# Vessel: Efficient Plain Text File Format for Ship Hull Geometry

Alexander Degtyarev, Ivan Gankevich(✉), Anton Gavrikov, Artemii Grigorev, Vasily Khramushin, and Ivan Petriakov

Saint Petersburg State University, Saint Petersburg, Russia
{a.degtyarev,i.gankevich,v.khramushin}@spbu.ru,
{st047437,st016177,st049350}@student.spbu.ru
http://www.shipdesign.ru/

**Abstract.** Initially, digital geometric model of a ship hull is assumed to maintain continuity in solving traditional ship theory problems, ship hydromechanics and seakeeping in severe storm waves. In the research reported here we consider using a table of plaza ordinates (transversal projection of frames) supplemented with a description of sterns as a means of describing digital geometric model. This description, which we call Vessel (VSL) format, allows for later addition of compartments, appendages and superstructures. These more complicated ship structures and their characteristics (e.g. ship compartments and superstructures are characterised by template-based modelling) are added to initial ship hull model in separate files from the working directory of a particular experiment. We show how this model is converted into three-dimensional triangular mesh suitable for ship motion simulations. VSL format is generally more efficient than industry standard IGES format and is distinguished by the simplicity and capability of editing by hand.

**Keywords:** Ship lines · Ship blueprint · Ship theory · Ship design · Ship hydromechanics · Storm seakeeping · Coons surface · Hermite spline · VSL · IGES

## 1 Introduction

Initial geometric model of a ship hull is close to traditional hull blueprints which are naturally rendered as a table of plaza ordinates in ship theory. Such a model quite reliably characterises ship hull lines and above-water ship hull parts, while preserving continuity in hydrostatic and hydrodynamic calculations in ship theory and hydromechanics. This is important for verification of newly created numerical experiments with a multitude of historical series of ship calculations, model basin experiments and sea trials. Obvious advantage of this digital format is that it is relatively simple and has compact data layout, with the data prepared and refined as plain text strings representing stern and frame coordinate sequences. We call this format Vessel (or VSL for short).

© Springer Nature Switzerland AG 2019
S. Misra et al. (Eds.): ICCSA 2019, LNCS 11622, pp. 729–739, 2019.
https://doi.org/10.1007/978-3-030-24305-0_54

Numerous variants of ship hulls were systematically collected in VSL format in Saint Petersburg State University in Vessel database [3]. The database is maintained with the help of Hull software suite [5] which allows for editing ship hull coordinates and calculates certain hydrostatic characteristics. In this paper we describe VSL format, compare and contrast its efficiency and performance with industry standard IGES format, and outline advantages and disadvantages of using plain text for creating and editing ship hull geometry by hand.

## 2 Methods

### 2.1 Vessel Format

Formalised description of ship hull geometry is done using plain text data which include ship name, hull dimensions, and successive description of aft, main (table of plaza ordinates) and bow sections. When digital ship hull model is first added to Vessel database the following design characteristics are specified in the comments: displacement, wetted surface area, hull volume ratio coefficient, date and time of file creation. The diagram of a VSL file is presented in Table 1, projections and three-dimensional model of Aurora cruiser are presented in Figs. 8 and 9.

**Table 1.** VSL file format diagram.

<table>
<tr><td colspan="3">1. Technical vessel description (lines starting with <code>//</code> or <code>;</code> are comments)</td></tr>
<tr><td colspan="3">2. Format magic number (30) and hull model name in angle brackets <code><...></code></td></tr>
<tr><td colspan="3">3. The number of frames and middle frame number</td></tr>
<tr><td colspan="3">4. Hull dimensions (length, beam, draft)</td></tr>
<tr><td>The number of points on a curve</td><td colspan="2">5. $X(z)$ — sternpost contour abscissas as a function of applicates</td></tr>
<tr><td rowspan="4"></td><td colspan="2">6. $Y(z)$ — transom width ordinates as a function of applicates</td></tr>
<tr><td>Frame abscissas</td><td>7. $Y(z)$ — frame curves as functions of applicates of general hull line</td></tr>
<tr><td colspan="2">8. $Y(z)$ — bulbous bow width ordinates as a function of applicates</td></tr>
<tr><td colspan="2">9. $X(z)$ — stern contour abscissas as a function of applicates</td></tr>
<tr><td colspan="3">10. Design characteristics (displacement, wetted surface, volume ratio coefficient)</td></tr>
</table>

