV in the wave zone during launch and recovery from a small surface vessel," *Ocean Engineering*, vol. 235, p. 109382, Jun. 2021, doi: <https://doi.org/10.1016/j.oceaneng.2021.109382>
- [10] C. Kuo, "A Controlled Handling Method For Effective Offshore Support Operations," *Offshore Technology Conference*, May 1978, doi: <https://doi.org/10.4043/3318-ms>
- [11] G. Wu, M. Zhao, Y. Cong, Z. Hu, and G. Li, "Algorithm of Berthing and Maneuvering for Catamaran Unmanned Surface Vehicle Based on Ship Maneuverability," *Journal of Marine Science and Engineering*, vol. 9, no. 3, p. 289, Mar. 2021, doi: <https://doi.org/10.3390/jmse9030289>
- [12] A. Łebkowski and W. Koznowski, "Analysis of the Use of Electric and Hybrid Drives on SWATH Ships," *Energies*, vol. 13, no. 24, p. 6486, Dec. 2020, doi: <https://doi.org/10.3390/en13246486>
- [13] E. V. Lewis, *Principles of Naval Architecture: Stability and strength*. Society of Naval Architects & Marine Engineers, 1988.
- [14] E. V. Lewis, *Principles of Naval Architecture: Resistance, Propulsion and Vibration*. Society of Naval Architects & Marine Engineers, 1988.
- [15] J. S. Carlton, *Marine propellers and propulsion*. 2018. doi: 10.1016/c2014-0-01177-x.
- [16] M. K. Iz, "The powering performances of large waterjet and propeller driven catamarans at medium-speed," *UTAS Research Repository*, Jan. 2019, doi:

10.25959/100.00034064.

[17] A. Eslamdoost, “The hydrodynamics of Waterjet/Hull interaction,” 2014. [Online]. Available:

<http://publications.lib.chalmers.se/publication/208838-the-hydrodynamics-of-waterjethull-interaction>

[18] G. De Melo Rodriguez, I. Echevarrieta, and J. M. Serra, “Improving the efficiency of a high speed catamaran through the replacement of the propulsion system,” *TransNav the International Journal on Marine Navigation and Safety of Sea Transportation*, vol. 9, no. 4, pp. 531–535, Jan. 2015, doi: 10.12716/1001.09.04.09.

[19] “Trajectory tracking using Sliding Mode Control for Autonomous Surface Vessel,” *IEEE Conference Publication | IEEE Xplore*, Dec. 01, 2016.

<https://ieeexplore.ieee.org/stamp/stamp.jsp?tp=&arnumber=7859948>

[20] S. A. Wicaksono, D. Chrismianto, S. Sunusi, H. Boesono, and I. Setiyanto, “Technical analysis of azimuth thruster and conventional systems,” *Indian Journal of Science and Technology*, vol. 12, no. 39, pp. 1–8, Oct. 2019, doi: 10.17485/ijst/2019/v12i39/147589.

[21] “RINA Rules,” *RINA.org*. <https://www.rina.org/en/rules>

[22] “Corvus Orca ESS | Products | Corvus Energy,” *Corvusenergy.com*, 2016.

<https://corvusenergy.com/products/corvus-orca-ess#specs>

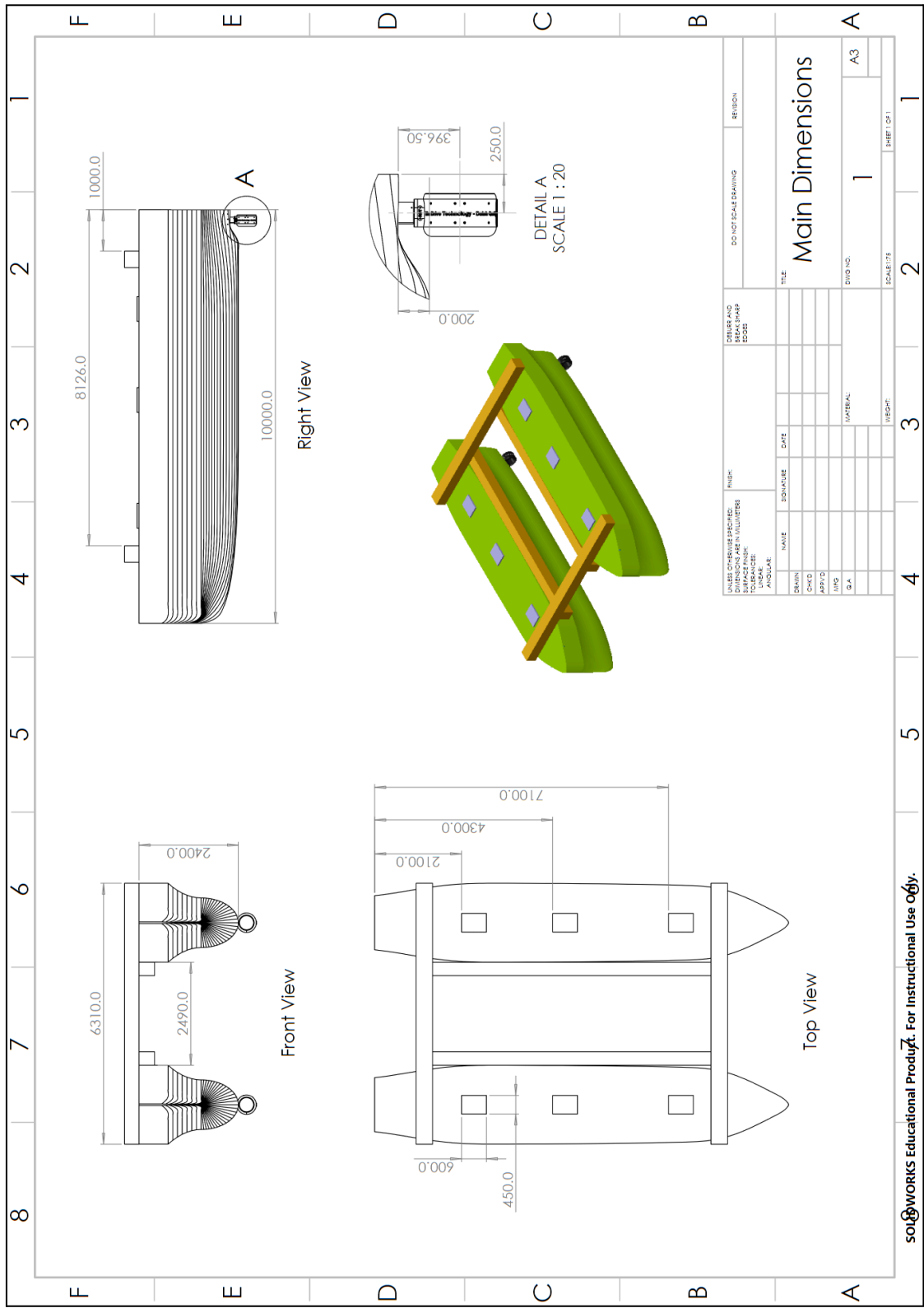
[23] “Electric propulsion POD boat - POD 25.0-S - Rim Drive Technology,” *Rim Drive Technology*, Oct. 2025. <https://rimdrivetechology.nl/product/electric-steerable-pod-25-0/> (accessed Dec. 01, 2025).

