

VascularModelDeformation の使い方



概要

- VascularModelDeformationは、チューブ形状の流体解析モデル(テトラ・プリズム複合メッシュ)を変形するために開発したソフトウェアです。
- もともとの解析形状(*.stl)と、その中心線点群 (*.txt) ,
変形のための目標中心線点群(*.txt)の3つを入力として使います。
- 変形前後の中心線から移動量・変形量を計算し、それによって
メッシュ全体を移動・変形させ、新たな解析モデルを得ることができます。

環境構築

- Windows 10 or 11 (開発者は11を使用)
- Python3.11 インストールしてパスを通してください
- Visual Studio2022

requirements.txt のある階層で、venv を用いて仮想環境を作成してください

https://qiita.com/shun_sakamoto/items/7944d0ac4d30edf91fde

`pip install -r requirements.txt`

でGmshをはじめ必要なライブラリが使用できるようになります。

- LocalPath.txt のパスを適切に書き直して下さい。

1行目 : python3.11のパス

2行目 : テトラメッシュを切りなおすスクリプト のパ:

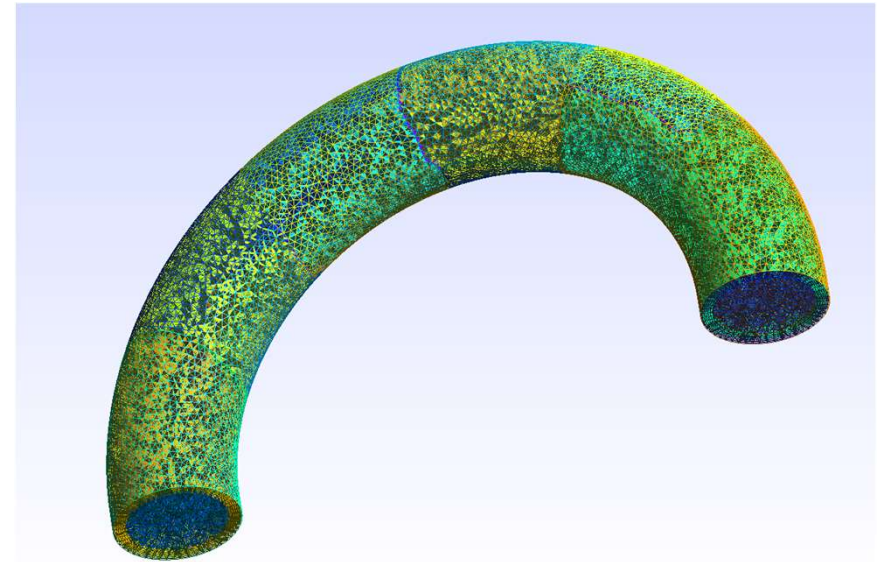
```
C:\git\VMD\venv\Scripts\python
C:\git\VMD\assets\makemesh_inner.py
```

操作手順

- まず適当にフォルダを作成し、「makemesh.py」と、
入力データとなる「WALL.stl」、「centerline.txt」、「centerlineFinal.txt」
の4つのファイルを入れてください(WALL.stlはこの名前である必要があります)。
- makemesh.py内のパラメータ(メッシュサイズや、プリズム層の層数など)を
必要に応じて変更してください

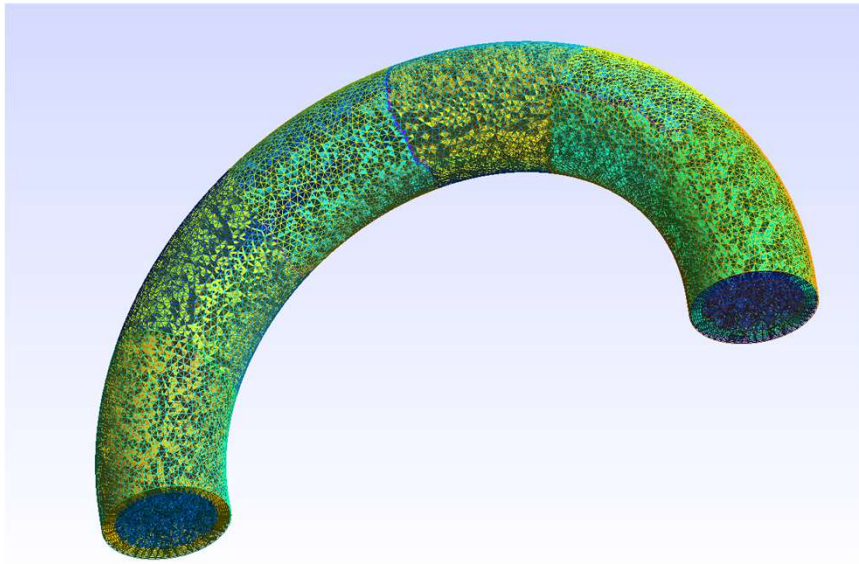
STEP1. makemesh.py を実行

- 基準となる表面形状(WALL.stl)データを入力とし、Gmshにより基準解析モデル MeshOriginal.mshを出力する。
(WALL.stlはASCII形式とバイナリ形式のどちらでも可)