Ship hull is divided into three sections (Fig. 1): aft, main and bow sections. Main section consists of frames each of which is defined by a collection of points lying in transverse plane. Smooth curve that goes though all of these points is

created by cubic Hermite spline interpolation. Each frame may not have the same number of points (but usually do), and there are no additional points between endpoints of subsequent frames in longitudinal plane. Aft and bow sections consist of frames in transverse plane and a curve in longitudinal plane that defines the shape of the ship hull in this plane. This curve go through (usually) all frames and defines intermediate points between endpoints of subsequent frames. If it does not go between some frames, then there are no intermediate points between them. Curves are not closed and define only left part of the ship hull, the full ship hull model is created by mirroring each point of the curve with respect to longitudinal axis and connect corresponding curve endpoints by straight lines.

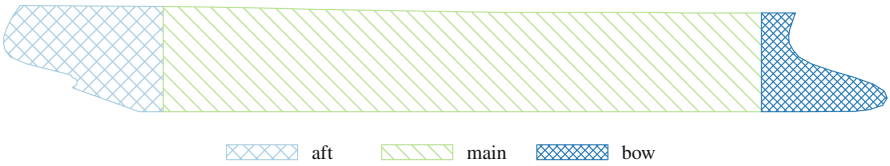


**Fig. 1.** Three sections of a ship hull as defined by VSL format.

In accordance with initial purpose of a digital ship hull model, Hull programme makes basic calculations of certain characteristics of a ship hull, curved elements and ship stability diagrams by fixing the centre of gravity with respect to general line, centre of buoyancy, metacentre or current waterline (Fig. 2). The programme also performs wave resistance calculations for arbitrary travel speeds taking into account radiation intensity and interference of ship waves with respect to waterline as a function of Froude number.

### 2.2 Hydrostatic Calculations for Historic, Traditional and Contemporary Ship Hulls

As an illustrative example of simulating seakeeping ship characteristics we consider ship stability diagrams for contemporary and historical ships, and for a ship that is purposely optimised for improved seakeeping in storm waves.

Figure 3 shows that Aurora cruiser ship hull has excellent seakeeping characteristics in severe storm waves conditions, which guarantee safe small oscillations and smoothness of roll by reducing metacentric height — initial ship hull stability. Additional positive stability appears for large roll angles and arbitrary heaving.

More complicated usage scenario of computational model is optimisation of ship lines to achieve certain form of wave resistance curve (Fig. 4). In the calculations we build radiation intensity and ship waves interferences along the current waterline. By changing ship lines we reduce resistivity extrema for relative speeds (on Froude) on the order of 0.3 and 0.5 without sacrificing propulsion for minimum wave formation for speeds 0.4 and 0.2.