[24] “BlueCool S-Series,” *Webasto.com*, 2026.

<https://www.webasto.com/en-us/cooling/marine-self-contained-air-conditioning-units/bluecool-s-series.html#carousel-7779c0a393-item-a01bdc6b0e-tabpanel> (accessed Apr. 30, 2026).

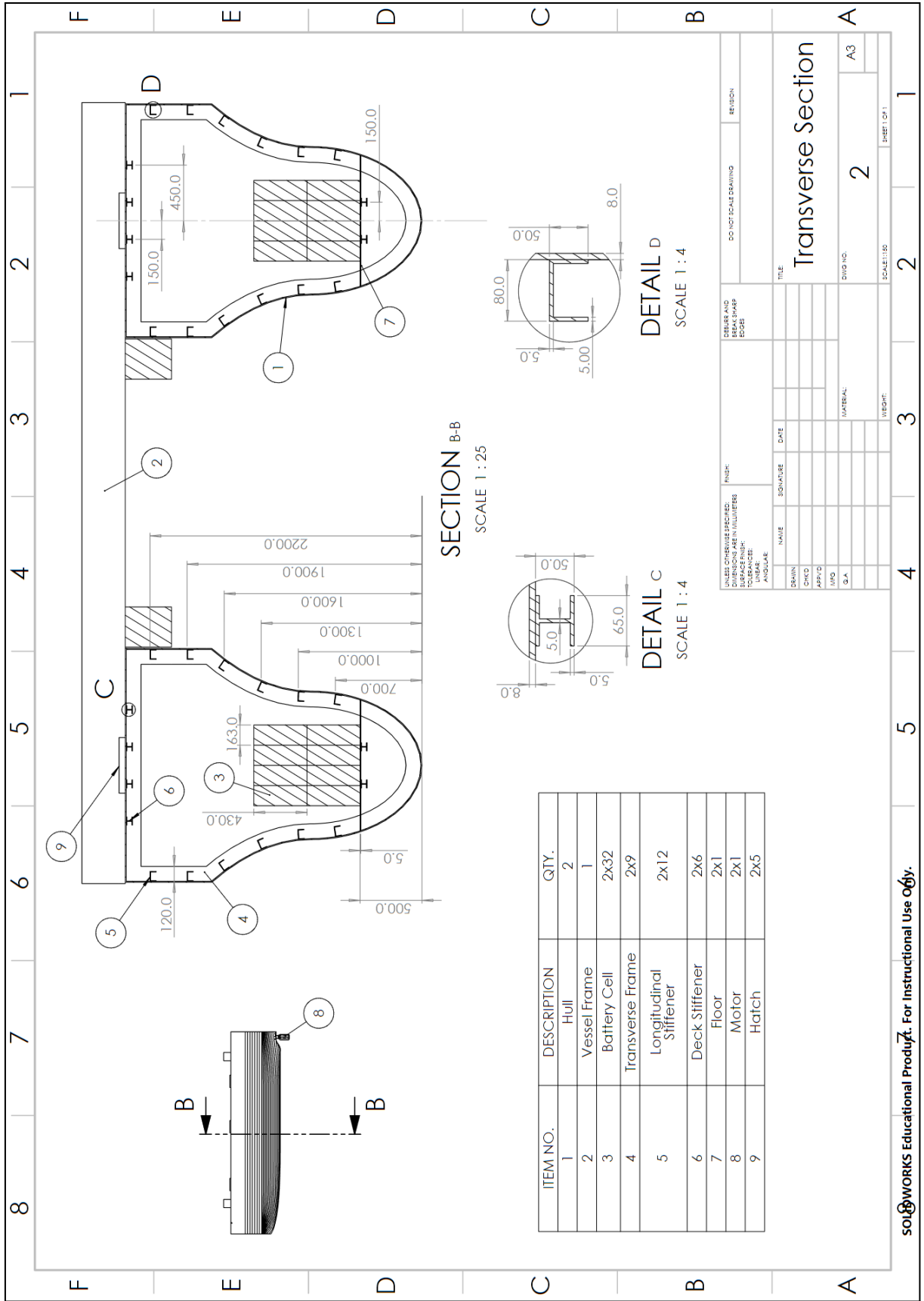
Appendix A. Drawings⁴

Overall dimensions and thruster placement:

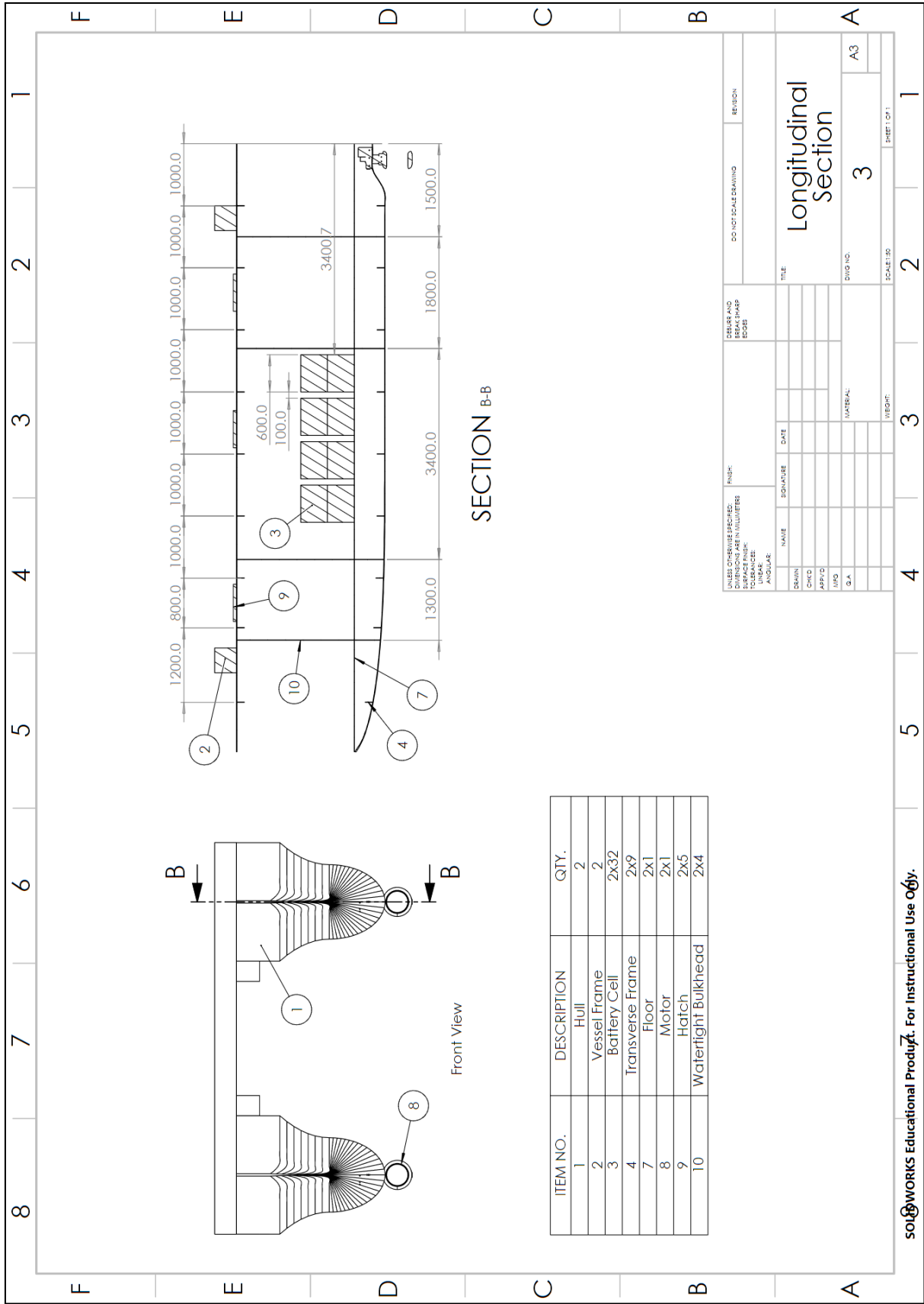


⁴ Note: All units are in mm.

