

Davor Bolf

E-mail: dbolf@riteh.hr

University of Rijeka, Faculty of Engineering, Vukovarska 58, Rijeka, Croatia

Peter Rogelj

E-mail: peter.rogelj@upr.si

University of Primorska, Faculty of Mathematics, Natural Sciences and Information Technologies, Glagoljaška 8, 6000 Koper, Slovenia

Adrian Tomić

E-mail: atomic.worldpeace@gmail.com

Albert Zamarin

E-mail: zamarin@riteh.hr

University of Rijeka, Faculty of Engineering, Vukovarska 58, Rijeka, Croatia

Application of a General Purpose Software Package on Shear Forces and Bending Moment Calculations in Ship Structure

Abstract

The calculation of shear forces and bending moments are the basis of every ship design documentation. It is essential for determining the strength of the ship and the reliability of the structure itself. Nowadays, due to a sheer volume of data, this calculation is performed exclusively with the help of specialized software packages. Such shear force and bending moment curves can often be obtained within the software packages specially designed for checking the ship's design according to specific classification societies' rules. Usually, they already have built-in formulas and guidelines to execute that task properly. However, such software packages often do not meet all the end user's needs. For example, for the initial project stage, special software packages with the functionality of easy designing of a ship form are usually used, while the aforementioned packages are used in later design stages when performing stability and structure checks. This transition often requires converting files into the appropriate type from one software to another. The Rhinoceros program is widespread in the shipbuilding profession, and it is often used in the initial stages of the project. It also allows automation of the process with the help of integrated scripts. Therefore, the authors have decided to examine the possibility of using Python scripts within the Rhinoceros software package. The mentioned script will calculate the transverse forces and bending moment based on the previously modelled hull.

Keywords: shear forces, bending moment, Rhinoceros, Python

1. Introduction

It is of utmost importance to know the exact loads and material characteristics used in the calculation to design any ship structure properly. The loads on a ship can be classified according to the level of structure at which they act, [1], i.e., the hydrostatic pressure is acting on the ship's hull on a global level, while windlass forces, various equipment weight and operating forces tend to transfer the load on a smaller area of the structure; thus these forces can be considered local. The calculation of shear forces and bending moment acting along the length of a ship is essential for designing a longitudinal structure, [2], therefore, being an essential step in calculating the ship's longitudinal strength. However, these forces can and are usually also used in later design stages while calculating the strength of the localized structural elements.

In order to calculate ship longitudinal strength (and shear forces and bending moment), the users can choose among several specialized software packages already available on the market, usually as a part of the classification society. Some of them are software packages tied to a particular classification society, like BV Veristar, LR SEASAFE, [3], [4], [5], and others are independent, i.e., EcoLoad Master from Greenship technology company, [6]. However, even such software abides by the rules of the classification societies. Specialized hydrostatic programs are also usually capable of calculating the still water, shear force and bending moment distribution along the ship length, [1]. However, they usually do not provide open-source access, and the customization of the inputs and outputs is mostly limited. The small independent design offices often rely on software packages for 3D modelling and easy drawing extraction such as Rhinoceros (Rhino) and AutoCAD, often not specially customized for shipbuilding application, using them as a comprehensive tool for designing the hull structure, machinery, systems and equipment, [7], rather than investing in several specialized software programs.

Rhino is usually used as starting tool, often in the initial design phase of the project. The modelling in Rhino is based on NURBS (Non-Uniform Rational B-Splines) mathematical model. The modelling is relatively easy and intuitive, as it can create the simplest geometrical forms and organic 3D shapes and free surfaces. Therefore, it is often used to model the ship hull and design the ship superstructure in the early stages of the design process, [8].

