サマースクール2022

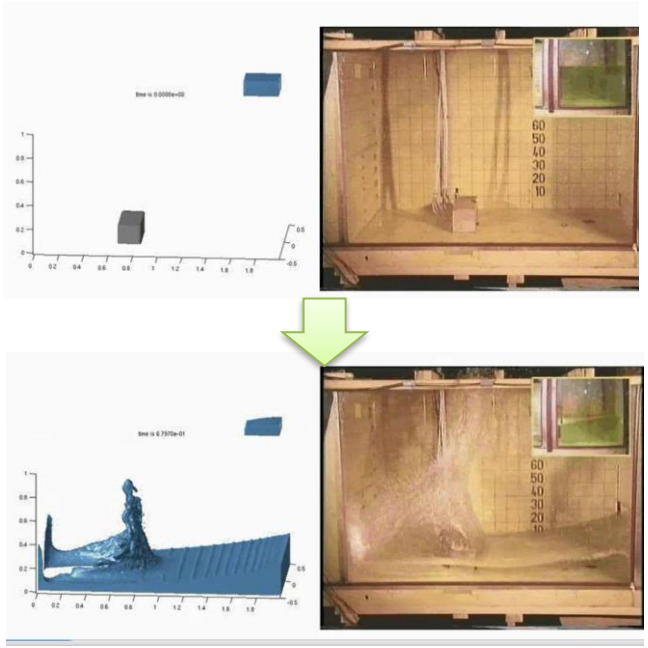
Track3

2022年8月27日(土)

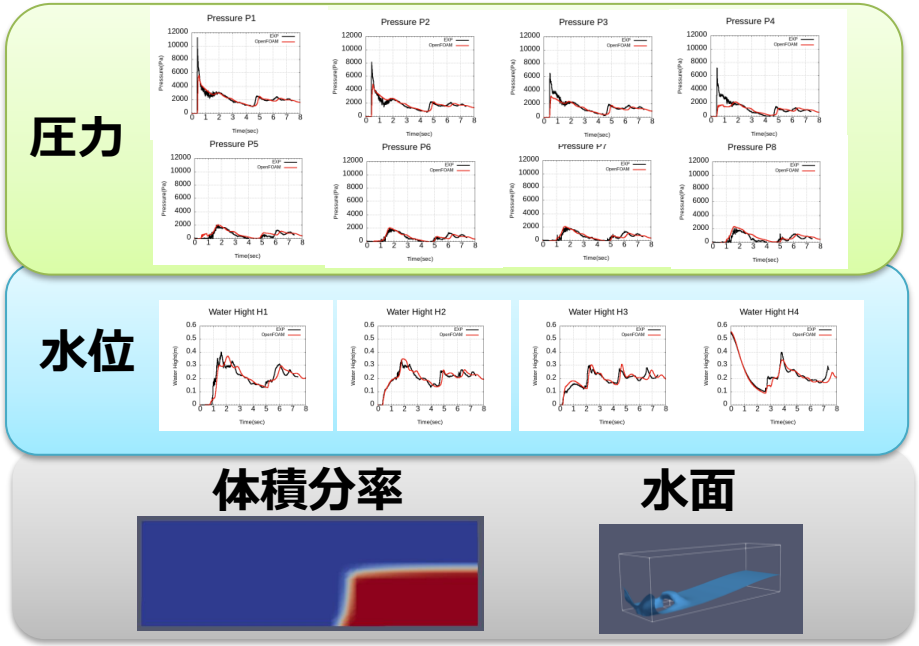
概要

- OpenFOAMによる混相流(気液二相流)の流体解析について学ぶ。
  解析対象は実験データとの結果比較のため「**3次元ダムブレイク**」を扱う。
- 結果処理の方法を学ぶ。

3次元ダムブレイク（実験）



OpenFOAMと実験データとの比較



環境

- DEXCS2021
- OpenFOAM v2106(ESI版)
- ParaView 5.7.0(pvpython python3.8.10)
- Gnuplot 5.2

$コマンド

操作方法の説明

目次

1. V＆Vについて

2. 3次元ダムブレイクの実験データ

3. OpenFOAMによる3次元ダムブレイクの計算実行(interFoam)

4. postProcessユーティリティを使用したデータ処理

5. OpenFOAMによる3次元ダムブレイクの妥当性検証

6. ParaViewによる可視化

7. 実習 モデルの修正

1. V＆Vについて

V&Vについて

工学シミュレーションの品質保証とV&V

V＆V（Verification & Validation)とは

検証(Verification)

要求事項を満たす解析計画のとおりに解析が実施されていること及び適切な解析が実施されていることを確認する活動。検証は、解析が論理的及び数学的に正しく行なわれたことを確認するプロセスである。

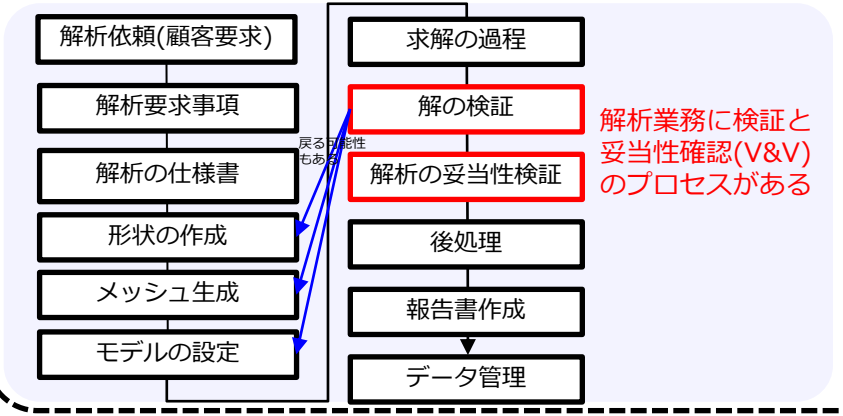
妥当性確認(Validation)

入力データ及び数値計算に関わる一連の検証過程を確認した上で、問題の理想化及び解析結果が、顧客の用途に対して許容される範囲で、工学的に妥当であることを確認する活動。解析結果は何らかの独立した手法を用いて妥当性を確認されなくてはならない。

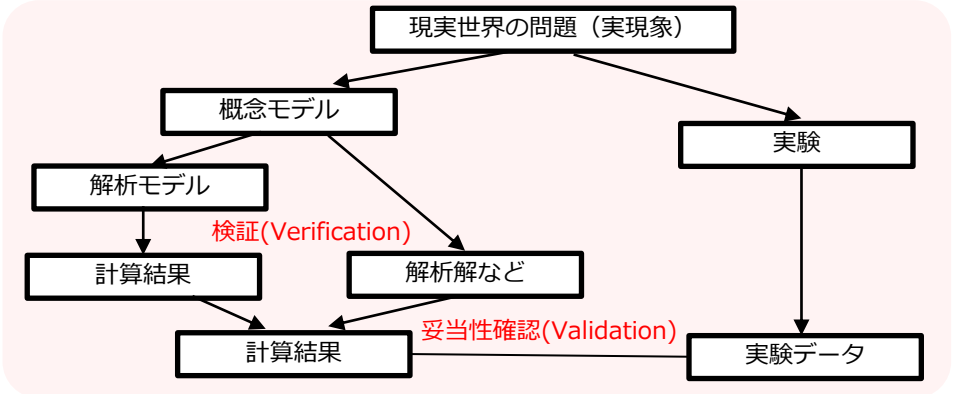
品質マネジメントV&V → モデリング&シミュレーションにおけるV&V

作業にミスが無く完ぺきに仕事すればOKか？

品質V&Vを取り入れた解析業務



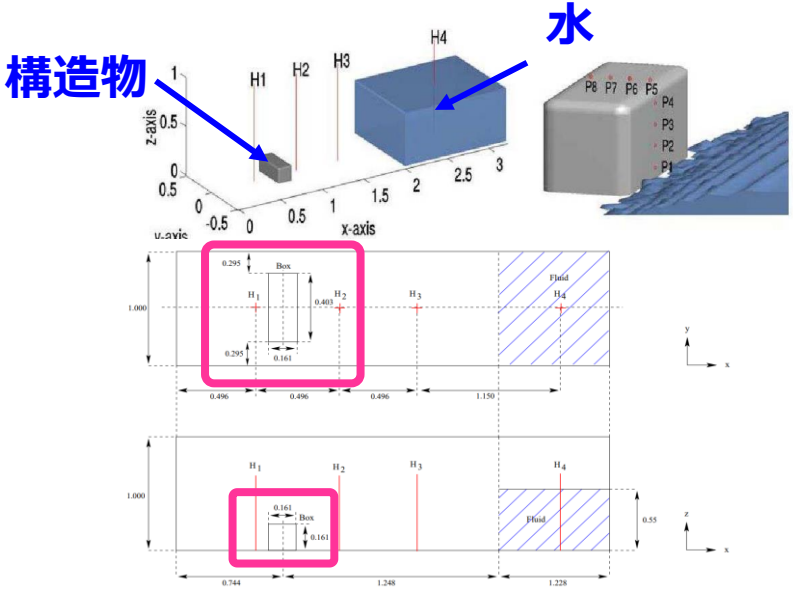
現実世界の問題の本質部分をモデル化→妥当な結果が得られるかどうか

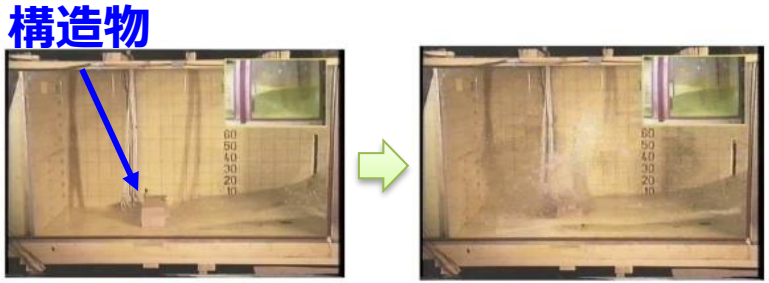


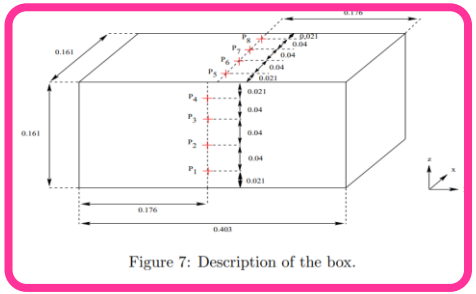
2. 3次元ダムブレイクの実験データ

3次元ダムブレイクの実験

- 高さ0.55mの貯水部がありゲートを急開すると段波を発生
- 貯水部の左側には構造物が固定
  - P1~P8：圧力測定位置（圧力計）
  - H1~H4：水面高さ測定位置（水位計）







By R. Issa and D. Violeau. 3D schematic dam break and evolution of the free surface.
URL：https://www.spheric-sph.org/tests/test-02

実験(3次元ダムブレイク)データの入手

[こちらの実験データ](https://www.spheric-sph.org/tests/test-02)
https://www.spheric-sph.org/tests/test-02

Test 02

3-D schematic dam break and evolution of the free surface

(R. Issa & D. Violeau)

(1)Zipファイルの保存

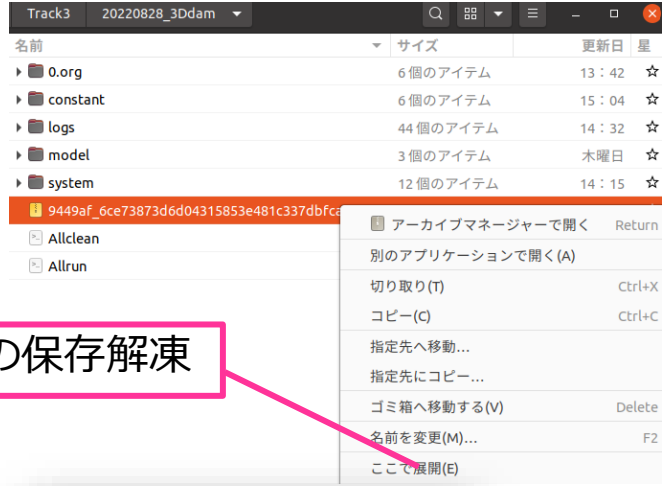
Download the test case details here (SPHERIC_Test2.zip - 2.5MB).

Archive includes:

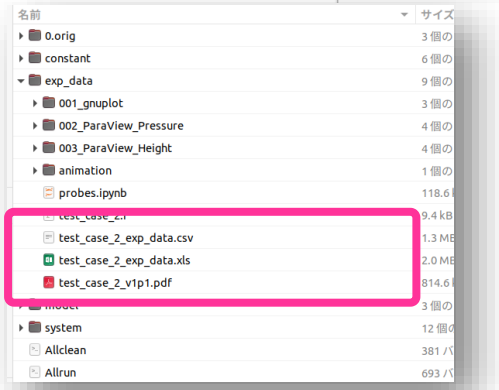
1. Test Description (test_case_2_v1p1.pdf)
2. Geometry preparation file (test_case_2.f)
3. Experimental data (test_case_2_exp_data.xls)

View animation here (Marine Institute of Netherlands) (1.6MB)

(3)フォルダ名変更
9449af_6ce73873d6d04315853e481c337dbfca
↓
exp_data



(2)Zipファイルの保存解凍



実験(3次元ダムブレイク)データの入手

exp_data

test_case_2.f	9.4 kB
test_case_2_exp_data.xls	5.0 MB
test_case_2_vip1.pdf	814.6 kB

exp_data/test_case_2_exp_data.xls

圧力(p1~p8)

水面高さ(H1~H4)

	A	B	C	D	E	F	G	H	I	J	K	L	M	N
1	Time (s)	P1 (Pa)	P2 (Pa)	P3 (Pa)	P4 (Pa)	P5 (Pa)	P6 (Pa)	P7 (Pa)	P8 (Pa)		H1 (m)	H2 (m)	H3 (m)	H4 (m)
2	2.193E-05	-45.9333333	-29.4	-69.8666667	-54.6666667	-55.6	-45.9333333	-41.8	-51.9333333		0.000266667	0.000533333	0.0004	0.546666667
3	0.001028906	-29.4	-11.4666667	-52.8666667	-35.4	-28.4666667	-18.4	-14.2666667	-27.1333333		0.0002	0.000533333	0.0004	0.547666667
4	0.002010061	-8.73333333	11	-31.7333333	-10.1333333	3.666666667	12.4	15.13333333	2.266666667		0.0002	0.000533333	0.0004	0.548533333
5	0.003017037	11.93333333	32.13333333	-13.8	12.4	33.53333333	42.26666667	44.13333333	30.8		0.0002	0.000533333	0.0004	0.549266667
6	0.004024013	23.4	45.46666667	-5.06666667	26.66666667	57.86666667	66.6	67.53333333	55.6		0.0002	0.000533333	0.0004	0.549933333
7	0.005030988	24.8	49.13333333	-2.26666667	33.06666667	73.06666667	81.33333333	81.33333333	72.6		0.0002	0.000533333	0.0004	0.550466667
8	0.006012144	20.66666667	45.46666667	-3.66666667	32.6	78.13333333	86.4	86.4	81.33333333		0.0002	0.000533333	0.0004	0.550933333
9	0.00701912	13.33333333	38.13333333	-9.66666667	28.93333333	74.86666667	84.06666667	83.6	81.33333333		0.0002	0.000533333	0.0004	0.551133333
10	0.008026096	0.933333333	26.2	-20.6666667	17.93333333	60.2	69.4	68.46666667	68		0.0002	0.000533333	0.0004	0.551133333
11	0.009033071	-13.8	11	-33.5333333	1.4	33.53333333	43.66666667	41.8	44.13333333		0.0002	0.000533333	0.000333333	0.551
12	0.010014227	-33.0666667	-9.66666667	-51.9333333	-19.2666667	0	11	8.733333333	12.4		0.0002	0.000533333	0.0004	0.550666667
13	0.011021203	-55.6	-34.4666667	-74.8666667	-44.1333333	-35.8666667	-22.5333333	-24.3333333	-18.4		0.0002	0.000533333	0.0004	0.5502
14	0.012028178	-66.1333333	-46.8666667	-85.9333333	-58.3333333	-59.2666667	-46.8666667	-48.2666667	-41.8		0.0002	0.000466667	0.0004	0.5494
15	0.013009334	-62.9333333	-45.4666667	-82.2666667	-60.6666667	-71.6666667	-58.3333333	-60.6666667	-56.5333333		0.0002	0.000466667	0.0004	0.548466667

※入手したデータのH1とH3が逆であったため修正した。

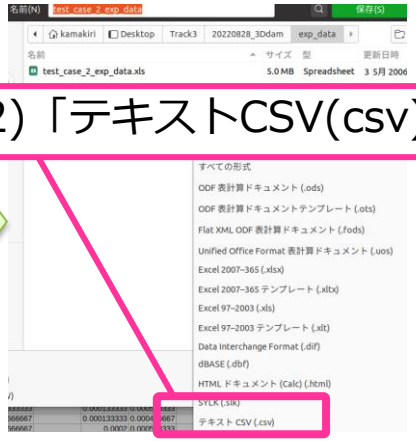
csvファイルに変換

※後ほどgnuplotで読み込むためcsvファイルに変換

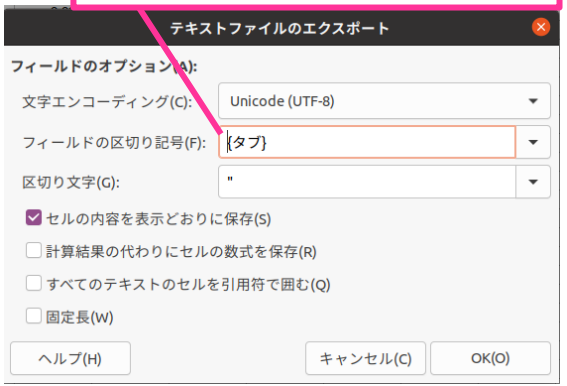
exp_data/test_case_2_exp_data.xls

(1)「ファイル」＞「名前を付けて保存」

(2)「テキストCSV(csv)」を選択

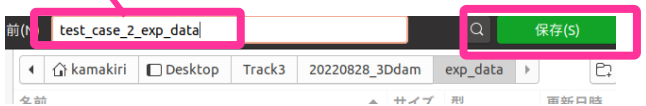


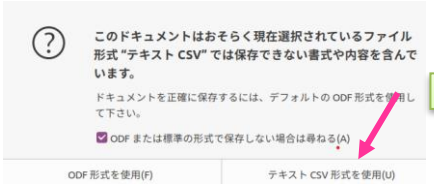
(3){タブ}を選択
※タブ区切り

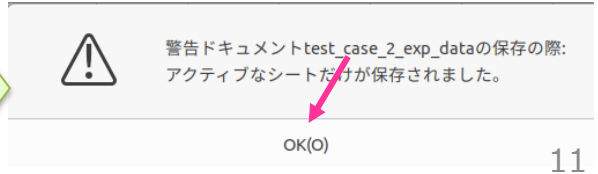


※gnuplotのデフォルトがタブ区切り
カンマ区切りの場合はgnuplotを起動後「 set datafile separator “,” 」を実行

(4)「test_case_2_exp_data」
という名前で保存







テキストエディターで開くと・・・

exp_data/test_case_2_exp_data.csv

Time (s)	P1 (Pa)	P2 (Pa)	P3 (Pa)	P4 (Pa)	P5 (Pa)	P6 (Pa)	P7 (Pa)	P8 (Pa)	H1 (m)	H2 (m)	H3 (m)	H4 (m)
2.19299528509964E-05	-45.9333333333333	-29.4	-69.8666666666667	-54.6666666666667	-55.6	-45.9333333333333	-41.8	-51.9333333333333	0.0004	0		
0.001028905622865	-29.4	-11.4666666666667	-52.8666666666667	-35.4	-28.4666666666667	-18.4	-14.2666666666667	-27.1333333333333	0.0004	0		
0.002010061403904	-8.73333333333333	11	-31.7333333333333	-10.1333333333333	3.66666666666667	12.4	15.1333333333333	2.26666666666667				
0.003017037073918	11.9333333333333	32.1333333333333	-13.8	12.4	33.5333333333333	42.2666666666667	44.1333333333333	30.8	0.0004	0		
0.004024012743932	23.4	45.4666666666667	-5.06666666666667	26.6666666666667	57.8666666666667	66.6	67.5333333333333	55.6	0.0004	0		
0.005030988413946	24.8	49.1333333333333	-2.26666666666667	33.0666666666667	73.0666666666667	81.3333333333333	81.3333333333333	72.6				
0.006012144194985	20.6666666666667	45.4666666666667	-3.66666666666667	32.6	78.1333333333333	86.4	86.4	81.3333333333333	0.0004	0		
0.007019119864998	13.3333333333333	38.1333333333333	-9.66666666666667	28.9333333333333	74.8666666666667	84.0666666666667	83.6	81.33333				
0.008026095535013	0.933333333333333	26.2	-20.6666666666667	17.9333333333333	60.2	69.4	68.4666666666667	68	0.0004	0.00053333333333		
0.009033071205026	-13.8	11	-33.5333333333333	1.4	33.5333333333333	43.6666666666667	41.8	44.1333333333333	0.000333333333333	0		
0.010014226986066	-33.0666666666667	-9.66666666666667	-51.9333333333333	-19.2666666666667	0	11	8.73333333333333	12.4	0.0004	0		
0.01102120265608	-55.6	-34.4666666666667	-74.8666666666667	-44.1333333333333	-35.8666666666667	-22.5333333333333	-24.3333333333333	-18.4				
0.012028178326094	-66.1333333333333	-46.8666666666667	-85.9333333333333	-58.3333333333333	-59.2666666666667	-46.8666666666667	-48.266666666666					
0.013009334107133	-62.9333333333333	-45.4666666666667	-82.2666666666667	-60.6666666666667	-71.6666666666667	-58.3333333333333	-60.666666666666					
0.014016309777147	-57.8666666666667	-41.8	-77.6666666666667	-59.2666666666667	-75.3333333333333	-61.6	-60.6666666666667	-60.2	0.0004	0		
0.015023285447161	-53.2666666666667	-38.6	-74	-57.8666666666667	-69.4	-55.6	-53.2666666666667	-56.5333333333333	0.0004	0.00046666666666		
0.016030261117175	-50.5333333333333	-36.7333333333333	-71.2	-56.5333333333333	-60.2	-46.8666666666667	-42.2666666666667	-50.0666666666667				
0.017011416898215	-45.4666666666667	-29.4	-65.2666666666667	-51.4666666666667	-46.8666666666667	-34.9333333333333	-29.4	-39.5333333333333				
0.018018392568228	-36.7333333333333	-18.4	-57.8666666666667	-41.3333333333333	-29.4	-21.1333333333333	-15.1333333333333	-28.4666666666667				
0.019025368238242	-30.8	-9.66666666666667	-52.8666666666667	-31.7333333333333	-11	-4.6	0	-13.8	0.0004	0.000466666666667	0.000133	
0.020032343908255	-24.8	-2.73333333333333	-48.2666666666667	-22.5333333333333	6	13.8	16.5333333333333	2.73333333333333	0.0004	0		
0.021013499689295	-21.1333333333333	2.26666666666667	-46.4	-16.0666666666667	19.7333333333333	29.4	31.7333333333333	19.7333333333333				
0.022020475359309	-16.0666666666667	6.4	-43.2	-8.73333333333333	34.4666666666667	45.4666666666667	47.3333333333333	39.0666666666667				
0.023027451029323	-5.06666666666667	15.1333333333333	-33.5333333333333	3.66666666666667	46.8666666666667	58.3333333333333	59.2666666666667					
0.024034426699338	4.6	24.3333333333333	-23.4	14.2666666666667	52.8666666666667	62	62	59.2666666666667	0.0004	0.00046666666666		
0.025015582480376	8.73333333333333	28.4666666666667	-18.4	18.8666666666667	51.9333333333333	59.7333333333333	59.7333333333333	60.2				
0.02602255815039	8.26666666666667	28.4666666666667	-18.4	17.9333333333333	45.9333333333333	54.6666666666667	54.2	55.6	0.0004	0		
0.027029533820404	1.4	21.1333333333333	-22.0666666666667	12.4	33.5333333333333	43.2	42.2666666666667	45.4666666666667	0.0004	0		
0.028010689601444	-7.33333333333333	11.4666666666667	-28.4666666666667	3.66666666666667	16.0666666666667	24.8	24.3333333333333	27.133333				
0.029017665271457	-19.7333333333333	0.933333333333333	-39.0666666666667	-8.73333333333333	-5.06666666666667	2.26666666666667	1.4	2.733333				
0.030024640941472	-33.0666666666667	-11.4666666666667	-52.8666666666667	-23.8666666666667	-25.7333333333333	-20.6666666666667	-21.6	-22.0666				
0.031031616611485	-48.2666666666667	-27.1333333333333	-68.9333333333333	-41.3333333333333	-44.6	-39.5333333333333	-40.4	-41.8	0.0004	0		
0.032012772392524	-62.4666666666667	-43.2	-83.1333333333333	-57.8666666666667	-57.8666666666667	-51.9333333333333	-51	-55.1333333333333				

タブ区切りになっている

３．OpenFOAMによる3次元ダムブレイクの計算実行(interFoam)

interFoamにおける支配方程式

連続の式

$$\frac{\partial \rho}{\partial t} + \nabla \cdot (\rho \boldsymbol{u}) = 0$$

運動方程式

$$\frac{\partial \rho \boldsymbol{u}}{\partial t} + \nabla \cdot (\rho \boldsymbol{u}\boldsymbol{u}) = -\nabla p + \mu \nabla^2 \boldsymbol{u} + \rho \boldsymbol{g} + \boldsymbol{F_\sigma}$$

表面張力

VOF関数の保存方程式

$$\frac{\partial \alpha}{\partial t} + \nabla \cdot (\alpha \boldsymbol{u}) + \nabla \cdot (\alpha(1-\alpha)\boldsymbol{u_r}) = 0$$

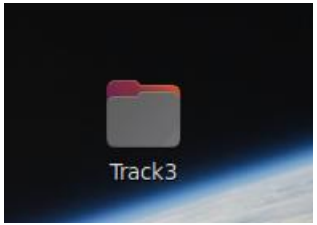
界面圧縮項としてモデル化

$\boldsymbol{u}$：速度ベクトル
t：時間
p：圧力
α：体積分率(*VOF*関数)
$\rho = \rho_1 \alpha + \rho_2 (1-\alpha)$：密度
$\mu = \mu_1 \alpha + \mu_2 (1-\alpha)$：粘度
$\boldsymbol{g}$：重力
F_σ：表面張力
$\boldsymbol{u_r}$：相対速度ベクトル