Transverse section:



Longitudinal section:



Appendix B. Scripts and input files

3. Extraction of hull splines for use in *MICHLET*:

Option Explicit

Sub Main()

```
    Dim SECTIONS, POINTS_PER_CURVE, T
    T = 1.3
    SECTIONS = 33
    POINTS_PER_CURVE = 20
    Dim objs, bbox, minX, maxX, i, stepX, fso, file, ii, lowestIndex
    Dim bboxMinY, bboxMaxY, bboxMinZ, bboxMaxZ, planeWidth, planeHeight, Ycenter

    objs = Rhino.GetObjects("Pick one hull", 8 + 16, True, True)
    If IsNull(objs) Then Exit Sub

    bbox = Rhino.BoundingBox(objs)
    If IsNull(bbox) Then
        Rhino.Print "Couldnt get bounding box."
        Exit Sub
    End If

    minX = bbox(0)(0)
    maxX = bbox(1)(0)
    stepX = (maxX - minX) / (SECTIONS - 1)

    bboxMinY = bbox(0)(1)
    bboxMaxY = bbox(3)(1)
    bboxMinZ = bbox(0)(2)
    bboxMaxZ = bbox(4)(2)
    Ycenter = (bboxMinY + bboxMaxY) / 2
    planeWidth = (bboxMaxY - bboxMinY)
    planeHeight = T

    Dim path1, path2
    path1 = "D:\Capstone_project\michlet\michlet933\sploff1.csv"
    path2 = "D:\Capstone_project\michlet\michlet933\sploff2.csv"
    Set fso = CreateObject("Scripting.FileSystemObject")
    Set file = fso.CreateTextFile(path1, True)
    file.WriteLine 3
    Rhino.EnableRedraw False
    For i = 0 To SECTIONS - 1
        Dim xPos, plane, planeSurf, obj, crvs, crv, pts, pt
        If i = 0 Then
            xPos = minX + 0.1 * stepX
        ElseIf i = SECTIONS - 1 Then
            xPos = maxX - 0.1 * stepX
        Else
            xPos = minX + i * stepX
        End If

        file.WriteLine "BEGIN"
        file.WriteLine FormatNumber(xPos, 4)
        plane = Rhino.PlaneFromFrame(Array(xPos, Ycenter, bboxMinZ), Array(0, 1, 0), Array(0, 0, 1))
        planeSurf = Rhino.AddPlaneSurface(plane, planeWidth, planeHeight)

        If IsNull(planeSurf) Then
            Rhino.Print "Couldnt create plane X = " & CStr(xPos)
        Else
            For Each obj In objs
                crvs = Rhino.IntersectBreps(obj, planeSurf)
                If Not IsNull(crvs) Then
                    For Each crv In crvs
                        pts = Rhino.DivideCurve(crv, POINTS_PER_CURVE)
                        For ii=0 To Ubound(pts)
                            If pts(ii)(1) < pts(lowestIndex)(1) Then
                                lowestIndex = ii
                            End If
                        Next
                    Next
                If lowestIndex > 0 Then
```

```

                                For ii=UBound(pts) To 0 Step -1
                                file.WriteLine FormatNumber(pts(ii)(2), 6) &
                                Next
                                Else
                                For ii=0 To UBound(pts)
                                file.WriteLine FormatNumber(pts(ii)(2), 6) &
                                Next
                                End If
                                Next
                                End If
                                Rhino.DeleteObjects(crvs)
                                Next
                                Rhino.DeleteObject(planeSurf)
                                End If
                                file.WriteLine "END"
                                Next
                                file.WriteLine "EOF"
                                fso.CopyFile path1, path2, True
End Sub

Call Main()

```

4. Input file (in.mlt) for *MICHLET*:

```

# =====
#
# ===== INPUT FILE TYPE AND SUBTYPE =====
# Input File Type (0=Standard)
0
# Input File Subtype (0=Standard)
0
# ===== OUTPUT FILE TYPE AND SUBTYPE =====
# Output File Type (0=Standard)
0
# Output File Subtype (0=Standard)
0
# ===== COURSE AND VESSEL TYPE =====
# Course Particulars (0=None)
0
# Number of Hulls (1, 2,..., or 5)
2
# ===== PHYSICAL QUANTITIES =====
# Gravitational Acceleration (m/sec/sec) (min 9.6, max 9.9)
9.80665
# ===== WATER PROPERTIES =====
# Water Density (kg/cubic metre) (min 995.0, max 1030.0)
1025.9
# Water Kin. Viscosity (sq. m/sec * 10^-6) (min 0.5, max 2.0)
1.18831
# Base Eddy Kin. Viscosity (non-dimensional, min 1.0)
10.0
# Water Depth (metres) (max=10000.0)
10000.0
# ===== AIR PROPERTIES =====
# Air Density (kg/cubic metre) (min 0.9, max 2.0)
1.226
# Air Kin. Viscosity (sq. m/sec * 10^-6) (min 10.0, max 20.0)
14.4
# Wind Speed (m/sec)
0.0
# Wind Direction (degrees)
0.0
# ===== CALCULATION PARAMETERS =====
# Minimum Speed (m/sec) (min 0.01, max 51.9)
0.1
# Maximum Speed (m/sec) (max 52.0)
3.0

```

```

# Number of Speeds (min 2, max 101)
31
# Leeway Parameters (0=None)
0
# Wave Drag Ntheta
320
# Skin Friction Method (0=None, 1=ITTC1957, 2=Grigson)
2
# Viscous Form Factor Method (0=None, 3=Dual)
3
# Viscous Drag Form Factor
1.0
# Wave Drag Form Factor
1.0
# Pressure Signature Method (0=None, 1=Slender body)
1
# ===== SHIP CALCULATION PARAMETERS =====
# Number of Offset Stations (rows) (odd integer: min 5, max 81)
33
# Number of Offset Waterlines (columns) (odd integer: min 5, max 81)
33
# Ship Loading Type
3
# Ship Loading Formula Parameters
1.0,0.0,0.0
# ===== WAVE ELEVATION PARAMETERS =====
# Sectorial Cuts and Patches
# R0
200.0
# R1
1200.0
# Beta
22.5
# Nr
101
# Nbeta
101
# Rectangular Cuts and Patches
# x0
200.0
# x1
800.0
# y0
-300.0
# y1
300.0
# Nwx
101
# Nwy
101
# Beaches and Walls
# x0
200.0
# x1
2200.0
# y0
500.0
# z0
-1.0
# z1
0.0
# Slope
90.0
# Nbx
11
# Nbz
11
# ===== FIRST HULL =====
# Offsets
-2
# Displacement Volume (cubic metres)
11.3
# Length (metres)

