Veleučilište u Virovitici

ET^{2}eR vol. IV, br. 1, 2022., str. 1-7

UDK: 510.5
Izvorni znanstveni rad (Original Scientific Paper)

Evaluation of the quad oriented meshing algorithms in Gmsh

Luka Grubišić[1], Domagoj Lacmanović[2], Josip Tambača[3]

[1]University of Zagreb, Faculty of Science, Department of Mathematics, Bijenička 30, 10000 Zagreb, Croatia, luka.grubisic@math.hr
[2] University of Zagreb, Faculty of Science, Department of Mathematics, Bijenička 30, 10000 Zagreb, Croatia, domagoj.lacmanovic@math.hr
[3] University of Zagreb, Faculty of Science, Department of Mathematics, Bijenička 30, 10000 Zagreb, Croatia, josip.tambaca@math.hr

Sažetak

U ovom radu procijenjena je izvedbu triju algoritama za generiranje četverostruko dominiranih površinskih mreža geometrijskih modela u strukturnoj analizi broda. Uspoređuju se algoritmi koji se temelje na kombinatornom pristupu pakiranja ploha paralelogramima i algoritmi koji se temelje na varijantama Delaunayeva naprednog frontalnog pristupa. Specifičnost meširanja u strukturnoj analizi brodova je po tome što su ograničenja na algoritam definirana pravilima klasifikacijskih grupa brodova koja sadrže linearna ograničenja na geometriju. Linearna ograničenja izravno su u suprotnosti s načelima Delaunayeva naprednog frontalnog pristupa. Predstavljen je specifičan pokazatelj kvalitete za procjenu kvalitete takvih mreža i uspoređena tri dostupna algoritma u Gmsh-u. Na kraju, procijenjena je izvedbu strategije preduvjetovanja za ove algoritme i argumentirana njezinu prikladnost za dani zadatak.

Ključne riječi

zareza aproksimacije konačnih elemenata; wire modeli; algoritmi frontalne mreže; packing algoritmi

Abstract

In this paper we benchmark the performance of three algorithms for generating quad dominated surface meshes of the geometric models in ship structural analysis. We compare algorithms based on the combinatorial approach of packing the surfaces with parallelograms and those algorithms which are based on the variants of the Delaunay advancing front approaches. The meshing in ships structural analysis is specific in that the restrictions on the algorithm are defined by the rules of the ship classification societies which contain linear restrictions on the geometry. Linear restrictions are directly at odds with the principles of the Delaunay advancing front approaches. We present a specific quality indicator for assessing the quality of such meshes and compare three available algorithms in Gmsh. Finally, we assess the performance of our preconditioning strategy for these algorithms and argue for its suitability for the task.

Keywords

finite element approximations; wire models; frontal mesh algorithms; packing algorithms

Introduction

This paper presents a report on the development of a fully automatic meshing algorithm for ship structural analysis. This solution is aimed at small design bureaus for developing coarse finite element models of ships' structures in the early stages of the design development. These wire models have to be generated fully automatically since they are needed in the phase of the design when a designer searches for solution candidates and so a generated model, after being numerically assessed can likely be discarded and then modified only in order to be meshed again. Subsequently, this procedure has to be fast and sufficiently regular, according to the rules for analyzing ship structural safety.

According to the guidelines of the Ship Structure Committee (Wang et al. 2019) mesh generators can be classified in the following three categories, see Table 1. Let us point out the Ship Structure Committee is the US inter departmental authority which issues design and safety guidelines for the US Coast Guard, the US Navy, American Bureau of Shipping, the US Maritime Administration, and the US Military Sealift Command.

The manual meshing approach is the only approach which does not assume a processed geometric model of the ship's structure. It is however very expensive in terms of engineering time and is measured in weeks or months of high-level work. The automatic and interactive meshing approaches both require a cleaned-up geometry, which in turn incurs many hours of expert engineering work. This work is like the work necessary for the manual meshing but is speeded up by the tools available in modelling software such as a general software suit as Altair [1], or a dedicated ship structure analysis tool like MAESTRO [7]. The meshing in Altair suit is performed by the Altair Hypermesh which among other algorithms offers the HM-Automesh[1] algorithm. In the case of MAESTRO, meshing is done using Rhino3d[2] or NAPA-Steel[3].