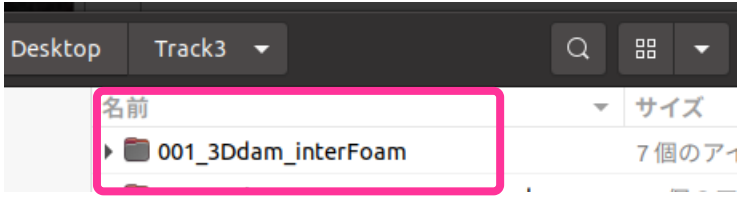
※乱流モデルなし

OpenFOAMの計算ファイル

デスクトップに「Track3」という名前で保存



「Track3」フォルダ内に「001_3Ddam_interFoam」を使用



$FOAM_TUTORIALS/multiphase/interFoam/laminar/damBreak/damBreak のチュートリアルを編集して作成した

環境変数　/usr/lib/openfoam/openfoam2106/tutorials/

OpenFOAM Terminalの起動

(1)「of-v2106 Terminal」を起動

OpenFOAM Terminalが起動する

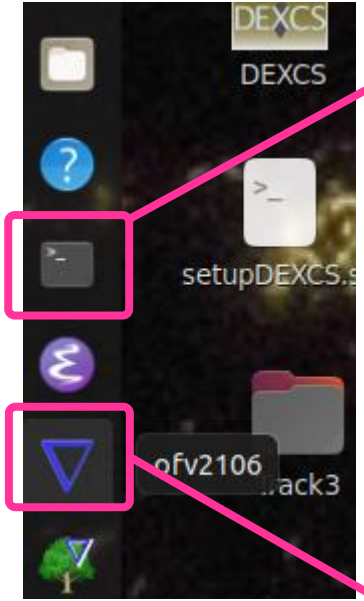
(2)フォルダの移動

cd /home/**ユーザー名**/Desktop/Track3/

```
OpenFOAMv2106 is enabled.
kamakiri@kamakiri-VirtualBox ~ $ cd '/home/kamakiri/Desktop/Track3'
```

各時設定したユーザー名

※補足



「 OS標準のターミナル」からは直接OpenFOAMは使えない

```
kamakiri@kamakiri-VirtualBox:~$ simpleFoam -help

コマンド 'simpleFoam' が見つかりません。次の方法でインストールできます:

sudo apt install openfoam
```

以下のコマンドを端末で実行

$source ~/.bashrc-OF4v2106

```
kamakiri@kamakiri-VirtualBox:~$ source ~/.bashrc-OF4v2106
OpenFOAMv2106 is enabled.
kamakiri@kamakiri-VirtualBox:~$ simpleFoam -help

Usage: simpleFoam [OPTIONS]
Options:
  -case <dir>          Specify case directory to use (instead of cwd)
  -decomposeParDict <file>
                       Use specified file for decomposePar dictionary
  -dry-run             Check case set-up only using a single time step
  -dry-run-write       Check case set-up and write only using a single time step
  -parallel            Run in parallel
  -postProcess         Execute functionObjects only
  -doc                 Display documentation in browser
  -help                Display short help and exit
  -help-full           Display full help and exit

Steady-state solver for incompressible, turbulent flows.

Using: OpenFOAM-2106 (2106) - visit www.openfoam.com
Build: _40a7d5d6-20210727 (patch=211215)
Arch:  LSB;label=32;scalar=64
```

※「~」は「 /home/user名」
DEXCSのバージョンによって上記のコマンドは変わる

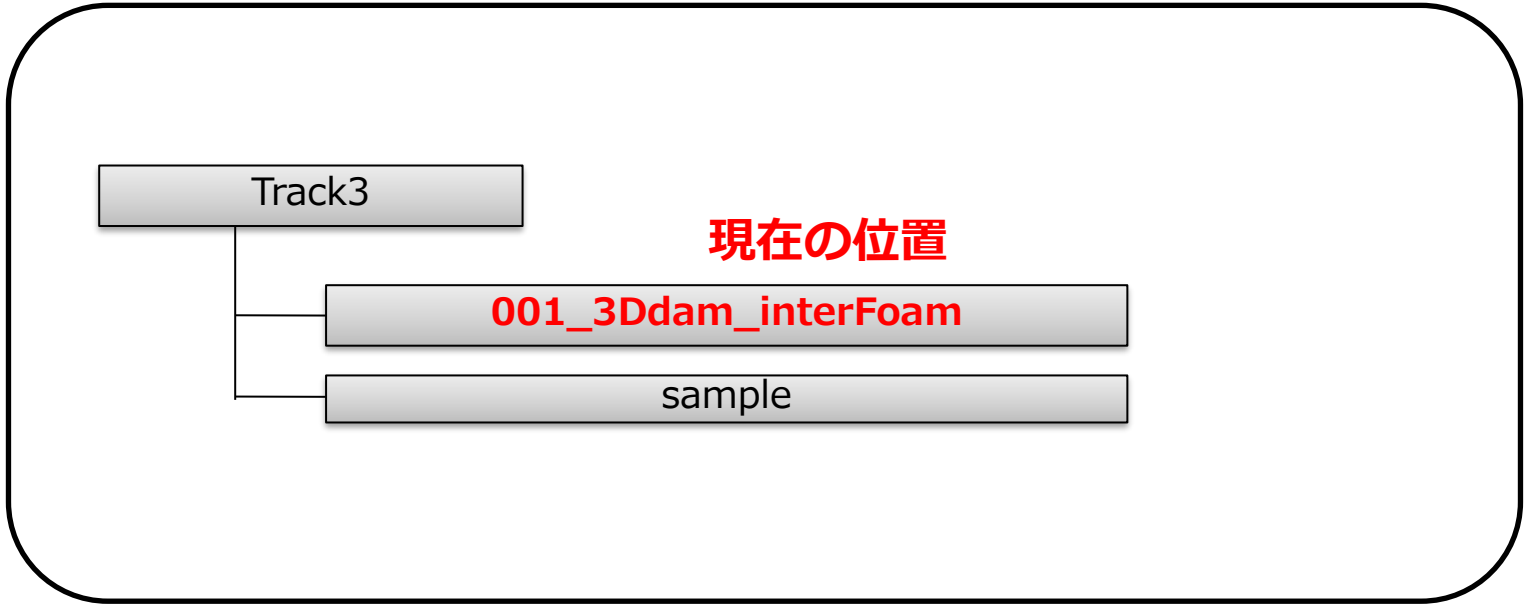
OpenFOAM ターミナルはOpenFOAMが使える。

```
OpenFOAMv2106 is enabled.
kamakiri@kamakiri-VirtualBox ~ $ simpleFoam -help

Usage: simpleFoam [OPTIONS]
Options:
  -case <dir>          Specify case directory to use (instead of cwd)
  -decomposeParDict <file>
                       Use specified file for decomposePar dictionary
  -dry-run             Check case set-up only using a single time step
  -dry-run-write       Check case set-up and write only using a single time step
  -parallel            Run in parallel
  -postProcess         Execute functionObjects only
  -doc                 Display documentation in browser
  -help                Display short help and exit
  -help-full           Display full help and exit

Steady-state solver for incompressible, turbulent flows.

Using: OpenFOAM-2106 (2106) - visit www.openfoam.com
Build: _40a7d5d6-20210727 (patch=211215)
Arch:  LSB;label=32;scalar=64

kamakiri@kamakiri-VirtualBox ~ $
```

以下のコマンドで「**001_3Ddam_interFoam**」フォルダに移動

フォルダを移動

```
$cd 001_3Ddam_interFoam
```



解析手順

形状作成

FreeCAD

メッシュ作成

```
blockMesh                    # ベースメッシュ作成
surfaceFeatureExtract        # 特徴線の抽出
snappyHexMesh -overwrite     # メッシュ生成
```

解析条件の設定

流速、圧力、体積分率の初期値の設定
スキーム、解析制御の設定

計算実行

OpenFOAM

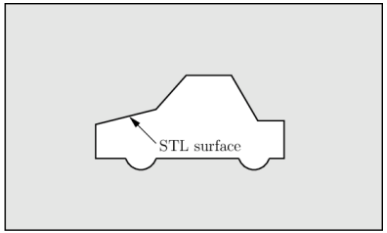
ポスト処理

gnuplot

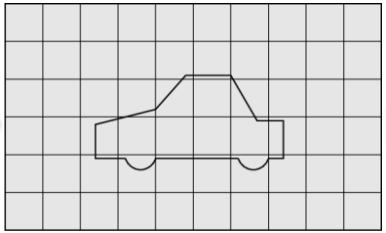
ParaView

メッシュ作成までの流れ

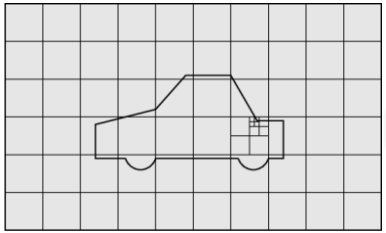
snappyHexMesh



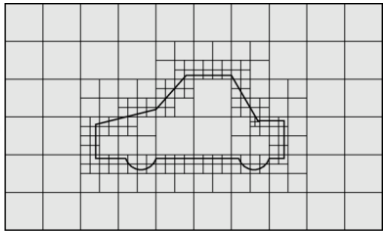
stl表面形状を作成し
「constant/triSurface」
に保存（**FreeCAD**）



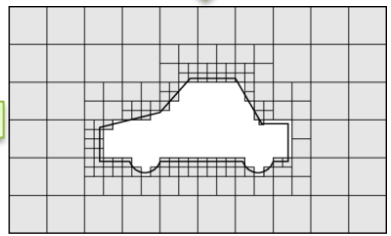
基準セルの生成
(**blockMesh**)



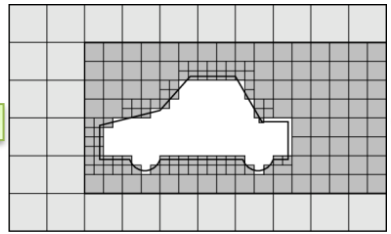
特徴線・面（輪郭）で分割
輪郭データは
surfaceFeatureExtractを使ってSTL 形状ファイルから生成



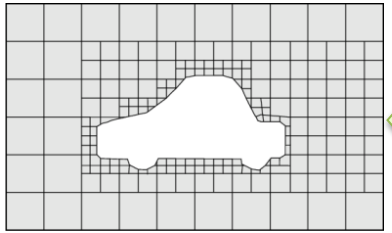
指定された表面における
分割のためにセルが選択



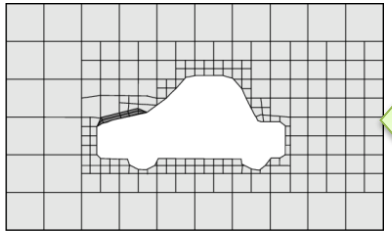
不要セルの削除



特定領域内のセルの分割



メッシュを平滑化するために
セルの頂点を表面に移動

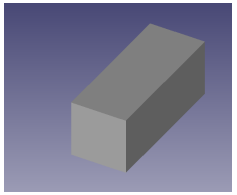


メッシュレイヤ

メッシュ作成

形状(stlファイル)

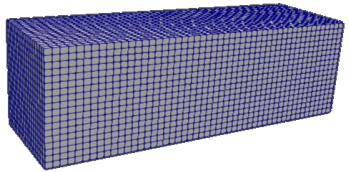
構造物の形状



binaryやascii形式のSTL形式による表面形状データをケースフォルダの「constant/triSurface」に置く。

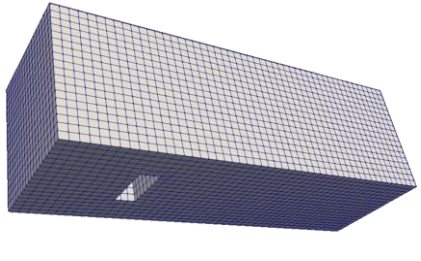
blockMesh

ベースメッシュ作成



「system/blockMeshDict」を使用して解析領域の範囲とメッシュ密度の基準を決めるために六角形の基礎メッシュを作成しておく。

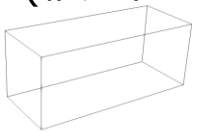
snappyHexMesh



「system/snappyHexMeshDict」に適切な内容を入力する。

surfaceFeatureExtract

特徴線抽出(構造物の輪郭)



「system/surfaceFeatureExtractDict」を使用して特徴線を作成

解析手順

形状作成

メッシュ作成

解析条件の設定

計算実行

ポスト処理

FreeCAD

```
blockMesh                    # ベースメッシュ作成
surfaceFeatureExtract        # 特徴線の抽出
snappyHexMesh -overwrite     # メッシュ生成
```

流速、圧力、体積分率の初期値の設定
スキーム、解析制御の設定

OpenFOAM

gnuplot

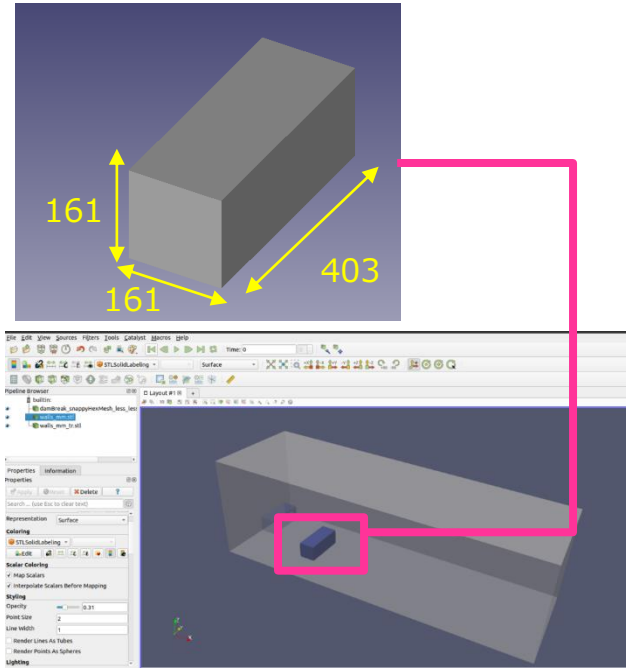
ParaView

形状作成

FreeCADで構造物の形状を作成（作成済み）

作成した形状ファイルは「constant/triSurface」に保存

```
.
├── 0.orig
│   ├── U
│   ├── alpha.water
│   └── p_rgh
├── Allclean
├── Allrun
├── constant
│   ├── g
│   ├── transportProperties
│   ├── triSurface
│   │   ├── walls.stl
│   │   ├── walls_mm.stl
│   │   ├── walls_mm_tr.eMesh
│   │   ├── walls_mm_tr.obj
│   │   └── walls_mm_tr.stl
│   └── turbulenceProperties
├── model
│   ├── model_all.FCStd
│   ├── model_all.FCStd1
│   └── walls.stl
└── system
    ├── blockMeshDict
    ├── controlDict
    ├── createPatchDict
    ├── decomposeParDict
    ├── fvSchemes
    ├── fvSolution
    ├── interfaceHeight
    ├── probes
    ├── residuals
    ├── setFieldsDict
    ├── snappyHexMeshDict
    └── surfaceFeatureExtractDict
```



解析手順

形状作成

メッシュ作成

解析条件の設定

計算実行

ポスト処理

FreeCAD

```
blockMesh                    # ベースメッシュ作成
surfaceFeatureExtract        # 特徴線の抽出
snappyHexMesh -overwrite     # メッシュ生成
```

流速、圧力、体積分率の初期値の設定
スキーム、解析制御の設定

OpenFOAM

gnuplot

ParaView

メッシュ作成(blockMesh)

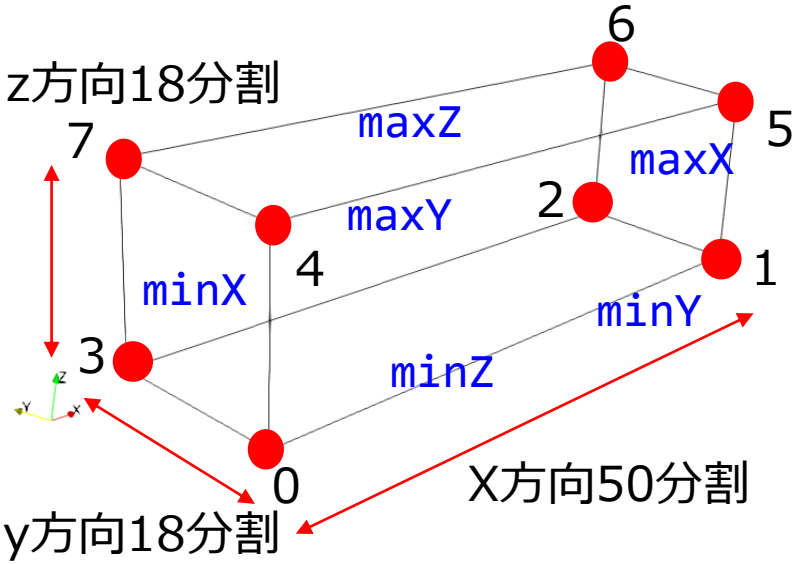
system/blockMeshDict

```
scale   1;

vertices
(
    (0 -0.5 0)      //0
    (3.22 -0.5 0)   //1
    (3.22 0.5 0)    //2
    (0 0.5 0)       //3
    (0 -0.5 1)      //4
    (3.22 -0.5 1)   //5
    (3.22 0.5 1)    //6
    (0 0.5 1)       //7
);

blocks
(
    hex (0 1 2 3 4 5 6 7) (50 18 18) simpleGrading (1 1 1)
);

edges
(
);
```

節点座標

分割数
(X, Y, Z)

分割拡大比率
※今回は均等に分割



※C++言語では2種類のコメント方法が提供

単一行コメント	複数行コメント
//コメントアウト	/* コメントアウト1 コメントアウト2 */

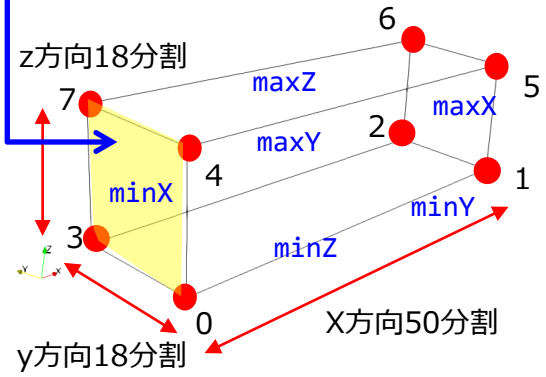
メッシュ作成(blockMesh)

system/blockMeshDict

```
boundary   境界
(
    minX
    {
        type wall;
        faces
        (
            (0 4 7 3)
        );
    }
    maxX
    {
        type wall;
        faces
        (
            (2 6 5 1)
        );
    }
    minY
    {
        type wall;
        faces
        (
            (3 7 6 2)
        );
    }
    maxY
    {
        type wall;
        faces
        (
            (1 5 4 0)
        );
    }
    minZ
    {
        type wall;
        faces
        (
            (0 3 2 1)
        );
    }
    maxZ
    {
        type patch;
        faces
        (
            (4 5 6 7)
        );
    }
);
```

(例)

```
minX   面の名前
  {
      type wall;  面のタイプ
      faces       面の定義
      (
          (0 4 7 3) 面を構成する節点番号
      );
  }
```



面のタイプ

- patch (パッチ)
- wall (壁)
- symmetryPlane (対称面)
- cyclic (周期境界)
- cyclicAMI (不整合周期境界)
- cyclicAMI (不整合周期境界)
- wedge (2 次元軸対称)
- empty (2 次元)

※maxZは大気圧解放境界：「type patch」
その他はすべり無しの壁：「type wall」

メッシュ作成(blockMesh)

以下のコマンドでblockMeshを実行

```
$ blockMesh
```

```
kamakiri@kamakiri-VirtualBox ~/Desktop/runFolder/001_3Ddam_interFoam $ blockMesh
/*---------------------------------------------------------------------------*\
| =========                 |                                                 |
| \\      /  F ield         | OpenFOAM: The Open Source CFD Toolbox           |
|  \\    /   O peration     | Version:  2106                                  |
|   \\  /    A nd           | Website:  www.openfoam.com                      |
|    \\/     M anipulation  |                                                 |
\*---------------------------------------------------------------------------*/
Build  : _40a7d5d6-20210727 OPENFOAM=2106 patch=211215
Arch   : "LSB;label=32;scalar=64"
Exec   : blockMesh
Date   : Aug 17 2022
Time   : 21:27:33
Host   : kamakiri-VirtualBox
PID    : 10195
I/O    : uncollated
Case   : /home/kamakiri/Desktop/runFolder/001_3Ddam_interFoam
nProcs : 1
trapFpe: Floating point exception trapping enabled (FOAM_SIGFPE).
fileModificationChecking : Monitoring run-time modified files using timeStampMaster (fileModific
nSkew 5, maxFileModificationPolls 20)
allowSystemOperations : Allowing user-supplied system call operations

// * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * //
Create time

Creating block mesh from "system/blockMeshDict"
```

```
There are no merge patch pairs

Writing polyMesh with 0 cellZones
----------------
Mesh Information
----------------
  boundingBox: (0 -0.5 0) (3.22 0.5 1)
  nPoints: 18411
  nCells: 16200
  nFaces: 50724
  nInternalFaces: 46476
----------------
Patches
----------------
  patch 0 (start: 46476 size: 324) name: minX
  patch 1 (start: 46800 size: 324) name: maxX
  patch 2 (start: 47124 size: 900) name: minY
  patch 3 (start: 48024 size: 900) name: maxY
  patch 4 (start: 48924 size: 900) name: minZ
  patch 5 (start: 49824 size: 900) name: maxZ

End
```

エラーなく終了

ベースメッシュの作成完了

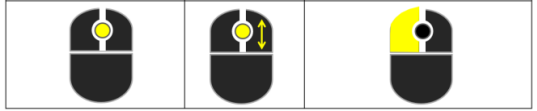
メッシュ作成(blockMesh)

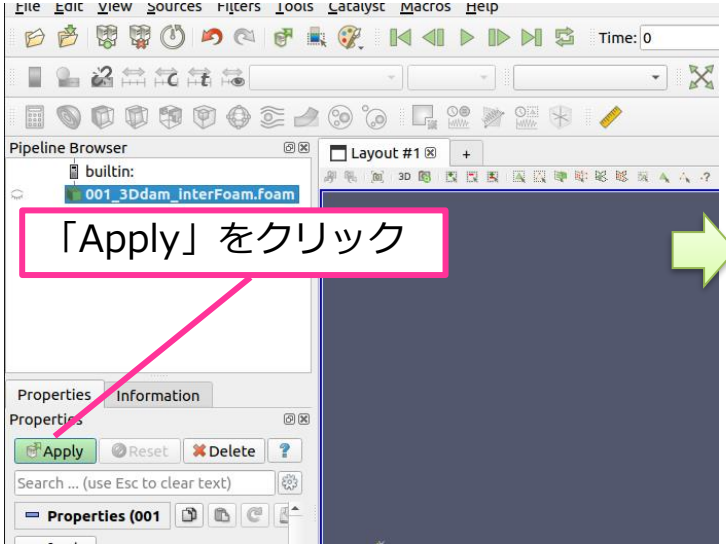
以下のコマンドを実行しParaViewを起動

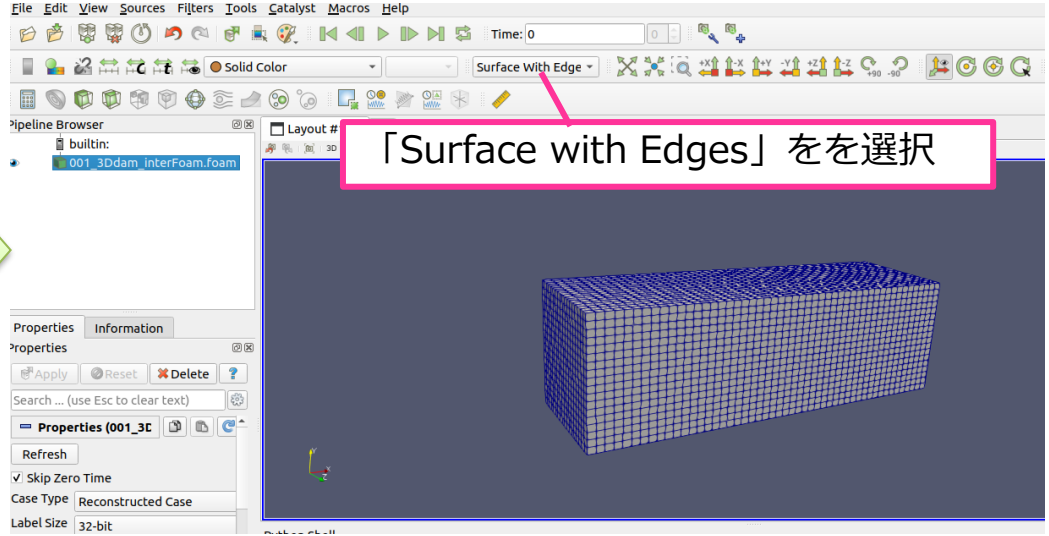
```
$ paraFoam
```

マウス操作

平行移動	拡大・縮小	回転
マウス中ボタン+ドラッグ	マウスホイール	マウス左ボタン+ドラッグ







メッシュ形状に問題がないことを確認

メッシュ作成(surfaceFeatureExtract)

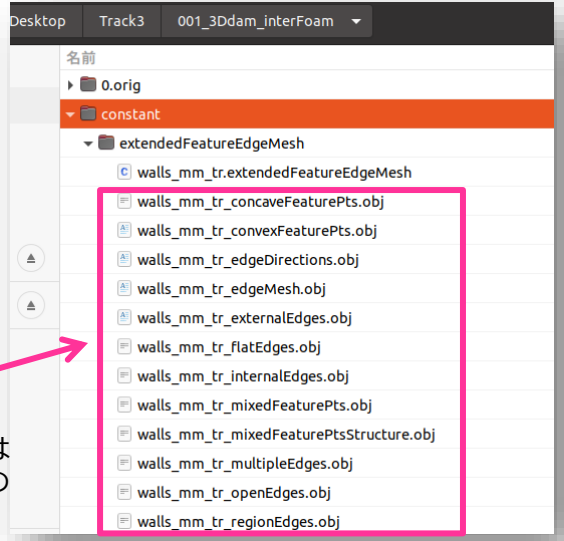
snappyHexMeshにおいて**特徴線 (feature)** を利用
特徴線はメッシュを形状に合わせるために使われるものでsurfaceFeaturesで作成

system/surfaceFeatureExtractDict

constant/triSurface/walls_mm_tr.stl

```
walls_mm_tr.stl        構造物stlファイル
{
    extractionMethod     extractFromSurface; 表面形状ファイルから抽出

    extractFromSurfaceCoeffs
    {
        includedAngle    150;  辺に隣接する2つの三角形の法線
                               がなす角度がincludedAngle以下
    }                          であれば特徴線とみなす

    subsetFeatures
    {
        nonManifoldEdges     no;  連続でないエッジの扱いについて指定。

        openEdges            yes; 開かれたエッジの扱いについて指定。
                                  yesの場合は１つの面にしか属して
    }                             いない線も特徴線とする。

    writeObj                 yes; 特徴線をobj形式で出力するかどう
}                                 かを指定。
```

constant/extendedFeatureEdgeMesh



objファイルは
ParaViewでの
読み込み可能

メッシュ作成(surfaceFeatureExtract)