```

```

10
# Draft (metres)
1.3
# Longitudinal Separation (metres) (0.0 for a monohull)
0.0
# Lateral Separation Distance (metres) (0.0 for a monohull)
0.0
# Loading Type for this hull
3
# Loading Formula Parameters
0.672,0.21,0.64
# Trim Method
2
# Trim: Number of speeds (integer: greater than or equal to 2)
2
# Trim: speed, angle
0.1,0.0
3.0,0.0
# Sinkage Method
2
# Sinkage: Number of speeds (integer: greater than or equal to 2)
2
# Sinkage: speed, amount
0.0,0.0
3.0,0.0
# Heel Method
0
# Heel: Number of speeds (integer: greater than or equal to 2)
2
# Heel: speed, angle
0.0,0.0
3.0,0.0
# Appendages (0=None)
0
# Other Particulars (0=None)
0
# ===== SECOND HULL =====
# Offsets
-2
# Displacement Volume (cubic metres)
11.3
# Length (metres)
10
# Draft (metres)
1.3
# Longitudinal Separation (metres) (0.0 for a monohull)
0.0
# Lateral Separation Distance (metres) (0.0 for a monohull)
4.4
# Loading Type for this hull
3
# Loading Formula Parameters
0.672,0.21,0.64
# Trim Method
2
# Trim: Number of speeds (integer: greater than or equal to 2)
2
# Trim: speed, angle
0.1,0.0
3.0,0.0
# Sinkage Method
2
# Sinkage: Number of speeds (integer: greater than or equal to 2)
2
# Sinkage: speed, amount
0.0,0.0
3.0,0.0
# Heel Method
0
# Heel: Number of speeds (integer: greater than or equal to 2)
2
# Heel: speed, angle
0.0,0.0

```

```

3.0,0.0
# Appendages (0=None)
0
# Other Particulars (0=None)
0

```

5. Extraction of points for Offset Table generation and use in *ArchimedesMB*:

Option Explicit

```

Sub Main()
    Dim HullSepDist
    HullSepDist = 2.490 ' min horizontal dist between hulls
    Dim SECTIONS, POINTS_PER_CURVE
    SECTIONS = 40
    POINTS_PER_CURVE = 50
    Dim IsHullOpen
    IsHullOpen = False ' If the hull surface is open (not closed)
    Dim objs, bbox, minX, maxX, i, j, stepX, txt, fso, txtfile
    Dim bboxMinY, bboxMaxY, bboxMinZ, bboxMaxZ, planeWidth, planeHeight

    objs = Rhino.GetObjects("Pick left hull", 8 + 16, True, True)
    If IsNull(objs) Then Exit Sub

    bbox = Rhino.BoundingBox(objs)
    If IsNull(bbox) Then
        Rhino.Print "Couldnt get bounding box."
        Exit Sub
    End If

    minX = bbox(0)(0)
    maxX = bbox(1)(0)
    stepX = (maxX - minX) / (SECTIONS - 1)

    bboxMinY = bbox(0)(1)
    bboxMaxY = bbox(3)(1)
    bboxMinZ = bbox(0)(2)
    bboxMaxZ = bbox(4)(2)

    planeWidth = (bboxMaxY - bboxMinY) * 2.0
    planeHeight = (bboxMaxZ - bboxMinZ) * 2.0

    txt = "D:\Capstone_project\for_archimedes.hll"

    Set fso = CreateObject("Scripting.FileSystemObject")

    Set txtfile = fso.CreateTextFile(txt, True)
    txtfile.WriteLine " <?xml version=""1.0"" encoding=""UTF-8"" ?> "
    txtfile.WriteLine "<HULL>"
    txtfile.WriteLine "<HULL_NAME>"
    txtfile.WriteLine "Blank Hullform"
    txtfile.WriteLine "</HULL_NAME>"
    txtfile.WriteLine "<DENSITY> 1.025 </DENSITY>"
    txtfile.WriteLine "<AP> 0 </AP>"
    txtfile.WriteLine "<FP> 0 </FP>"
    txtfile.WriteLine "<FWD_DRAUGHT_MARK> 0 </FWD_DRAUGHT_MARK>"
    txtfile.WriteLine "<MID_DRAUGHT_MARK> 0 </MID_DRAUGHT_MARK>"
    txtfile.WriteLine "<AFT_DRAUGHT_MARK> 0 </AFT_DRAUGHT_MARK>"
    txtfile.WriteLine "<APPENDAGE_COUNT> 2 </APPENDAGE_COUNT>"
    txtfile.WriteLine "<APPENDAGE>"
    txtfile.WriteLine "<APPENDAGE_NAME>"
    txtfile.WriteLine "New Appendage [0]"
    txtfile.WriteLine "</APPENDAGE_NAME>"
    txtfile.WriteLine "<ACTIVE> 1 </ACTIVE>"
    txtfile.WriteLine "<FREE_FLOODING> 0 </FREE_FLOODING>"
    txtfile.WriteLine "<APPENDAGE_COLOUR> 0 </APPENDAGE_COLOUR>"
    txtfile.WriteLine "<SECTION_COUNT> " & SECTIONS & " </SECTION_COUNT>"

    Rhino.EnableRedraw False
    For j = 0 To 1

```

```

For i = 0 To SECTIONS - 1
    Dim xPos, plane, planeSurf, obj, crvs, crv, pts, pt
    If i = 0 Then
        xPos = minX + 0.1 * stepX
    ElseIf i = SECTIONS - 1 Then
        xPos = maxX - 0.1 * stepX
    Else
        xPos = minX + i * stepX
    End If

    txtfile.WriteLine "<SECTION>"
    txtfile.WriteLine "<SECTION_NAME>"
    txtfile.WriteLine "S" & i
    txtfile.WriteLine "</SECTION_NAME>"
    txtfile.WriteLine "<YCOORD>" & FormatNumber(xPos, 4) & "</YCOORD>"
    txtfile.WriteLine "<OFFSET_COUNT>" & POINTS_PER_CURVE + Abs(IsHullOpen) & "
</OFFSET_COUNT>"

    txtfile.WriteLine "<OFFSETS>"
    plane = Rhino.PlaneFromFrame(Array(xPos, bboxMinY, bboxMinZ), Array(0, 1, 0), Array(0, 0, 1))
    planeSurf = Rhino.AddPlaneSurface(plane, planeWidth, planeHeight)

    If IsNull(planeSurf) Then
        Rhino.Print "Couldnt create plane X = " & CStr(xPos)
    Else
        For Each obj In objs
            crvs = Rhino.IntersectBreps(obj, planeSurf)
            If Not IsNull(crvs) Then
                For Each crv In crvs
                    pts = Rhino.DivideCurve(crv, POINTS_PER_CURVE)
                    If Not IsNull(pts) Then
                        For Each pt In pts
                            txtfile.WriteLine
FormatNumber(pt(1) + j * (HullSepDist + bboxMaxY - bboxMinY), 6) & " " & FormatNumber(pt(2), 6)
                            Next
                        End If
                    Next
                End If
            End If
            Rhino.DeleteObjects(crvs)
        Next
        Rhino.DeleteObject(planeSurf)
    End If
    txtfile.WriteLine "</OFFSETS>"
    txtfile.WriteLine "</SECTION>"
Next
txtfile.WriteLine "</APPENDAGE>"
If j = 0 Then
    txtfile.WriteLine "<APPENDAGE>"
    txtfile.WriteLine "<APPENDAGE_NAME>"
    txtfile.WriteLine "New Appendage [1]"
    txtfile.WriteLine "</APPENDAGE_NAME>"
    txtfile.WriteLine "<ACTIVE> 1 </ACTIVE>"
    txtfile.WriteLine "<FREE_FLOODING> 0 </FREE_FLOODING>"
    txtfile.WriteLine "<APPENDAGE_COLOUR> 0 </APPENDAGE_COLOUR>"
    txtfile.WriteLine "<SECTION_COUNT>" & SECTIONS & "</SECTION_COUNT>"
End If
Next
txtfile.WriteLine "</HULL>"
End Sub

Call Main()