The algorithms available in Rhino3d and NAPA-Steele are primarily interactive and allow automatic meshing of large (selected by hand) sections of the geometry which are then automatically joined into the full wired mesh model.

TABLE 1: TAXONOMY OF MESHING APPROACHES

	Meshing approachs		
	Description	Properties	CAD model requirement
(1)	Automatic (or batch) meshing with predefined parameters	Minimum user involvement but generates more elements and nodes than the other approaches.	A clean CAD geometry is needed to achieve accuracy.
(2)	Mapped (or interactive) meshing with more predefined parameters and pre-trimmed partitions	Mapped meshing requires more time and user involvement compared with automatic meshing, but mesh quality can be significantly improved, especially around the areas of interest.	A clean CAD geometry is needed to achieve accuracy.
(3)	Manual meshing section by section	Most time and user involvement . A full control over the mesh design and quality can be achieved.	Not dependent on the quality of the CAD geometry.

Source: Report SSC 475 [15].

Note that all the afore mentioned approaches assume that the geometry of the model has been corrected for element overlaps and element connectivity errors. This process is called topology refinement and is performed by hand in the section-by-section manner using tools for geometric modelling.

On the other hand, it is our aim to present and assess the performance of a fully automated mesher which combines an automatic topology refinement

---

[1] https://www.altair.com/hypermesh/

[2] https://www.rhino3d.com/

[3] https://www.napa.fi/software-and-services/ship-design/structural-design/

together with a fully automatic quad-oriented mesh generation. This procedure has been realized by combining the Boolean geometry operations from the Open CASCADE geometric kernel together with the quad oriented meshing algorithms implemented in Gmsh (Geuzaine and Remacle, 2009) and (Remacle et al., 2013). Open CASCADE is a geometric CAD kernel which is fully integrated with Gmsh and can be controlled using python module pygmsh[4]. All of these routines are open source and are released under GPLv3 license which allows for their commercial use. The preconditioned fully automatic mesh generation pipeline has been realized using these building blocks and has been released under the same GPLv3 license under the name pyREMAKEmsh[5]. The underlying algorithm to control the generation of a quad oriented mesh, called the virtual stiffener algorithm, has been presented in the paper (Grubišić et al., 2021).

In this paper we will evaluate the use of three alternative algorithms to produce the quad oriented mesh, after a pass of the virtual stiffener algorithm. First, we will consider the surface packing algorithm, called packing for parallelograms, from (Baudouin et al., 2014). Then we will consider the Delaunay marching front algorithm combined with the Blossom quad recombination algorithm from (Remacle et al. 2013). Packing for parallelograms is the simplified version of the Delaunay frontal mesh algorithm. However, its simplicity allows sometimes for better generation of the quadrilateral dominated meshes with linear geometric constraints. Finally, since the recombination algorithm might leave some triangles in the mesh, if recombining all the triangles would lead to quads of low quality. In such cases, to generate full-quad meshes, Gmsh offers the option of the full-quad recombination algorithm. This algorithm typically generates a mesh with more than 99% of quadrilaterals. This is achieved by iterating the processing pipeline consisting of subdivision, recombination, and topological smoothing. We will show, using statistical indicators, that even though full recombination generates a mesh with almost all quadrilateral elements the quality of these elements is lesser than with other approaches and produces a mesh with many more nodes. Subsequently this mesh might lead to the less accurate computation of nodes with some finite element choices, such as low degree shell elements.