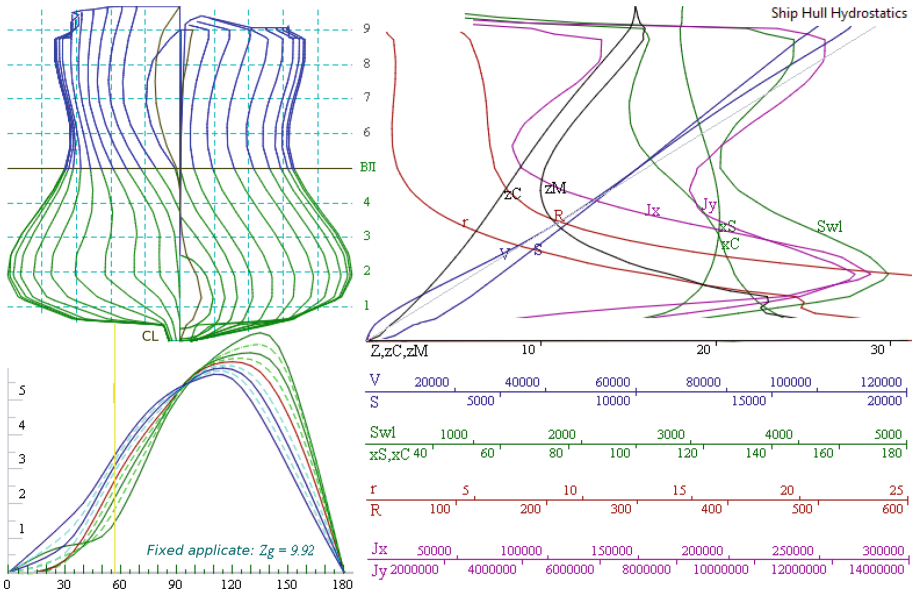


**Fig. 2.** Ship stability calculations for a ship with small moments of inertia for the area of the current waterline (MICW-85). For a design draught roll moments are small up to 30° wave slope, and stability increases in response to any change in draught.

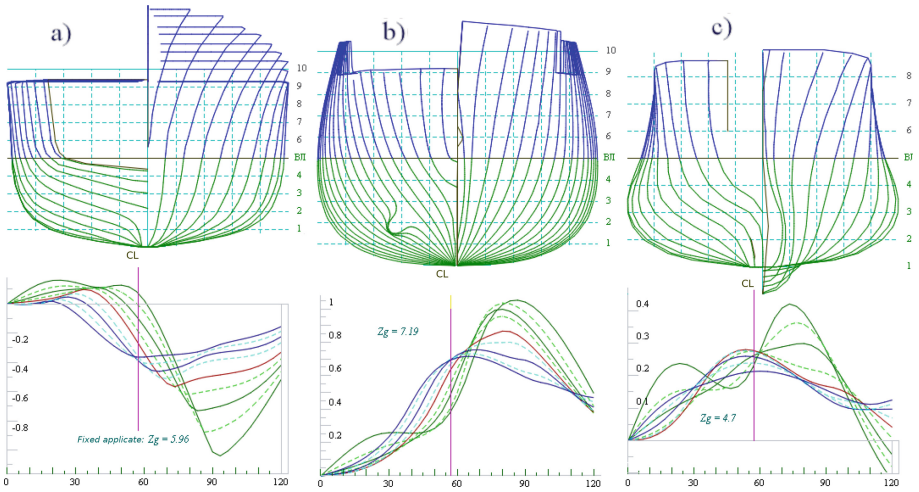


**Fig. 3.** Transversal projection of a ship hull (top), static ship stability diagrams for a fixed centre of gravity $Z_g$ and metacentric height equal to 1% of a beam for contemporary (a), historical (b) and improved storm seakeeping (c) ships.