以下のコマンドでsurfaceFeatureExtractを実行

```
$ surfaceFeatureExtract
```

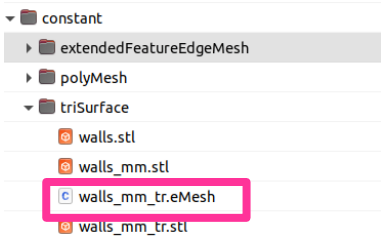
```
kamakiri@kamakiri-VirtualBox ~/Desktop/runFolder/001_3Ddam_interFoam $ surfaceFeatureExtract
/*---------------------------------------------------------------------------*\
| =========                 |                                                 |
| \\      /  F ield         | OpenFOAM: The Open Source CFD Toolbox           |
|  \\    /   O peration     | Version:  2106                                  |
|   \\  /    A nd           | Website:  www.openfoam.com                      |
|    \\/     M anipulation  |                                                 |
\*---------------------------------------------------------------------------*/
Build  : _40a7d5d6-20210727 OPENFOAM=2106 patch=211215
Arch   : "LSB;label=32;scalar=64"
Exec   : surfaceFeatureExtract
Date   : Aug 17 2022
Time   : 21:39:06
Host   : kamakiri-VirtualBox
PID    : 10325
I/O    : uncollated
Case   : /home/kamakiri/Desktop/runFolder/001_3Ddam_interFoam
nProcs : 1
trapFpe: Floating point exception trapping enabled (FOAM_SIGFPE).
fileModificationChecking : Monitoring run-time modified files using timeStampMaster (fileModificationSkew 5, maxFileM
ificationPolls 20)
allowSystemOperations : Allowing user-supplied system call operations

// * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * //
Create time
```

```
Writing extendedEdgeMesh components to "/home/kamakiri/Desktop/runFolder/001_3Ddam_interFoam/constant/extendedFeature
Writing 8 convex feature points to "/home/kamakiri/Desktop/runFolder/001_3Ddam_interFoam/constant/extendedFeatureEdge
urePts.obj"
Writing 0 concave feature points to "/home/kamakiri/Desktop/runFolder/001_3Ddam_interFoam/constant/extendedFeatureEdg
aturePts.obj"
Writing 0 mixed feature points to "/home/kamakiri/Desktop/runFolder/001_3Ddam_interFoam/constant/extendedFeatureEdgeM
ePts.obj"
Writing 0 mixed feature point structure to "/home/kamakiri/Desktop/runFolder/001_3Ddam_interFoam/constant/extendedFea
xedFeaturePtsStructure.obj"
Writing 12 external edges to "/home/kamakiri/Desktop/runFolder/001_3Ddam_interFoam/constant/extendedFeatureEdgeMesh/w
j"
Writing 0 internal edges to "/home/kamakiri/Desktop/runFolder/001_3Ddam_interFoam/constant/extendedFeatureEdgeMesh/wa
"
Writing 0 flat edges to "/home/kamakiri/Desktop/runFolder/001_3Ddam_interFoam/constant/extendedFeatureEdgeMesh/walls_
Writing 0 open edges to "/home/kamakiri/Desktop/runFolder/001_3Ddam_interFoam/constant/extendedFeatureEdgeMesh/walls_
Writing 0 multiple edges to "/home/kamakiri/Desktop/runFolder/001_3Ddam_interFoam/constant/extendedFeatureEdgeMesh/wa
"
Writing 0 region edges to "/home/kamakiri/Desktop/runFolder/001_3Ddam_interFoam/constant/extendedFeatureEdgeMesh/wall
Writing 12 edge directions to "/home/kamakiri/Desktop/runFolder/001_3Ddam_interFoam/constant/extendedFeatureEdgeMesh/
obj"

Writing featureEdgeMesh to "/home/kamakiri/Desktop/runFolder/001_3Ddam_interFoam/constant/triSurface/walls_mm_tr.eMes
ExecutionTime = 0 s  ClockTime = 0 s

End
```

エラーなく終了

特徴線の抽出の完了

.eMeshファイルがconstant/triSurface/以下に作成



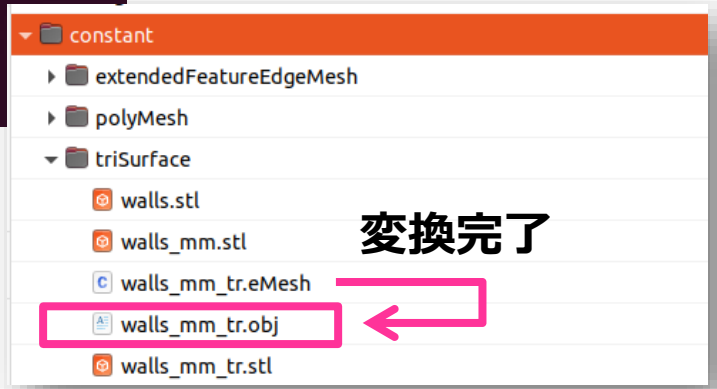
メッシュ作成(surfaceFeatureExtract)

.eMeshファイルはParaViewで開くことができないので以下のコマンドでobjファイルに変換することができる

```
$surfaceFeatureConvert constant/triSurface/walls_mm_tr.eMesh constant/triSurface/walls_mm_tr.obj
```

```
kamakiri@kamakiri-VirtualBox ~/Desktop/Track3/001_3Ddam_interFoam $ surfaceFeatureConvert constant/triSurface/walls_mm_tr.eMesh constant/triSurface/walls_mm_tr.obj
/*---------------------------------------------------------------------------*\
| =========                 |                                                 |
| \\      /  F ield         | OpenFOAM: The Open Source CFD Toolbox           |
|  \\    /   O peration     | Version:  2106                                  |
|   \\  /    A nd           | Website:  www.openfoam.com                      |
|    \\/     M anipulation  |                                                 |
\*---------------------------------------------------------------------------*/
Build  : _40a7d5d6-20210727 OPENFOAM=2106 patch=211215
Arch   : "LSB;label=32;scalar=64"
Exec   : surfaceFeatureConvert constant/triSurface/walls_mm_tr.eMesh constant/triSurface/walls_mm_tr.obj
Date   : Aug 18 2022
Time   : 15:24:50
Host   : kamakiri-VirtualBox
PID    : 18944
I/O    : uncollated
Case   : /home/kamakiri/Desktop/Track3/001_3Ddam_interFoam
nProcs : 1
trapFpe: Floating point exception trapping enabled (FOAM_SIGFPE).
fileModificationChecking : Monitoring run-time modified files using timeStampMaster (fileModificationSkew 5, maxFileModificationPolls 20)
allowSystemOperations : Allowing user-supplied system call operations

// * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * //

Read edgeMesh "constant/triSurface/walls_mm_tr.eMesh"
points      : 8
edges       : 12
boundingBox : (0.6635 -0.2015 0) (0.8245 0.2015 0.161)
```

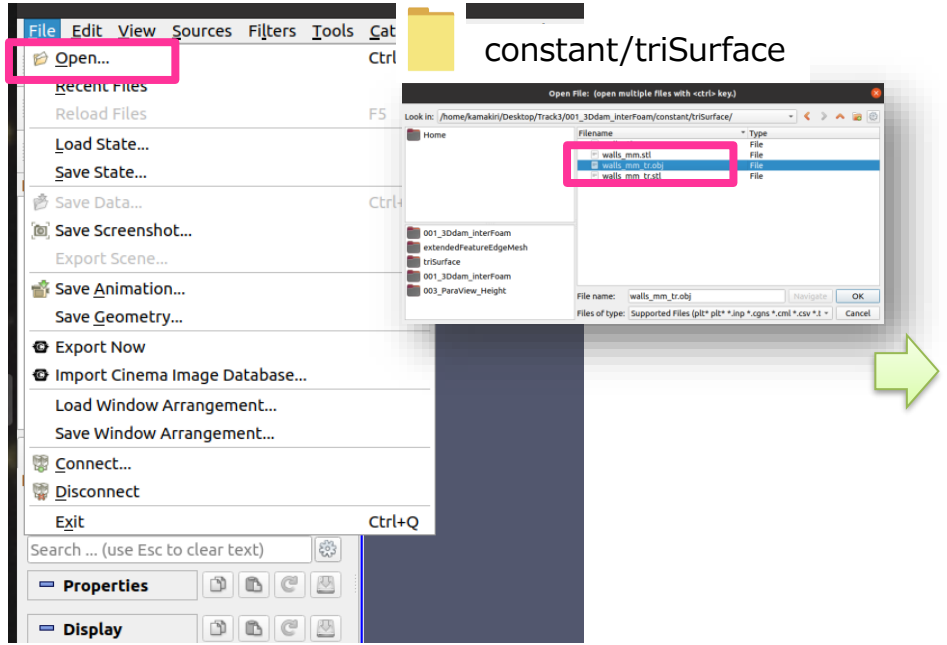
```
allowSystemOperations : Allowing user-supplied system call operations

// * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * //

Read edgeMesh "constant/triSurface/walls_mm_tr.eMesh"
points      : 8
edges       : 12
boundingBox : (0.6635 -0.2015 0) (0.8245 0.2015 0.161)


writing "constant/triSurface/walls_mm_tr.obj" without scaling
points      : 8
edges       : 12
boundingBox : (0.6635 -0.2015 0) (0.8245 0.2015 0.161)

End
```

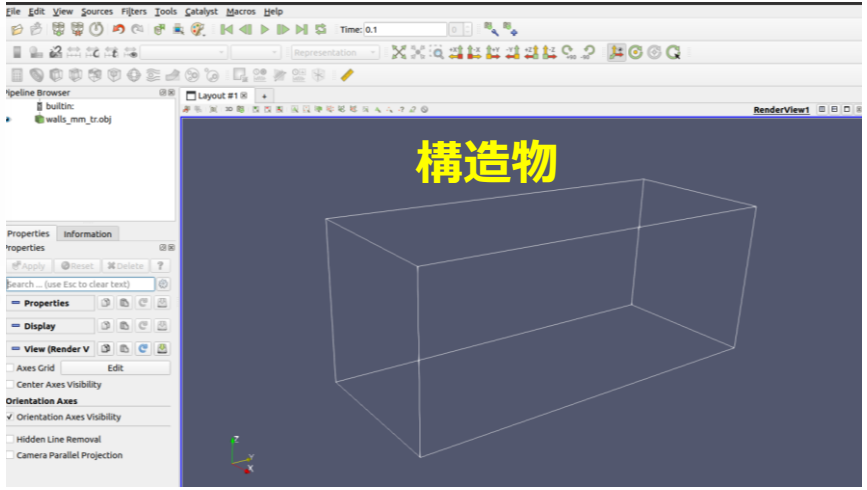


メッシュ作成(surfaceFeatureExtract)

以下のコマンドを実行しParaViewを起動後「constant/triSurface/walls_mm_tr.obj」を読み込む

```
$ paraview
```





特徴線の確認

メッシュ作成(snappyHexMesh)

system/snappyHexMeshDict

```
geometry
{
  walls_mm_tr.stl
  {
    type triSurfaceMesh;
    name walls; //面の名前
  }
}
```

構造物stlファイル

```
castellatedMeshControls
{
  features (
    {
        file "walls_mm_tr.eMesh";
        level 0;
    }
  );
```

surfaceFeatureExtractで作成した特徴線を指定(constant/triSurface)

```
refinementSurfaces
 {
   walls
   {
     level (0 0);
     patchInf
     {
        type wall;
     }
   }
 }
 refinementRegions
 {
 }
 locationInMesh (2.0 0.5 0.5);
```

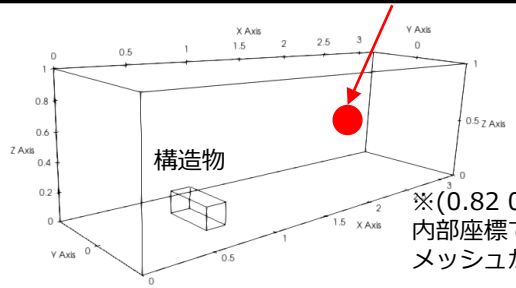
境界メッシュの細分化

再分割の段階設定
※今回は再分割は行わない

メッシュ内部の座標を指定する

constant/triSurface/walls_mm_tr.stl

```
solid walls
  facet normal -1.000000 0.000000 0.000000
    outer loop
      vertex 824.500000 698.000000 0.000000
      vertex 824.500000 295.000000 0.000000
```



※(0.82 0.0 0.0)とすると構造物内部座標であるため構造物内部にメッシュが作成される

メッシュ作成(snappyHexMesh)

以下のコマンドでsnappyHexMeshを実行

$snappyHexMesh -overwrite

※オプションとして「-overwrite」を付け結果の上書きを行う。
snappyHexMeshをオプションなしで実行した場合、実行ステップごとに時刻フォルダができる。できたメッシュをチェックして問題なければ0以外の時刻フォルダを削除してから改めて上書きで snappyHexMesh を実行する。

```
kamakiri@kamakiri-VirtualBox ~/Desktop/Track3/001_3Ddam_interFoam $ snappyHexMesh -overwrite
/*---------------------------------------------------------------------------*\
| =========                 |                                                 |
| \\      /  F ield         | OpenFOAM: The Open Source CFD Toolbox           |
| \\    /   O peration     | Version:  2106                                  |
| \\  /    A nd           | Website:  www.openfoam.com                      |
| \\/     M anipulation  |                                                 |
\*---------------------------------------------------------------------------*/
Build  : _40a7d5d6-20210727 OPENFOAM=2106 patch=211215
Arch   : "LSB;label=32;scalar=64"
Exec   : snappyHexMesh -overwrite
Date   : Aug 17 2022
Time   : 22:43:04
Host   : kamakiri-VirtualBox
PID    : 12435
I/O    : uncollated
Case   : /home/kamakiri/Desktop/Track3/001_3Ddam_interFoam
nProcs : 1
trapFpe: Floating point exception trapping enabled (FOAM_SIGFPE).
fileModificationChecking : Monitoring run-time modified files using timeStampMaster (fileModificationSkew 5, maxFileModificationPolls 20)
allowSystemOperations : Allowing user-supplied system call operations

// * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * //
Create time

Create mesh for time = 0

Read mesh in = 0.21 s
```

```
Writing mesh to time constant
Wrote mesh in = 0.24 s.
Layers added in = 0.24 s.
Checking final mesh ...
Checking faces in error :
    non-orthogonality > 65  degrees                          : 0
    faces with face pyramid volume < 1e-13                   : 0
    faces with concavity > 80  degrees                       : 0
    faces with skewness > 4   (internal) or 20  (boundary)   : 0
    faces with interpolation weights (0..1)  < 0.05          : 0
    faces with volume ratio of neighbour cells < 0.01        : 0
    faces with face twist < 0.05                             : 0
    faces on cells with determinant < 0.001                  : 0
Finished meshing without any errors
Finished meshing in = 3.13 s.
End
```

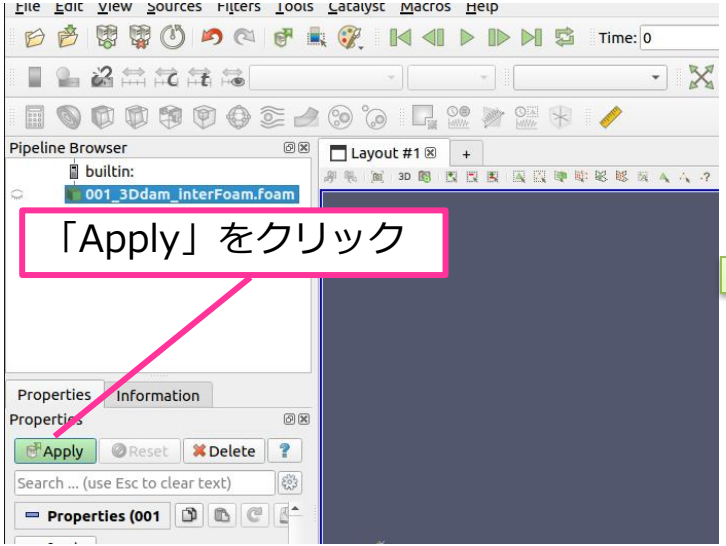
エラーなく終了

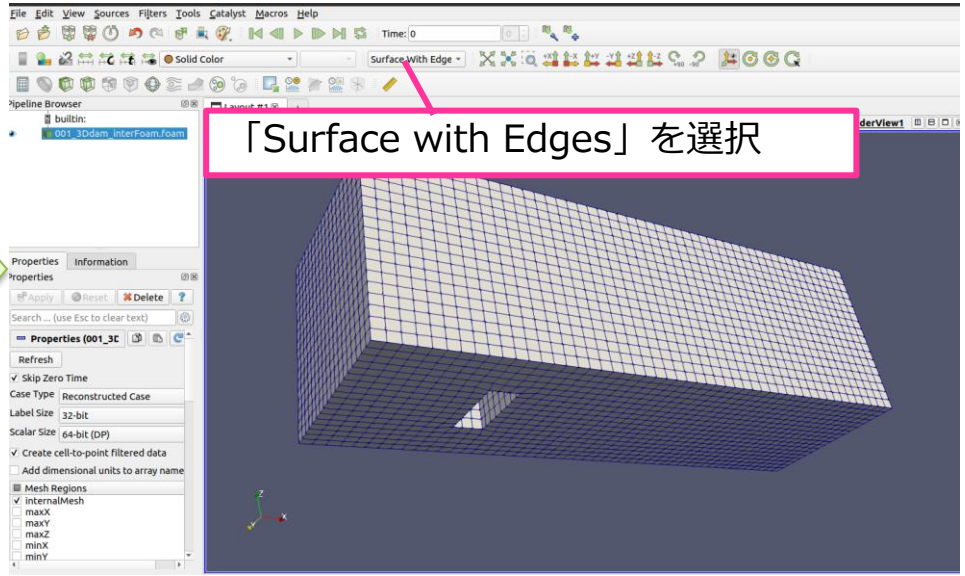
メッシュ作成の完了

メッシュ作成(snappyHexMesh)

以下のコマンドを実行しParaViewを起動

```
$ paraFoam
```





メッシュ形状に問題ないことを確認

メッシュ作成(checkMesh)

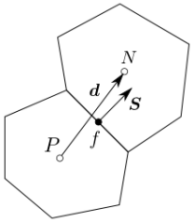
メッシュに関する情報を表示するユーティリティcheckMesh

以下のコマンドを実行しメッシュ情報の確認

```
$ checkMesh
```

```
Checking geometry...
    Overall domain bounding box (0 -0.5 0) (3.22 0.5 1)
    Mesh has 3 geometric (non-empty/wedge) directions (1 1 1)
    Mesh has 3 solution (non-empty) directions (1 1 1)
    Boundary openness (-1.35158e-16 4.91481e-16 4.48838e-16) OK.
    Max cell openness = 2.12825e-16 OK.
    Max aspect ratio = 1.36317 OK. アスペクト比の最大値
    Minimum face area = 0.00205919. Maximum face area = 0.00482982.  Face area magnitudes OK.
    Minimum cell volume = 0.000183424. Maximum cell volume = 0.000282483.  Total volume = 3.20955.  Cell volumes OK.
    Mesh non-orthogonality Max: 19.2201 average: 0.938323
    Non-orthogonality check OK.
    Face pyramids OK.
    Max skewness = 0.293715 OK. skewness(歪度，ゆがみ)の最大値
    Coupled point location match (average 0) OK.

Mesh OK.

End
```

メッシュ品質に関係するパラメーターとして non-orthogonality（非直交性）がある



面とセル中心間を結ぶ線の直交性の度合いを表している。
注目セルの中心Pと隣接セルの中心 Nとを結んだベクトルdと両セルが共有する面 f の法線ベクトル S が作る角度。
目安として80°を超える場合はその部分のメッシュは作り直した方が良い。

解析手順

形状作成

メッシュ作成

解析条件の設定

計算実行

ポスト処理

FreeCAD

```
blockMesh                    # ベースメッシュ作成
surfaceFeatureExtract        # 特徴線の抽出
snappyHexMesh -overwrite     # メッシュ生成
```

流速、圧力、体積分率の初期値の設定
スキーム、解析制御の設定

OpenFOAM

gnuplot

ParaView

解析条件の設定

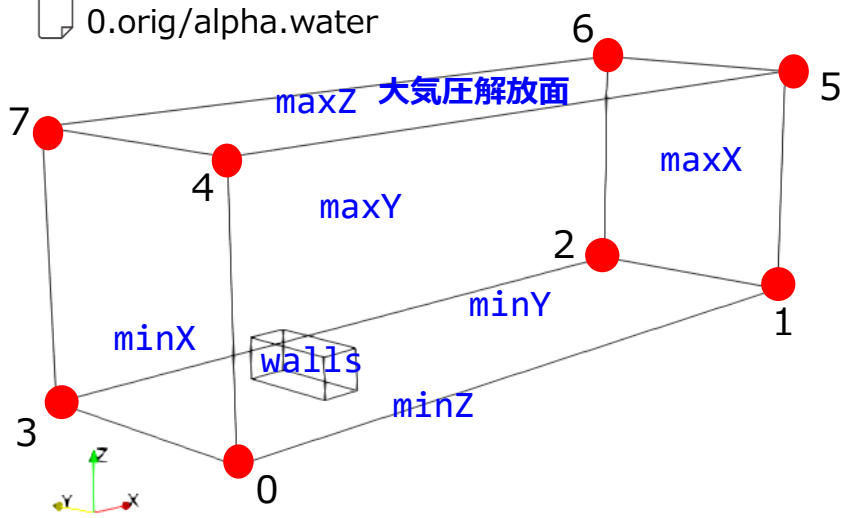
流速U、圧力P、体積分率αは「0.orig」フォルダで設定

- **maxZのみ大気圧解放面の圧力境界**
- **それ以外はすべり無し**

0.rig/U

0.orig/p_rgh

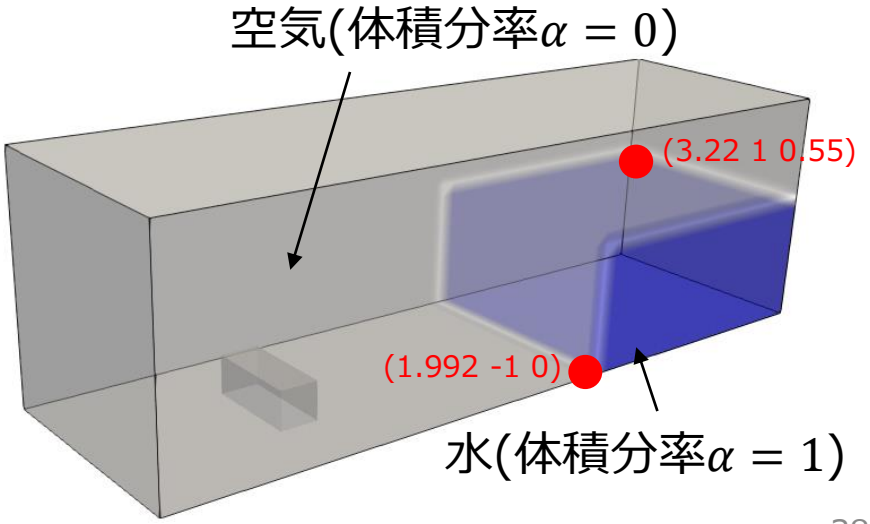
0.orig/alpha.water



体積分率の初期値は「system/setFieldsDict」ファイルで設定

(1.992 -1 0)と(3.22 1 0.55)を対角線にもつ直方体を水領域とする

system/setFieldsDict



解析条件設定

- **maxZのみ大気圧解放面の圧力境界**
- **それ以外はすべり無し**

0.orig/U

```
boundaryField
{
    maxZ
    {
        type            pressureInletOutletVelocity;
        value           uniform (0 0 0);
    }

    ".*"
    {
        type            noSlip;
    }
}
```

デフォルト設定
.*は正規表現（何でもいい1文字か複数文字）

pressureInletOutletVelocity
圧力規定境界条件下の流入
流出用速度境界条件
流出時: パッチ法線方向の速度勾配0の自由流出
流入時: 境界に隣接する内部格子の速度のパッチ法線方向の成分

noSlip
すべり無し

0.orig/p_rgh

圧力はpではなく
p_rgh =p – ρghを解く

```
boundaryField
{
    maxZ
    {
        type            totalPressure;
        p0              uniform 0;
    }

    ".*"
    {
        type            fixedFluxPressure;
        phi             phiAbs;
        value           uniform 0;
    }
}
```

デフォルト設定
.*は正規表現（何でもいい1文字か複数文字）

totalPressure
全圧規定(全圧=静圧+速度圧)
全圧p=0
※非圧縮仮定では圧力は差圧のみを解いている

fixedFluxPressure
速度の境界条件から定まる流束となるように圧力勾配を規定

0.orig/alpha.water

```
boundaryField
{
    maxZ
    {
        type            inletOutlet;
        inletValue      uniform 0;
        value           uniform 0;
    }
    ".*"
    {
        type            zeroGradient;
    }
}
```

デフォルト設定
.*は正規表現（何でもいい1文字か複数文字）

inletOutlet
流入時の値を指定できる流出境界条件
流出時: パッチ法線方向の勾配0(自由流出)

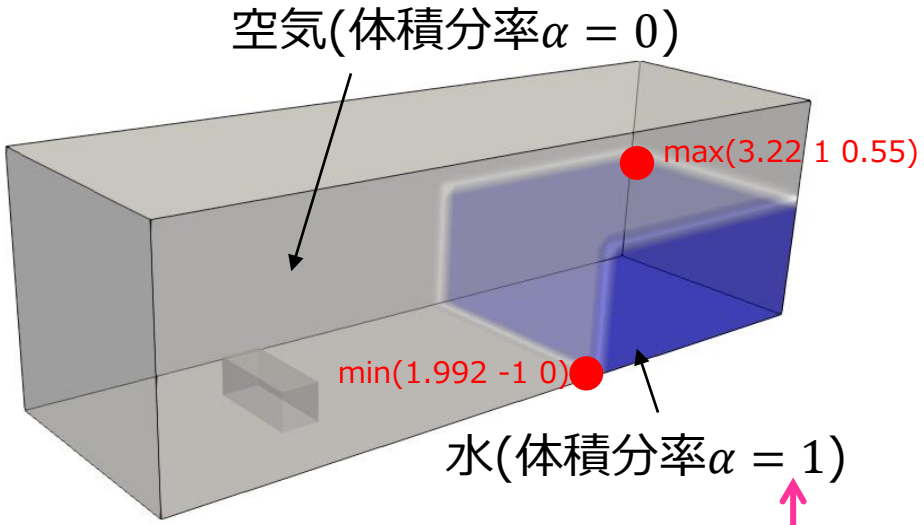
zeroGradient
勾配を0
自由流出を仮定すると流出時の境界条件は法線方向の勾配を0とするのが一般的

解析条件設定

初期値の設定

system/setFieldsDict

```
defaultFieldValues    デフォルト設定
(
    volScalarFieldValue alpha.water 0   スカラー値
    volVectorFieldValue U (0 0 0)
);

regions    領域指定でスカラー値やベクトル設定
(
    boxToCell
    {
        box (1.992 -1 0) (3.22 1 0.55);  領域(min) (max)
        fieldValues
        (
            volScalarFieldValue alpha.water 1   スカラー値
        );
    }
);
```