As a benchmark let us note that according to the Hypermesh training resources[6] its Automesh algorithm is considered to have run successfully if it has generated a mesh which has 80-90% of elements satisfying the constraints and starting from the fully cleaned CAD geometry (eg. by using hand topology refinement). In comparison, pyREMAKEmsh will generate meshes with almost 95% of elements satisfying the restrictions while starting from the uncleaned CAD geometry.

2. Methodology

A geometry of the ships' structure will be represented by the dictionary of geometric entities. These will be points, rods, surfaces, and openings. The points are elements which describe positions where loads during the finite element simulation are going to be applied or the measurements will be taken. Rods describe the panel stiffeners or pillars. The set of surfaces is divided in the set of web surfaces. Those are restricted to be parallelograms with one dimension much smaller than the other and are used to reinforce plates into stiffened panels. The Set of regular surfaces contains convex quadrilaterals defined by four co-planar nodes. Finally, the set of warped surfaces contains closed loop surfaces defined by four not necessarily co-planar corners. An example of a geometry can be seen on Picture 1.

A mesh T of the geometry (dictionary) G is a union of tessellations into regular quadrilaterals or triangles of all of the elements in the dictionary. The task of generating a mesh is challenging due to the conformity constraints which are placed on the union of tessellations. Furthermore, these rules are formally prescribed and do not necessarily correspond with purely geometric and combinatorial algorithms.

The rules for generating surface meshed are prescribed by ship classification societies such as (Det Norske Veritas and Germanischer Lloyd, 2015) or (Nersesian and Mahmood, 2009). The essential requirement is that there are no hanging nodes in the mesh on connected geometries. That is to say that any two elements from the mesh can only intersect on a joint edge or on a joint vertex. Further restrictions address the restrictions on the internal angles of the elements, the aspect ratio of the sides of triangles and quadrilaterals and the relationship between the element size and the thickness attribute of the actual physical element which it is modelling.

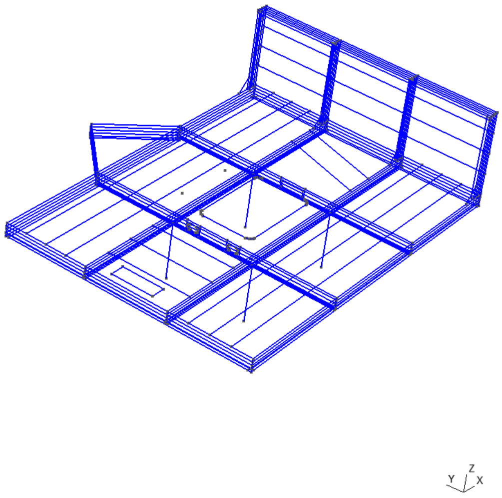
---

[4] https://github.com/nschloe/pygmsh

[5] https://github.com/PMF-ZNMZR/pyREMAKEmsh

[6] https://altairuniversity.com/wp-content/uploads/2014/02/HM_Automeshingintro.pdf

FIGURE 1: GEOMETRY G$_1$



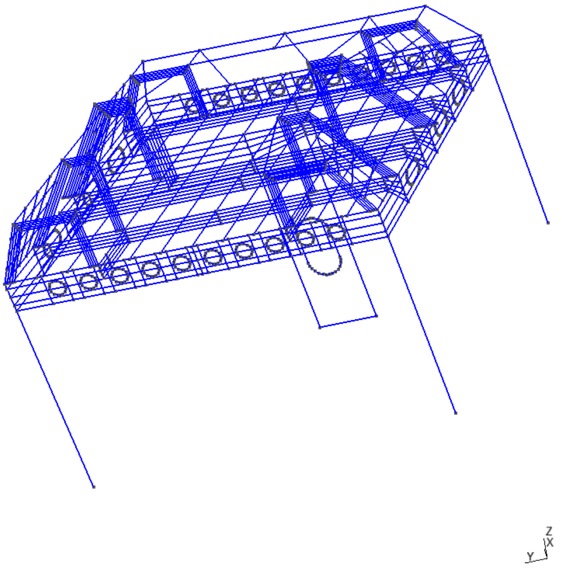
Source: generated by pyREMAKEmsh.

