Geosci. Model Dev. Discuss.,
https://doi.org/10.5194/gmd-2018-203-AC2, 2018

Geoscientific
Model Development
Discussions
Open Access

Interactive comment on "OceanMesh2D 1.0: MATLAB-based software for two-dimensional unstructured mesh generation in coastal ocean modeling" *by* Keith J. Roberts et al.

Keith J. Roberts et al.

krober10@nd.edu

Received and published: 28 December 2018

Dear Reviewer,

Thank you for taking the time to read through our manuscript and look over the software. Overall, while the reviewer appears to agree that the information in the manuscript and the software is useful, it is suggested that possibly a more algorithmic focus should be applied and/or assessment of simulation results of coastal ocean models as related to the mesh resolution heuristics. However, these are not the intended or desired directions that we want to take with this paper. The intent of this open-source software package is to be an integrated end-to-end implementation of automated coastal model generation from geospatial data to robust/stable computations that can be performed efficiently given the vast amount of data and complexity of the geometry. The paper aims to describe this software package and how it achieves the above stated intent by integrating together various algorithms and mesh size heuristics. In other words, it is the whole package that is much more significant than its individual components.

Reviewer comments are presented in red italics, author responses are included in plain-text.

A new mesh generation library: OceanMesh2D is described, focusing on the construction of multi-scale unstructured triangulations for applications in coastal ocean modelling. Adapting the well-known DISTMESH algorithm (Persson and Strang), and building on top of other open-source contributions for various mesh-based and geo-spatial processing tasks, the authors present a MATLAB-based meshing library designed to automate the unstructured grid generation work-flow for coastal ocean modelling configurations. In addition to a description of their MATLAB-based implementation, the authors present a variety of mesh-resolution heuristics to control element size throughout the domain. As well as a number of existing resolution functions appropriate for coastal modelling (distance-to-coast, barotropic wave-length scaling, etc) a set of new metrics (Rossby radius filtered bathymetric gradients, channel thalweg scaling, etc) are introduced âĂŤ focusing on better resolution of various dynamical processes and/or topographic features in unstructured models. While much useful information is contained in the paper, I am overall somewhat unsure what its focus is or should be. Currently, I feel the authors have provided a detailed description of their MATLAB-based implementation, with much specific discussion of various classes and routines to be found in the OceanMesh2D code-base. To me, this reads a little like a software user manual.

Your summary of the manuscript here is accurate and detailed. We appreciate that you noted that the information in the paper is useful. The focus of the paper is indeed related strongly to the description of our implemented software which is one of the major motivating factors for us to submit this manuscript to GMD in the “Model description paper” article style. It is not our intention to provide a generic algorithmic outline of mesh generation, and we fully intend for this to be specific to our software. However, the style of the software as implemented is not necessarily strongly tied to MATLAB and could also be readily implemented in for example Python. For example, we provide a class based workflow provenance which can be implemented in any programming language that supports the OOP style. We also avoid paid MATLAB toolboxes and use open source codes instead which would make the software more easily portable into other programming languages. Furthermore, we provide a user manual on the OceanMesh2D GitHub website that guides a user through generating meshes using the software, presenting the actual command lines and scripts that are used and so on. This is contrast to this GMD manuscripts which introduces the general structure of the software, mesh generation workflow, and demonstrates the various functionality of the software through presentation of example meshes.

If the authors intend to focus on algorithmic innovations, I suggest that a higher level and more mathematically-focused description of the algorithms be presented. While detailed discussions of various MATLAB functionality and the availability of open source code may undoubtedly be useful to model users, I do not feel that algorithmic discussions need to be focused on any particular implementation, and that in fact to do so may diminish adoption and re-implementation by other authors.

Please see our response to reviewer # 1 in regard to how we plan to revise the manuscript in order to more clearly show our algorithmic contributions to this problem. The organizational changes are detailed in that response. In the revised manuscript, we plan to highlight our multiscale meshing technique (currently section 5.2) earlier on in the manuscript and to revise the presentation of Algorithm 2 for shoreline simplification purposes. Reviewer #1 suggests that parts of the paper, such as the algorithmic psuedo-code describing Algo. 2, may be best put into the user guide and the presentation should be focused on a more pictorial overview, and we agree. We would want this paper to be accessible to a wide-range of *readers* (perhaps less so readers who want to re-implement the algorithms and more so to users of the software), and thus we prefer to avoid a mathematical and formulaic structure to the manuscript.

If algorithmic innovations are to be the focus of this paper, I suggest it may be necessary to better compare against (and demonstrate improvement over) existing coastal meshing strategies and packages âĂŤ highlighting the impact of any new algorithmic techniques.