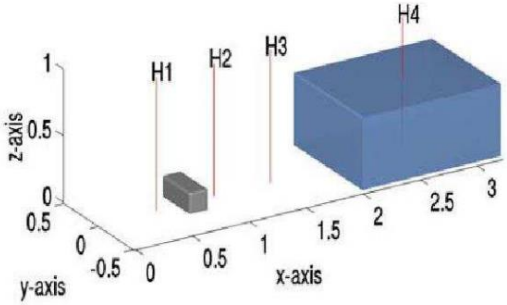
解析条件設定

重力の設定

constant/g

```
dimensions      [0 1 -2 0 0 0 0];
value           (0 0 -9.81);
```

重力方向(0 0 -9.81)



物性値の設定

constant/transportProperties

```
phases (water air);

water
{
   transportModel  Newtonian;  ニュートン流体
   nu              1e-06;      動粘性係数
   rho             1000;       密度
}

air
{
   transportModel  Newtonian;
   nu              1.48e-05;
   rho             1;
}

sigma           0.07;  2相間の表面張力
                       単位[N/m]（[kg/s2 ]）
```

乱流モデルの設定

constant/turbulenceProperties

```
simulationType  laminar;     乱流モデルなし
```

laminar	層流（乱流モデルなし）
RAS	レイノルズ平均に基づく乱流モデル
LES	LES (Large Eddy Simulation) モデル

解析条件設定

離散化スキーム

system/fvSchemes

```
ddtSchemes    時間項の離散化スキーム
{
    default        Euler;    オイラー(1次精度)
}

gradSchemes   勾配項の離散化スキーム
{
    default        Gauss linear;  線形補間(中心差分、2次精度)
}
divSchemes    移流項・発散項の離散化スキーム
{
    div(rhoPhi,U)   Gauss upwind;      Uの移流項 風上差分(1次精度)
    div(phi,alpha)  Gauss vanLeer;     体積分率の移流項 vanLeer制御関数TVD
    div(phirb,alpha) Gauss linear;     界面圧縮項関連 線形補間
    div(((rho*nuEff)*dev2(T(grad(U))))) Gauss linear;
}
```

```
laplacianSchemes   ラプラシアン項の離散化スキーム
{
    default        Gauss linear corrected;   格子の非直交性考慮あり
}                                            （非直交性補正あり）

interpolationSchemes    格子界面の補間スキーム
{
    default        linear;    線形補間
}

snGradSchemes    界面法線方向勾配の補間スキーム
{
    default        corrected;  格子が非直交性の場合補正を制限なく
}                              行う（laplacianSchemesスキームと合
                               わせておくとよい）
wallDist
{
    method meshWave;
}
```

解析条件設定

代数方程式ソルバ設定

system/fvSolution

```
solvers    線形ソルバ設定
{
   "alpha.water.*"   体積分率のVOF解法設定
   {
      nAlphaCorr       2;  解法反復回数
      nAlphaSubCycles 1;サブサイクル回数
      cAlpha           1;      界面圧縮項関連
                               （圧縮パラメータ）
      MULESCorr        yes;
      nLimiterIter     5;

      solver          smoothSolver;
      smoother        symGaussSeidel;
      tolerance       1e-8;
      relTol          0;
   }

   "pcorr.*"    格子移動時の流束補正用
   {            に対する線形ソルバ
      solver          PCG;
      preconditioner  DIC;
      tolerance       1e-5;
      relTol          0;
   }

   p_rgh   静水圧を除いた圧力のポアソン方程式に対
   {       する線形ソルバ
      solver          PCG;  前処理付き共役勾配法
      preconditioner  DIC;対角ベース不完全Cholesky分解法
      tolerance       1e-07収束判定残差
      relTol          0.05;  収束判定残差（初期値との差）
   }

   p_rghFinal   静水圧を除いた圧力（反復最終時）
   {
      $p_rgh;        p_rghの設定を再利用
      relTol         0;  0とすればtoleranceだけの判定
   }

   U        Uの輸送方程式に対する線形ソルバ
   {
      solver          smoothSolver;      平滑型のソルバ
      smoother        symGaussSeidel;対称ガウスザイデル
      tolerance       1e-06;
      relTol          0;
   }
}
```

※全ての変数で代数方程式の初期残差がtolerance以下になるか、イテレーションループの初期残差との差がrelTol以下になればイテレーションを抜ける

```
PIMPLE  PIMPLE法の設定
{
   momentumPredictor    no;  運動量輸送方程式を解いて運動量予測値を算出
   nOuterCorrectors     1;  PIMPLE法の内反復回数
   nCorrectors          3;     PIMPLE法の外反復回数
   nNonOrthogonalCorrectors 0;非直交補正反復回数
}

relaxationFactors
{
   equations
   {
      ".*" 1;
   }
}
```

解析条件設定

計算の制御

system/controlDict

```
application     interFoam;   使用するソルバ
startFrom       startTime;
startTime       0;           解析開始時間(0秒から)
stopAt          endTime;
endTime         8;           計算時間(8秒まで)
deltaT          0.001;       時間刻み
                             (自動制御の場合は初期値)
writeControl    adjustable;
writeInterval   0.1;         計算の記録間隔
purgeWrite      0;
writeFormat     ascii;
writePrecision  6;
```

```
writePrecision  6;
writeCompression off;
timeFormat      general;
timePrecision   6;
runTimeModifiable yes;
adjustTimeStep  yes;
maxCo           0.5;         最大クーラン数
maxAlphaCo      0.5;         相界面(0<相率<1)
                             での最大クーラン数
maxDeltaT       1;           最大時間刻み
```

```
functions // 関数定義
{
    #includeFunc probes; // 壁面での圧力時刻歴
    #includeFunc interfaceHeight; // 水面高さの時刻歴

    #includeFunc residuals; // 残差
}
```

後述する

解析条件設定

以下のコマンドで初期値の設定を実行

```
$cp -r 0.orig 0
$setFields
```

「0.orig 0」ファイルを「0」ファイルとしてコピー
※setFieldsコマンドの前に「0」ファイルがないとエラーとなる

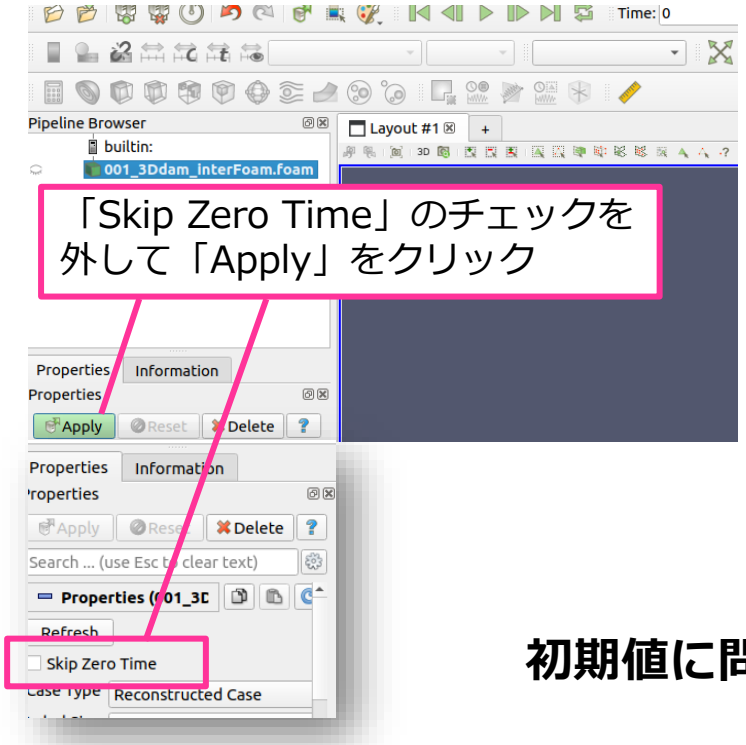
```
kamakiri@kamakiri-VirtualBox ~/Desktop/Track3/001_3Ddam_interFoam $ cp -r 0.orig 0
kamakiri@kamakiri-VirtualBox ~/Desktop/Track3/001_3Ddam_interFoam $ setFields
/*---------------------------------------------------------------------------*\
| =========                 |                                                 |
| \\      /  F ield         | OpenFOAM: The Open Source CFD Toolbox           |
|  \\    /   O peration     | Version:  2106                                  |
|   \\  /    A nd           | Website:  www.openfoam.com                      |
|    \\/     M anipulation  |                                                 |
\*---------------------------------------------------------------------------*/
Build  : _40a7d5d6-20210727 OPENFOAM=2106 patch=211215
Arch   : "LSB;label=32;scalar=64"
Exec   : setFields
Date   : Aug 17 2022
Time   : 23:06:51
Host   : kamakiri-VirtualBox
PID    : 14475
I/O    : uncollated
Case   : /home/kamakiri/Desktop/Track3/001_3Ddam_interFoam
nProcs : 1
trapFpe: Floating point exception trapping enabled (FOAM_SIGFPE).
fileModificationChecking : Monitoring run-time modified files using timeStampMaster (fileModificationSk
allowSystemOperations : Allowing user-supplied system call operations

// * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * //
Create time
```

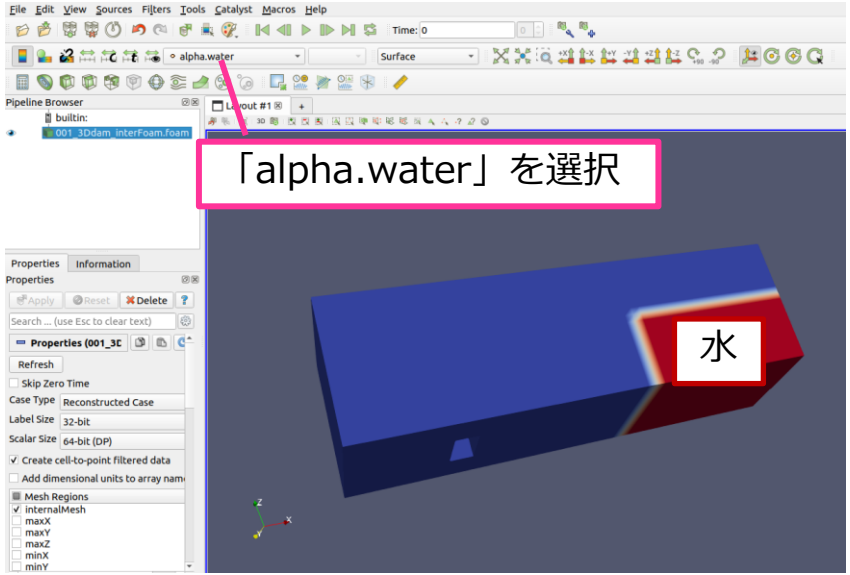
```
// * * * * * * * * * * * * * * * * * * *
```

解析条件設定

以下のコマンドを実行しParaViewを起動

```
$ paraFoam
```





初期値に問題ないことを確認

解析手順

形状作成

FreeCAD

メッシュ作成

```
blockMesh                    # ベースメッシュ作成
surfaceFeatureExtract        # 特徴線の抽出
snappyHexMesh -overwrite     # メッシュ生成
```

解析条件の設定

流速、圧力、体積分率の初期値の設定
スキーム、解析制御の設定

計算実行

OpenFOAM

ポスト処理

gnuplot

ParaView

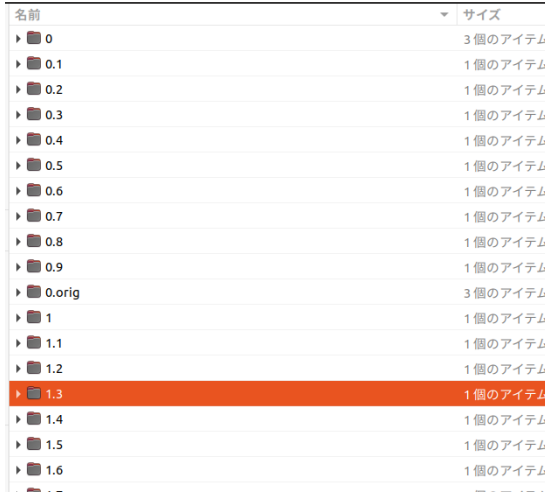
計算実行

以下のコマンドで計算実行

$interFoam

※並列計算なし

```
kamakiri@kamakiri-VirtualBox ~/Desktop/Track3/001_3Ddam_interFoam $ interFoam
/*---------------------------------------------------------------------------*\
| =========                 |                                                 |
| \\      /  F ield         | OpenFOAM: The Open Source CFD Toolbox           |
|  \\    /   O peration     | Version:  2106                                  |
|   \\  /    A nd           | Website:  www.openfoam.com                      |
|    \\/     M anipulation  |                                                 |
\*---------------------------------------------------------------------------*/
Build  : _40a7d5d6-20210727 OPENFOAM=2106 patch=211215
Arch   : "LSB;label=32;scalar=64"
Exec   : interFoam
Date   : Aug 17 2022
Time   : 23:41:30
Host   : kamakiri-VirtualBox
PID    : 14974
I/O    : uncollated
Case   : /home/kamakiri/Desktop/Track3/001_3Ddam_interFoam
nProcs : 1
trapFpe: Floating point exception trapping enabled (FOAM_SIGFPE).
fileModificationChecking : Monitoring run-time modified files using timeStampMaster (fileModificationSkew 5, maxFileModificationPolls 20)
allowSystemOperations : Allowing user-supplied system call operations

// * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * //
Create time

Create mesh for time = 0

Selecting dynamicFvMesh staticFvMesh

PIMPLE: Operating solver in PISO mode
```



エラーなく計算実行されることを確認

計算の中断は「Ctrl+C」

OpenFOAM計算実行

フォルダ内を初期化したいときに使う

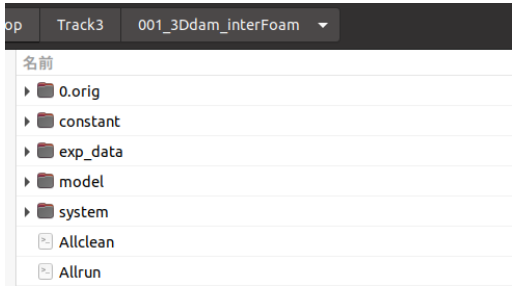
Allcleanスクリプトでフォルダ内の初期化

```
$./Allclean
```

Allclean

```
#!/bin/sh
cd "${0%/*}" || exit                                # Run from this directory
. ${WM_PROJECT_DIR:?}/bin/tools/CleanFunctions      # Tutorial clean functions
#------------------------------------------------------------------------------

cleanCase0

if [ -d ./logs ]; then
    rm -r logs
fi
```

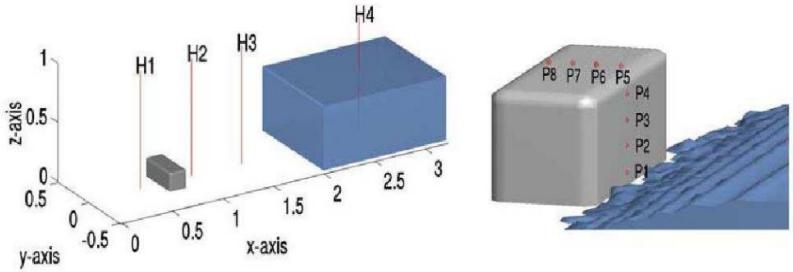
フォルダ内が初期化



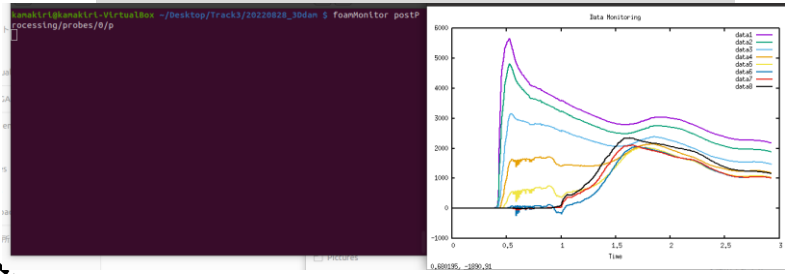
測定データの設定

計測位置P1~P8の圧力データとH1~H4の水位データをモニタリングする

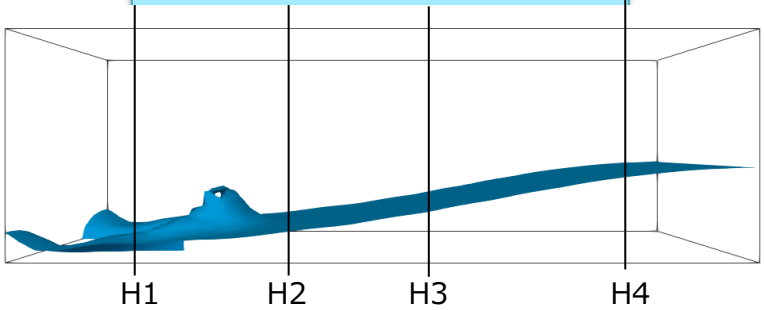
P1~P8の圧力データ



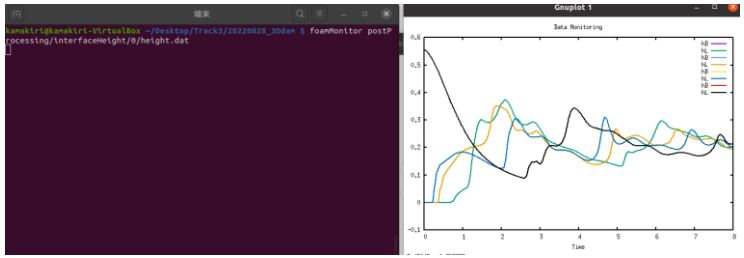
$foamMonitor postProcessing/probes/0/p



H1~H4の水位データ



$foamMonitor postProcessing/interfaceHeight/0/height.dat



圧力測定

controlDictファイル内でfunction objectsによりポスト処理の設定を記述する

system/controlDict

```
(省略)

application     interFoam;

(省略)

maxDeltaT       1;

functions // 関数定義
{
        #includeFunc probes; // 壁面での圧力時刻歴

        #includeFunc interfaceHeight; // 水面高さの時刻歴

        #includeFunc residuals; // 残差
}
```

圧力測定

インクルード(読み込み)

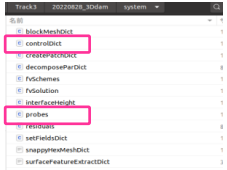
system/probes

```
//probes 設定の挿入
#includeEtc "caseDicts/postProcessing/probes/probes.cfg"
fields (p p_rgh); // プローブする場
probeLocations
(
    (0.8245 0 0.021) // プローブ座標 P1
    (0.8245 0 0.061) // プローブ座標 P2
    (0.8245 0 0.101) // プローブ座標 P3
    (0.8245 0 0.141) // プローブ座標 P4
    (0.8035 0 0.162) // プローブ座標 P5
    (0.7635 0 0.162) // プローブ座標 P6
    (0.7235 0 0.162) // プローブ座標 P7
    (0.6835 0 0.162) // プローブ座標 P8
);
interpolationScheme cellPoint; // 補間スキーム
```

計測する物理量

計測位置の指定

セル値による線形重み付け補間



#includeFunc <fileName>

reads the file of name <*fileName*>, searched from the case *system* directory, followed by the *$FOAM_ETC* directory;

ファイルを開いて確認

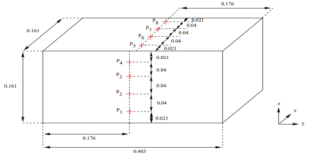


Figure 7: Description of the box.

水面高さ

※v2006以前のバージョンでは並列計算時に出力されない問題

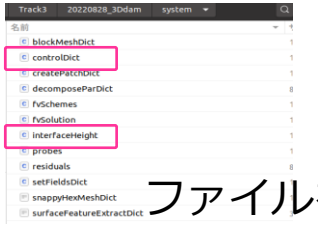
controlDictファイル内でfunction objectsによりポスト処理の設定を記述する

system/controlDict

水面高さの測定

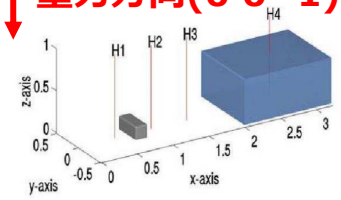
```
(省略)

application     interFoam;

(省略)

maxDeltaT       1;

functions // 関数定義
{
        #includeFunc probes; // 壁面での圧力時刻歴
        #includeFunc interfaceHeight; // 水面高さの時刻歴

        #includeFunc residuals; // 残差
}
```

インクルード(読み込み)



ファイルを開いて確認

重力方向(0 0 -1)



system/interfaceHeight

interface Height (リンク) v2006から導入

```
interfaceHeight
{
    type            interfaceHeight;
    libs            (fieldFunctionObjects);
    locations       (
                    (0.496 0 0)
                    (0.992 0 0)
                    (1.488 0 0)
                    (2.638 0 0)
                    );
    alpha           alpha.water;
    liquid          true;
    direction       (0 0 -1);
    interpolationScheme     cellPoint;
    writePrecision  8;
    writeToFile     true;
    useUserTime     true;
    region          region0;
    enabled         true;
    log             true;
    timeStart       0;
    timeEnd         1000;
    executeControl  timeStep;
    executeInterval 1;
    writeControl    timeStep;
    writeInterval   1;
}
```

計測位置の指定

重力方向を指定

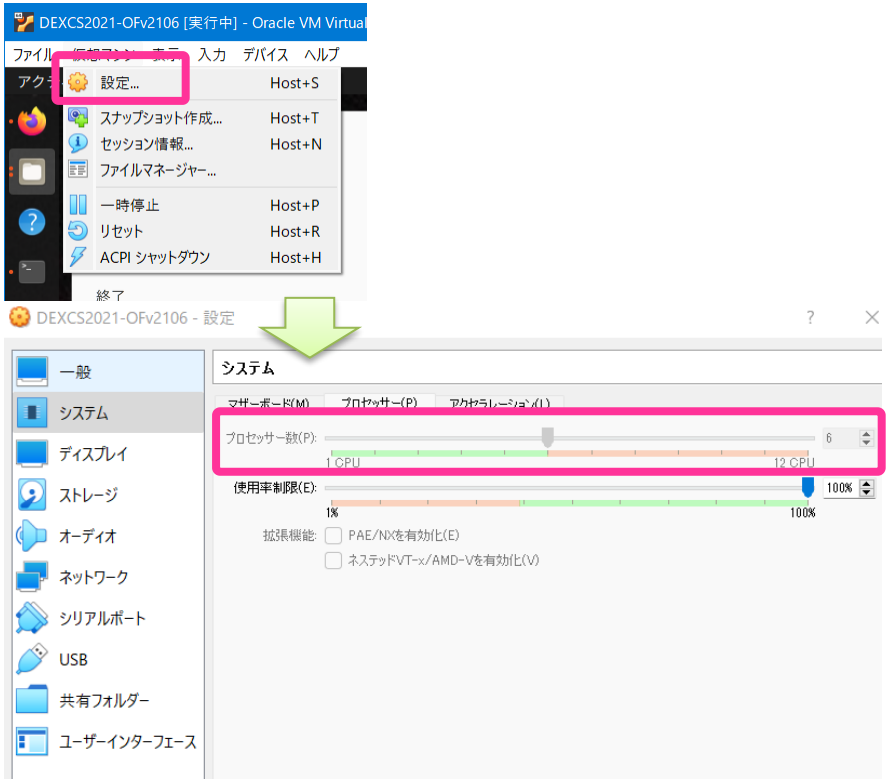
並列計算

「仮想マシン」>「設定」>「プロセッサー数」を確認

system/decomposeParDicct

```
numberOfSubdomains   4;

method               scotch;
```

他に何も設定しなくても分割
プロセッサー数をよりnumberOfSubdomainsが大きい
とエラーになるので注意。



OpenFOAM計算実行

Allrunスクリプトを実行

```
$./Allrun &
```

バックグラウンド実行

Allrun

```
#!/bin/sh
cd "${0%/*}" || exit                                # Run from this directory
. ${WM_PROJECT_DIR:?}/bin/tools/RunFunctions        # Tutorial run functions
#------------------------------------------------------------------------------

restore0Dir                                         # 0.origファイルをコピー(cp -r 0.orig 0を実施する関数を含む)
touch post.foam                                     # post.foamの空ファイルを作成
runApplication blockMesh                            # ベースメッシュ作成
runApplication surfaceFeatureExtract                # 特徴線の抽出
runApplication snappyHexMesh -overwrite             # メッシュ生成
runApplication setFields                            # 体積分率の初期値
runApplication decomposePar                         # 領域分割
runParallel $(getApplication)                       # interFoamソルバ実行(並列計算)
reconstructPar                                      # 分割領域の結合
rm -r processor*                                    # processor*フォルダ削除
runApplication foamLog log.$(getApplication)        # logファイル作成（残差など）
#------------------------------------------------------------------------------
```

※並列計算のコマンド
mpirun -np 4 interFoam -parallel

解析手順

形状作成

FreeCAD

メッシュ作成

```
blockMesh                    # ベースメッシュ作成
surfaceFeatureExtract        # 特徴線の抽出
snappyHexMesh -overwrite     # メッシュ生成
```

解析条件の設定

流速、圧力、体積分率の初期値の設定
スキーム、解析制御の設定

計算実行

OpenFOAM

ポスト処理

gnuplot

ParaView

4. postProcessユーティリティを使用したデータ処理

OpenFOAMで行うデータ処理

OpenFOAMのポスト処理機能の実行方法

<u>Run-Time Processing</u>
シミュレーション実行中に行うデータ処理
- シミュレーション中のデータをモニターできる
- ソルバの実行前にcontrolDictに設定する

<u>Post-Processing</u>
シミュレーション実行後に行うデータ処理
- controlDictに設定する

```
$postProcess -list
```
で設定できるデータ処理を確認

計算結果

controlDictで設定したfunction objectによりpostProcessingにデータが格納される

計算結果

圧力データ

postProcessing/probes/0/p

```
# Probe 0 (0.8245 0 0.021)
# Probe 1 (0.8245 0 0.061)
# Probe 2 (0.8245 0 0.101)
# Probe 3 (0.8245 0 0.141)
# Probe 4 (0.8035 0 0.162)
# Probe 5 (0.7635 0 0.162)
# Probe 6 (0.7235 0 0.162)
# Probe 7 (0.6835 0 0.162)
#       Probe             0             1             2             3             4             5             6             7
#        Time
   0.00120482       3.51925       3.18574       2.70206       2.12904       1.58101       1.18204       0.970707       0.844719
   0.00265769     -0.451132     -0.749895      -1.14187      -1.45141      -1.64954      -1.71513      -1.70609      -1.62011
   0.00439595      0.729691      0.420587    0.00105787     -0.387416     -0.690317     -0.855673     -0.912463      -0.88963
   0.00647429       1.01514       0.70343       0.27736     -0.130347     -0.458722     -0.648204     -0.720619     -0.712969
    0.0089355       1.00324      0.691635      0.265871     -0.141196      -0.46871     -0.657299     -0.728931     -0.720652
    0.0118731      0.996645      0.685099      0.259517     -0.147236     -0.474316     -0.662428     -0.733564      -0.72492
    0.0153981       1.00429      0.692693      0.267007     -0.140293     -0.468148     -0.657007     -0.728602     -0.720468
    0.0196282       1.03011      0.718296      0.292043     -0.117019     -0.447152     -0.638078     -0.710809     -0.703837
    0.0246515       1.05846      0.746432      0.319601    -0.0914138     -0.424157     -0.617537     -0.691786     -0.686375
```