We summarize the guidelines which should be met when modeling a physical structure. These restrictions correspond to specifications for generating a coarse mesh.

- The use of 2 node line elements and 4 or 3 node shell elements is sufficient for hull structure representation.
- Quadrilateral shell elements - with inner angles below 45° or above 135° between edges should be avoided.
- Quadrilateral shell elements with high aspect ratio as well as distorted elements should be avoided. The aspect ratio is to be kept close to 1 but should not exceed 3 for 4 node elements.
- The use of triangular shell elements is to be kept to a minimum.
- A web surface should be meshed with at least four elements per shorter dimension.
- In the area where high stresses are expected the aspect ratio of shell elements is to be kept close to 1 and the use of triangular elements is to be avoided.

This last statement on the avoidance of triangular elements in high stress areas needs to be further clarified. In class guidelines, such as (Det Norske Veritas and Germanischer Lloyd, 2015), there are examples of meshes which adhere to these rules (cf. pages 30 and 71 of (Det Norske Veritas and Germanischer Lloyd, 2015)). Subsequently we conclude that avoidance of triangular elements is to be quantified as keeping the percentage of triangles in those areas as low as possible. We set the threshold on the percentage of triangles in the mesh to 5% and we use this as quantitative realization of the optimization instruction to avoid triangular elements as much as it is allowed by the geometry.

FIGURE 2: GEOMETRY G$_2$ REPRESENTING A SECTION OF A SUPERSTRUCTURE. WE SEE VIRTUAL STIFFENERS PLACED AROUND OPENINGS IN WEB SURFACES.



Source: generated by pyREMAKEmsh

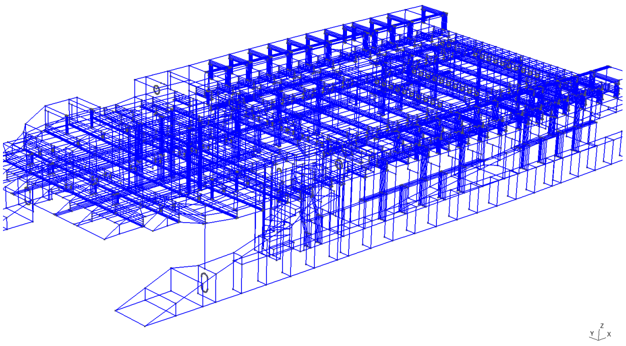
For the fine mesh zone, the rules are slightly extended. In such a zone the following requirements are placed on the local mesh and its transition to the global coarse mesh.

- A uniform quadratic mesh is to be used with a smooth transition leading up to the fine mesh zone.
- Finite element size it to be limited to 50 mm x 50 mm.
- The extent of the fine mesh zone is to be not less than 10 elements in all directions from the area under investigation.
- The use of extreme aspect ratio (greater than 3) and distorted elements (corner angles below 60° and greater than 120°) are to be avoided.
- The use of triangular elements is to be avoided.

- The transition between shell elements and beam elements is to be modeled so that the overall stiffener deflection is retained.
- Openings - the first two layers of elements around the opening are to be modeled with mesh size not greater than 50 mm x 50 mm.
- Face plates - of openings, primary supporting members and associated brackets are to be modeled with at least two elements across their width on either side.