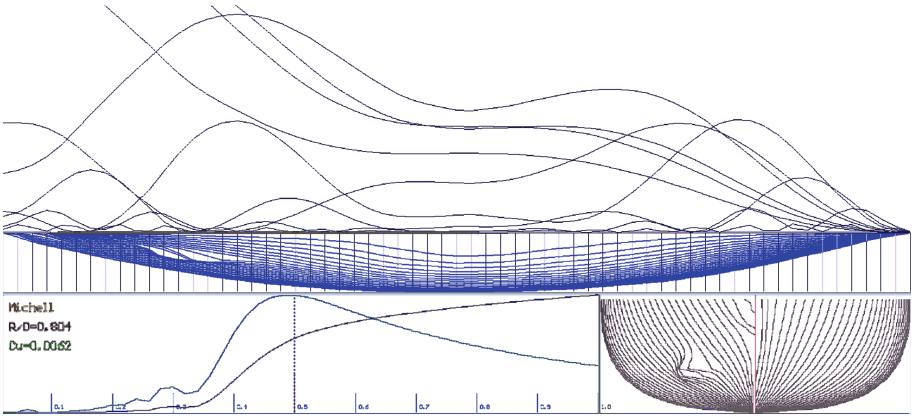


**Fig. 4.** Wave resistance, radiation intensity and ship waves intensity along the ship hull.

### 2.3 Triangulation of a Ship Hull Given by a Collection of Curves

In the original programme [5] that visualises ship lines and calculates hydrostatic characteristics, ship hull is described by a collection of curves; however, for a programme that simulates ship dynamics in rough sea this description is not convenient. A better representation would be a collection of triangles (a *triangular mesh*) that approximates analytic ship hull geometry. At the centre of each triangle pressure force induced by ocean waves is applied, and then these forces are used to calculate ship motion. Triangles is a better representation because they

- do not require recomputation every frame (like analytic curves),
- have simple formula for area (which is needed for pressure force calculation), and
- the same representation is used internally by graphical accelerators that visualise simulation frames.

In the following paragraphs we describe how analytically given ship hull is transformed into a fully connected collection of triangles.

Ship hull is transformed from analytic to discrete form by using *intermediate representation* — two-dimensional rectangular array of points, which makes it easy to obtain triangular mesh. Each ship hull section is transformed to such an array, and then these arrays are concatenated. Each row of the resulting array represents a frame, and each frame is an array of points of this frame. Since the array have to be rectangular, the number of points equal the maximum number of points among all original frames in VSL file. After that, endpoints are added to make curves closed, and the whole array is mirrored with respect to longitudinal axis. Then each rectangular patch of the resulting array is divided into two triangles to obtain triangular mesh. Duplicate vertices and faces, that may have

been introduced by mirroring or making curves closed, are removed from the mesh. So, intermediate representation is easy to transform to a triangular mesh, but we have to interpolate frame curves to generate the same number of points in order to transform them to such a representation.

It is straightforward to transform main section to a rectangular array of points. First, we determine the maximum number of points in a frame across all ship hull frames. Then for each frame we use cubic Hermite spline interpolation to generate the specified number of points. If each frame has the same number of point, we generate the same points that the original frame had, because the spline goes through all of them.

In contrast to the main section, transformation of bow and aft sections is done in multiple stages. During the first stage we generate intermediate points between subsequent frames using the curve describing the ship hull shape in longitudinal plane. For that purpose we create a polygon in longitudinal plane that consists of all points of this curve, all points of the frame that is the closest to the curve endpoints, but does not cross the curve, and all endpoints of the frames that are between this frame and the curve (it happens when first and last endpoint of the curve are close to different frames). Frame points that lie outside this polygon are removed. Polygon points that lie between subsequent frames become intermediate points. At this stage we decompose a bow/aft section into a collection of "rectangular" patches vertical sides of which are curves created from subsequent frames and horizontal sides of which are curves created from intermediate points (Fig. 5).

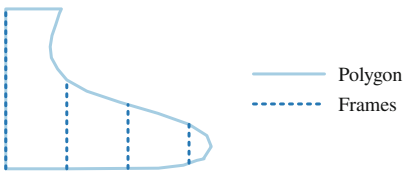


**Fig. 5.** Longitudinal slice of bow section of a ship hull that shows horizontal curves with large curvature.

A patch that is defined by four curves is called Coons patch [4]. Coons patch is a parametric surface $S(u, v)$ between four curves $c_0(u)$, $c_1(u)$, $d_0(v)$, $d_1(v)$:

$$
\begin{aligned}
&S(u,v) = C_1(u,v) + C_2(u,v) - C_3(u,v),\\
&C_1(u,v) = v'c_0(u) + vc_1(u), \quad v' = 1 - v,\\
&C_2(u,v) = u'd_0(v) + ud_1(v), \quad u' = 1 - u,\\
&C_3(u,v) = c_0(0)u'v' + c_0(1)uv' + c_1(0)u'v + c_1(1)uv,\\
&c_0(0) = d_0(0), \quad c_0(1) = d_1(0), \quad c_1(0) = d_0(1), \quad c_1(1) = d_1(1).
\end{aligned}
$$