圧力(p1~p8)

水面高さデータ

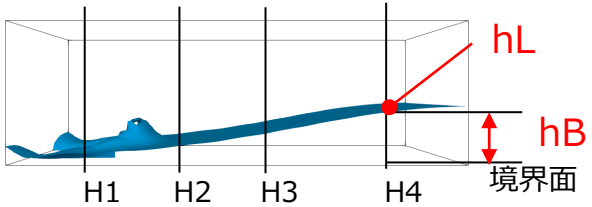
postProcessing/interfaceHeight/0/height.dat

```
# Location 0  : (4.960000e-01 0.000000e+00 0.000000e+00)
# Location 1  : (9.920000e-01 0.000000e+00 0.000000e+00)
# Location 2  : (1.488000e+00 0.000000e+00 0.000000e+00)
# Location 3  : (2.638000e+00 0.000000e+00 0.000000e+00)
# hB          : Interface height above the boundary
# hL          : Interface height above the location
# Location    0                   1                   2                   3
# Time        hB          hL      hB          hL      hB          hL      hB          hL
0.00120482    0.000000e+00  0.000000e+00  0.000000e+00  0.000000e+00  0.000000e+00  0.000000e+00  5.555557e-01  5.555557e-01
0.00265769    0.000000e+00  0.000000e+00  0.000000e+00  0.000000e+00  0.000000e+00  0.000000e+00  5.555517e-01  5.555517e-01
0.00439595    0.000000e+00  0.000000e+00  0.000000e+00  0.000000e+00  0.000000e+00  0.000000e+00  5.555414e-01  5.555414e-01
0.00647429    0.000000e+00  0.000000e+00  0.000000e+00  0.000000e+00  -1.570643e-43  -1.570643e-43  5.555209e-01  5.555209e-01
0.0089355     0.000000e+00  0.000000e+00  0.000000e+00  0.000000e+00  -1.987036e-40  -1.987036e-40  5.554855e-01  5.554855e-01
0.0118731     0.000000e+00  0.000000e+00  -1.387632e-72  -1.387632e-72  -1.073663e-38  -1.073663e-38  5.554273e-01  5.554273e-01
0.0153981     0.000000e+00  0.000000e+00  -5.659985e-68  -5.659985e-68  -7.697728e-37  -7.697728e-37  5.553350e-01  5.553350e-01
0.0196282     0.000000e+00  0.000000e+00  -4.587809e-63  -4.587809e-63  -8.312911e-35  -8.312911e-35  5.551920e-01  5.551920e-01
0.0246515     -1.779911e-97  -1.779911e-97  -1.021755e-59  -1.021755e-59  -2.005294e-33  -2.005294e-33  5.549767e-01  5.549767e-01
```

水面高さ(H1~H4)

hB : lowest boundary(境界面からの距離)
hL : location(座標)

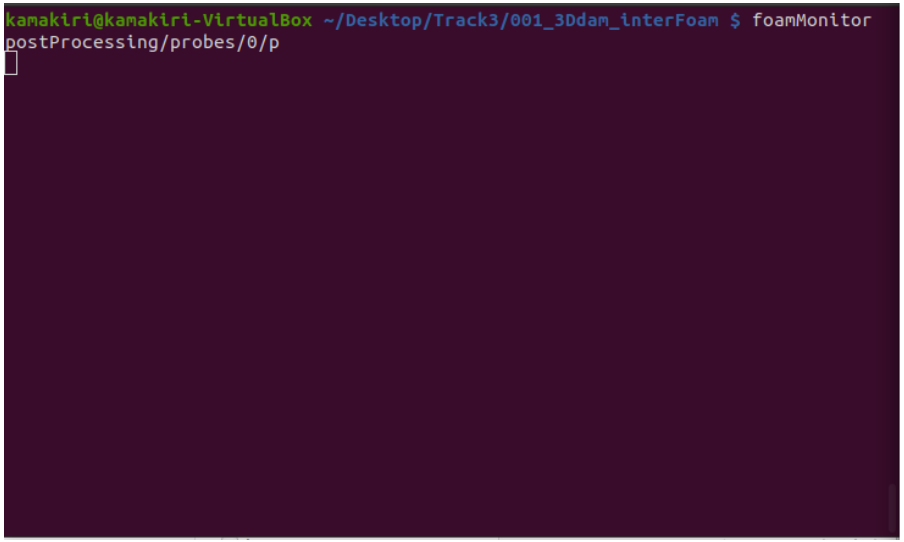
境界面がz=0
∴hL=hB
今回はhBを使う

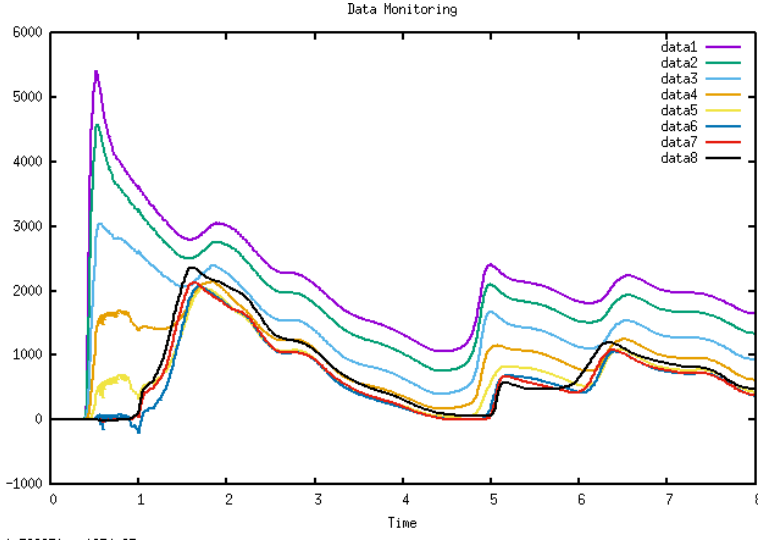


Run-Time Processing

以下のコマンドにより結果のグラフ化

```
$foamMonitor postProcessing/probes/0/p
```



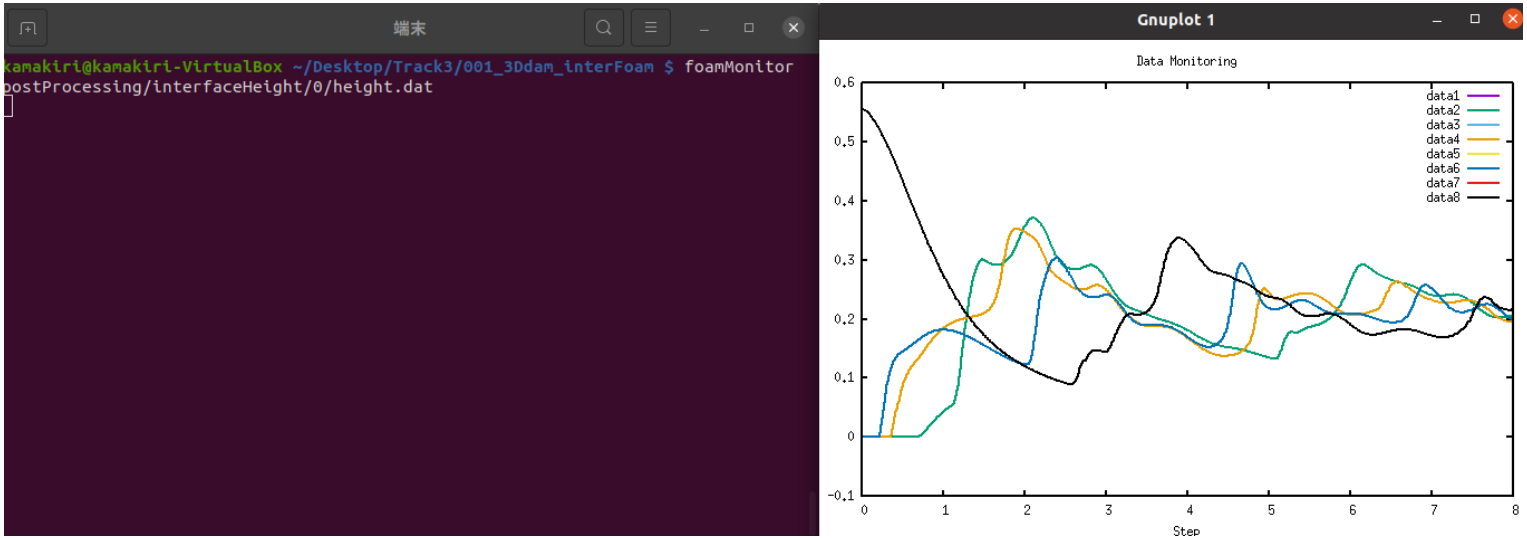


ターミナル上で「Ctrl+C」でfoamMonitorを中断する

Run-Time Processing

以下のコマンドにより結果のグラフ化

```
$foamMonitor postProcessing/interfaceHeight/0/height.dat
```



ターミナル上で「Ctrl+C」でfoamMonitorを中断する

Run-Time Processing

※補足
foamMonitorの使用方法はhelpオプションで確認ができる

$foamMonitor -help

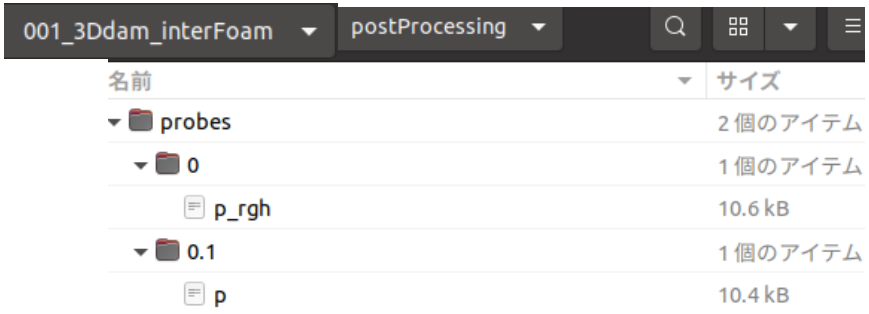
```
kamakiri@kamakiri-VirtualBox ~/Desktop/Track3/001_3Ddam_interFoam $ foamMonitor -help

Usage: foamMonitor [OPTION] <file>
options:
  -h | -help            prints the usage
  -i | -idle <time>     stops if <file> unchanging for <time> sec (default = 60)
  -l | -logscale        plots data (y-axis) on log scale, e.g. for residuals
  -r | -refresh <time>  refreshes display every <time> sec (default = 10)
  -y | -yrange <range>  sets data (y-axis) <range>, format "[0:1]"
  -g | -grid            draws grid lines on the plot

Monitor data with Gnuplot from time-value(s) graphs written by OpenFOAM
e.g. by functionObjects
- requires gnuplot, gnuplot_x11

Example:
  foamMonitor -l postProcessing/residuals/0/residuals.dat
```

Post-Processing

シミュレーション実行後に行うデータ処理

```
$postProcess -func probes
```

postProcessingが作成される




```
$postProcess –funcs '(probes interfaceHeight)'
```

複数データを処理したい場合はリストにする

※ interfaceHeightはv2106では使用不可。今後のバージョンに期待

Post-Processing

※補足
postProcessの使用方法はhelpオプションで確認ができる

```
$postProcess -help
```

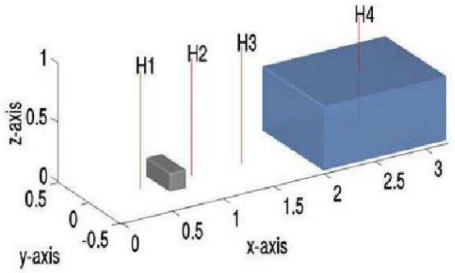
```
kamakiri@kamakiri-VirtualBox ~/Desktop/Track3/001_3Ddam_interFoam $ postProcess -help

Usage: postProcess [OPTIONS]
Options:
  -case <dir>       Specify case directory to use (instead of cwd)
  -constant         Include the 'constant/' dir in the times list
  -decomposeParDict <file>
                    Use specified file for decomposePar dictionary
  -dict <file>      Read control dictionary from specified location
  -field <name>     Specify the name of the field to be processed, e.g. U
  -fields <list>    Specify a list of fields to be processed, e.g. '(U T p)'
  -func <name>      Specify the name of the functionObject to execute, e.g. Q
  -funcs <list>     Specify the names of the functionObjects to execute, e.g.
                    '(Q div(U))'
  -latestTime       Select the latest time
  -list             List the available configured functionObjects
  -noZero           Exclude the '0/' dir from the times list
  -parallel         Run in parallel
  -profiling        Activate application-level profiling
  -region <name>    Specify alternative mesh region
  -time <ranges>    List of ranges. Eg, ':10,20 40:70 1000:', 'none', etc
  -doc              Display documentation in browser
  -help             Display short help and exit
  -help-full        Display full help and exit

Execute the set of functionObjects specified in the selected dictionary or on
the command-line for the selected set of times on the selected set of fields

Using: OpenFOAM-2106 (2106) - visit www.openfoam.com
Build: _40a7d5d6-20210727 (patch=211215)
Arch:  LSB;label=32;scalar=64
```

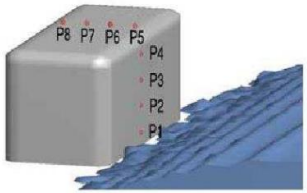
5. OpenFOAMによる3次元ダムブレイクの妥当性検証

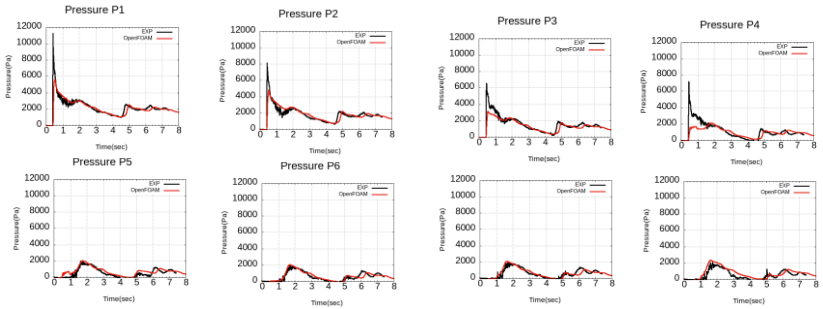
実機データとの比較

OpenFOAMで3次元ダムブレイクの解析を実行し実機データとの比較を行う

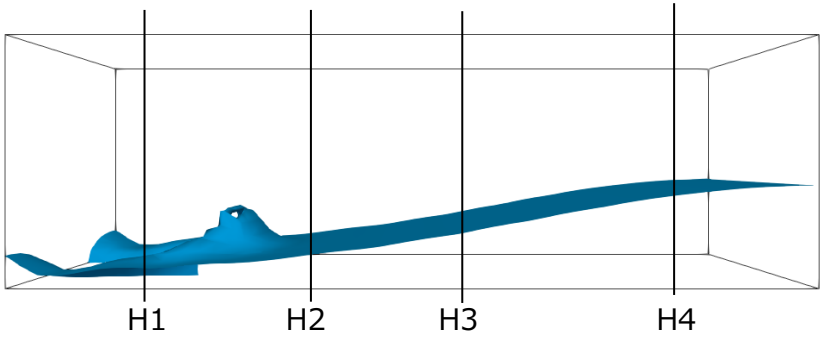
P1~P8の圧力データ

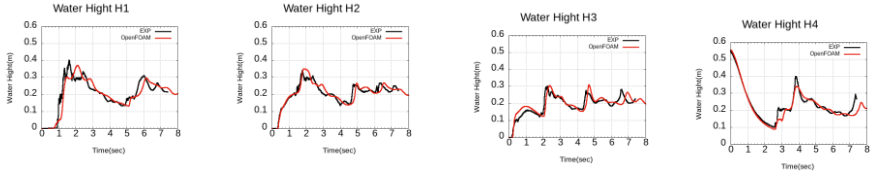






H1~H4の水位データ

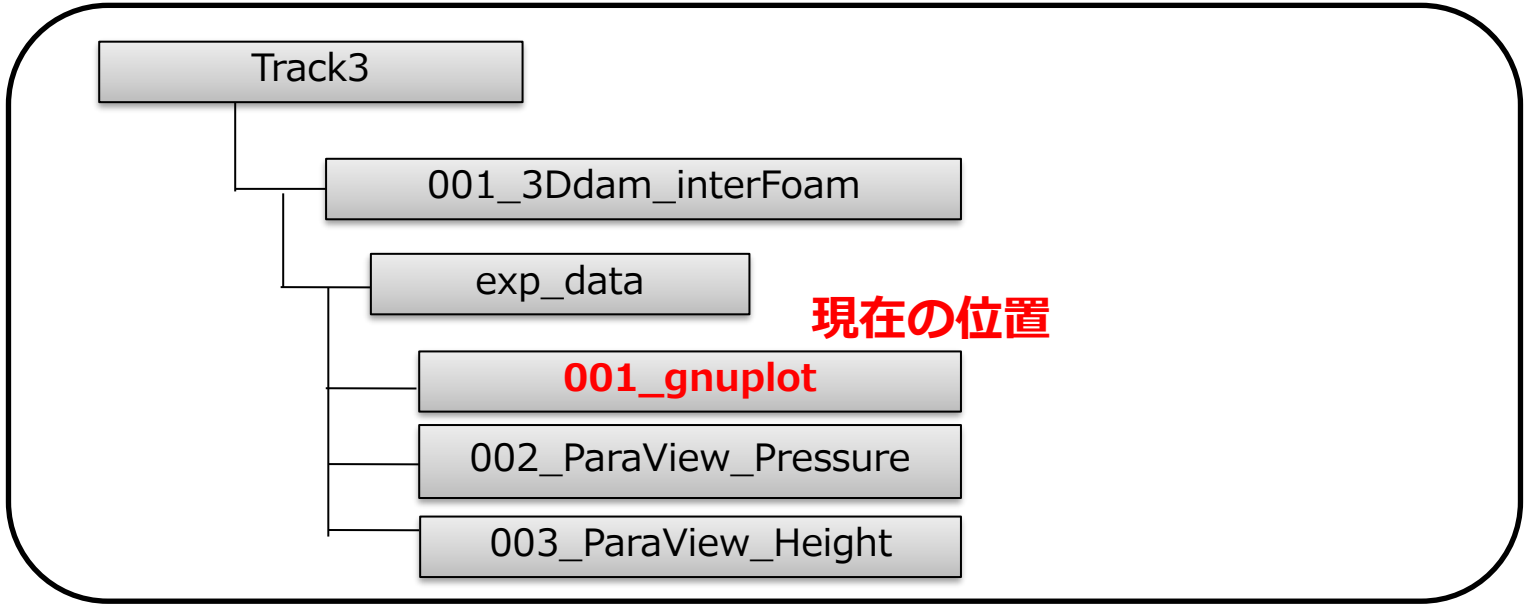




以下のコマンドで「**001_gnuplot**」フォルダに移動

フォルダを移動

```
$cd exp_data/001_gnuplot
```



gnuplot

◻ gnuplotとは

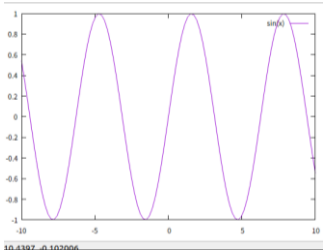
gnuplotは2次元および3次元のグラフを描画するためのフリーウェア

CUI

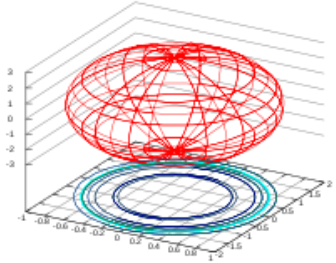
```
kamakiri@kamakiri-VirtualBox ~/Desktop/Track3/20220828_3Ddam/exp

        G N U P L O T
        Version 5.2 patchlevel 8    last modified 2019-12-01

        Copyright (C) 1986-1993, 1998, 2004, 2007-2019
        Thomas Williams, Colin Kelley and many others

        gnuplot home:     http://www.gnuplot.info
        faq, bugs, etc:   type "help FAQ"
        immediate help:   type "help"  (plot window: hit 'h')

Terminal type is now 'qt'
gnuplot> plot sin(x)
```

コマンド入力

2次元



3次元

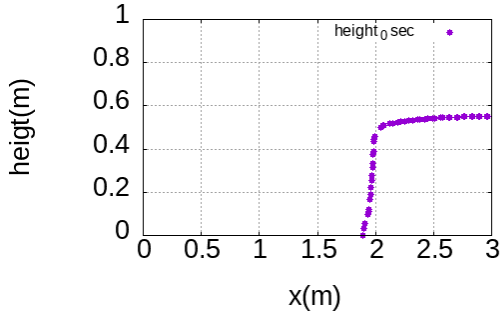


スクリプトファイル

```
set xlabel 'x(m)'  offset 0,-1              # x-axis
set ylabel 'heigt(m)'  offset -5,0          # y-axis
set xrange[0:        3 ]                    # x range
set yrange[0:        1 ]                    # height range
set grid
set xlabel font 'Arial,20'
set ylabel font 'Arial,20'
set tics font 'Arial,20'
set datafile separator ','
set lmargin 15
set tmargin 2
set rmargin 10
set bmargin 5
set term gif animate optimize delay 100  size 600,350    # 出力をgifアニメに設定
set output "heitht.gif"  # 出力ファイル名の設定

do for [i=0:8]{
    plot 'height_'.i.'sec.csv' us 6:8 ps 1 pt 7 title 'height_'.i.'sec
    pause 1
}
pause -1
```

スクリプト実行

$gnuplot “scriptファイル”



【操作マニュアル】
http://dsl4.eee.u-ryukyu.ac.jp/DOCS/gnuplot/node4.html

- 画像ファイルPNG,EPS,SVG,JPEGなどに対応
- バッチファイルにまとめて実行可能

gnuplotのスクリプト

gnuplot_run_P.plt　※拡張子は無くても良い

```
if (exist("n")==0 || n<0) n=0 #変数の初期化

point=n+1
sample=sprintf("%d",point+1) #数値を文字列(整数型)に変換

# 画像の保存
set terminal png
set output "graph_P".point.".png"

# 画枠のマージン
set lmargin 15
set tmargin 5
set rmargin 10
set bmargin 5
set mxtics #set mxtics 10

# グリッドを表示
set grid

# 座標軸レンジとラベル
set yrange[0:12000]
set xlabel font "Arial,16"
set ylabel font "Arial,16"
set tics font "Arial,20"
set ylabel 'Pressure(Pa)' offset -2,0
set xlabel 'Time(sec)' offset 0,-1

# グラフタイトル
set title "Pressure P".point offset -5,0 font "Arial,24"

plot "../test_case_2_exp_data.csv" us 1:@sample with lines lt 8 lw 3 title "EXP", "../../postProcessing/probes/0/p" us 1:@sample with lines lt 7 lw 3 title "OpenFOAM"

if (n<7) pause 0.1; n=n+1; ¥
        reread        # スクリプトの再読み込み


n=-1
#end of script
```

圧力

gnuplot_run_H.plt　※拡張子は無くても良い

```
if (exist("n")==0 || n<0) n=0 #変数の初期化

point=n+1 #変数
sample=sprintf("%d",point+9) #数値を文字列(整数型)に変換
sampleCAE=sprintf("%d",2*point+1) #数値を文字列(整数型)に変換

# 画像の保存
set terminal png
set output "graph_H".point.".png"

# 画枠のマージン
set lmargin 15
set tmargin 5
set rmargin 10
set bmargin 5
set mxtics #set mxtics 10

# グリッドを表示
set grid

# 座標軸レンジとラベル
set yrange[0:0.6]
set xlabel font "Arial,16 : "
set ylabel font "Arial,16"
set tics font "Arial,20"
set ylabel 'Water Hight(m)' offset -2,0
set xlabel 'Time(sec)' offset 0,-1

# グラフタイトル
set title "Water Hight H".point offset -5,0 font "Arial,24"

plot "../test_case_2_exp_data.csv" us 1:@sample with lines lt 8 lw 3 title "EXP", "../../postProcessing/interfaceHeight/0/height.dat" us 1:@sampleCAE with lines lt 7 lw 3 title "OpenFOAM"

if (n<3) pause 0.1; n=n+1; ¥
        reread        # スクリプトの再読み込み


n=-1
#end of script
```

水面高さ

(1)gnuplot起動状態
以下のコマンドでファイルのプロット

```
$gnuplot "gnuplot_run_P.plt"
$gnuplot "gnuplot_run_H.plt"
```

圧力グラフ画像
水面高さグラフ画像

```
001_3Ddam_interFoam/exp_data/001_gnuplot $ gnuplot "gnuplot_run_P.plt"
001_3Ddam_interFoam/exp_data/001_gnuplot $ gnuplot "gnuplot_run_H.plt"
```

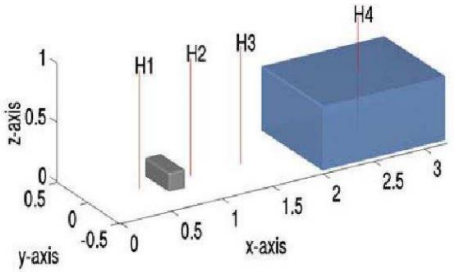
(2)画像ファイルが出力

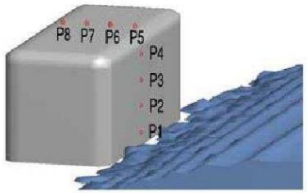
graph_H1.png	
graph_H2.png	8.8 kB
graph_H3.png	8.8 kB
graph_H4.png	8.8 kB
graph_P1.png	7.9 kB
graph_P2.png	7.9 kB
graph_P3.png	7.8 kB
graph_P4.png	7.7 kB
graph_P5.png	7.5 kB
graph_P6.png	7.5 kB
graph_P7.png	7.5 kB
graph_P8.png	7.6 kB