The main meshing algorithm which are available in Gmsh are all based on the notion of the crossfield. A crossfield is a heuristic function which describes the local orientation of a mesh. It is defined by the boundaries of the surfaces and then it is propagated from the boundaries towards the interior. There are two main variants of the advancing front Delaunay mesh generation algorithm. One is based on the triangular Delaunay mesher in the L^∞ norm which is then followed by the Blossom recombination algorithm (Edmonds, 1965). The Blossom recombination algorithm is a graph theoretic algorithm which produces a perfect matching of the nodes in a graph. Here we take the triangles in the mesh to be the nodes of the graph and two triangles sharing an edge are then connected by an edge in the sense of the topology of this graph. The perfect matching algorithm produces a quadrilateral dominated mesh by joining triangles into quadrilaterals while optimizing a measure of the closeness of their interior angles to the right angle. The algorithm is stopped as soon as the matching of further triangles would decrease the quality measure based on their internal angles. There is also an option of a full recombination algorithm which aims to match all triangles (almost always more than 99% of quadrilaterals in a mesh is achieved). This option, as we will see does not necessarily lead to higher quality meshes. The challenge in tessellating the geometry into regular quadrilaterals comes from complex interactions between web plates and surface plates and also from the introduction of the curvature in the geometry from circular openings. The virtual stiffener algorithm from (Grubišić et al., 2021) is designed to control the scope of the influence of a curved surface on the orientation of a mesh. The virtual stiffener is a stiffener which is introduced before the Boolean geometry operations but is not included in the dictionary as a new element. The virtual stiffeners are adaptively introduced around openings on a web surface. Further restrictions on this adaptive algorithm are placed because of the presence of warped plates in the geometry. These elements are treated differently by the Open CASCADE kernel and so the algorithm needs to be adapted. A further aspect of the algorithm is in selecting the longitudinal virtual stiffeners on the web plates so that the number of degenerate elements, due to the interaction of a virtual stiffener and an opening, is kept to a minimum.

Picture 3: Geometry G_3 representing a superstructure of a large yacht



Source: generated by pyREMAKEmsh

We use the following statistics to assess the quality of a mesh. First, we introduce the notion of the regular quadrilaterals and the distorted elements. An element is distorted if it is quadrilateral with aspect ratio larger than 3, or with at least one angle less than 10^0 or more than 170^0. Distorted triangles are triangles with aspect ratio 1:5 or more or with one internal angle less than 10^0 and more than 170^0. The regular quadrilateral is the quadrilateral with internal angles between 80^0 and 100^0. The quantity

- $\xi(T)$ is the percentage of regular quadrilaterals in T
- $\tau(T)$ is the percentage of elements which are not triangles
- $\rho(T)$ is the percentage of distorted elements

We also study these indicators by restricting the percent-ages solely to web surfaces. In this case we use the notation $\xi_w(T)$, $\tau_w(T)$ and $\rho_w(T)$. We denote the number of elements in T by $|T|$.

3. Results

We evaluate the performance of the algorithms by comparing the local mesh quality indicators for web surfaces $\xi_w(T)$, $\tau_w(T)$ and $\rho_w(T)$ and global mesh quality indicators $\xi(T)$, $\tau(T)$ and $\rho(T)$. We use the geometries G_1, G_2 and G_3 which are presented on Pictures 1, 2 and 3.

As a benchmark for comparing the results of the experiments as a criterion for assessing the result we use the guidelines for the design piplenine from (Altair University, 2021) where it is stated that an automatically generated mesh which has up to 75%-90% good elements is acceptable as a first step in future interactive mesh refinement. We define the notion of a regular quadrilateral as a quadrilateral whose internal angles are between 80^0 and 100^0. A regular triangle has an angle in the range between 80^0 and 100^0 and an aspect ratio of less than 1:5.

TABLE 2: GEOMETRY G_1

	Quality indicators			
	|T|	*ξ(T)*	*τ(T)*	*ρ(T)*
Packing for parallelograms simple recombination	11,902	80.18 %	96 %	1.77 %
		Web: 75.95 %	Web: 94.83 %	Web: 7.60 %
L^∞ frontal mesh generator with Blossom recombination	11,489	78.02 %	96 %	1.46 %
		Web: 71.94 %	Web: 95.93 %	Web: 8.97 %
L^∞ frontal mesh generator with full recombination	19,481	66.58 %	99.8 %	15.45 %
		Web: 58.27%	Web: 99.7 %	Web: 15.05 %

Source: pyREMAKEmsh.