Here $C_1$ is linear interpolation between points $c_0$ and $c_1$, $C_2$ is linear interpolation between points $d_0$ and $d_1$, and $C_3$ is bilinear interpolation between corner points

of the patch. $C_1$ and $C_2$ are ruled surfaces — surfaces between two curves, that are generated by interpolating corresponding curve points. Bicubic interpolation can be used instead of bilinear to get the same derivative when joining multiple Coons patches together, but for a grid of interior points linear interpolation is enough.

Using these formulae we generate a grid of interior points between curves of each bow/aft patch during the second stage. Our first approach was to use one Coons patch for each subsequent pair of frames, but we have found that some ship hulls have horizontal curves with large curvatures which cause linear interpolation to produce cusps on the resulting surface (Fig. 6). We solved this problem by using multiple smaller Coons patches that were arranged vertically for each subsequent pair of frames. This approach removes the cusps, but the vertical size of the patch have to be small enough to reduce the effect of the large curvature of the horizontal curve. After generating grids of intermediate points with Coons patches for each subsequent frame, we concatenate them to obtain rectangular array for bow/aft sections.

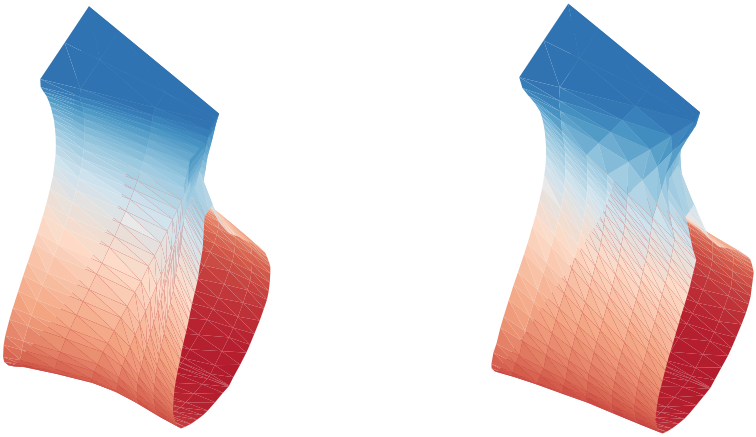

**Fig. 6.** Part of the bow section of the ship hull. The section with single Coons patch spanning the whole frame (left). The section with multiple smaller Coons patches arranged vertically (right).

After transforming each section of the ship hull into a rectangular array and concatenating them, we obtain an array of dimensions $m \times n$. We then use two-dimensional cubic Hermite spline interpolation to generate a grid of $(2m-1) \times (2n-1)$ points. Additional points increase surface smoothness and provide better approximation for the original ship hull. Finally, we transform the resulting grid into a triangular mesh to obtain three-dimensional discrete ship hull model, which is ready for the use in ship motion simulation and visualisation.

## 3 Results

To prove VSL viability for using it as an alternative to another format that uses analytic curves called IGES [7], we measured triangulation performance and measured how many vertices and faces are generated when analytic curves and surfaces are transformed into them. We do not have the same ship hulls in both formats, so we resorted to measuring performance relative to vertex and face count. To work with IGES format we use OpenCASCADE library [6]. Benchmark results are presented in Table 2.

**Table 2.** VSL performance in comparison to IGES for different ship hulls.

| Ship | Format | Import time, s | No. of vertices | No. of faces |
|---|---|---|---|---|
| MICW | VSL | 0.019 | 5457 | 10912 |
| Aurora | VSL | 0.054 | 14653 | 29306 |
| 5415 | IGES | 1.300 | 21088 | 41841 |
| KVLCC2 | IGES | 24.000 | 57306 | 114110 |
| KCS | IGES | 32.500 | 626188 | 1243307 |

We chose MICW (a hull with reduced moments of inertia for the current waterline) and Aurora cruiser ship hulls in VSL format and tanker (KVLCC2), container ship (KCS) and combat ship (5415) in IGES format that are freely available on the Internet[1]. VSL format showed higher performance in comparison to IGES format considering the number of vertices and faces. The resulting triangular meshes are shown in Fig. 7.