The benefit of Rhino is the ability to interact and automate the design process using some of the well-known and widely used programming languages, such as Python, C++ or Visual Basic, [9]. This automated modelling process is often started and implemented using the scripting process, capable of executing all 3D modelling commands without using the graphical user interface. The connection between the modelling and scripting in Rhino can be achieved using *rhinoscriptsyntax* module package, [10]. The authors have used Rhinoceros 7, which has integrated IronPython package with the ability to connect Python and .Net framework, to evaluate the possibility of calculating the shear forces and bending moments using the Python script with 3D modelling tool on

Windows operating platform. Three different calculation methods were investigated and compared to the results obtained with a standard software package used in the well-known design office.

2. Basic principles of calculating longitudinal shear force and moment distribution

The concept of the “hull girder” model falls under the primary level of ship structural loadings and responses. The ship is simplified and idealized as the thin-wall box beam with several assumptions and simplifications: there is only one variable with loads and deflections having the single value at any cross-section, the hull girder remains elastic with relatively small deflections, any dynamic forces and effects are either neglected or substituted with equivalent static forces, and finally, the vertical bending is predominant, and finally, bending strain is linear, [1].

In order to calculate the resultant load on the ship q_x , the weight distribution curve, q_t and buoyancy distribution curve, q_u , must be determined. The weight distribution curve is obtained by adding the weights acting on each cross-section along the length of the hull, and the buoyancy curve is usually obtained by calculating the volume of the immersed area of the hull on that same cross-sections. The static equilibrium is achieved if the buoyancy force is equal to the weight of the ship, as shown in the following equation:

$$\int_0^L q_t dx = \int_0^L q_u dx, \quad (1)$$

where L is the length of the ship, and the position of the longitudinal centre of buoyancy must be in coincidence with the longitudinal centre of gravity, [1].

Thus, it is possible to calculate and draw shear force Q_x and bending moment M_x on any given position x along the length of the ship (Figure 1) according to equations 2 and 3:

$$Q_x = Q_u - Q_t = \int_0^x (q_u - q_t) dx = - \int_0^x q_x dx, \quad (2)$$

$$M_x = M_u - M_t = \int_0^x \int_0^x (q_u - q_t) dx dx = - \int_0^x \int_0^x q_x dx dx, \quad (3)$$

where Q_u , M_u are shear force and moment respectively, resulting from buoyancy, and Q_t , M_t are shear force and moment respectively, resulting from weight distribution.

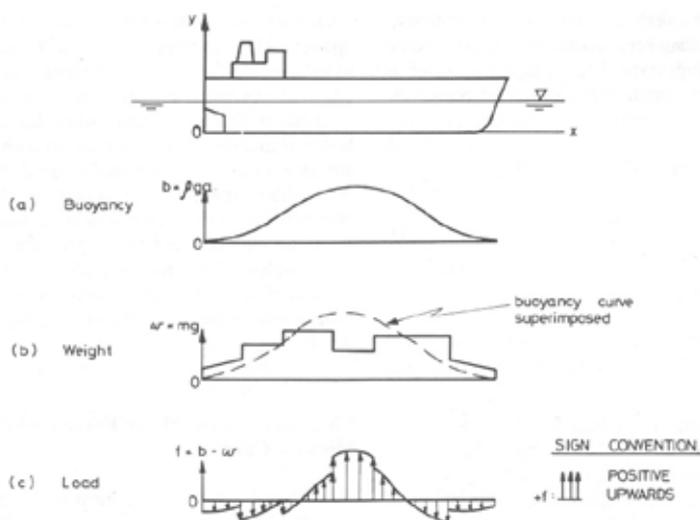


Figure 1: Summary of hull girder bending [1]

The starting step of shear forces and moment calculation is determining weight distribution along the length of the ship. The loads are usually grouped to form logical units, such as hull weight, machinery section, fuel, water, crew and provisions, passengers and cargo, plus a safety margin for the weights that can be underestimated, [2]. The weight of the hull can easily be extracted by using a preliminary 3D model of the structure or in early design stages by any given equation for preliminary calculation of the weight i.e. from literature sources, [2], [11]. Other loads, which are not positioned through the complete length of the ship, are then added to the hull weight by positioning them on the exact longitudinal position with correct weight length and weight amplitude, thus forming the weight distribution diagram.