```

6. Static balancing (based on script developed by K. Kostas⁵):

```

'Assumptions
'Longitudinal axis is world x
'Transverse axis is world y
'Vertical axis is world z
'Length units: meters, Weight units: tonnes

```

⁵ konstantinos.kostas@nu.edu.kz

'The hull needs to be a solid model

'If there are any problems with the wl curve, increase tolerance

Option Explicit

' Calculate position of balance

' Input: Weight, LCG, TCG and VCG

' Output: Draft, heel and trim angles (heel: transverse, trim: longitudinal)

Sub CalcBalance()

Dim hull,w

Dim e:e = 0.001 'Tolerance

hull = Rhino.GetObjects("Pick ship hull:", 16)

If Not IsNull(hull) Then

Dim bb:bb = Rhino.BoundingBox(hull)

Dim LCG,TCG,VCG

Dim maxw:maxw = Rhino.SurfaceVolume(hull(0))(0) + Rhino.SurfaceVolume(hull(1))(0)

Dim maxLCG:maxLCG = bb(1)(0) - e

Dim minLCG:minLCG = bb(0)(0) + e

Dim maxTCG:maxTCG = Rhino.Max(Array(bb(0)(1), bb(3)(1)))

Dim minTCG:minTCG = Rhino.Min(Array(bb(0)(1), bb(3)(1)))

Dim maxVCG:maxVCG = Rhino.Max(Array(bb(0)(2), bb(4)(2)))

Dim minVCG:minVCG = Rhino.Min(Array(bb(0)(2), bb(4)(2)))

w = Rhino.Getreal("Input weight for calculation (W>" & e & " W<" & maxw & ")", e, maxw)

LCG = Rhino.Getreal("Input the longitudinal position of the center of gravity (LCG>" & minLCG & " LCG<" & maxLCG & ")", minLCG, maxLCG)

TCG = Rhino.Getreal("Input the transverse position of the center of gravity (TCG>" & minTCG & " TCG<" & maxTCG & ")", minTCG, maxTCG)

VCG = Rhino.Getreal("Input the vertical position of the center of gravity (VCG>" & minVCG & " VCG<" & maxVCG & ")", minVCG, maxVCG)

Dim CG:CG = Array(LCG, TCG, VCG)

Rhino.AddPoint CG

Dim CGv:CGv = Array(0, 0, maxVCG - minVCG)

Dim BalanceNotFound:BalanceNotFound = True

Dim counter:counter = 0

Dim t0,Ft0,t1,Ft1,Status,Data,CGv1,CGv2

Dim f0,Ff0,f1,Ff1

Dim y0,Fy0,y1,Fy1

Dim dt:dt = 0.001

Dim df:df = 0.01

Dim dy:dy = 0.01

t0 = (bb(4)(2) - bb(0)(2)) / 2 'initial estimation at ship's half depth

f0 = 0

y0 = 0

t1 = t0

f1 = f0

y1 = y0

Ft0 = 2 * e

Ff0 = 2 * e

Fy0 = 2 * e

Dim j:j = 0

While BalanceNotFound And j < 50

While Abs(Ft0) > e And counter < 50

Ft0 = VolumeCalc(hull, t0, f1, y1)(0) - w

Ft1 = VolumeCalc(hull, t0 + dt, f1, y1)(0) - w

t1 = t0 - Ft0 / ((Ft1 - Ft0) / dt)

t0 = t1

counter = counter + 1

Wend

counter = 0

While Abs(Ff0) > e And counter < 50

CGv1 = Rhino.VectorRotate(CGv, f0, Array(0, 1, 0))

CGv1 = Rhino.VectorRotate(CGv1, y1, Array(1, 0, 0))

Data = VolumeCalc(hull, t1, f0, y1)

Ff0 = pointToLineSegDistance(Array(Data(1), Data(2), Data(3)), CG,

Rhino.VectorAdd(CG, CGv1))(0)

Data = VolumeCalc(hull, t1, f0 + df, y1)

Ff1 = pointToLineSegDistance(Array(Data(1), Data(2), Data(3)), CG,

Rhino.VectorAdd(CG, CGv1))(0)

f1 = f0 - Ff0 / ((Ff1 - Ff0) / df)