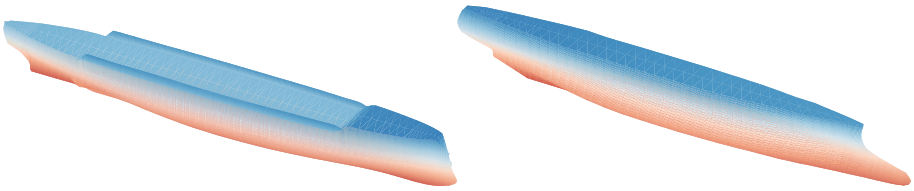

**Fig. 7.** Final Aurora and MICW three-dimensional triangular meshes.

## 4 Discussion

### 4.1 Advantages

*Faster Triangulation.* Faster triangulation is advantageous for performing ship motion simulations on a regular workstation. These machines are usually not

[1] https://simman2014.dk/ship-data/.

very powerful and do not have a lot of memory for complicated mesh preprocessing. IGES importer uses multiple threads for triangulation, whereas VSL does everything in one thread. Yet, it achieves higher performance on a regular workstation.

*Ease of Editing.* Most of the VSL files from Vessel database were written by hand. Points for frames were generated in a drawing programme by using legacy blueprints as an overlay and approximating curve projections by points. Then points were exported in plain text format and subsequently converted to VSL format by hand. Thus, there is no need in complicated and expensive CAD programme to produce Vessel files.

*Compatibility with Legacy Blueprints.* Legacy ship hull blueprints were mostly done by drawing projections of ship lines on a paper, and VSL incorporates longitudinal projections for bow and aft and transversal projection for the main section. It is easy to convert legacy blueprint to VSL file by overlaying frame points on the drawing. It is also easy for people who used to legacy blueprints to reason about ship hull characteristics.

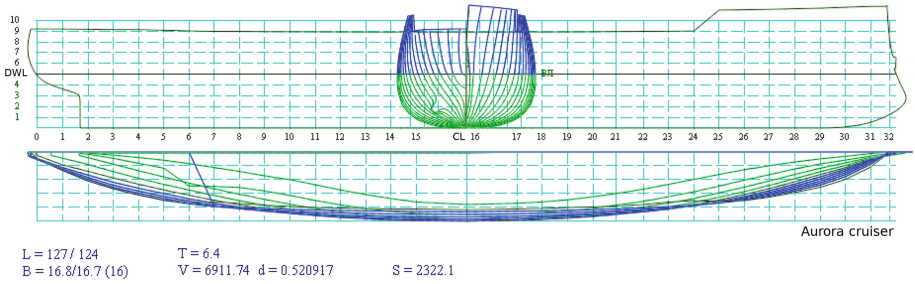


**Fig. 8.** Aurora cruiser digital model.

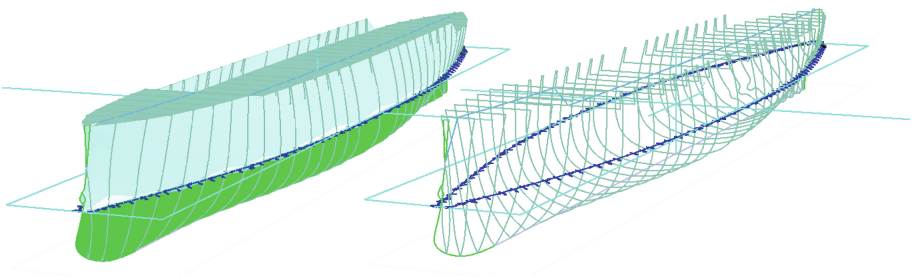

**Fig. 9.** Aurora cruiser hull is prepared for a numerical ship hydromechanics experiment. Although, the number of frame points in the main section is relatively small, it allows for smooth approximation of hull surface with the desired accuracy.