Before calculating the buoyancy distribution diagram, it is essential to determine the correct waterline using the convergence method described in the literature, [2]. The proposed starting waterline must be corrected with one parallel translation and one angular rotation. Buoyancy can be calculated and checked against the weight distribution upon creating the waterline. When the equilibrium is achieved, meaning that the weight of the ship is equal to the total upwards buoyancy and these two forces coincide, [1], shear forces and moment can be calculated. The principle method flowchart is presented in Figure 2.

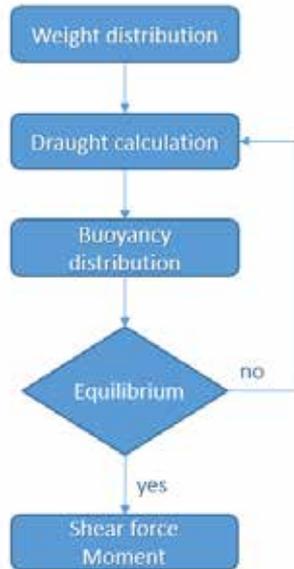


Figure 2: Design flowchart for shear force and bending moment calculation

The method mentioned above can be applied for still water as well as for wave loads. The method usually includes integration of the load curve using numerical integration formula i.e. trapezoidal rules or Simpson's One-Third Rule, [1], which can be used for hand calculation and for calculation executed using some specialized computer program.

3. Using *rhinoscriptsyntax* with Python code within Rhinoceros model

The shear force and moment calculation was used to test and verify calculation capability with an integration of Python within the Rhino models. The design principle from Figure 2 was incorporated into the Python script embedded within the Rhino model. To verify the script, authors have created the hull shape using Rhino modelling tools and have tried three different approaches, resulting in the creation of three versions of Python scripts. Version A and version B mainly focus on an analytical calculation using the numerical integration and following the workflow and equations from the literature, [2]. In contrast, version C used the graphical solution, relying mainly on the command available in Rhino and executed through the functions from *rhinoscriptsyntax* module. Versions A and B focus on calculating the buoyancy by using a determined number of sections through the hull length, where for version A, the number of sections was 20, and for version B, the number was user-defined. Upon defining the hull sections,

the total buoyancy of the hull was calculated using Simpson's One Third Rule. In version C, the hull geometry was graphically closed with a waterline surface, and by using the *rhinoscriptsyntax* module, the volume formed by the hull surface and closing waterline was formed. The flowchart of version C is shown in Figure 3, and usage some of the most used functions are briefly listed and explained through this chapter.

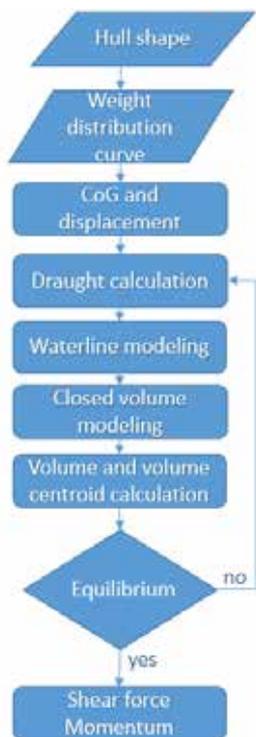


Figure 3: Design flowchart for version C

The Python script modules *math* and *rhinoscriptsyntax* were imported into the program using the **import** command. The *rhinoscriptsyntax* module enables the usage of Rhino command through Python scripting language, and *math* module enables some mathematical functions to be executed without the need of scripting these functions.