TABLE 3: GEOMETRY G_2

	Quality indicators			
	|T|	*ξ(T)*	*τ(T)*	*ρ(T)*
Packing for parallelograms simple recombination	11,830	61.06 %	89.8 %	5.3 %
		Web: 42.31 %	Web: 83.12 %	Web: 14.41 %
L^∞ frontal mesh generator with Blossom recombination	11,802	57.52 %	90.1 %	5.74 %
		Web: 39.93 %	Web: 84.89 %	Web: 14.77 %
L^∞ frontal mesh generator with full recombination	22,211	46.57 %	99.9 %	23.56 %
		Web: 38.17 %	Web: 99.9 %	Web: 38.40 %

Source: pyREMAKEmsh.

TABLE 4: GEOMETRY G_3

	Quality indicators			
	|T|	*ξ(T)*	*τ(T)*	*ρ(T)*
Packing for parallelograms simple recombination	128,070	77.81 %	94.4 %	1.95 %
		Web: 73.78 %	93.03 %	Web: 8.97 %
L^∞ frontal mesh generator with Blossom recombination	129,405	76.14 %	94.6 %	1.91 %
		Web: 70.97 %	Web: 92.96 %	Web: 7.94 %
L^∞ frontal mesh generator with full recombination	232,720	60.38 %	99.8 %	18.37 %
		Web: 57.37 %	Web: 99.81 %	Web: 23.37 %

Source: pyREMAKEmsh.

Let us briefly outline the choice of the geometries for these tests. The geometry G_1 is the standard test geometry consisting of a section of a deck and a bulkhead. The dictionary does not contain the warped plates. The geometry G_2 is the only geometry in this test suit which contains the warped plates. The geometry G_3 is chosen for its size and complexity.

From the Tables 2,3 and 4 we see that in all of the geometries the advancing front algorithms preconditioned by the virtual stiffener and Boolean topology refinement algorithm from (Grubišić et al., 2021) produces meshes with fewer than 10 % triangles – in the case when warped plates are in the geometry – and fewer than 5 % triangles when there are no warped plates in the geometry. The algorithm seems to scale well with the size of the model, since the geometry G_3. Has been meshed with as many as 130,000 elements. The Blossom recombination algorithm typically produces meshes with fewest degrees of freedom. However, packing for parallelograms together with simple recombination algorithm produces meshes which have the largest percentage of regular quadrilaterals. The statistics for the web plates indicate that resolving the geometry around openings is the main challenge in generating these meshes. However, the number of degenerated elements on web surfaces has been observed to be small. Most degenerate elements in the mesh appear due to the unintended overlaps in the geometry. Note that the overall ratio of degenerate elements is within the mesh optimization requirements as posed by the classification societies, (Huges and Paik, 2010).

The full recombination algorithm with mesh refinement can produce almost fully quadrilateral meshes. However, the price is the significant number of degenerate elements, as measured by $\rho(T)$. Furthemore, the overall number of elements is almost double of what the packing for parallelograms and the L^∞ Delaunay algorithm can achieve. Also, the ratio of regular quadrilaterals is significantly lower. Moreover, the number of degenerate elements on web surfaces has considerably increased. This has mostly stemmed from the relatively large ratio of quadrilateral elements with bad aspect ratio.

4. Discussion and conclusion

In this paper we have evaluated the three algorithms for the fully automatic meshing (including the automatic topology refinement) of geometries describing ships' structures. The simplified marching front algorithm called packing for parallelograms generated meshes with the largest ratio of regular quadrilaterals, while being close to the optimal L^∞ Delaunay mesh generator in terms of other indicators. Subsequently, it is our recommendation that the packing for parallelograms with virtual stiffener preconditioning and geometry refinement based on Boolean operations is the method of choice to automatically generate coarse meshes for ship structural analysis.

Acknowledgements

This work has been supported in part by EU Regional Development Funds under the project: KK.01.2.1.01.0124 - Development of efficient methodology for finite element method based structural analysis of marine structures – REMAKE. Thanks are due to the project coordinator Bonum d.o.o team for help during preparation of an automatic meshing module.

