VOXEL MESHER : A QUICK INTRODUCTION

J.Cugnoni, 7.2022

What is VoxelMesher?

It is a “research” code that I wrote a while back to generate FE meshes from 3D segmentation images or voxel models. It uses CGAL and TETGEN meshing libraries, so check the licencing terms of those libs (Tetgen is free for non commercial use, contact Tetgen author for other use cases).

Getting started

Install CAELinux 2020 first, then…

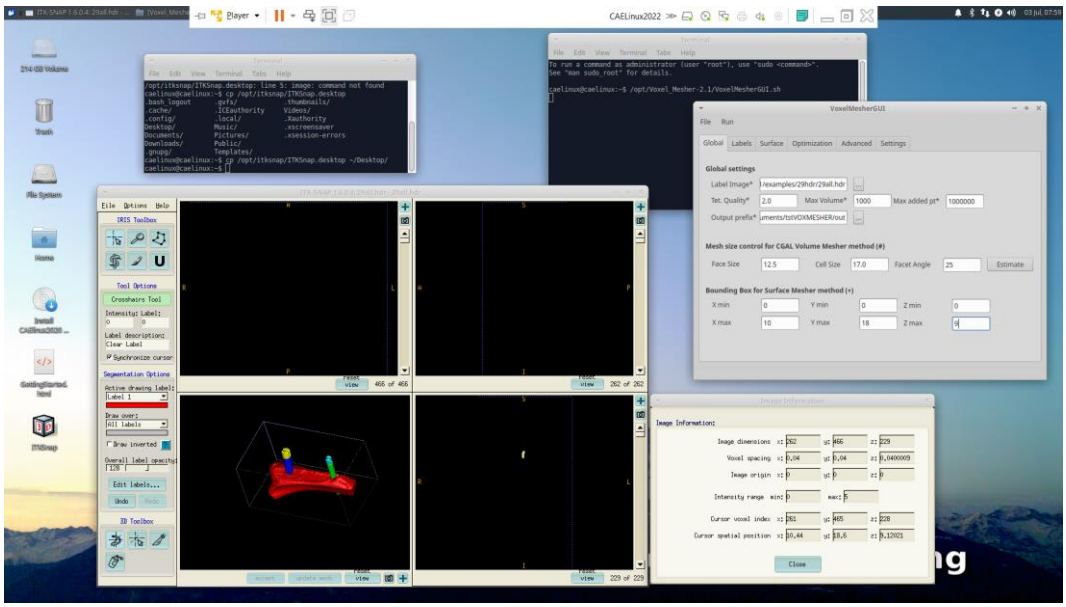

Fig 1. : Getting started, running ITKsnap and VoxelMesherGUI in CAELinux 2020 Lite

In CAELINUX2020 :

1. Open terminal and run VoxelMesherGUI :
   /opt/Voxel_Mesher-2.1/VoxelMesherGUI.sh
2. Create shortcut for ITKSnap (to view segmented image):
   cp /opt/itksnap/ITKSnap.desktop ~/Desktop
3. Run ITKSnap: double click on the ITKSnap icon on Desktop.
4. Load Grey scale image, for example File -> Open Grey scale image:
   /opt/Voxel_Mesher-2.1/examples/29hdr/29all.hdr
5. Load segmentation image, for example File -> Open Segmentation image:
   /opt/Voxel_Mesher-2.1/examples/29hdr/29all.hdr
6. Take note of the image dimensions / nb voxels and voxel size : File -> Image info. Here the image is approximately 10x18x9 mm with a voxel size of 0.04x0.04x0.04 mm.
7. Now in VoxelMesherGUI: select the “Label image” to discretize, for example:
   /opt/Voxel_Mesher-2.1/examples/29hdr/29all.hdr
8. Choose the prefix of the out files that we be created (all files will start with this prefix): for example here /home/caelinux/Documents/output

9. Set the maximum element “Max volume” (tetrahedra ~ L^3 / 10, where L is the max edge length). For example here, I don’t want to constrain the element volume too much so I set it to 0.5 mm^3.
10. Set the maximum face and cell diameter, here I have chosen 0.25mm. Leave angle limit between faces to 25° (it controls the refinement in curved areas).
11. Enter the bounding box for the region to be meshed , here it is the whole image (this is where you need the image size taken from ITKsnap).