The ship particulars length over all (L_{oa}), hull breadth (B_{oa}) and a total height of the hull surface were defined through the function *BoundingBox(obj)*, where *obj* is 3D object obtained with *GetObject()* function through the user selection of the hull surface in the Rhino model. *GetObject()* function was frequently used to determine the graphical objects selected by the user in the graphical user interface (GUI) of Rhino software. The same function was later used to select the weight distribution curve and determine its parameters by selecting the curve using the object type filtering expression within

the *rhinoscriptsyntax* module, within the *if* statement in form (*if rs.IsCurve(graph):*), where *graph* is the selected diagram. The points of the curve object were also extracted by creating the new list of points with the help of the *CurvePoints(graph)*. The weight distribution curve was then plotted in scale and closed. Redrawn and scaled curve was used to calculate the area under the curve, equal to the ship's displacement, with *CurveArea()* function. The centre of gravity coincides with the curve area centroid, easily extracted with *CurveAreaCentroid()* function. The same functions and principles were applied to extract the frame area and centre of buoyancy.

The waterline was created at half of the hull shape height and interpolated by the Rhino script in several iterations. The final version of intersecting waterline plane was created using the bounding box limiting planes and calculated waterline curve. The waterline surface was created with the surface command *AddSrfPt* and used as the intersection surface to create the volume with the hull surface model. With syntax line *rs.Command("-_CreateSolid")*, the selected curves (waterline and hull surface) joined in the solid, closed volume and thus, enabling extraction of the volume and centre of buoyancy with use of functions *SurfaceVolume()* and *SurfaceVolumeCentroid()*.

These values were compared with total displacement and longitudinal centre of gravity extracted from the weight distribution diagram. The shear force and momentum curves were drawn when the equilibrium was achieved.

The final frame area curve was calculated and drawn using previously described methods upon achieving the equilibrium. Two curves, frame area curve and weight distribution curve, were adjusted to have the same number of values on the abscissa axis and then the unknown values were interpolated using the linear interpolation. The curves were subtracted and plotted into the Rhino 3D model space.

4. Python within Rhino script execution method

From the user's point of view, several steps were taken. In order to commence calculation, a proper hull shape had to be selected. Hull shape geometry and weight distribution curve (diagram) were created (or imported) to Rhino. The imported or created geometry was checked for gaps prior to the script execution. If needed, surfaces were recreated using the *Patch* and *Loft* commands in Rhino and *ShowEdge* command crucial for visualization of any possible gaps in the model. The weight distribution diagram was inserted into the same model as the hull shape.

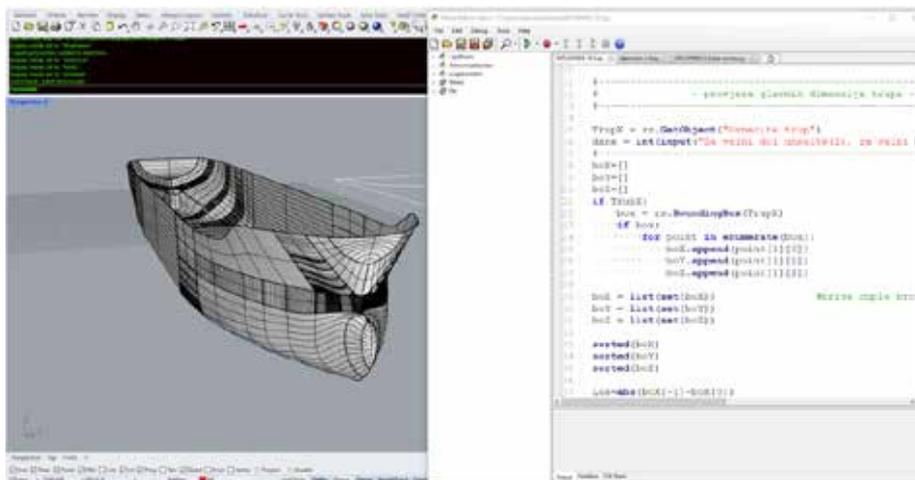


Figure 4: Rhino GUI with Python script editor [8]

Upon inspection and after finalizing model preparation, the initiated the script within *Tools* toolbar by choosing the *PythonScript -> Run...* command and selecting the appropriate Python script from the working folder. The model and loaded script are visible in Figure 4.