### 4.2 Disadvantages

The main disadvantage of VSL format is that a triangular mesh generated from it approximates ship hull surface only with a certain accuracy, but cannot represent it exactly. This is due to the fact that we choose the same number of points for each frame to be able to create rectangular array of all points and convert it to a mesh. Obvious solution to this problem is to use non-rectangular array, but it does not come without disadvantages. Non-rectangular array complicates triangle generation as we need to compute intermediate points for smaller patches, and it would be more difficult to make the resulting surface as smooth as the original, especially at intersections of frames with different number of points.

A simpler approach is to use rectangular array as the original ship hull representation, not an intermediate one. In that case each array element represents a geometric vertex and in addition the type of spline surface that represents ship hull surface *exactly* is specified in the file. That way it is easy to create triangular mesh that would be an exact representation of the original analytic surface when the number of grid points tends to infinity, and the resulting surface would be as smooth as the original one with all derivatives and surface normals computed using analytic representation. Similar approach is followed by FastShip [2] in which ship hull is defined by a B-spline surface, that in turn is defined by control points, allowing ship hull designer and ship motion simulation programmes to use exactly the same ship hull representation. One direction of future work is to incorporate B-splines into existing format.

## 5 Conclusion

In computational experiments we used theoretical blueprints of Aurora cruiser which has excellent seakeeping characteristics and for which there can be no dangerous situations in severe storm waves. We presented the more complicated geometric ship hull form which is optimised for storm seakeeping. This ship hull has relatively small moments of inertia for the area of the current waterline (MICW). For this hull mutual compensation of external force excitations was proved in seakeeping trials in severe storm waves in model basin of Leningrad shipbuilding institute under the guidance of Prof. Alexander Nikolaevich Kholodilin [1].

Although, traditional plaza table of ordinates with a few frames has relatively small hull representation accuracy, it gives satisfactory representation of ship hydrostatic and hydrodynamic characteristics in computational experiments using multiprocessor machines and supercomputers. For research stages we used formalised simulation and trochoidal models of storm sea waves, for which digital ship hull models are quite optimal.

We described efficient plain text format for three-dimensional ship hull geometry called VSL. This format uses longitudinal and transversal projections of ship lines given in analytic form. This format is easy to write by hand and easy to use for converting legacy ship hull blueprints into digital form. We demonstrated efficiency and viability of this format in comparison to IGES. Although, there are more precise alternatives for storing ship hull geometry, and VSL format gives

only approximate representation of the original ship hull, it is distinguished by the simplicity and efficiency.

**Acknowledgements.** Research work is supported by Saint Petersburg State University (grant no. 26520170 and 39417213).

## References

1. Search studies of ship seakeeping in storm waves, 3 edn. Lambert Academic Publishing (2018). (in Russian)
2. Alion Science Inc.: FastShip: basic ship design (2019). http://www.proteusengineering.com/fastship.htm
3. Bogdanov, A., Khramushin, V.: Vessel: blueprints for the analysis of hydrostatic characteristics, stability and propulsion of the ship (2015). http://www1.fips.ru/fips_servl/fips_servlet?DB=EVM&DocNumber=2015621368&TypeFile=html. (in Russian)
4. Coons, S.A.: Surfaces for computer-aided design of space forms. Technical report, Massachusetts Institute of Technology (1967)
5. Khramushin, V.: Analytic ship hull shape construction, wave resistance calculations, theoretical blueprint feature curve calculations, and ship stability diagrams (2010). http://www1.fips.ru/fips_servl/fips_servlet?DB=EVM&DocNumber=2010615849&TypeFile=html. (in Russian)
6. Open Cascade SAS: Open CASCADE technology, 3D modeling & numerical simulation (2019). https://www.opencascade.com/
7. Smith, B.M., Wellington, J.: IGES, a key interface specification for CAD/CALM systems integration. In: Computer-Aided Geometry Modeling, pp. 279–319 (1983)