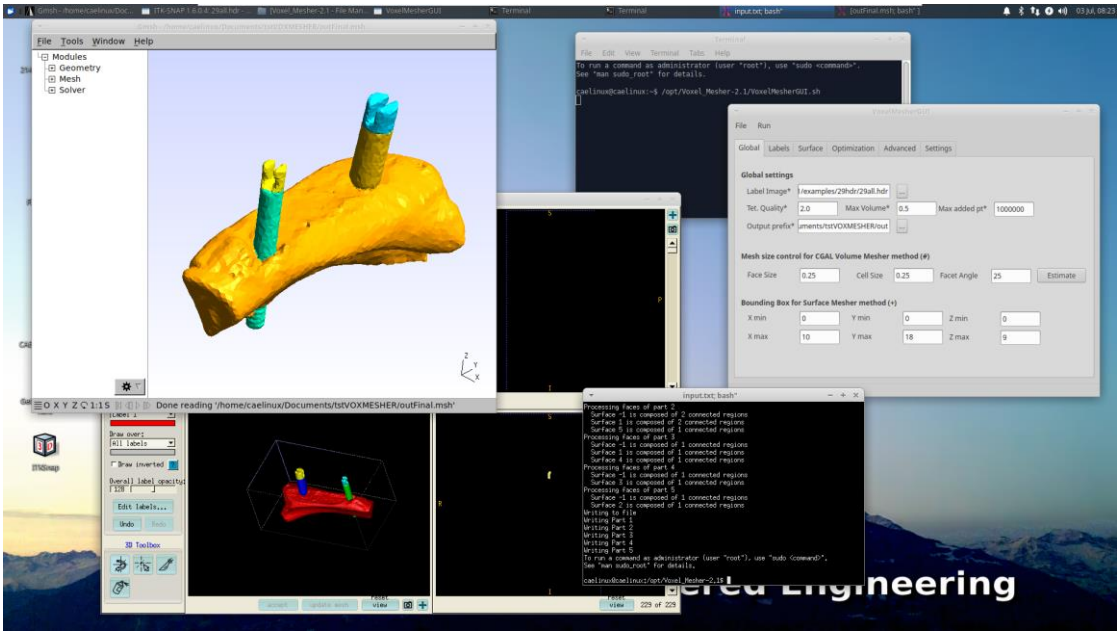

Fig 2. Mesh generation phase and visualization in GMSH

12. In the “Labels” tab, you can set volume, cell and face size limits for each region (or color) of the segmentation image, for example for local mesh refinement. But the most important aspect is that you can “Exclude” some labels such as label 0 (transparent / empty cells have a color / label set = 0). Just enter 1 in the Exclude column for label 0, leave the rest by default.
13. In the other tabs, you can let all the values to default. In optimization, you can set time limits for the different optimizers (for example 10 to 60 s) if computation of the mesh seem to take forever.
14. Finally save the input file, menu File -> Save settings. For example /home/caelinux/Documents/inputVoxelMesher.txt
15. You can open the input file in a text editor and read Voxel Mesher doc in /opt/Voxel_Mesher-2.1 if you want to learn more.
16. run VoxelMesher: menu Run -> Run VoxelMesher
17. watch the terminal to see the progress. The output mesh will be created in different formats: linear tetrahedral mesh in Nastran BDF, Gmsh MSH, Comsol MPHTXT, Tetgen .node,.face,.ele and quadratic mesh in Abaqus INP (two versions: one single part or a model with each region as parts of an assembly).
18. To view the final result: Run -> Visualize in GMSH.
19. Then you can for example convert the mesh to 2nd order in GMSH and export it in other formats, for example UNV to be used in Salome for example. In GMSH: File -> Export, format UNV, save option: save all = on, save node groups = on. And import the mesh in Salome MESH module (File -> Import, Universal File format UNV)

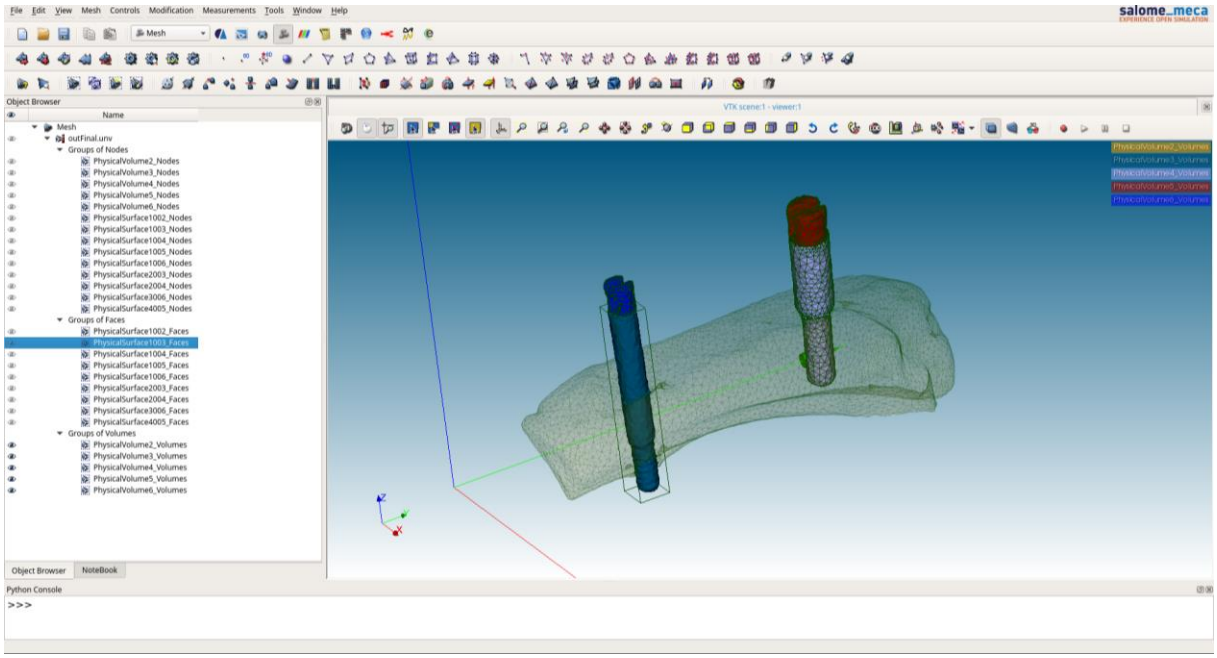

Fig 3: End result in Salome, including groups for volumes for each region, group of faces for each interfaces between the volumes, and group of nodes for each interface too.