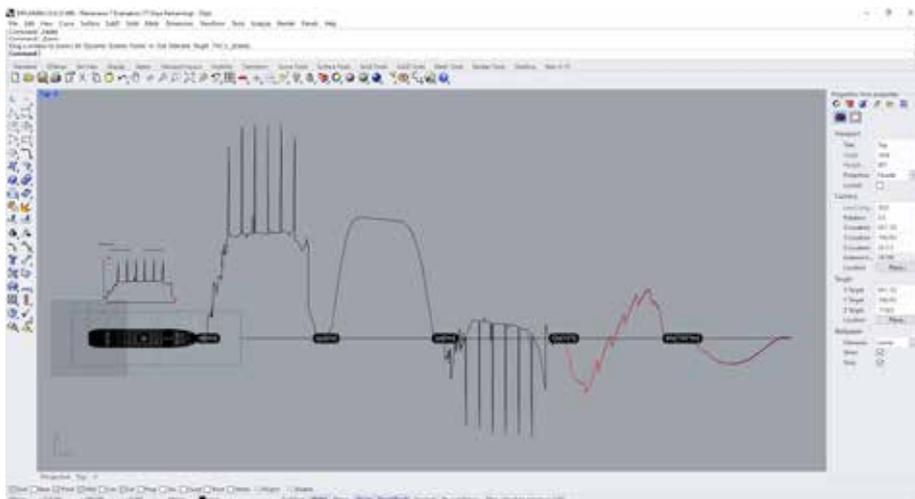


Figure 5: Output curves in Rhino 3D model space [8]

Several inputs were required prior to commencing the scripted calculation. The script guided the user through the necessary actions, selection of hull surface and

diagram selection process and diagram scaling. After the initial inputs, the script commenced the calculation and output diagrams were created in the Rhino 3D model space (Figure 5).

The script recreates the weight distribution diagram; it is only created using the appropriate scale. The frame area curve and load curve q_x were also created, as well as shear force and bending moment curves stacked from left to right in Figure 5. If needed, the moment envelope can be created by the user using the appropriate scale for moment (1 m in model space equals 100 tm).

This method was used to test all three versions and compare them to the approved and finalized calculation for the chemical tanker provided by the design office of the well-known shipyard, called the reference case. The weight distribution curve was taken from the shipyard's documentation, and the hull was remodelled according to the lines plan of the ship. The ship's particulars are shown in Table 1 and general arrangement of the vessel in Figure 6.

Table 1: Main particulars [8]

Length overall	195.20 m
Length between perpendiculars	187.30 m
Breadth moulded	32.20 m
Depth moulded	17.80 m
Design draught	12.00 m
Deadweight at design draught	49000 t

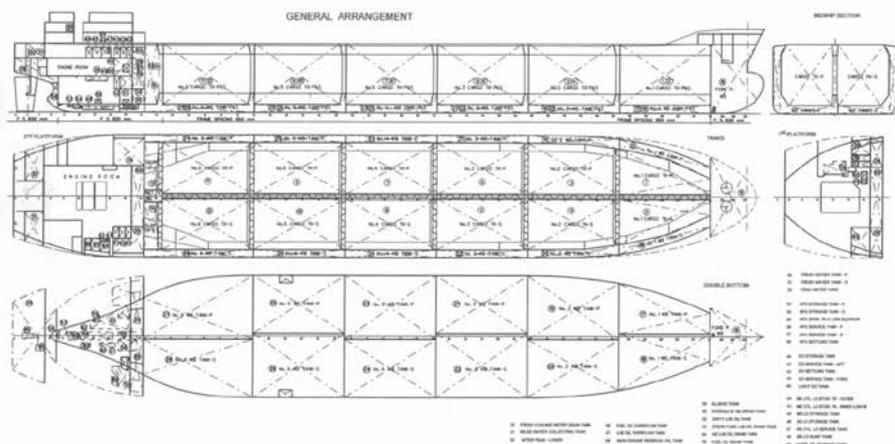


Figure 6: General arrangement of the case study ship [8]