References

1. Altair University. (2021): Practical Aspects of Finite Element Simulation. Altair Engineering, Inc. eBook https://altairuniversity.com/free-ebooks/free-ebook-practical-aspects-of-finite-element-simulation-a-study-guide. (8.10.2021)
2. Baehmann, P.L.; Wittchen, S.L.; Shephard, M.S.; Grice, K.R.; Yerry, M.A. (1987): Robust, geometrically based, automatic two-dimensional mesh generation. International Journal for Numerical Methods in Engineering, Vol 24, (pp 1043–1078)
3. Baudouin, T.C.; Remacle, J.F.; Marchandise, E.; Henrotte, F.; Geuzaine, C. (2014): A frontal approach to hex-dominant mesh generation. Advanced Modeling and Simulation in Engineering Sciences, Vol 1, p. 8
4. Blacker, T.D.; Stephenson, M.B. Paving. (1991): A new approach to automated quadrilateral mesh generation. International Journal for Numerical Methods in Engineering, Vol. 32, (pp. 811–847)
5. Det Norske Veritas and Germanischer Lloyd. (2015): Class Guidelines DNVGL-CG-0127. Finite element analysis.
6. Edmonds, J. (1965): Maximum matching and a polyhedron with 0,1-vertices. Journal of Research of the National Bureau of Standards Section B Mathematics and Mathematical Physics, p. 125.
7. Geuzaine, C. and Remacle, J.-F. (2009): Gmsh: a three-dimensional finite element mesh generator with built-in pre- and post-processing facilities. International Journal for Numerical Methods in Engineering. Vol. 79(11), (pp. 1309-1331)
8. Grubišić, L.; Lacmanović, D.; Tambača, J. (2021): Automatic mesh generation for structural analysis in naval architecture. Proceedings of the International Conference on Ships and Offshore Structures ICSOS 2021., pp. 10, Paper 13.
9. Hughes, O.; Paik, J.K. (2010): Ship Structural Analysis and Design; The Society of Naval Architects and Marine Engineers: New Jersey.
10. Jang, B.S.; Suh, Y.S.; Kim, E.K.; Lee, T.H. (2008): Automatic FE modeler using stiffener-based mesh generation algorithm for ship structural analysis. Marine Structures, Vol. 21, (pp 294–325)
11. Lévy, B.; Liu, Y. (2010): L^p Centroidal Voronoi Tessellation and Its Applications.ACM Trans. Graph., Vol. 29.
12. Nersesian, R.; Mahmood, S. (2009): International association of classification societies (pp. 765–774)
13. Owen, S. (2000): A Survey of Unstructured Mesh Generation Technology. 7th International Meshing Roundtable, p 3.
14. Pellenard, B.; Orbay, G.; Chen, J.; Sohan, S.; Kwok, W.; Tristano, J.R.(2014): QMCF: QMorph Cross Field-driven Quad-dominant Meshing Algorithm. Procedia Engineering, Vol. 82, (pp 338–350)
15. Remacle, J.F.; Lambrechts, J.; Seny, B.; Marchandise, E.; Johnen, A.; Geuzainet, C. (2012): Blossom-Quad: A non-uniform quadrilateral mesh generator using a minimum-cost perfect-matching algorithm. International Journal for Numerical Methods in Engineering, Vol. 89, (pp 1102 – 1119)
16. Remacle, J.F.; Henrotte, F.; Carrier-Baudouin, T.; Béchet, E.; Marchandise, E.; Geuzaine, C.; Mouton, T. (2013): A frontal Delaunay quad mesh generator using the L^∞ norm. International Journal for Numerical Methods in Engineering, Vol. 94, (pp 494–512)
17. Wang, E.D.; Bone, J.S.; Ma, M. and Dinovitzer, A. (2019): Guidelines for evaluation of marine finite element analyses. Ship Structure Committee. Report SSC 475.