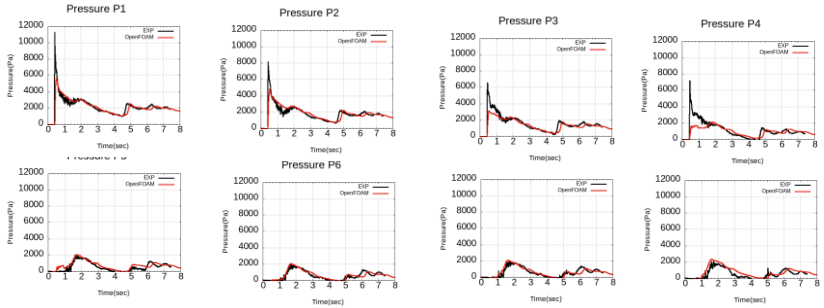
結果

グラフを確認して実験データとの差を確認してください。

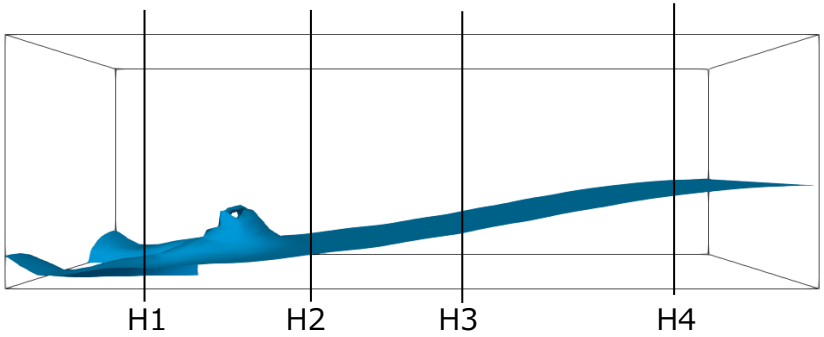
P1~P8の圧力データ

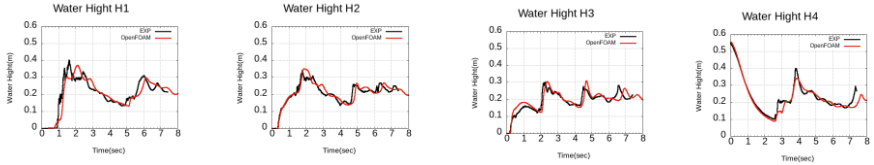






H1~H4の水位データ

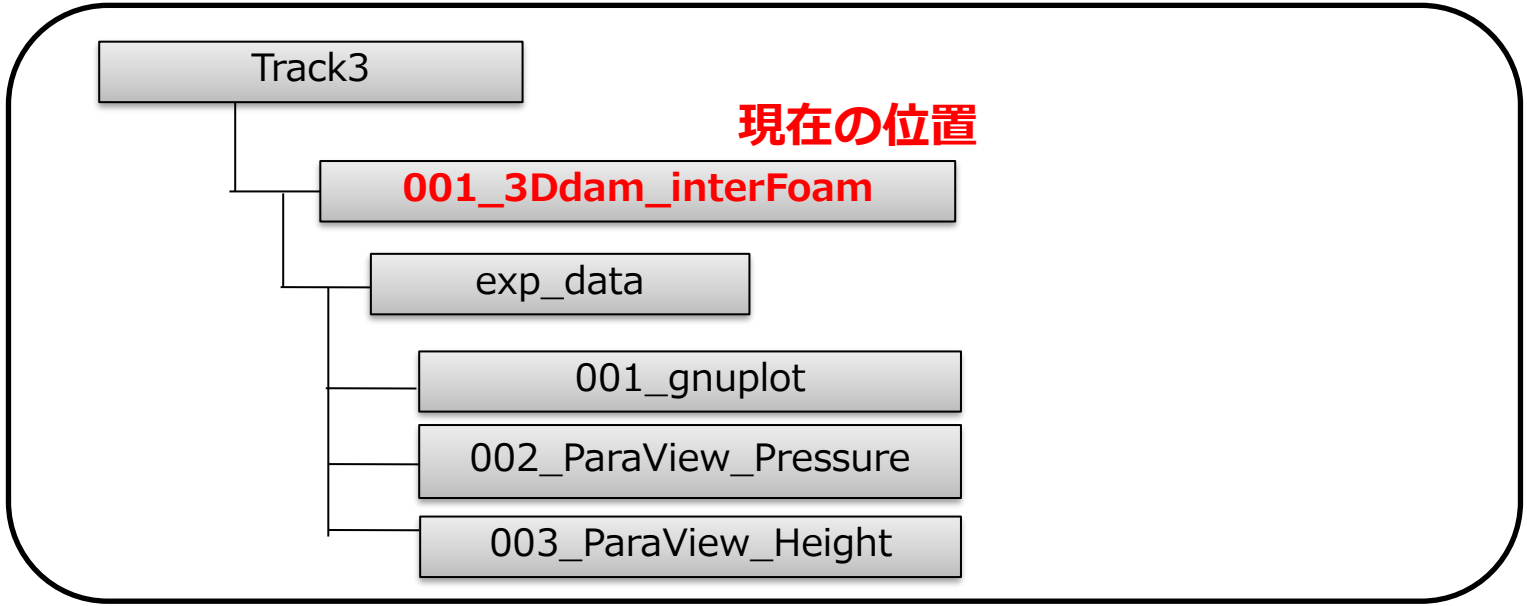




以下のコマンドで「**001_3Ddam_interFoam**」フォルダに移動

フォルダを移動

```
$cd ../../
```

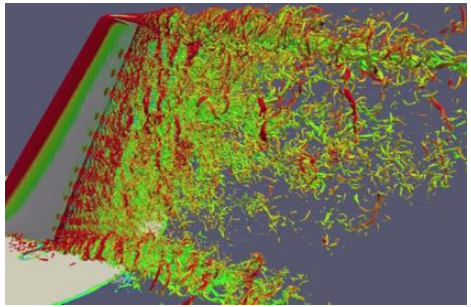


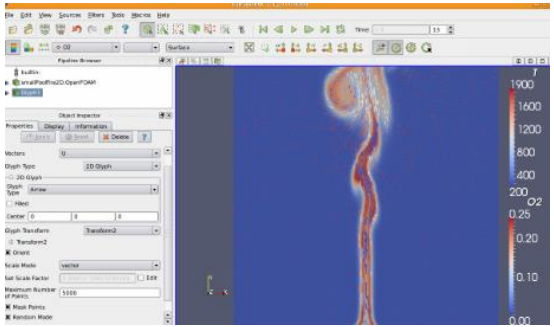
6. ParaViewによる可視化

ParaViewとは

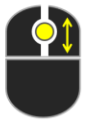
ParaView https://www.paraview.org/

ParaViewは科学技術可視化の並列化と対話操作のための自由に利用できるオープンソースプログラムである。





マウス操作

平行移動	拡大・縮小	回転
マウス中ボタン＋ドラッグ	マウスホイール	マウス左ボタン＋ドラッグ
		

書籍
はじめてのParaView

paraFoam
https://www.openfoam.com/documentation/user-guide/7-post-processing/7.1-parafoam

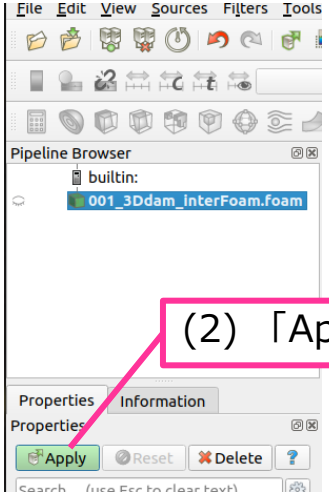
OpenFOAMの結果の読み込み方法(1)

方法1：paraFoamコマンドでOpenFOAMの結果を読み込む

(1)以下のコマンドでParaViewの起動

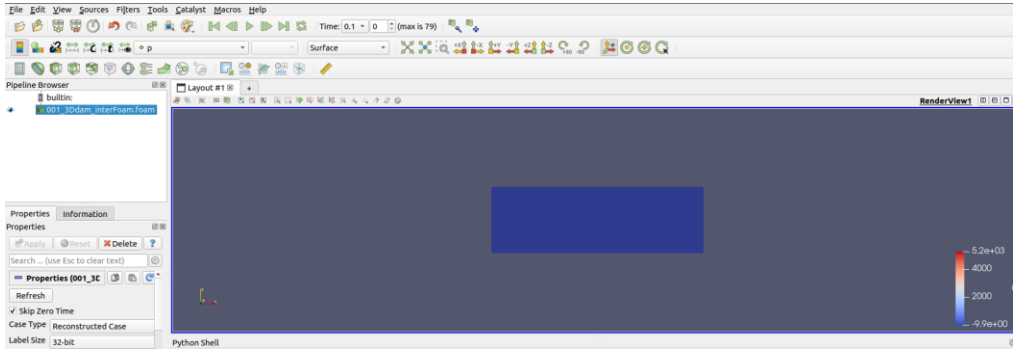
```
$paraFoam
```

```
kamakiri@kamakiri-VirtualBox ~/Desktop/Track3/001_3Ddam_interFoam $ paraFoam
Invalid $PV_PLUGIN_PATH and -plugin-path= not defined
No supplementary ParaView/OpenFOAM reader modules
Using builtin reader: paraFoam -vtk

Created temporary '001_3Ddam_interFoam.foam'
```



(2) 「Apply」

結果の読み込み

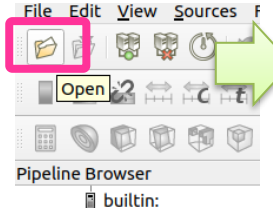


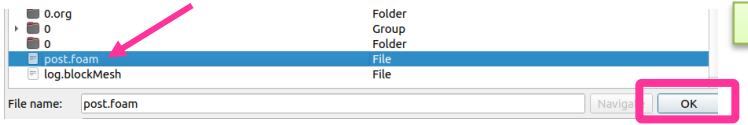
OpenFOAMの結果の読み込み方法(2)

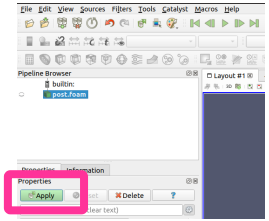
拡張子 “.foam” の空ファイルを読み込む方法

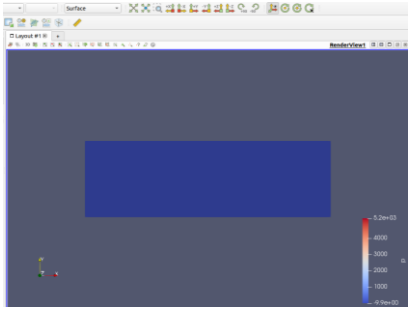
方法2：手動で.foamファイルを読み込む

$paraview	ParaViewの起動



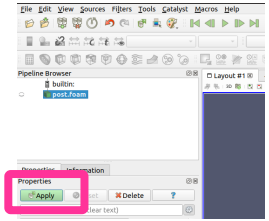


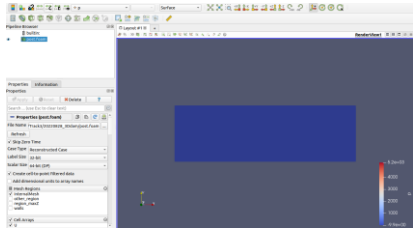




方法3：コマンドで.foamファイルを読み込む

$paraview post.foam	post.foamの読み込み指定でParaViewの起動



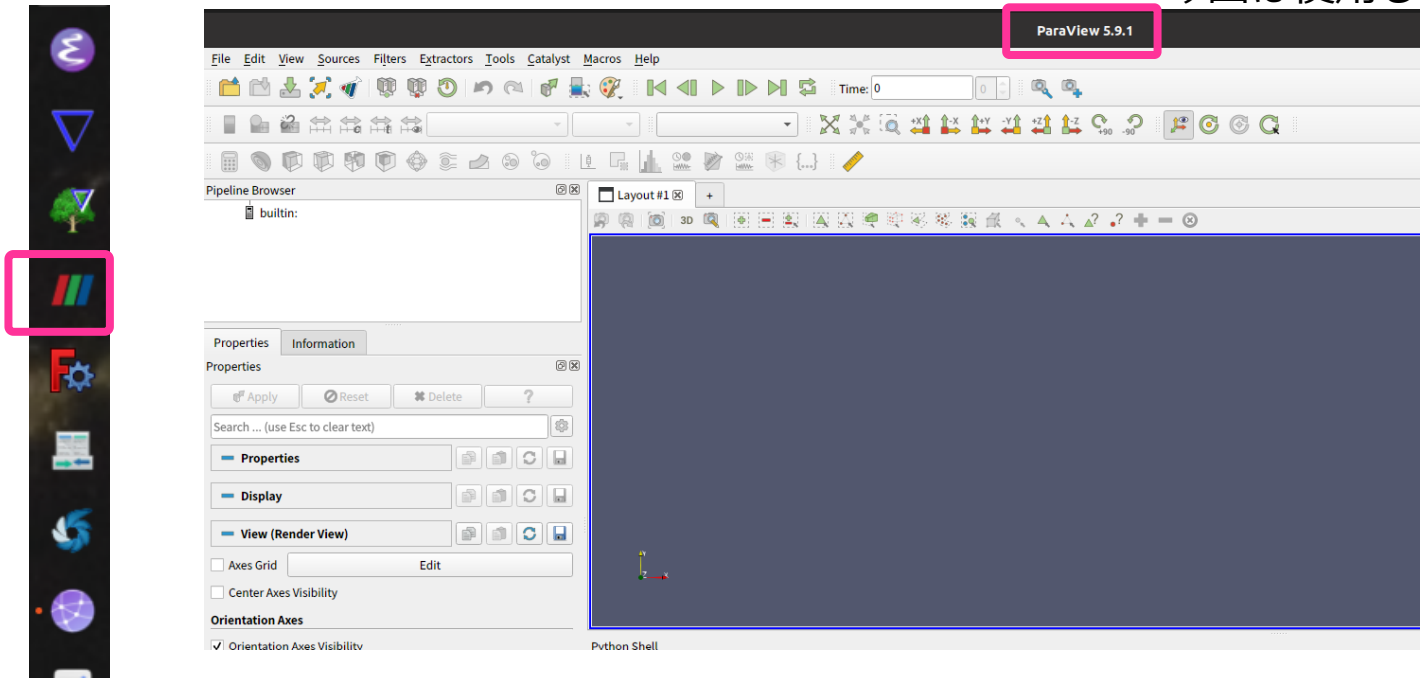


OpenFOAMの結果の読み込み方法(3)

方法4：ParaViewを手動で起動してOpenFOAMの結果を読み込む

拡張子 “.foam” の空ファイルを読み込む方法

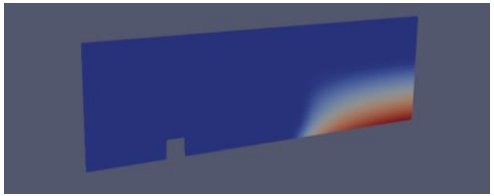
※バージョンが5.9.1
今回は使用しない



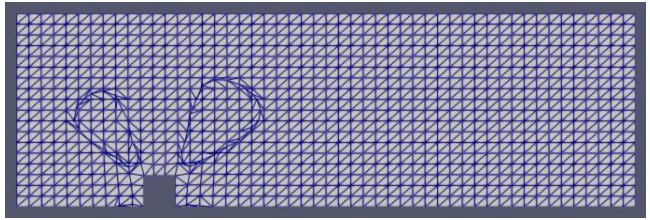
ParaViewによる結果

ParaViewで以下の操作方法を示す

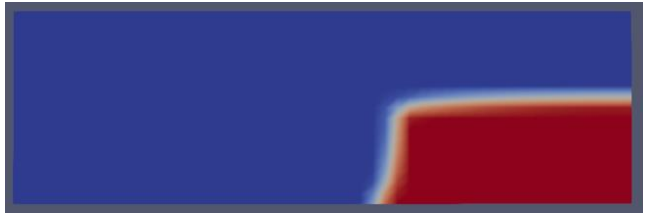
断面表示



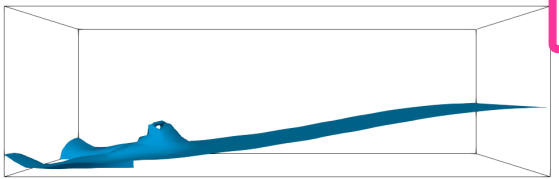
メッシュの確認



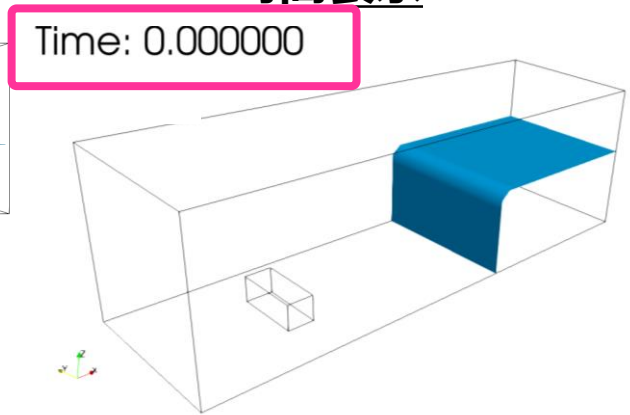
体積分率



コンター図

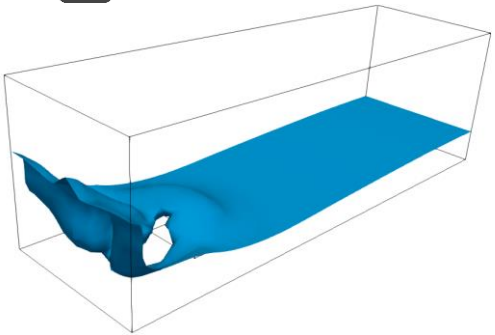


時間表示



アニメーション保存

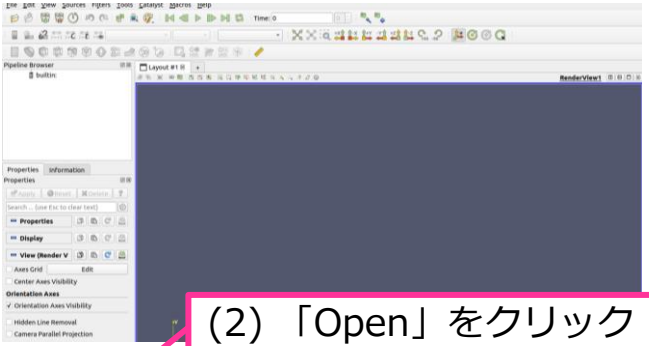
animation/ani.avi



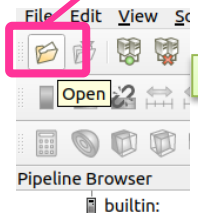
OpenFOAMの結果の読み込み

(1) 以下のコマンドによりParaViewの起動※**方法2：手動で.foamファイルを読み込む**

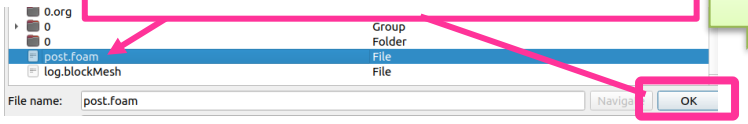
$paraview	ParaViewの起動

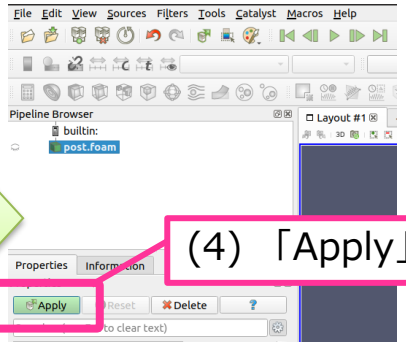


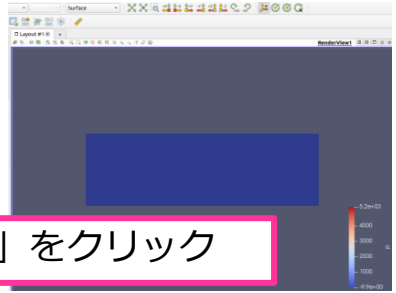
(2) 「Open」をクリック



(3) 「post.foam」を選択して「OK」をクリック

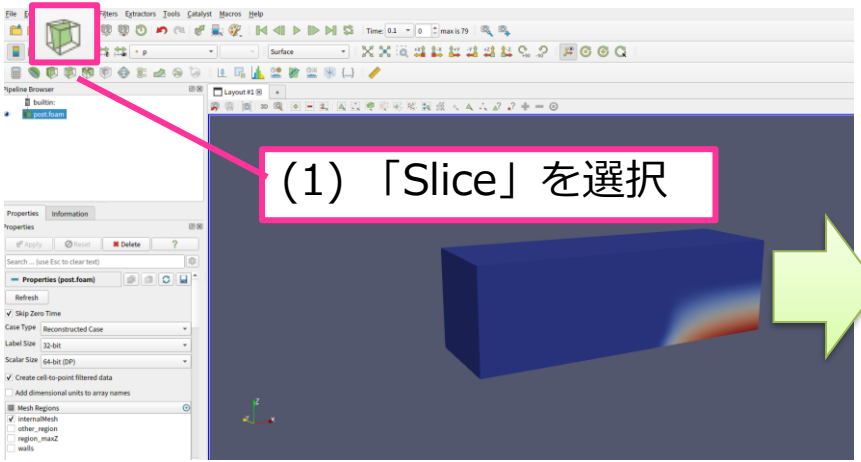


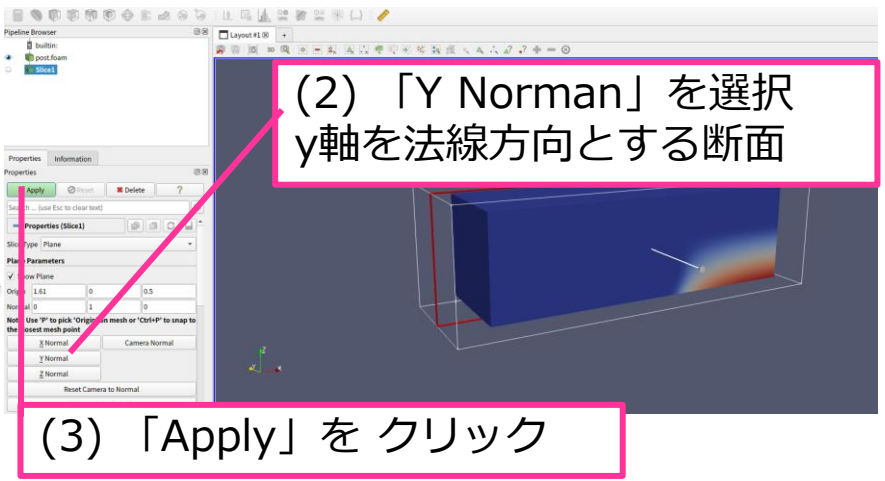


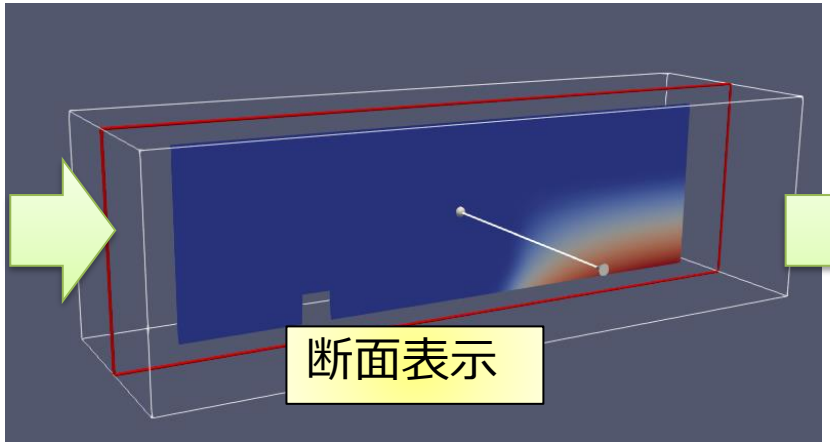


(4) 「Apply」をクリック

断面表示


(1) 「Slice」を選択

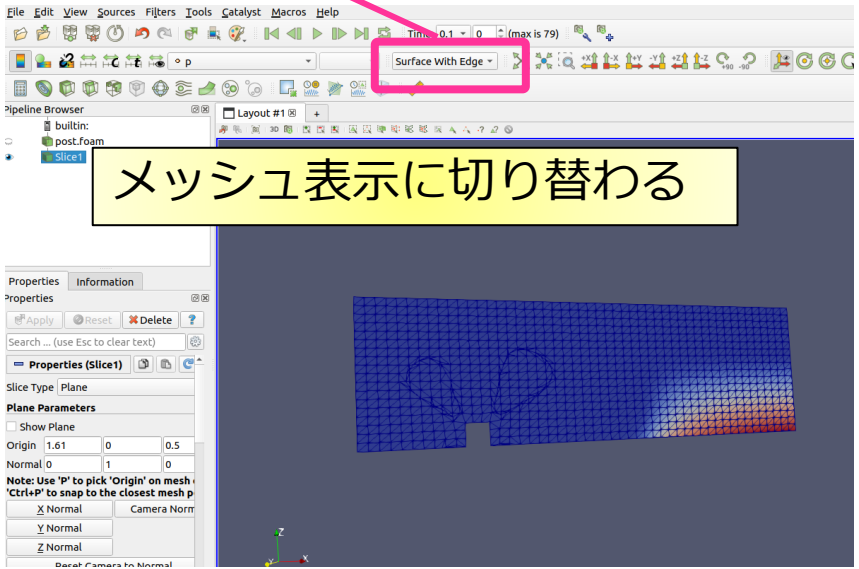

(2) 「Y Norman」を選択
y軸を法線方向とする断面
(3) 「Apply」を クリック


断面表示

(4) 「Show Plane」のチェックを
外すとPlaneが消える

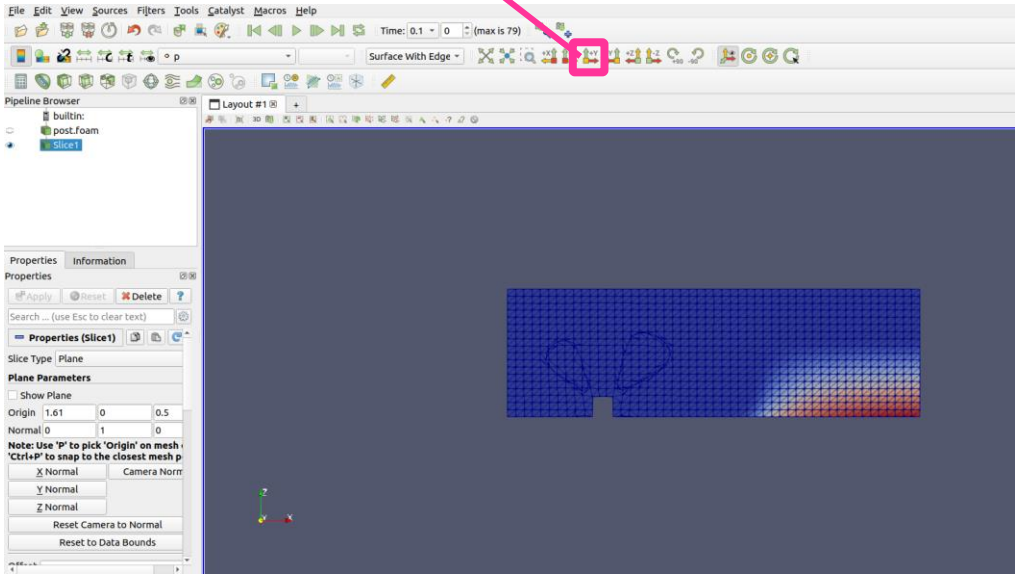
メッシュの確認

(1) 「Surface With Edges」を選択

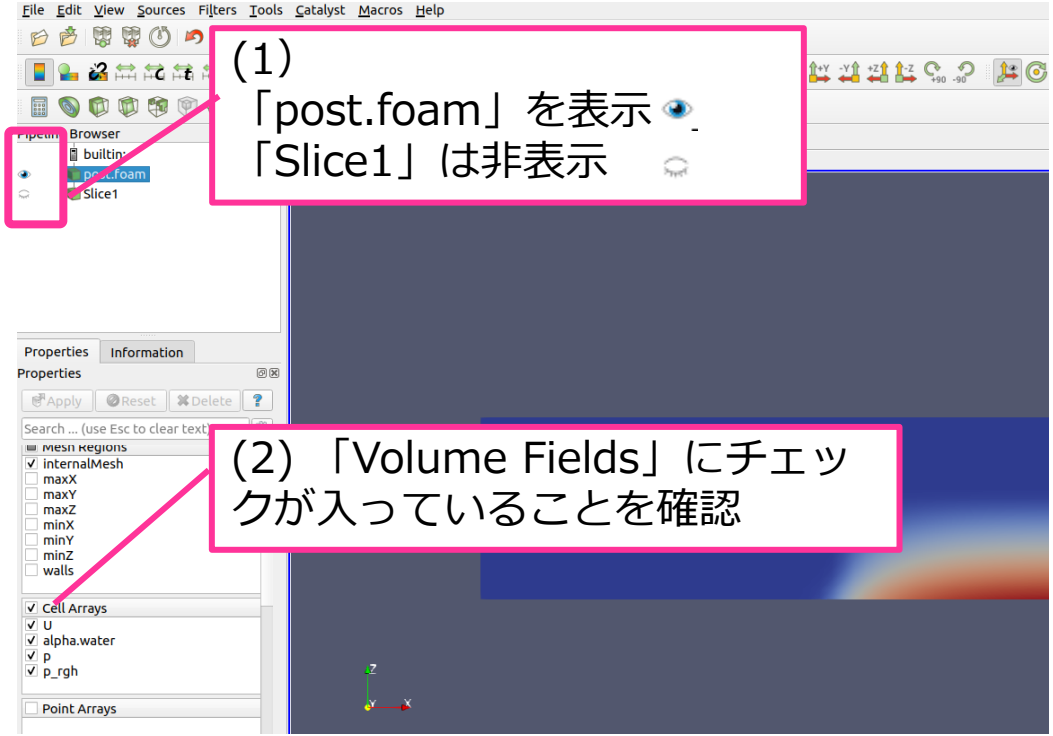


メッシュ表示に切り替わる

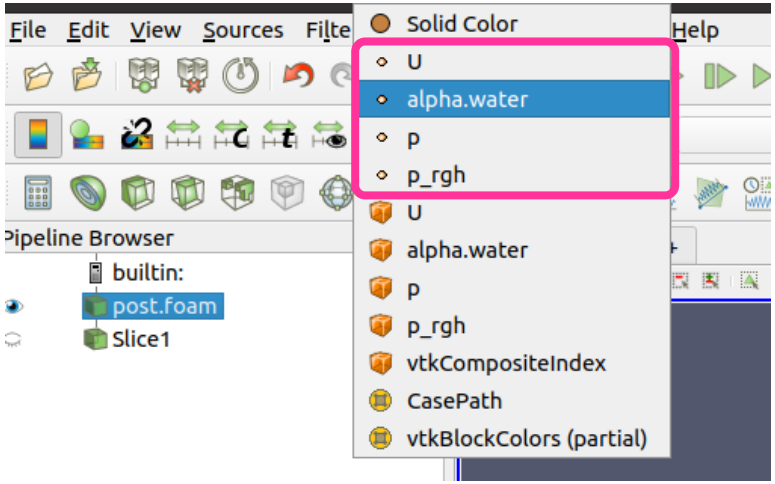
(2)「Set view direction to +Y」
x-z平面表示にする