5. Results and discussion

The shear force curves for each version were plotted and compared to the reference case, as seen in Figure 7. The comparison in the shear force diagram between the reference case and user versions of the program showed a 45.76 t greater value in favour of version C, while version B and version A had smaller values from the reference case by 82.12 t, and 337.14 t, respectively. Closer to the bow, the difference in maximum values was more significant. Version C had the biggest difference to the reference case with the value of 450.17 t, while version A had the smallest deviation of 117.20 t and version B intermediate 304.25 t. These values were smaller than the maximal value of the reference case in the bow area. The starting values of versions C and B were almost overlapping the reference curve, but the difference becomes more significant through the ship's length.

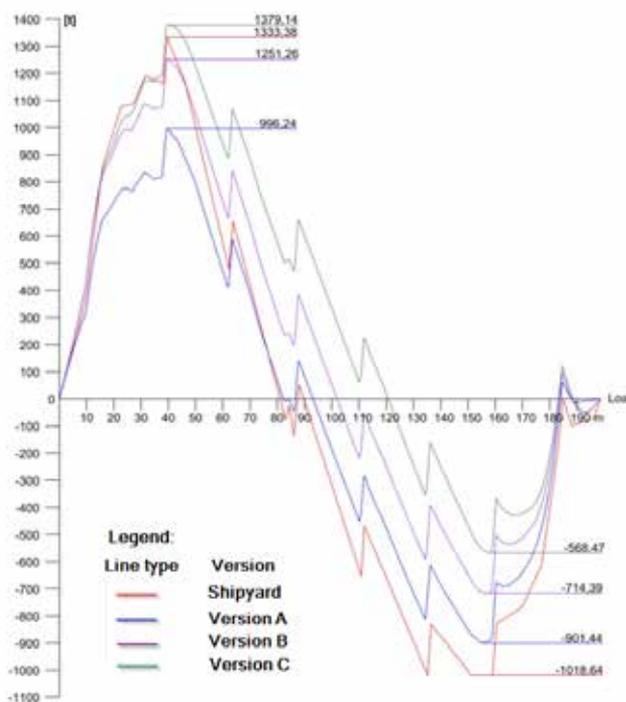


Figure 7: Comparison of shear force curves to reference case [8]

A similar issue was observed when looking at the bending moment curves (Figure 8). Version A, although with the most significant difference in maximum moment value (854 tm) but has the position of the momentum along the length of the ship almost at the same x coordinate, while version B and version C have maximal value closest to the

reference case (641.4 tm and 587.8 tm respectively). The bending moment maximum for version C is shifted well forward, as seen in Figure 8.

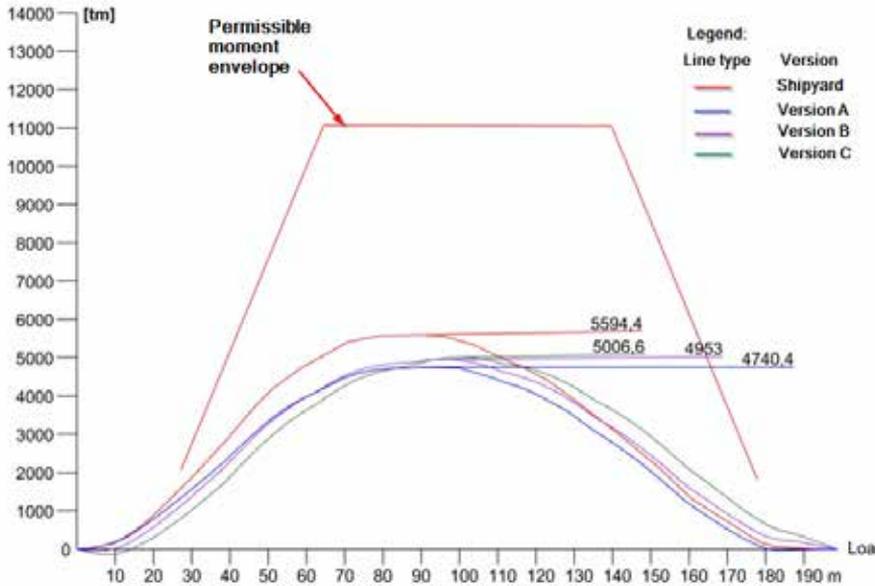


Figure 8: Comparison of bending moment curves to reference case [8]

