# Combining snappyHexMesh and enGrid

Oliver Gloth

*enGits GmbH, Langenbachstr. 3 - 79674 Todtnau, Germany*

In 2006 we first got into touch with OpenFOAM®. At the time, no reliable open-source tools existed which could generate anisotropic boundary layer grids. This was the major motivation to start our own open-source development (enGrid). Now we have 8 years of experience with this tool. For the vast majority of cases, we were able to deliver a suitable grid to our customers' problems or our own projects. In the few cases we have not been able to do this, snappyHexMesh came to our rescue. Saying that it always was an easy process, however, would be vastly exaggerated.

Looking back at our experiences with snappyHexMesh and enGrid, it appears fair enough to say they have their own strengths and weaknesses: In our opinion, the strength of snappyHexMesh is fast and reliable generation of isotropic hex grids. It might, occasionally, create small distortions in the vicinity of feature edges. For mainly isotropic grids, however, this is more of a cosmetic problem than a real issue. It has to be mentioned that this problem became smaller with every new release. In the presence of a boundary layer grid, however, such faults can lead to very distorted grids which completely destroy the solution.

On the other hand, enGrid is fairly reliable when it comes to growing prismatic layers. The far-field mesh is tetrahedral with the option to be converted to a polyhedral grid. This process does sometimes create skew or even concave cells in the transition zone between boundary layer and far-field. Choices are to either try and work with the distorted cells (e.g. by using more robust and less accurate discretisation schemes) or to work with the original tetrahedral grid. Both options are not very desirable and often lead to significant delays in a CFD project.

With snappyHexMesh and enGrid we have two open-source mesh generators of which one is good at creating boundary layer grids and the other at filling the remaining domain with high quality hex cells. It seems to be a natural step to go and try to combine the two. The main problem in achieving this is the very different nature of the cells. On one side we have triangular prisms, or the corresponding dual cells, and on the other end we have hex cells. Starting with version 2.1 of OpenFOAM the AMI functionality is available to couple arbitrary non-conforming patches. First tests with an enGrid boundary layer, and the rest of the domain meshed by snappyHexMesh, look very promising. Figure 1 shows a first computation using such an approach.

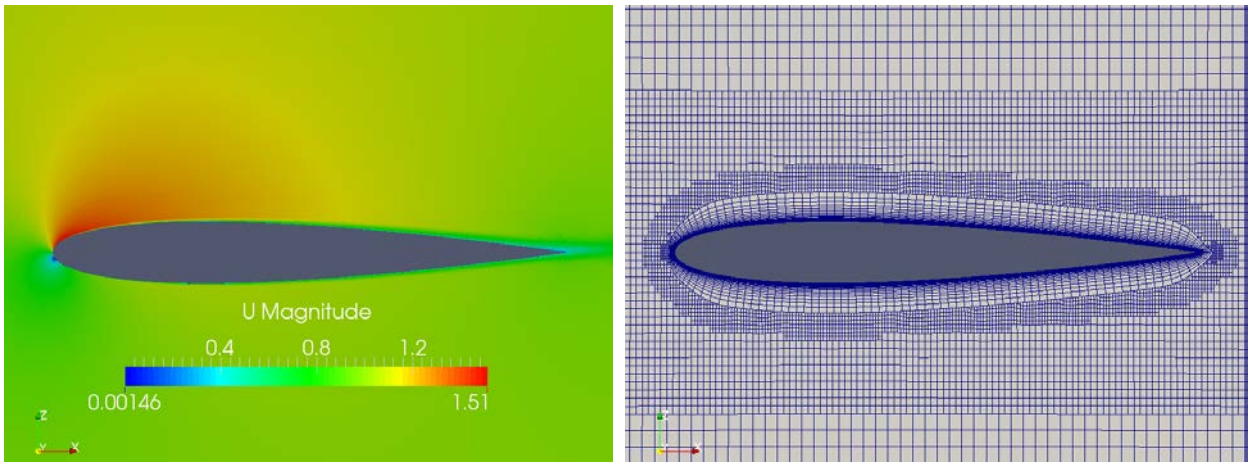


**Figure 1: A non-conforming combination (snappyHexMesh & enGrid)**

The convergence rate does not appear to suffer much from the rather unorthodox approach. Also, we still get a good parallel efficiency, because the coupling coefficients do not have to be recomputed on a regular basis, which would be the case for moving interfaces.

At the conference we plan to present a number of example grids and simulations. Where possible we will also present a performance and accuracy comparison with more traditional grids. If the approach proves useful for production simulations we will integrate a GUI into our meshing software enGrid. This GUI should enable the user to create a mesh without having to call all tools from the command line and merge the resulting grids by hand.