f0 = f1

counter = counter + 1

Wend

```

        counter = 0
        While Abs(Fy0) > e And counter < 50
            CGv2 = Rhino.VectorRotate(CGv, f1, Array(0, 1, 0))
            CGv2 = Rhino.VectorRotate(CGv2, y0, Array(1, 0, 0))
            Data = VolumeCalc(hull, t1, f1, y0)
            Fy0 = pointToLineSegDistance(Array(Data(1), Data(2), Data(3)), CG,
Rhino.VectorAdd(CG, CGv2))(1)
            Data = VolumeCalc(hull, t1, f1, y0 + dy)
            Fy1 = pointToLineSegDistance(Array(Data(1), Data(2), Data(3)), CG,
Rhino.VectorAdd(CG, CGv2))(1)
            y1 = y0 - Fy0 / ((Fy1 - Fy0) / dy)
            y0 = y1
            counter = counter + 1
        Wend
        Status = VolumeCalc(hull, t1, f1, y1)
        Data = pointToLineSegDistance(Array(Status(1), Status(2), Status(3)), CG, Rhino.VectorAdd(CG,
CGv2))
        Rhino.print j & " Vol:" & Status(0) & " f:" & f1 & " y:" & y1
        Rhino.print j & " dx:" & Data(0) & " dy:" & Data(1)
        If Abs(Status(0) - w) < e And Abs(Data(0)) < e And Abs(Data(1)) < e Then
            BalanceNotFound = False
        End If
        j = j + 1
        Ft0 = 2 * e
        Ff0 = 2 * e
        Fy0 = 2 * e
    Wend
    If Abs(y1) < e Then y1 = 0 End If
    If Abs(f1) < e Then f1 = 0 End If
    Rhino.print "Draft : "&t1&"m Heel angle:"&y1&" Trim angle:"&f1
    DrawWL hull, t1, f1, y1
End If
End Sub

Function VolumeCalc(Objects, T, phi, psi) 'T: draft, phi: trim angle, psi: heel angle
    'We assume that the longitudinal, transverse, and vertical axes correspond to x,y, and z, respectively
    Dim objsCopy, bbox, wlPlane, center, wlCenter, vols1, vols2, wlCurve, CB
    Dim wlCurve1, wlCurve2, wlCenter1, wlCenter2, wlArea1, wlArea2
    Rhino.enableRedraw(False)
    objsCopy = Rhino.CopyObjects(Objects)
    bbox = Rhino.BoundingBox(objsCopy)
    wlPlane = Rhino.AddSrfPt(Array(bbox(0), bbox(1), bbox(2), bbox(3)))
    wlPlane = Rhino.MoveObject(wlPlane, Array(0, 0, T))
    center = Rhino.SurfaceAreaCentroid(wlPlane)
    wlPlane = Rhino.ScaleObject(wlPlane, center(0), Array(10, 10, 1))
    If phi < 0 Or psi < 0 Then
        wlCurve1 = Rhino.IntersectBreps(objsCopy(0), wlPlane)(0)
        wlCurve2 = Rhino.IntersectBreps(objsCopy(1), wlPlane)(0)
        If Not IsNull(wlCurve1) Then
            wlArea1 = Rhino.CurveArea(wlCurve1)(0)
            wlCenter1 = Rhino.CurveAreaCentroid(wlCurve1)(0)
        Else
            wlArea1 = 0
            wlCenter1 = Array(0, 0, 0)
        End If
        If Not IsNull(wlCurve2) Then
            wlArea2 = Rhino.CurveArea(wlCurve2)(0)
            wlCenter2 = Rhino.CurveAreaCentroid(wlCurve2)(0)
        Else
            wlArea2 = 0
            wlCenter2 = Array(0, 0, 0)
        End If
        wlCenter = Array((wlArea1 * wlCenter1(0) + wlArea2 * wlCenter2(0)) / (wlArea1 + wlArea2), _
(wlArea1 * wlCenter1(1) + wlArea2 * wlCenter2(1)) / (wlArea1 + wlArea2), _
(wlArea1 * wlCenter1(2) + wlArea2 * wlCenter2(2)) / (wlArea1 + wlArea2))
        wlPlane = Rhino.RotateObject(wlPlane, wlCenter, phi, Array(0, 1, 0))
        wlPlane = Rhino.RotateObject(wlPlane, wlCenter, psi, Array(1, 0, 0))
        Rhino.deleteObject wlCurve1
        Rhino.DeleteObject wlCurve2
    End If

    vols1 = Rhino.SplitBrep(objsCopy(0), wlPlane, True)

```

```

vols2 = Rhino.SplitBrep(objsCopy(1), wPlane, True)

Dim min1,min2,bb1,bb2
If Not IsNull(vols1) Then
    bb1 = Rhino.BoundingBox(vols1(0))
    bb2 = Rhino.BoundingBox(vols1(1))
    If bb1(0)(2) < bb2(0)(2) Then
        min1 = 0
    Else
        min1 = 1
    End If
End If
If Not IsNull(vols2) Then
    bb1 = Rhino.BoundingBox(vols2(0))
    bb2 = Rhino.BoundingBox(vols2(1))
    If bb1(0)(2) < bb2(0)(2) Then
        min2 = 0
    Else
        min2 = 1
    End If
End If

Dim CB1,CB2,Vol1,Vol2,Vol
If Not IsNull(vols1) Then
    Rhino.CapPlanarHoles vols1(min1)
    CB1 = Rhino.SurfaceVolumeCentroid(vols1(min1))(0)
    Vol1 = Rhino.SurfaceVolume(vols1(min1))(0)
Else
    CB1 = Array(0, 0, 0)
    Vol1 = 0
End If

If Not IsNull(vols2) Then
    Rhino.CapPlanarHoles vols2(min1)
    CB2 = Rhino.SurfaceVolumeCentroid(vols2(min2))(0)
    Vol2 = Rhino.SurfaceVolume(vols2(min2))(0)
Else
    CB2 = Array(0, 0, 0)
    Vol2 = 0
End If

Vol = Vol1 + Vol2
CB = Array(0, 0, 0)
CB(0) = (Vol1 * CB1(0) + Vol2 * CB2(0)) / Vol
CB(1) = (Vol1 * CB1(1) + Vol2 * CB2(1)) / Vol
CB(2) = (Vol1 * CB1(2) + Vol2 * CB2(2)) / Vol
VolumeCalc = Array(Vol, CB(0), CB(1), CB(2))

Rhino.deleteObject wPlane
Rhino.deleteObjects vols1
Rhino.deleteObjects vols2
Rhino.enableRedraw(True)

End Function

Call CalcBalance()

Function DrawWL(object, T, phi, psi)
    If Not IsNull(object) Then
        Dim bbox: bbox = Rhino.BoundingBox(object)
        Dim wPlane:wPlane = Rhino.AddSrfPt(Array(bbox(0), bbox(1), bbox(2), bbox(3)))
        wPlane = Rhino.MoveObject(wPlane, Array(0, 0, T))
        Dim center:center = Rhino.SurfaceAreaCentroid(wPlane)
        wPlane = Rhino.ScaleObject(wPlane, center(0), Array(3, 3, 1))

        If phi <> 0 Or psi <> 0 Then
            Dim wCurve1,wCurve2,wCenter1,wCenter2, wArea1, wArea2, wCenter
            wCurve1 = Rhino.IntersectBreps(object(0), wPlane)(0)
            wCurve2 = Rhino.IntersectBreps(object(1), wPlane)(0)
            wCenter1 = Rhino.CurveAreaCentroid(wCurve1)(0)
            wCenter2 = Rhino.CurveAreaCentroid(wCurve2)(0)
            wArea1 = Rhino.CurveArea(wCurve1)(0)

```

```

        wlArea2 = Rhino.CurveArea(wlCurve2)(0)
        wlCenter = Array((wlArea1 * wlCenter1(0) + wlArea2 * wlCenter2(0)) / (wlArea1 + wlArea2),_
            (wlArea1 * wlCenter1(1) + wlArea2 * wlCenter2(1)) / (wlArea1 + wlArea2),_
            (wlArea1 * wlCenter1(2) + wlArea2 * wlCenter2(2)) / (wlArea1 + wlArea2))
        wlPlane = Rhino.RotateObject(wlPlane, wlCenter, phi, Array(0, 1, 0))
        wlPlane = Rhino.RotateObject(wlPlane, wlCenter, psi, Array(1, 0, 0))
        Rhino.deleteObject wlCurve
    End If
End If
End Function

Function pointToLineSegDistance(p, lstart, lend)
    Dim v:v = Rhino.PointSubtract(lend, lstart)
    Dim mag2:mag2 = v(0) * v(0) + v(1) * v(1) + v(2) * v(2)
    Dim w:w = Rhino.PointSubtract(p, lstart)
    Dim t:t = Rhino.VectorDotProduct(v, w) / mag2
    Dim pint:pint = Rhino.PointAdd(lstart, Rhino.VectorScale(v, t))
    pointToLineSegDistance = Array(p(0) - pint(0), p(1) - pint(1), p(2) - pint(2))
End Function