体積分率の分布

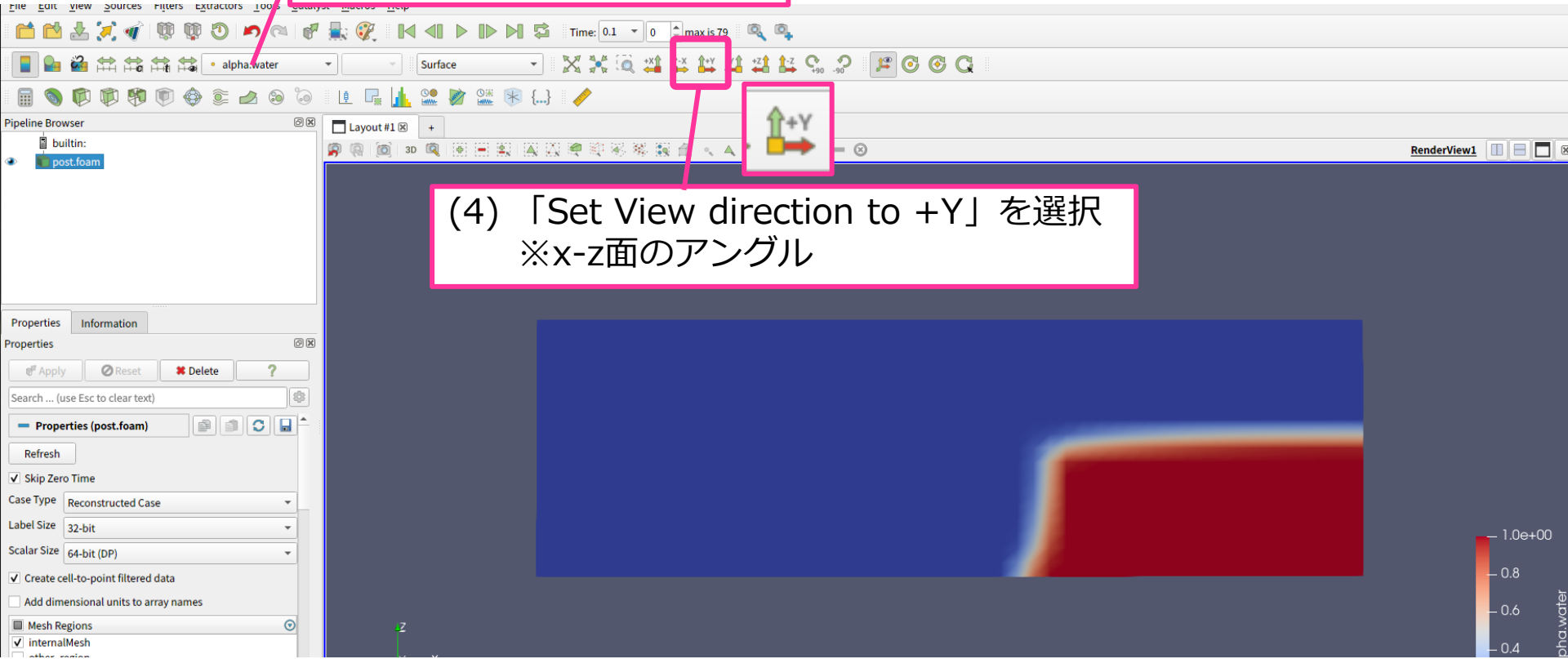


チェックを入れた物理量が表示



体積分率の分布

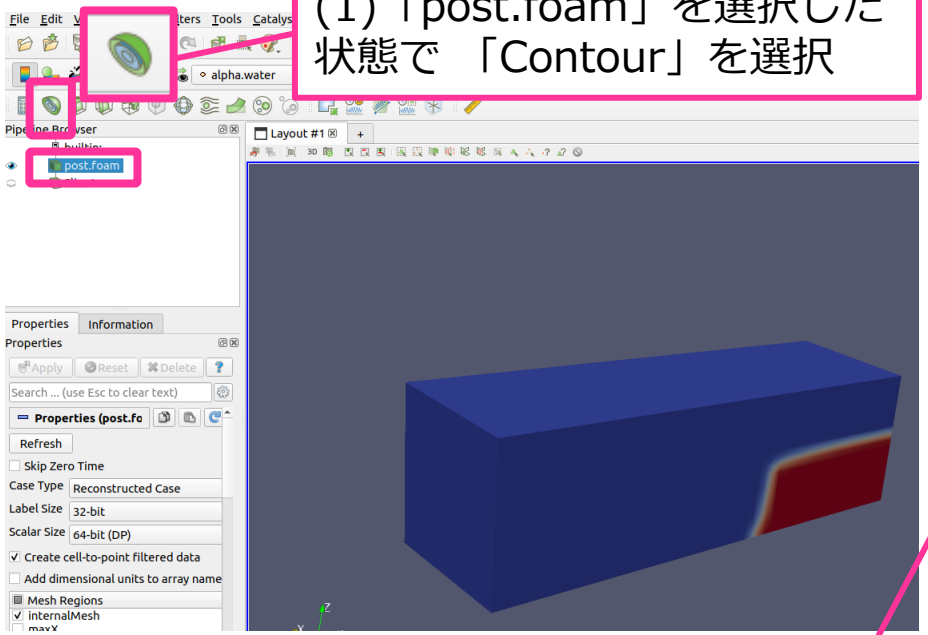
(3) 「alpha.water」を選択

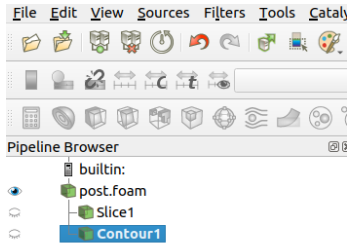


(4) 「Set View direction to +Y」を選択
※x-z面のアングル

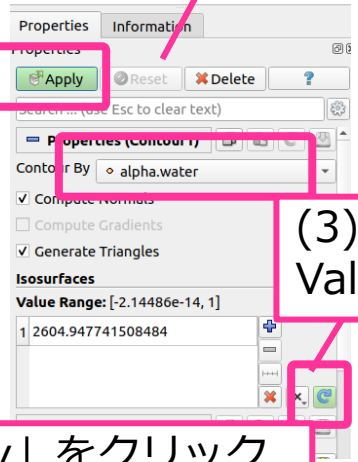
コンター

(1)「post.foam」を選択した状態で 「Contour」を選択

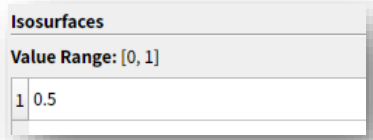




(2) 「alpha.water」を選択

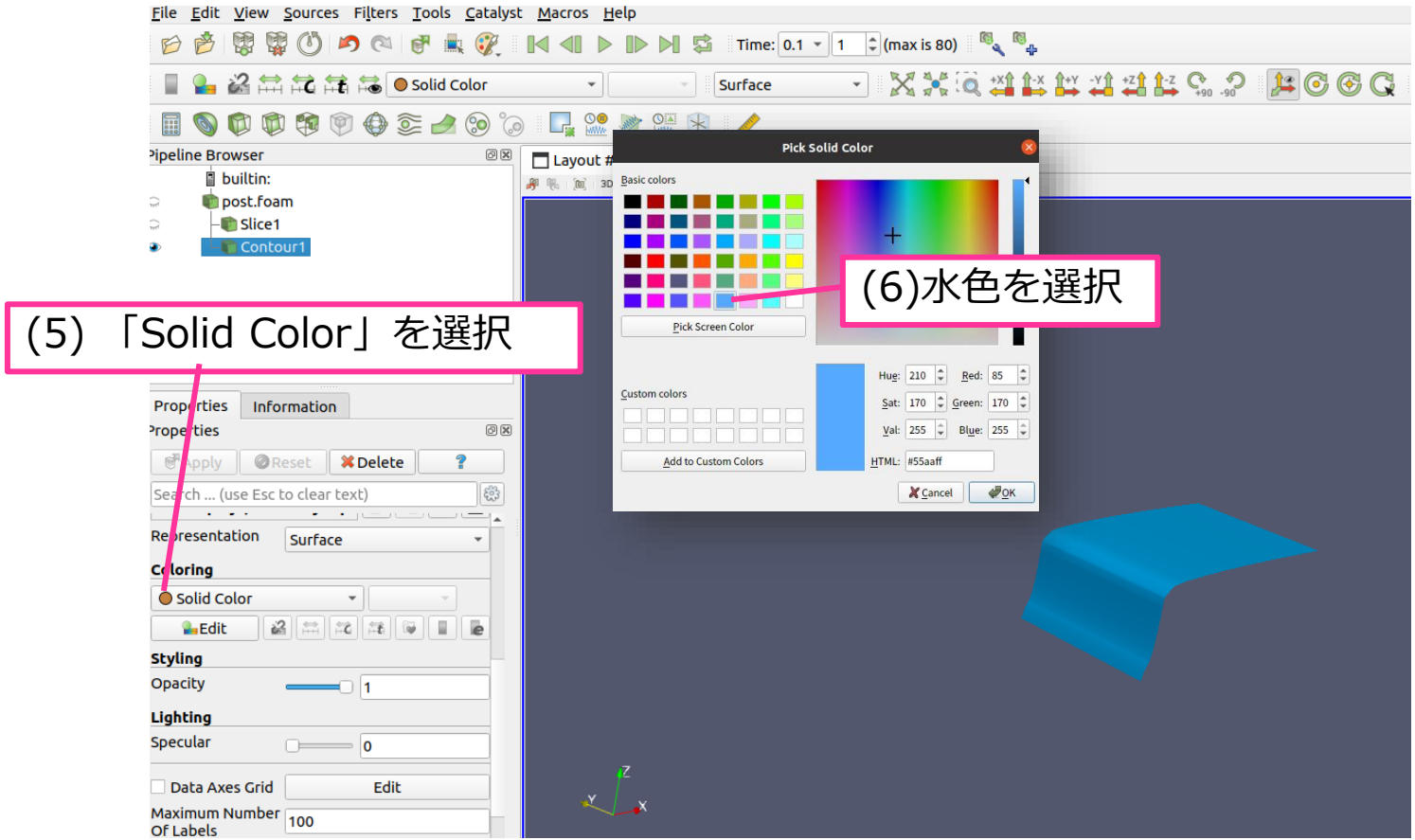


(3)更新する
Value Rangeの値が0.5になる

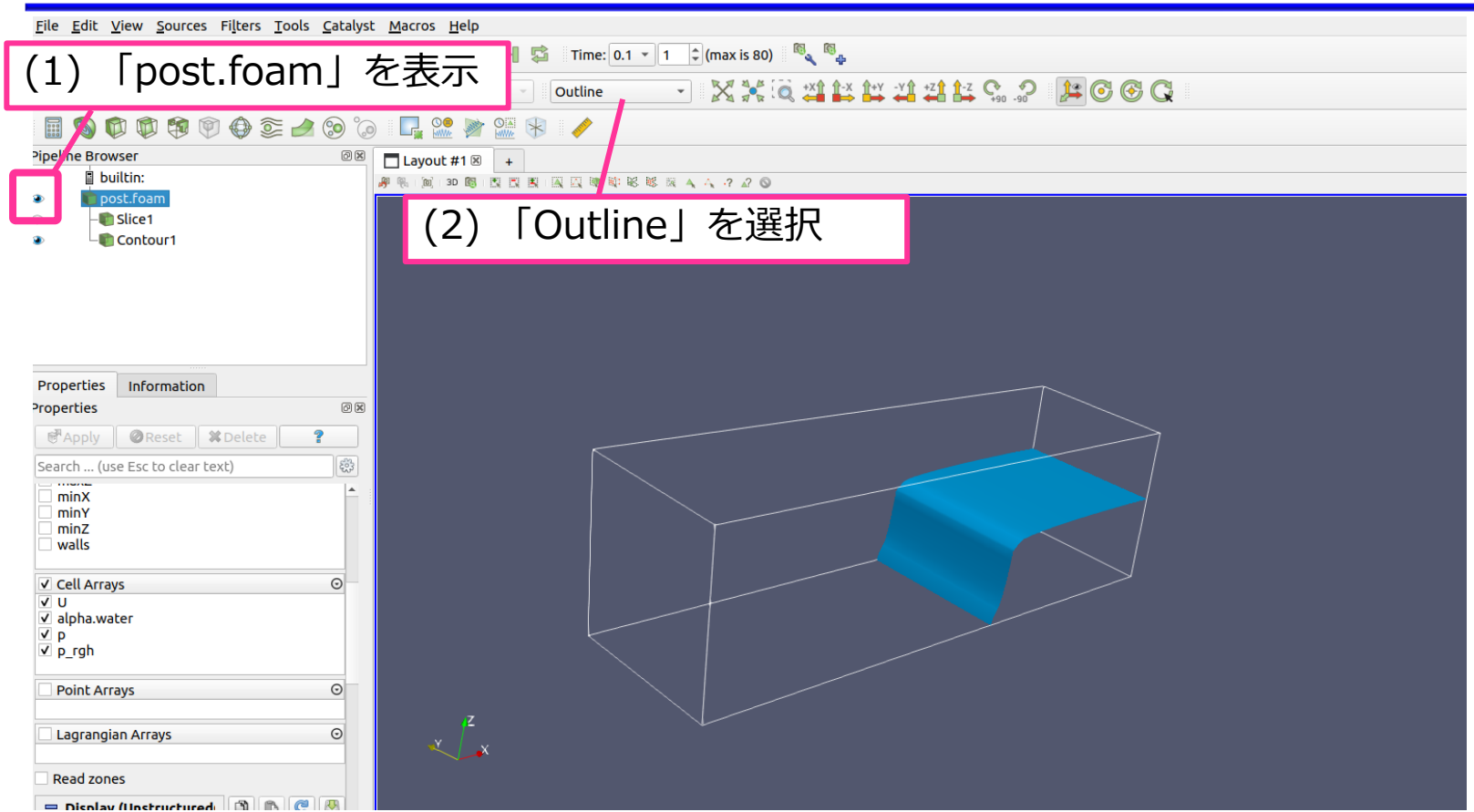


(4) 「Apply」をクリック

コンター

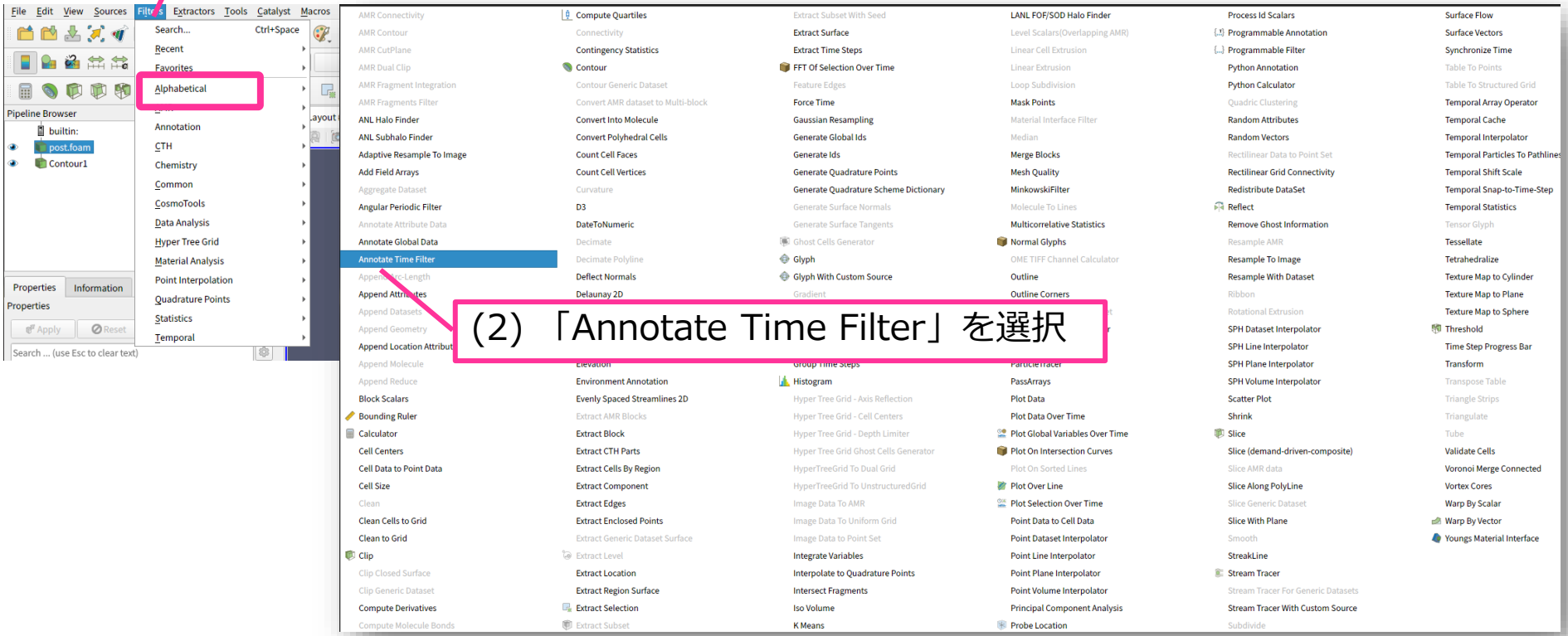

File Edit View Sources Filters Tools Catalyst Macros Help
Time: 0.1 1 (max is 80)
Solid Color
Surface
Pipeline Browser
builtin:
post.foam
Slice1
Contour1
Layout #
Pick Solid Color
Basic colors
Pick Screen Color
Custom colors
Add to Custom Colors
Hue: 210
Sat: 170
Val: 255
Red: 85
Green: 170
Blue: 255
HTML: #55aaff
Cancel
OK
(6)水色を選択
(5) 「Solid Color」を選択
Properties
Information
Apply
Reset
Delete
Search ... (use Esc to clear text)
Representation
Surface
Coloring
Solid Color
Edit
Styling
Opacity
1
Lighting
Specular
0
Data Axes Grid
Edit
Maximum Number Of Labels
100

計算領域の表示


File Edit View Sources Filters Tools Catalyst Macros Help
(1) 「post.foam」を表示
Time: 0.1 1 (max is 80)
Outline
Pipeline Browser
builtin:
post.foam
Slice1
Contour1
Layout #1
(2) 「Outline」を選択
Properties Information
Properties
Apply Reset Delete
Search ... (use Esc to clear text)
minX
minY
minZ
walls
Cell Arrays
U
alpha.water
p
p_rgh
Point Arrays
Lagrangian Arrays
Read zones