STEP2. Button2 (以降, c# コードの方)

- Gmshで作成したMeshOriginal.mshを入力とし、表面形状(gmsh22.stl)を出力



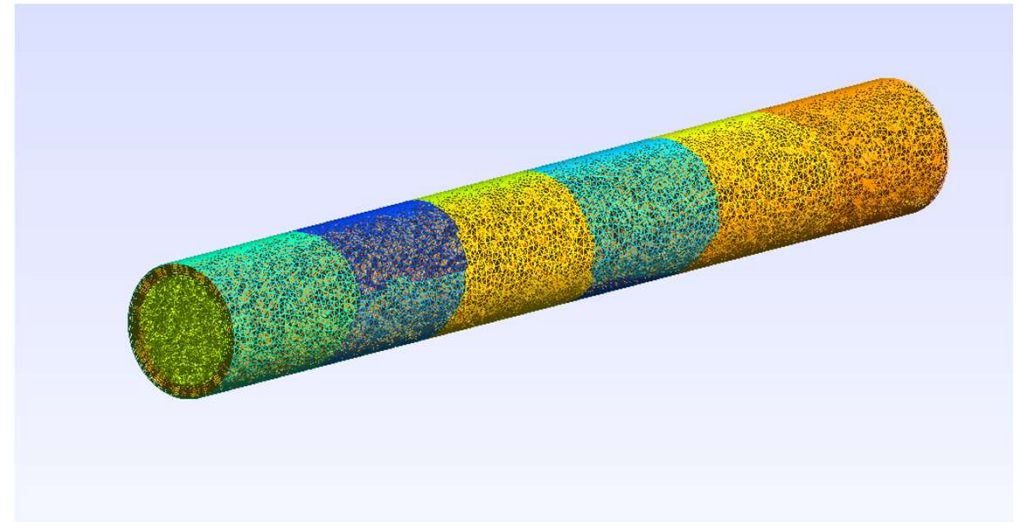
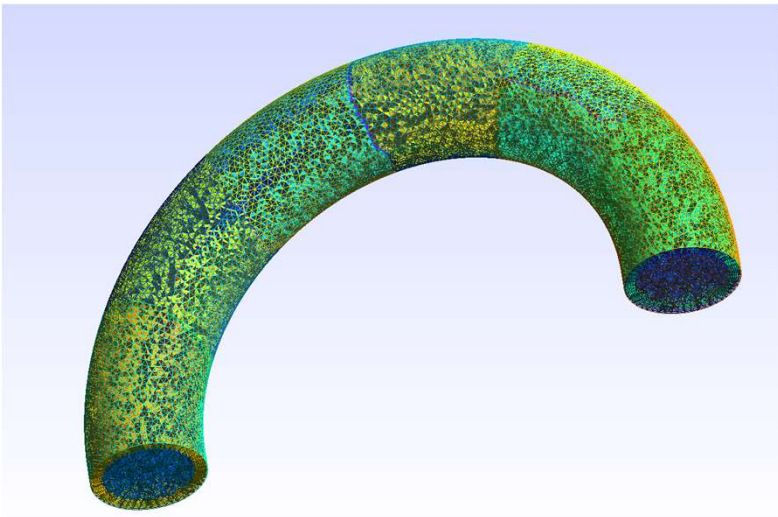
(※ここで得られるSTLデータは、もともとのWALL.stlとは三角形パッチの粗さが異なります。もともとのWALL.stlは、形状情報のみが重要で、そこからどのようにメッシュを作成するかはmakemesh.py内のパラメータによって決まります。そして makemesh.pyで作成されたメッシュモデルの、表面メッシュのみをそのまま抽出したのがgmsh22.stlです。)

STEP3 Button3

- 2回入力を求められるので、
基準中心線.txt → gmsh22.stl の順に入力。
- 中心線Node と、表面上の三角形パッチの対応関係を、
「test.ply」として出力

STEP4 Button4

- 入力を4回求められるので、
基準中心線 → 目標中心線 → test.ply → meshOriginal.msh
(もし「例外がスローされました」とでたら、実行を継続します という欄の「続行」をクリックしてください)
- 再び入力を求められるので、MeshInner.msh を選択。
- MeshMerge.msh (変形による解析モデル) が出力される



STEP5. 結果を確認する

- Gmsh をインストールし、パスを通していれば、ターミナルから「gmsh」と打つことでGmsh のGUI画面が立ち上がるので、「file」→「open」から生成したファイル(MeshMerged.msh)を選択し、変形後の形状やメッシュ精度を確認してください。
- あるいは、vtk ファイル(MeshMerged.vtk) も出力されているので、paraviewで可視化して下さい。