```

7. Input parameters for hull generation in *Fastships*:

```

# Hull form type
edgy_cs = False #If true then the hull will be aluminium construction friendly, with plates of very small double curvature
n_flat_sides_subm_edgy_cs = 3 # Number of flat plates to use at each side of hull

# Resolution data
n_cs = 1001
n_cs_verts_subm_full = 101

# Main dimensions
l_wl = 10 # length at waterline

max_width = 1.910
# Width data
width_stern = 1.000 # rear width
entrance_angle = 60*deg2rad
const_width_fwd_rel = 0.7 # this fraction of the hull (counting from the back)
# has constant width

# Draft data
draft_max = 0.9 # deepest point below the water
draft_transom = 1.0 * draft_max # draft of rear end
max_draft_aft_rel = 0.5
max_draft_fwd_rel = 0.5
draft_dist_el_exp_fwd = 3

# Cross-sectional superellipse data
cs_el_exp_aft = 2 # rear cross-section ellipse exponent
# (1=triangle, 2=circle, inf=square)
cs_el_exp_fwd = 2
const_cs_el_exp_aft_rel = 0.5
const_cs_el_exp_fwd_rel = 0.5

# Above-water data
n_cs_verts_abv_w_full = 100
abv_w_flare_h_aft = 0.5 # height of rear flare "shoulder"
abv_w_flare_h_fwd = 0.5 # height of front flare "shoulder"
abv_w_flare_w_aft = (max_width-width_stern)/2 # flare width (how much it protrudes)
const_abv_w_flare_fwd_rel = 0.5 # this fraction of the hull (counting from the back)
# has constant width
abv_w_flare_dist_se_exp = 3 #
h_main_deck = 1.0 # height of the deck

```

8. Extraction of contour points of sections from STL file

```

% this script extracts the contour points of the hull mesh and records them in separate files for each section
% STL file is the .obj mesh generated by Fastships (simply converted)
% the result is 51 text files with point coordinates (corresponding to 51 sections)

```

```

[triangles] = stlread('D:\Capstone_project\Hull.stl');
p = triangles.Points;
c = triangles.ConnectivityList;

points = [round(p(:,1),2) p(:,2:3)];
points = sortrows(points,1);

clean_points = [0 0 0];

for x = 0:0.2:10
    temp_points = points(abs(points(:,1)-x)<10^(-6),:);
    temp_points = round(temp_points,4);
    temp_points = unique(temp_points,'row');
    clean_points = [clean_points;temp_points];
end

ux = unique(clean_points(:,1));

for k = 1:length(ux)
    filename = "sections" + num2str(k) + ".txt";
    fid = fopen(filename,'w');
    idx = abs(clean_points(:,1)-ux(k)) < 1e-10; % tolerance
    S = clean_points(idx,:);

    cy = mean(S(:,2));
    cz = mean(S(:,3))+0.1*(max(S(:,3))-min(S(:,3)));
    vy = S(:,2) - cy;
    vz = S(:,3) - cz;
    ang = atan2d(vz, vy);
    [~, order] = sort(ang);

    S = S(order,:);
    if k==length(ux)-1
        plot(S(:,2),S(:,3))
    end
    fprintf(fid, '%f %f %f\r\n', S(1:end-1, :).');
    fprintf(fid, '%f %f %f', S(end, :).');
    fclose(fid);
end

```

9. Generation of cross-section curves in *Rhino* based on STL points:

```

Option Explicit

Call Main()
Sub Main()
    Dim filePath
    Dim fileItr
    Dim points()
    For fileItr=1 To 51

        filePath = "\\LENOVO\Capstone_project\python_test\sections" & CStr(fileItr) & ".txt"
        If IsNull(filePath) Then Exit Sub

        Dim fso, file, line
        Set fso = CreateObject("Scripting.FileSystemObject")
        Set file = fso.OpenTextFile(filePath, 1)

        ReDim points(-1)
        Dim count
        count = 0

        Do Until file.AtEndOfStream
            line = Trim(file.ReadLine)

            If line <> "" Then
                Dim parts
                parts = Split(line)

                If UBound(parts) >= 2 Then

```

```

ReDim Preserve points(count)

Dim pt
pt = Array(CDbl(parts(0))*1000, CDbl(parts(1))*1000, CDbl(parts(2))*1000)
points(count) = pt
count = count + 1
End If
End If
Loop
file.Close
If count < 2 Then
Rhino.Print "Not enough points to create a polyline."
Exit Sub
End If

ReDim Preserve points(count)
points(count) = points(0)

Rhino.AddPolyline points
Rhino.Print "Polyline created with " & count & " points."
Next
End Sub

```

10. Manual floodable length calculations:

a. Extraction of centers of buoyancy for different trims in *Rhino*:

```

Option Explicit
' This script records the following, separated by space: x_CB y_CB z_CB displacement waterline_area waterline_area_moment
' Resultant txt file is to be used
' The hull is repeatedly rotated around the intersection of the margin line and aft (front) perpendicular.
Call Main()
Sub Main()

Dim txt, fso, txtfile
txt = Rhino.SaveFileName("Save Results As", "Text Files (*.txt)|*.txt||", , "CB_and_Volumes.txt", "txt")
Set fso = CreateObject("Scripting.FileSystemObject")
Set txtfile = fso.CreateTextFile(txt, True)

Rhino.EnableRedraw False

Dim maxAng, angStep, marginZ, obj, bbox
maxAng = 8
angStep = 0.02
marginZ = 0.3
obj = Rhino.GetObject("Pick hull", 8 + 16, True, True)
bbox = Rhino.BoundingBox(obj)

Dim bboxMinY, bboxMaxY, bboxMinZ, bboxMaxZ
bboxMinY = bbox(0)(1)
bboxMaxY = bbox(3)(1)
bboxMinZ = bbox(0)(2)
bboxMaxZ = bbox(4)(2)

Dim rotAxisZ: rotAxisZ = bboxMaxZ - marginZ

Dim wlPlane: wlPlane = Rhino.AddSrfPt(Array(bbox(0), bbox(1), bbox(2), bbox(3)))
Dim wlPlaneTempCenter : wlPlaneTempCenter = Array(0.5 * (bbox(1)(0) + bbox(0)(0)), 0.5 * (bbox(3)(1) + bbox(0)(1)),
bbox(0)(2))
Rhino.ScaleObject wlPlane, wlPlaneTempCenter, Array(2, 2, 2)
Rhino.MoveObject wlPlane, Array(0, 0, 0), Array(0, 0, rotAxisZ)

Dim iterations, i
iterations = maxAng / angStep

Dim ang
Dim objCopy
Dim rotPoint

