Composite mesh generator for CFD problems

To cite this article: E I Kalinin et al 2016 IOP Conf. Ser.: Mater. Sci. Eng. 158 012047

View the article online for updates and enhancements.

You may also like

- Temperature evolution of 3d- and 4f-electron magnetic ordering in the ferrimagnetic Mn self-doped perovskite Yb0.667Mn0.333O3, Andreas Dönni, Vladihir Y Pomjakushin, Lei Zhang et al.
- Scanning tunneling microscopic investigation on morphology of magnetic Weyl semimetal YbMnBi2, Zhen Zhu, Zhen, Dong Yan et al.
- Red bone marrow dose calculations in radiotherapy of prostate cancer based on the updated VCH adult male phantom, Jinqin Ai, Tianwu Xie, Wenjuan Sun et al.
Composite mesh generator for CFD problems

E I Kalinin\textsuperscript{1}, A B Mazo\textsuperscript{1} and S A Isaev\textsuperscript{2}

\textsuperscript{1}Kazan (Volga region) Federal University, 18 Kremlevskaya Street, Kazan 420008, Tatarstan, Russia
\textsuperscript{2}Saint Petersburg State University of Civil Aviation, 38 Pilotov Street, Saint Petersburg 196210, Russia

E-mail: kalininei@yandex.ru

Abstract. In present paper a brief introduction of HybMesh grid generator which uses composite approach is given. The process of complicated area meshing using HybMesh generator consists of sequential building structured prototype grids in relatively simple geometry, mapping them to a non-regular domains and superposing to assemble resulting grid. Transitional areas between two superposed low level grids are filled with triangular cells. Currently only 2D algorithms of such approach are implemented; 3D grids can only be restored as a result of extrusion or revolution of 2D objects.

1. Introduction
Composite grid generation approach is based on meshing of given arbitrary domain by geometric union of lower level grids built in more primitive domains. Advantages of such approach are the simplicity of meshing domains with complicated geometry and convenient definition of appropriate mesh refinement. Furthermore, resulting grid is partly structured and this feature can be utilized for building robust numerical solution schemes. In current paper grid generator HybMesh which is based on composite approach is presented. Building a grid using HybMesh generator includes three basic steps: constructing structured prototype grids, mapping these grids to non-regular geometry (if necessary) and final superposition of low level grids into the final one. Resulting grid could be exported into vtk, Tecplot, gmsh or ANSYS Fluent file formats for further calculations.

HybMesh is a cross platform open source application which is written in C++ and Python programming languages. Currently HybMesh provides scripting interface based on python syntax. Latest version installation distributive, source code and documentation can be downloaded from git repository [1].

2. Algorithms
HybMesh 2D grid object is a set of cells which are defined as sequences of vertices. There is no restriction on the number of vertices used in a cell. All basic geometrical operations such as translation, scaling, reflecting are naturally supported. 2D grids could be clipped by a closed contour. 3D grids could be built by extrusion or revolution of 2D grid objects.

2.1. Grid superposition
This is the basic HybMesh operation. Generally it takes two independent grids which have non-zero domain intersection and composes them into a single grid. The domain of resulting grid is exactly equal to the domain of geometrical union of parent grid domains and its cells reflect the original grids cells everywhere except for a zone around the line of parent grids contact which is triangulated providing smooth cell size transition. This zone is later referenced as a buffer zone.

Order of superposition does matter. For clarity sake we call the first of two original grids the base grid and the second one – overlaid grid. Buffer is always built within the base grid. Cells of the overlaid grid are transferred to the resulting grid mostly untouched. So, as you can see in figure 1, by swapping the grid roles we obtain different resulting grid geometry.

![Figure 1. Basic superposition example.](image)

Buffer zone is constructed as an area of all base grid cells which contain a vertex located no further than given buffer zone size from the contact line. Larger buffer zone provides smoother triangle grid within the buffer (see figure 2).
2.2. Mapping
This procedure maps the domain containing a grid (base domain) to any other domain with equal connectivity (target domain) and uses this mapping to translate the grid. Mapping is built as a result of solution of the Laplace equation with boundary conditions of the first kind [2]. Boundary values are calculated using domain’s boundary mapping defined by user as a set of corresponding points at base and target boundaries. Boundary value problem is solved by a finite element method using auxiliary triangular grid built within a solution domain. There are no restrictions on the input grid and target domain except for their connectivities should be equal and boundary mapping for each of bounding contours should be provided. Example of the ring grid which was mapped to doubly connected domain could be seen in figure 3.

Figure 2. Superposition with different buffer sizes.
2.3. **Grid prototypes**

HybMesh provides the collection of instruments for building structured grid prototypes. This includes simple building blocks like structured quadrangular grid in rectangle, radial grid in circle/ring domains, structured grid in triangle domain, algorithms for building custom structured grid in curvilinear quadrangle or circle areas.

There is also a special algorithm for building a prototype grid around any given contour which allows user to define a mostly orthogonal boundary layer grid. Later this grid is used as any other low level grid so the connection of boundary grid with domain grid is implemented using general superposition algorithm.

3. **Examples**

Figure 4 illustrates a complicated shark-like multiply connected area meshed in HybMesh using three simple steps. First, grid prototypes for “body” and “fins” were built in appropriate triangular and hexagonal areas which were mapped to non-regular geometry. Then these grids were coupled using chain superposition procedure with non-zero buffer size. Finally, “gills”, “eye” and “mouth” cuts were built as boundary layer grids and superposed on the basic grid.

![Figure 3. Mapping of ring grid using two (a) and four (b) reference points.](image)
Figure 4. Result of chain superposition of multiple prototype grids.

A practical example of using HybMesh approach towards building a grid for typical CFD problem is shown in figure 5. Here external boundary layer grids were built around foil and generator contours. Then they were superposed on a structured substrate grid. Finally, a patch grid was added to the wake region of vortex generator to provide an appropriate cell size refinement.

Figure 5. Airfoil with circular vortex generator.
4. Further development
Further HybMesh grid generator development involves several aspects. New grid prototypes including those widely used in CFD problems (potential grids, H-grids, C-grids etc.) will be added. 3D superposition algorithm using tetrahedral cells as buffer fillers will be implemented. It is also planned to provide HybMesh with graphical user interface and enhance the set of supported import/export mesh formats.

Acknowledgments
This work was supported by Russian Fundamental Research Foundation (RFFI) Grant No 15-01-06172 A.

References
[1] HybMesh latest release https://github.com/kalininei/HybMesh/releases/latest
[2] Liseikin V D 2009 Grid Generation Methods (2nd ed.) (Springer Publishing Company Incorporated) p 171