While the smallest difference in shear force maximum values between the reference case and version C was 3,4% at the stern part, the percentage was increased in the bow area with a value difference of 44%. Version B has a 6% difference at the stern and 30% at the bow. The smallest difference of 10.5% for the bending moment values between version C and reference value cannot be omitted. Further investigation is needed to establish the reason of these deviations.

6. Conclusion

The authors have presented a methodology of determining the shear forces and bending moments based on scientific methods and tools that differ from the classical approach, primarily in using a generic 3D software package in conjunction with open source Python software to facilitate the calculation procedure.

The values were compared with the reference case, and even though the curve shapes match the reference case, there are differences in values that cannot be easily omitted. Authors have found several possible reasons that can influence the values:

Differences between the input geometry can have a significant influence on final values. The hull geometry was modelled in Rhino 3D using the lines plan, and the hull

shape was corrected and adjusted to have closed surfaces and work properly with the script. The geometry used by the authors was slightly simplified, especially around the bulbous bow, where the shape differs from the lines plan.

The authors have used a simplified interpolation method, linear interpolation, appropriate for the weight distribution curve. However, using this method on the buoyancy curve can lead to significant errors. In the authors' opinion, improving the iteration and integration methods can also lead to more accurate values.

To fully validate software and troubleshoot possible errors, several cases and several ships will have to be considered and studied in future work.

Acknowledgements

This research is supported by funds from the support research at the University of Rijeka under project UNIRI tehnic 18-159 "Development of Methodology for Ship Design and Production towards Industry 4.0. Concept"

References

1. Hughes, O. F. (1988) *Ship structural Design*. New Jersey, SNAME.
2. Uršić, J. (1991) *Čvrstoća broda I. dio*. Zagreb, Sveučilišna naklada d.o.o.
3. Bureau Veritas (2019) *Bureau Veritas Rules for the classification of Steel Ships, Part B Hull and Stability*, Paris, BV.
4. Marine and Offshore (2021) *Veristar Stability Software – Intact and Damage Calculation* Available from: <https://marine-offshore.bureauveritas.com/veristar-stability-software-intact-and-damage-calculation> [Accessed 28th October 2021].
5. LR SEASAFE (2021) *Providing certainty for critical marine calculation* Available from: <https://www.lr.org/en/lr-seasafe/> [Accessed 28th October 2021].
6. Green ship technologies - Marine Software (2019) *Marine Software*, Available from: <https://www.greenshiptech.com/marine-software/> [Accessed 28th October 2021].
7. Rhinocheros (2021) *Rhino for the Marine Industry* Available from: <https://www.rhino3d.com/for/marine/> [Accessed 28th October 2021].
8. Tomić, A. (2021) *Proračun uzdužne čvrstoće brod primjenom općih programskih paketa*, Croatia. Masters thesis, University of Rijeka, Faculty of Engineering.
9. van Oers, B., van Hees, M., Stapersma, D., Hopman, H. (2008) Combining a Knowledge System with Computer-Aided Design. *Ship Technology Research*. 55(2), 51-59.
10. Rhino Developer Docs (2021) *What is Rhino.Python*. Available from: <https://developer.rhino3d.com/guides/rhinopython/what-is-rhinopython/> [Accessed 28th October 2021].
11. Lamb, T. (2003) *Ship Design and Construction, Volumes I and II*. New Jersey, SNAME.