The end-to-end nature of the software makes a direct comparison in terms of speed and quality of mesh development with other similar mesh development tools detracting from the point of the manuscript (to describe the software package as a whole and how it helps provides an efficient solution to automatically build coastal ocean models). Based on the author's combined personal experiences, using a GUI-based software to develop a mesh such as the one showed in Example 3 often takes months of dedicated work to build and ensure numerical stability. A comparison with such a similar mesh development tool would thus require many assumptions (i.e., user competency, availability to paid software, post-processing tools). We enumerate why we think a direct comparison with similar software packages and some related algorithms is not appropriate for this manuscript:

1. The mesh generator is a modified DistMesh2D algorithm. The DistMesh2D al-

gorithm in its original state would not be able to terminate with a suitable mesh quality in any computable time for the meshes we are concerned with which span large ocean domains and have complicated fractal shorelines. This was a major point of this work; to improve a popular and simple algorithm for a specific purpose. Deluanay refinement schemes and the like may be much faster, but also much more complex. There were a number of aspects improved upon to the DistMesh2D which are described in the text that make using the DistMesh2D algorithm feasible for this kind of problem (i.e., improving the speed and memory overhead of the initial point distribution, speeding up the calculation of the signed distance function, using a more sophisticated force-repulsion equation, improving the termination criteria, adding periodic mesh improvement strategies before and after termination).

2. The authors contend a comparison could be made with the ADMESH generator in terms of wall-clock speed given the same meshing boundary, but this would be somewhat unfair to OceanMesh2D as the user-defined time performing the shoreline simplification and editing the mesh to enable simulation post-generation cannot be added into the timings objectively. Building a geometrically well-shaped triangulation in isolation of the other required steps for coastal model development is not a very useful timing result for the readers who have practical goals (i.e., building a stable model for simulation).

3. An algorithmic contribution regarding shoreline simplification is tied to the smoothing-based nature of the DistMesh2D algorithm and is not a standalone operation like other shoreline simplification algorithms. All other mesh generators (e.g., GMSH, JIGSAW, ADMESH, Mesh2D, SMS, BlueKenue, and the like), to the author's knowledge, require the user to input a pre-defined well-formed (i.e., spacing corresponds to the mesh size function) polygonal boundary to mesh. Our algorithm requires a polygonal boundary but it does not need to be well-formed and we have provided methods embedded in the various classes to repair coastal

boundary geometries. This simplification of the shoreline often requires external GIS software and cannot seamlessly include the effects of arbitrarily sized mesh size functions, which our software does. Based on this logic, it isn't clear why or how a comparison to pre-existing software should be made.

4. Another algorithmic contribution was the ability to define a mesh size function on *nested* structured grids with minimal effort from the user. The authors do not see how or why a comparison with an unstructured mesh size function approach would be insightful. The rationale behind this approach was that high resolution LiDAR datasets, that many modern models are built with, come in small rectangular tiles close to the shoreline. Given the resolution and size of these LiDAR datasets, it would make little sense to build a mesh size function on an unstructured background grid since the point spacing in the would be essentially uniformly fine. Additionally, the ability to build mesh size functions on a nested box partitioning of the domain using structured grids enables this approach to work for large domains with the available structured geospatial datasets. The authors do not see any similar packages in existence that can readily incorporate many DEMs, as illustrated in Example3, in an automatic sense into the coastal ocean model development process. Following from this, a comparison with another software package is not readily available.

I feel the discussion of mesh-resolution heuristics would be much enhanced by actual simulation results and comparisons. The authors have introduced a number of new mesh scaling functions based on, for example, filtered bathymetric gradients and channel thalweg resolution. While these ideas are interesting, and may be expected to improve model skill under certain conditions, it would be beneficial to prove this was actually the case in practice and to document the impact of mesh resolution selection and design on model output. Without studying the effect on model physics, I feel it is difficult to judge the performance or utility of any particular mesh resolution heuristic. It may be possible to undertake several multi-mesh comparison studies: demonstrating *that simulations run on meshes generated using the new resolution heuristics compare more favourably with high resolution numerical studies or observational data.*

We certainly agree that assessing the impact of these mesh resolution heuristics on model physics (and numerics) is important, but this is not the direction we would like to go with this manuscript. In fact, we are currently in preparation of a paper with the exact aim as you are suggesting here. The aim of this GMD manuscript is straightforwardly a "Model description" of our mesh generator software and how it can improve the model development process. There are some other papers on the usage of these mesh resolution heuristiscs and the software enables the user to turn on and off whatever subset of them they feel necessary, which is what most users do with all other mesh generator libraries. Combining the assessment of model physics due to mesh heuristics with a detailed description of the mesh generator would create an unwieldy manuscript. Based on proposed organizational changes (detailed in response to Reviewer #1), the new manuscript would more clearly highlight the algorithmic contributions.

Printer-friendly version

Discussion paper