```

```

rotPoint = Array(0, 0, rotAxisZ)

Dim submerged, bb1, bb2, split, wlCurve, wlCurvePlanar, wlArea, wlMoment, wlSurf
Dim CB, Vol
For i = 0 To iterations
    objCopy = Rhino.CopyObject(obj)
    ang = i * angStep

    Rhino.RotateObject objCopy, rotPoint, ang, Array(0, -1, 0), False
    wlCurve = Rhino.IntersectBreps(objCopy, wlPlane)(0)
    wlCurve = Rhino.MoveObject(wlCurve, Array(0, 0, -0.1))
    wlCurvePlanar = Rhino.ProjectCurveToSurface(wlCurve, wlPlane, Array(0, 0, 1))(0)
    wlArea = Rhino.CurveArea(wlCurvePlanar)(0)

    wlSurf = Rhino.AddPlanarSrf(wlCurvePlanar)(0)
    wlMoment = Rhino.SurfaceAreaMoments(wlSurf)(10)(1)

    Rhino.DeleteObject(wlCurve)
    Rhino.DeleteObject(wlCurvePlanar)
    Rhino.DeleteObject(wlSurf)
    split = Rhino.SplitBrep(objCopy, wlPlane, True)

    bb1 = Rhino.BoundingBox(split(0))
    bb2 = Rhino.BoundingBox(split(1))
    If bb1(0)(2) < bb2(0)(2) Then
        submerged = split(0)
    Else
        submerged = split(1)
    End If

    Rhino.CapPlanarHoles submerged
    CB = Rhino.SurfaceVolumeCentroid(submerged)(0)
    Vol = Rhino.SurfaceVolume(submerged)(0)

    txtfile.WriteLine CB(0) & " " & CB(1) & " " & CB(2) & " " & Vol & " " & wlArea & " " & wlMoment
    Rhino.DeleteObject objCopy
    Rhino.DeleteObjects split
    Rhino.Print(ang)
Next

rotPoint = Array(10, 0, rotAxisZ)
For i = 0 To iterations
    objCopy = Rhino.CopyObject(obj)
    ang = i * angStep

    Rhino.RotateObject objCopy, rotPoint, ang, Array(0, 1, 0), False
    wlCurve = Rhino.IntersectBreps(objCopy, wlPlane)(0)
    wlCurve = Rhino.MoveObject(wlCurve, Array(0, 0, -0.1))
    wlCurvePlanar = Rhino.ProjectCurveToSurface(wlCurve, wlPlane, Array(0, 0, 1))(0)
    wlArea = Rhino.CurveArea(wlCurvePlanar)(0)

    wlSurf = Rhino.AddPlanarSrf(wlCurvePlanar)(0)
    wlMoment = Rhino.SurfaceAreaMoments(wlSurf)(10)(1)

    Rhino.DeleteObject(wlCurve)
    Rhino.DeleteObject(wlCurvePlanar)
    Rhino.DeleteObject(wlSurf)
    split = Rhino.SplitBrep(objCopy, wlPlane, True)

    bb1 = Rhino.BoundingBox(split(0))
    bb2 = Rhino.BoundingBox(split(1))
    If bb1(0)(2) < bb2(0)(2) Then
        submerged = split(0)
    Else
        submerged = split(1)
    End If

    Rhino.CapPlanarHoles submerged
    CB = Rhino.SurfaceVolumeCentroid(submerged)(0)
    Vol = Rhino.SurfaceVolume(submerged)(0)

    txtfile.WriteLine CB(0) & " " & CB(1) & " " & CB(2) & " " & Vol & " " & wlArea & " " & wlMoment

```

```

Rhino.DeleteObject objCopy
Rhino.DeleteObjects split
Rhino.Print(ang)

Next
End Sub

```

b. Calculation of floodable length curve using the data recorded by *Rhino*:

```

% reference frame is fixed (independent on hull orientation)
CBV = readmatrix('Floodable\CB_and_Volumes.txt');

angles = (0:0.02:8)';
x_positions1 = zeros(length(angles),1);
lengths1 = zeros(length(angles),1);
mdif1 = zeros(length(angles),1);
x_positions2 = zeros(length(angles),1);
lengths2 = zeros(length(angles),1);
mdif2 = zeros(length(angles),1);
A_cross = 3313773.8/1000^2;
A_centroidZ = 1.41;

CG = [4.6891 0.0000 1.6226];
rho = 1025;
m = 22.766;

rot_point = [0,0,2.1]; % 2.1 is margin line height (2.4-0.3)

xCG_rel = CG(1) - rot_point(1);
zCG_rel = CG(3) - rot_point(3);

for i=1:length(angles)
    ang = angles(i);
    xCG = xCG_rel*cosd(ang) - zCG_rel*sind(ang);
    xCB = CBV(i,1);
    m_diff = 2*CBV(i,4)*rho/1000 - m;
    mdif1(i) = m_diff;
    lengths1(i) = 0.5*m_diff/(rho*A_cross/1000);
    x_rel = ((m+m_diff)*CBV(i,1) - m*xCG)/m_diff;
    x_positions1(i) = (x_rel + (A_centroidZ-rot_point(3))*sind(ang))/cosd(ang);

    if lengths1(i)<0 || x_positions1(i)<0
        lengths1(i) = NaN;
        x_positions1(i) = NaN;
        disp(ang)
        continue
    end

    if lengths1(i)/2 > x_positions1(i)
        lengths1(i) = 2*x_positions1(i);
    end
    if lengths1(i)/2 > 10-x_positions1(i)
        lengths1(i) = 2*(10-x_positions1(i));
    end
end

figure()
plot(x_positions1,lengths1)
hold on

rot_point = [10,0,2.1];

xCG_rel = rot_point(1) - CG(1);
zCG_rel = rot_point(3) - CG(3);

for i=1:length(angles)
    ang = angles(i);
    xCG = 10-(xCG_rel*cosd(ang) - zCG_rel*sind(ang));
    xCB = CBV(i+length(angles),1);
    m_diff = 2*CBV(i+length(angles),4)*rho/1000 - m;
    mdif2(i) = m_diff;
    lengths2(i) = 0.5*m_diff/(rho*A_cross/1000);

```

```

x_rel = ((m+m_diff)*CBV(i+length(angles),1) - m*xCG)/m_diff;
x_positions2(i) = 10-((10-x_rel) + (A_centroidZ-rot_point(3))*sind(ang))/cosd(ang);

if lengths2(i)<0 || x_positions2(i)>10
    lengths2(i) = NaN;
    x_positions2(i) = NaN;
    continue
end

if lengths2(i)/2 > x_positions2(i)
    lengths2(i) = 2*x_positions2(i);
end
if lengths2(i)/2 > 10-x_positions2(i)
    lengths2(i) = 2*(10-x_positions2(i));
end
end
plot(x_positions2,lengths2)
grid on
xlabel("x (m)"); ylabel("Floodable L (m)")
xlim([0 10])
ylim([0 4])

```

Appendix C. RINA Rules for Classification

The navigation coefficients were taken as 1, according to Table 1 of [21, Pt. B, p. 329]. The wave parameter and Froude's number were calculated using the given equations in [21, Pt. B, p. 328]:

$$C = (118 - 0.36L) \frac{L}{1000}$$

$$F = 0.164 \frac{V}{\sqrt{L}}$$

The distribution factor was taken as 1, referring to Table 2 of [21, Pt. B, p. 331]. The minimum yield stress of the hull steel was taken according to Table 1 of [21, Pt. B, p. 61] – $R_{eH} = 235 \text{ N/mm}^2$. The material factor was taken as 1, according to Table 2 of [21, Pt. B, p. 61].

Table 1C. Constants for the main calculations.

Navigation coefficient n	0.8
Navigation coefficient n_1	0.9
Wave parameter C	1.144
Froude's number F	0.104
Distribution factor F_m	1

Material factor k	1
---------------------	---

The reference value of the relative motion for midship section that were calculated for still water and inertial pressures calculation procedure using Table 4 in [21, p. 332]:

$$h_1 = 0.42nC(C_B + 0.7) = 0.520 \text{ m}$$