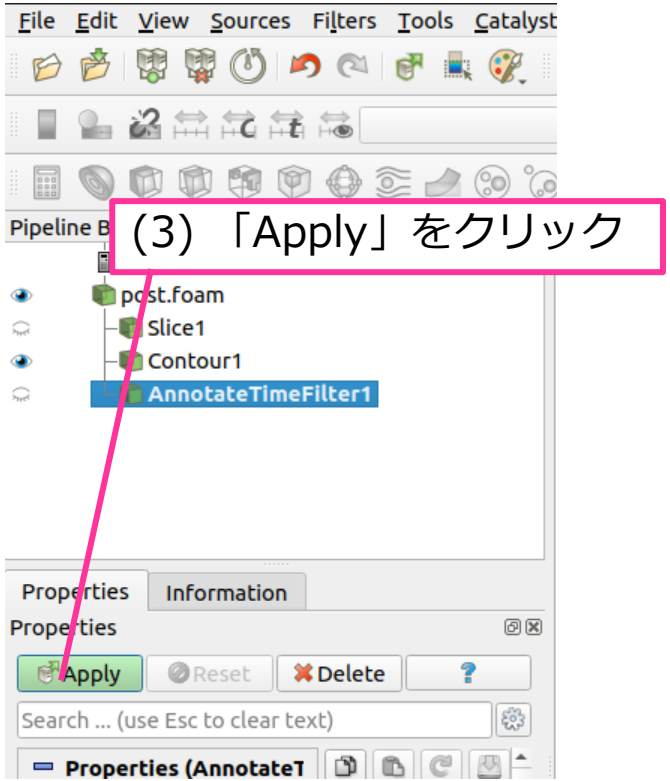
時間表示

(1) 「Filter」 > 「Alphabetical」

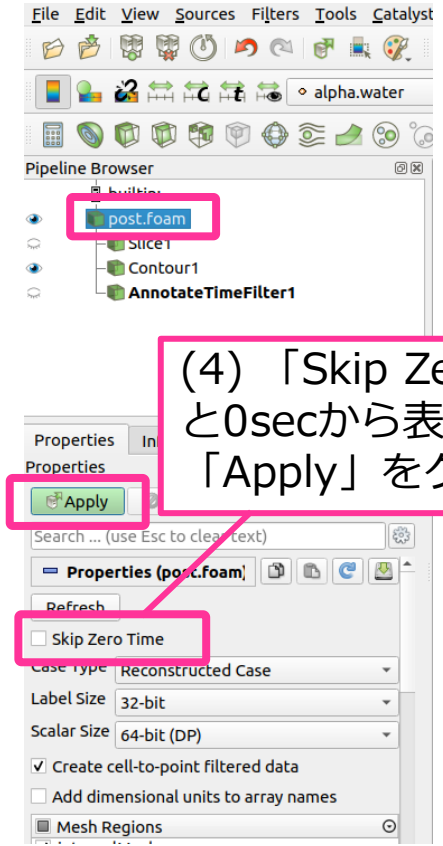


(2) 「Annotate Time Filter」を選択

時間表示

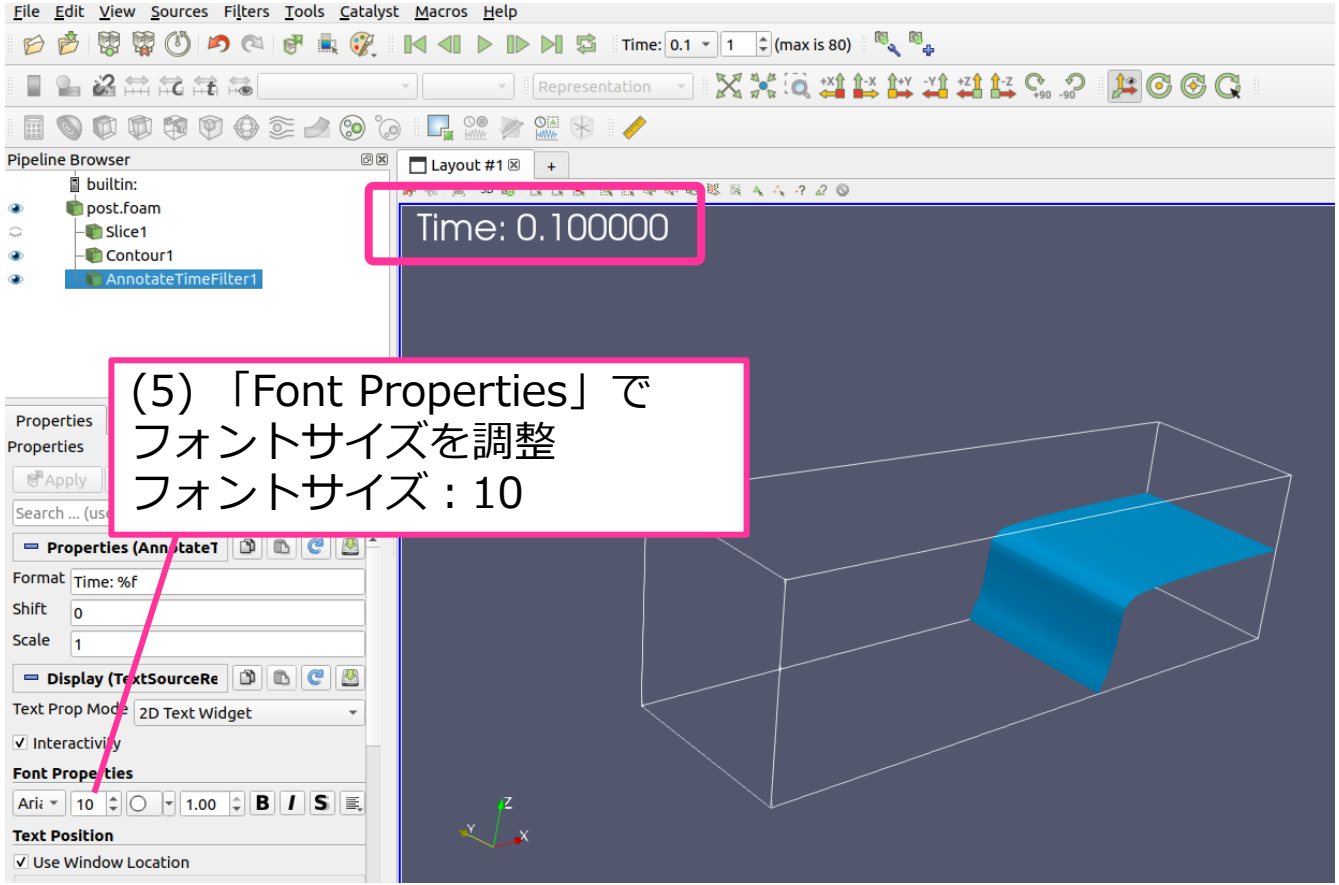


(3) 「Apply」をクリック



(4) 「Skip Zero Time」のチェックを外すと0secから表示される
「Apply」をクリック

時間表示

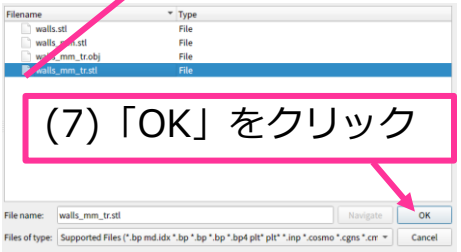


(5) 「Font Properties」で
フォントサイズを調整
フォントサイズ：10

時間表示

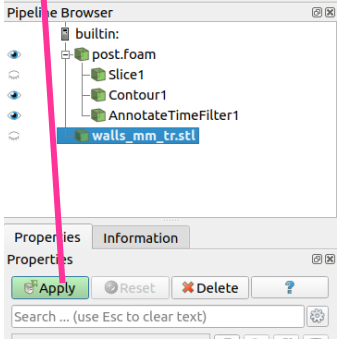
構造物も表示したい

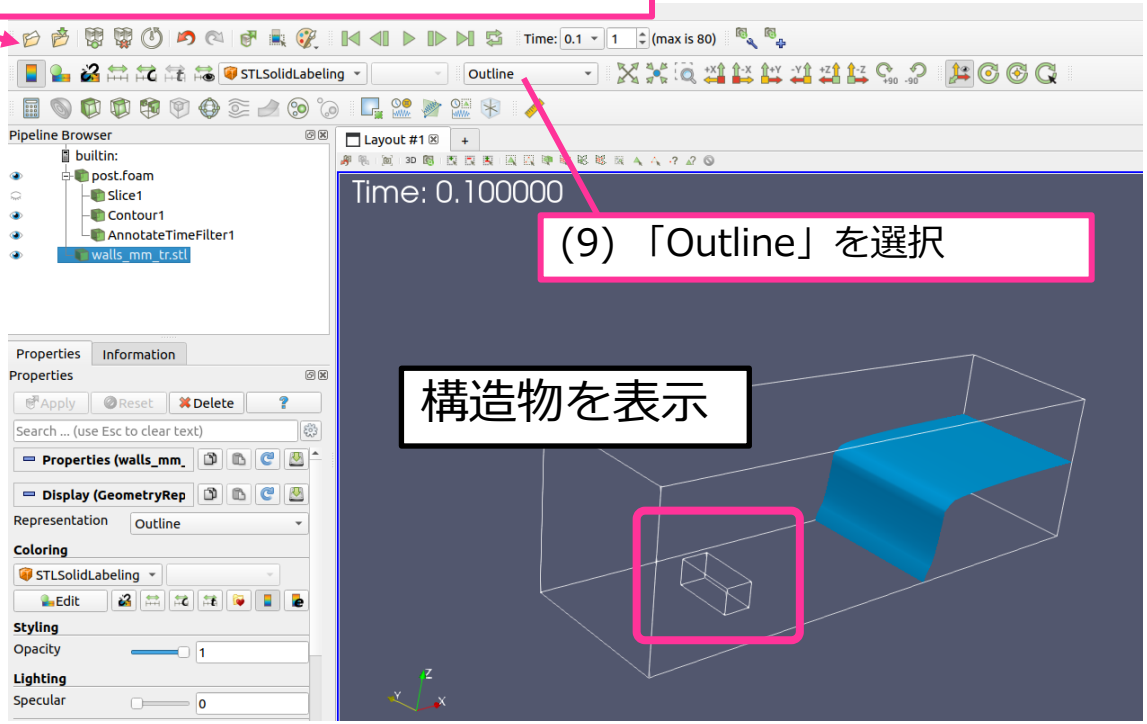
(6) 「Open」をクリック
constant/triSurface内の「walls_mm_tr.stl」を選択



(7)「OK」をクリック

(8) 「Apply」をクリック



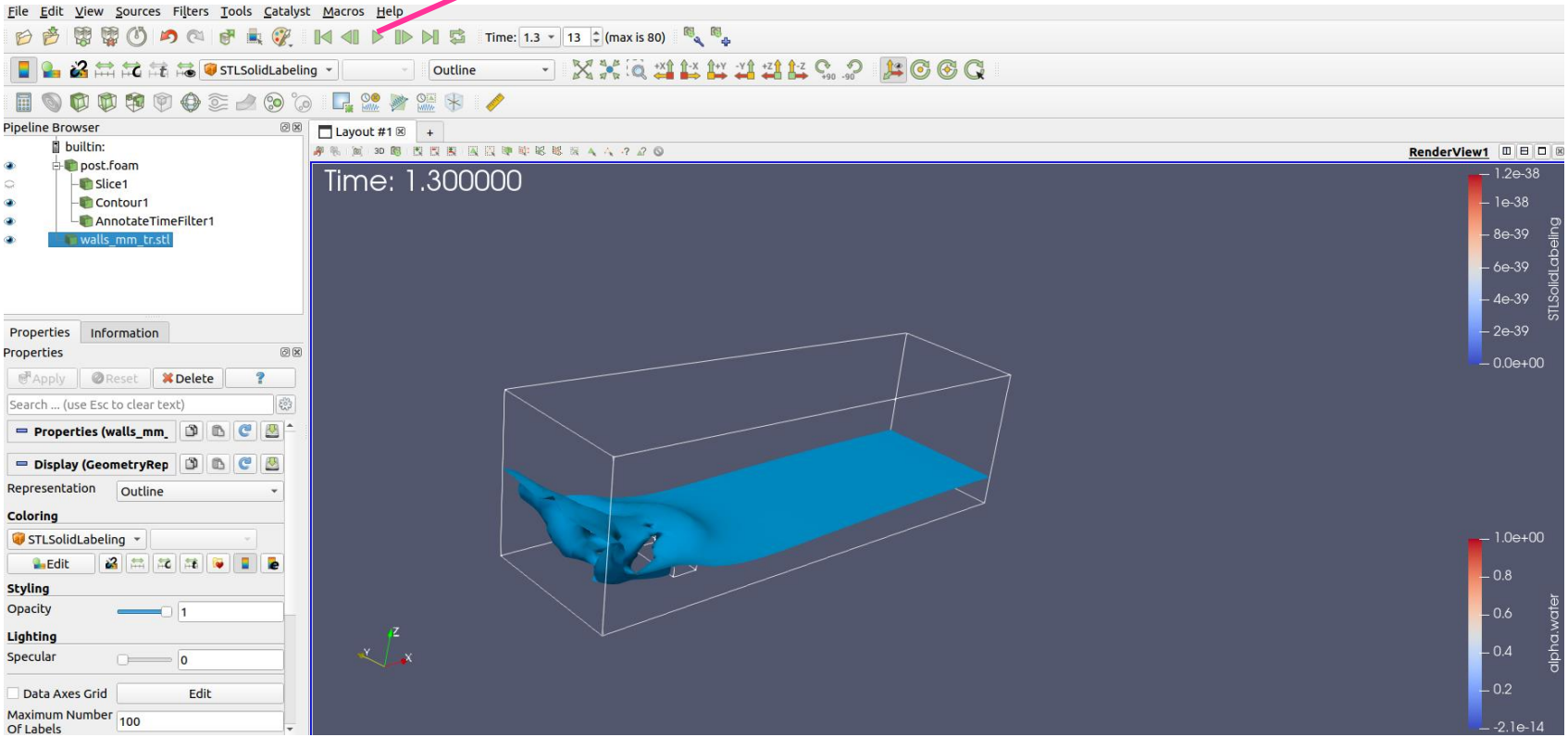


(9) 「Outline」を選択

構造物を表示

時間表示

(9) 「Play」をクリックして時間を進める



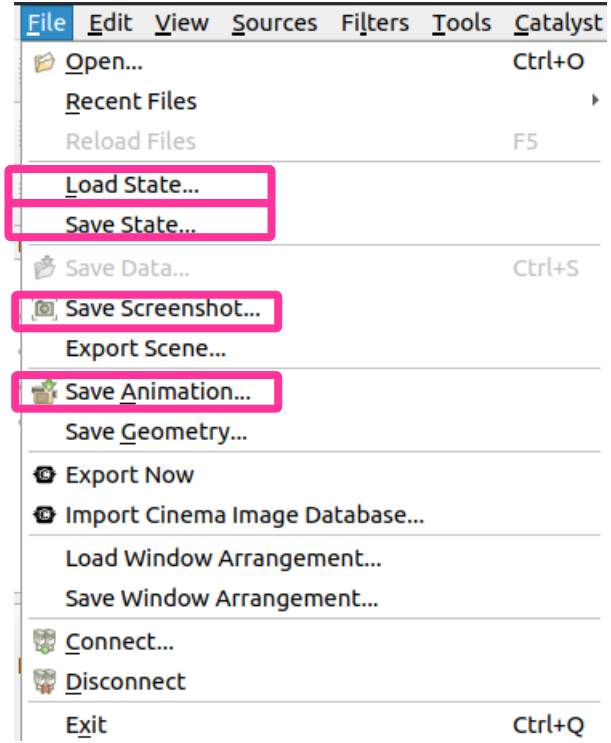
データの保存

可視化の各種設定の保存・読み込み

- Load State(読み込み)
- Save State(操作の保存)

作成した画像・アニメーションの保存

- Save Screenshot(画面の画像保存)
- Save Animation(アニメーション保存)

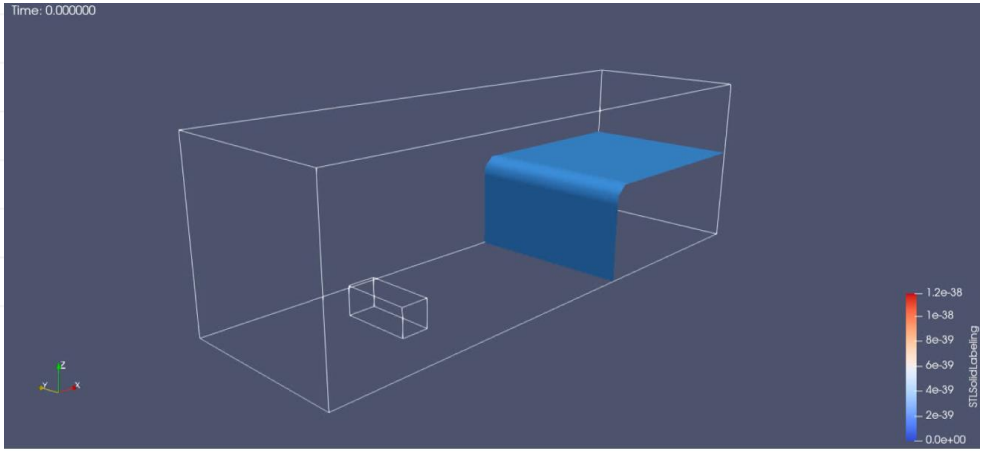


アニメーション保存

(1)「animation」フォルダを新規作成

次頁で作成

animation/ani.avi

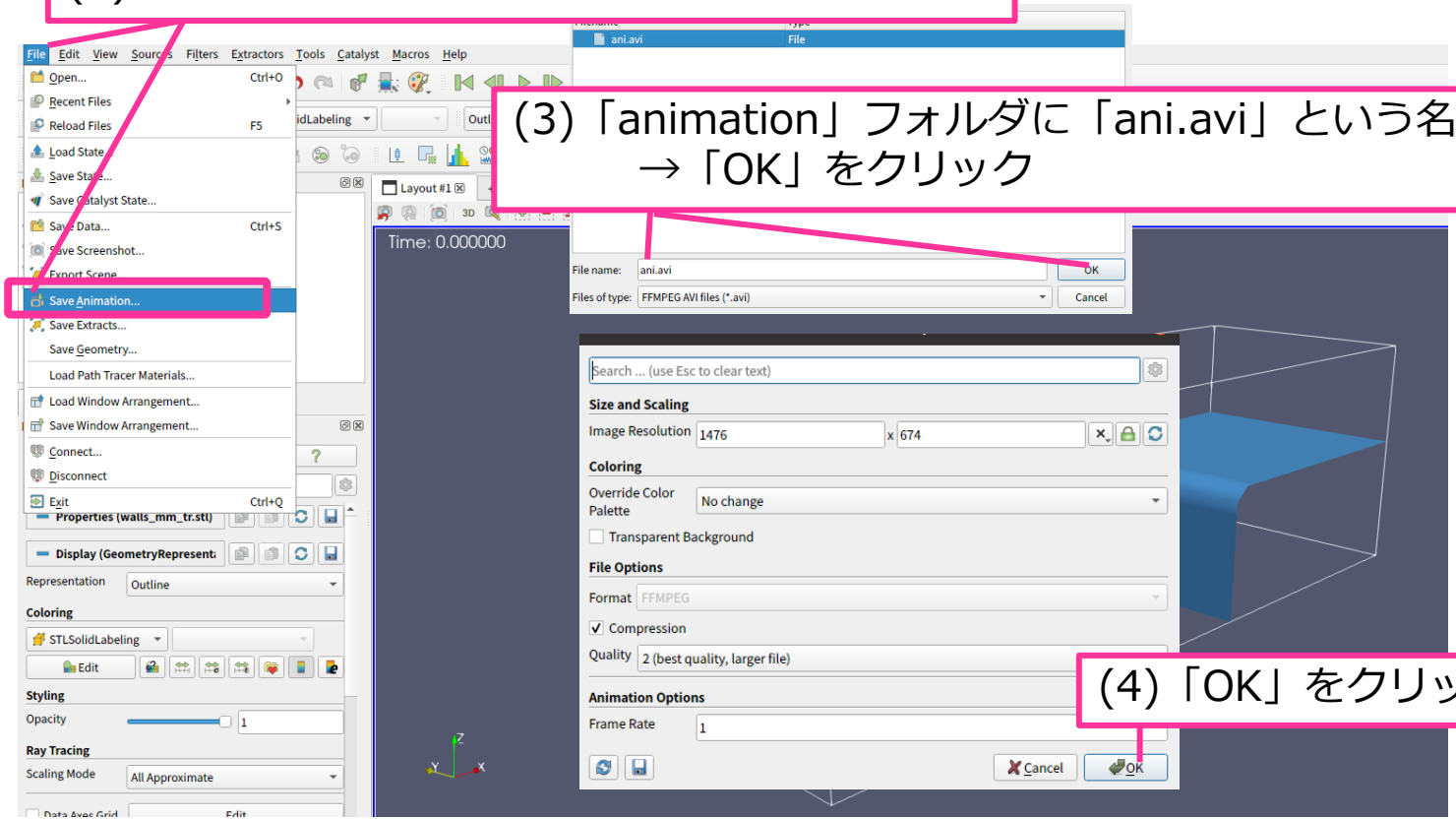


アニメーション保存

(2) 「File」>「Save Animation」をクリック

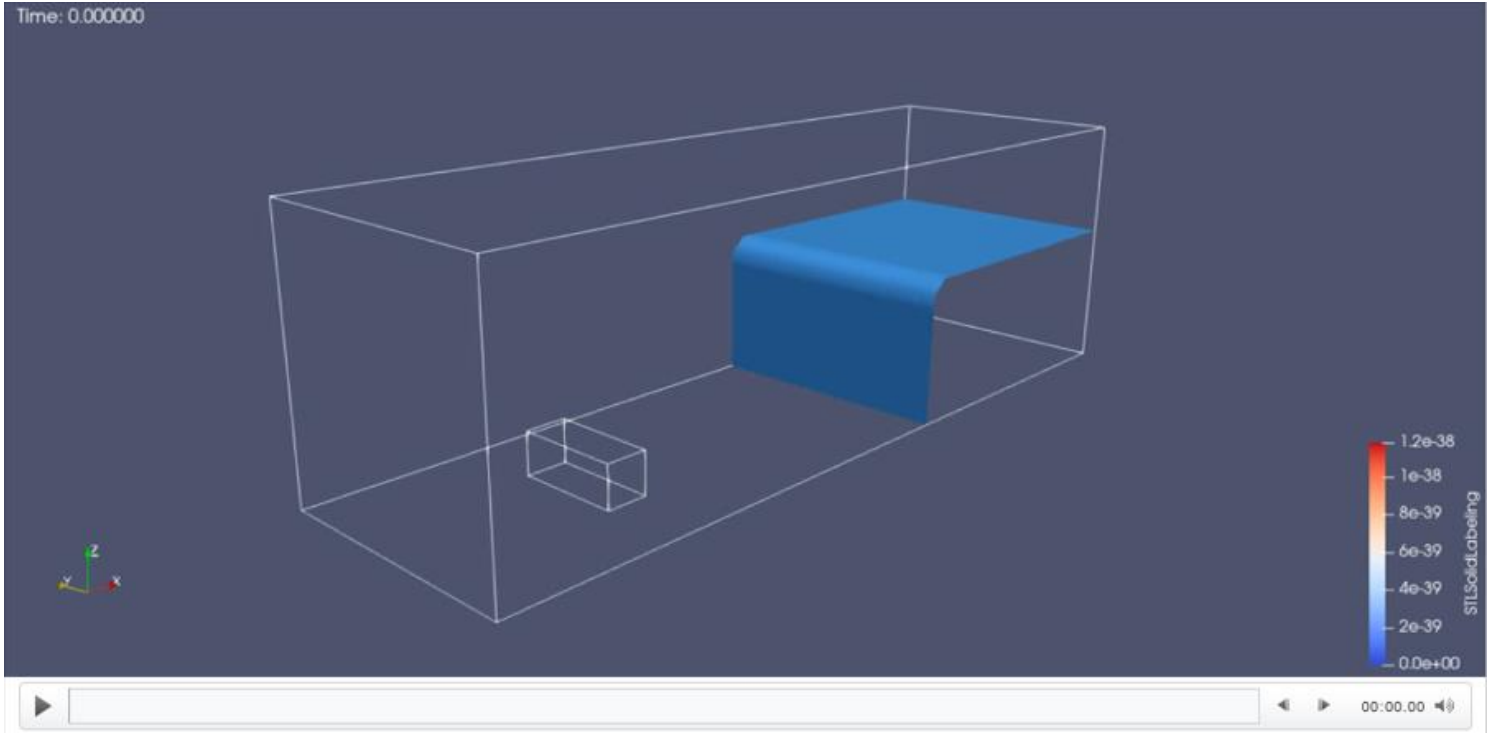
(3)「animation」フォルダに「ani.avi」という名前で保存
→「OK」をクリック

(4)「OK」をクリック



アニメーション保存

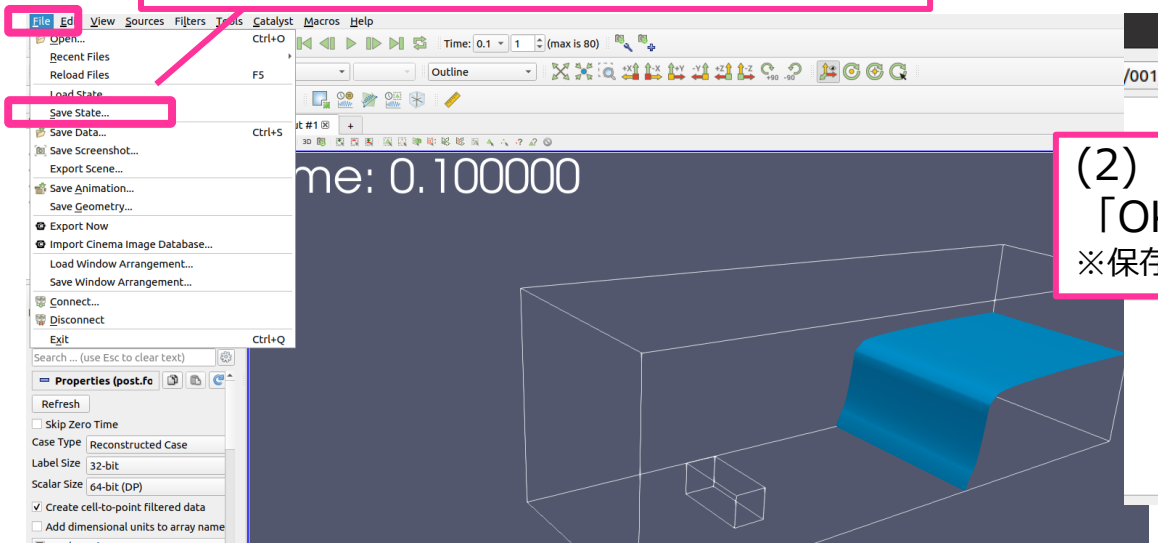
exp_data/animation/ani.avi

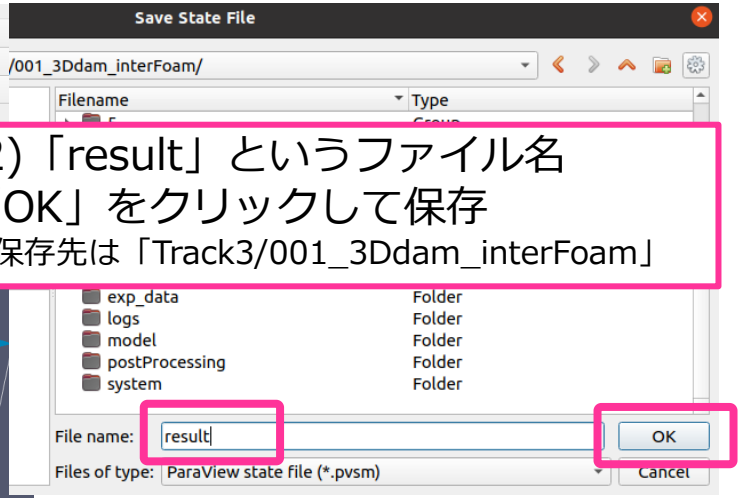


操作の保存

操作の保存ができる

(1)「File」>「Save State」を選択





(2)「result」というファイル名
「OK」をクリックして保存
※保存先は「Track3/001_3Ddam_interFoam」

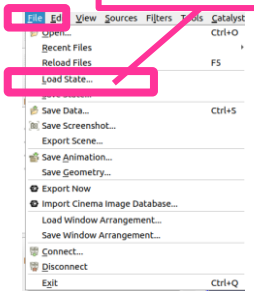
操作の読み込み

(1) 以下のコマンドによりParaViewの起動

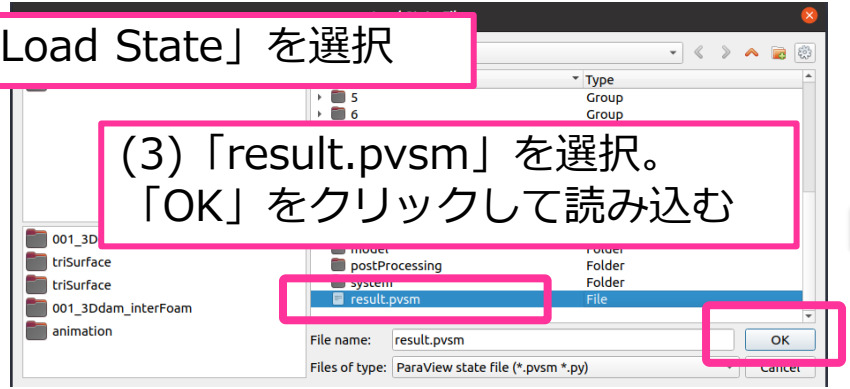
```
$paraview
```

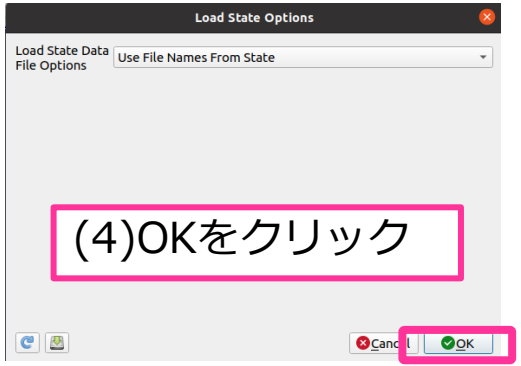
ParaViewの起動

(2)「File」>「Load State」を選択

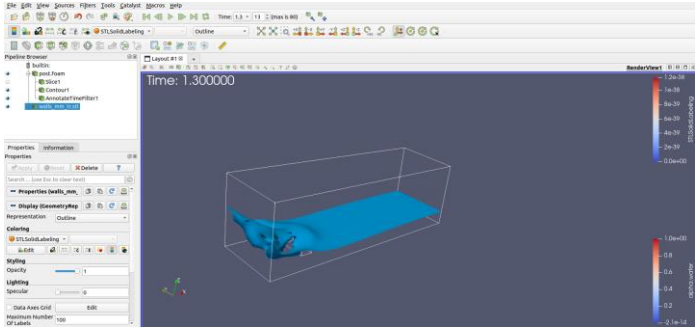


(3)「result.pvsm」を選択。
「OK」をクリックして読み込む





(4)OKをクリック

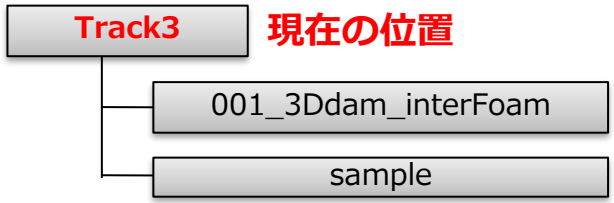


読み込み完了

7. 実習 モデルの修正

damBreakのtutorial (interIsoFoam)

interIsoFoamソルバを使用する

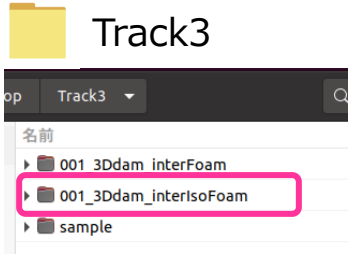


(1)以下のコマンドでTrack3に移動する

```
$cd ..
```

(2)以下のコマンドで、
$FOAM_TUTORIALS/multiphase/interIsoFoam/damBreakを
001_3Ddam_interIsoFoamという名前でコピーする

```
$cp -r  $FOAM_TUTORIALS/multiphase/interIsoFoam/damBreak 001_3Ddam_interIsoFoam
```

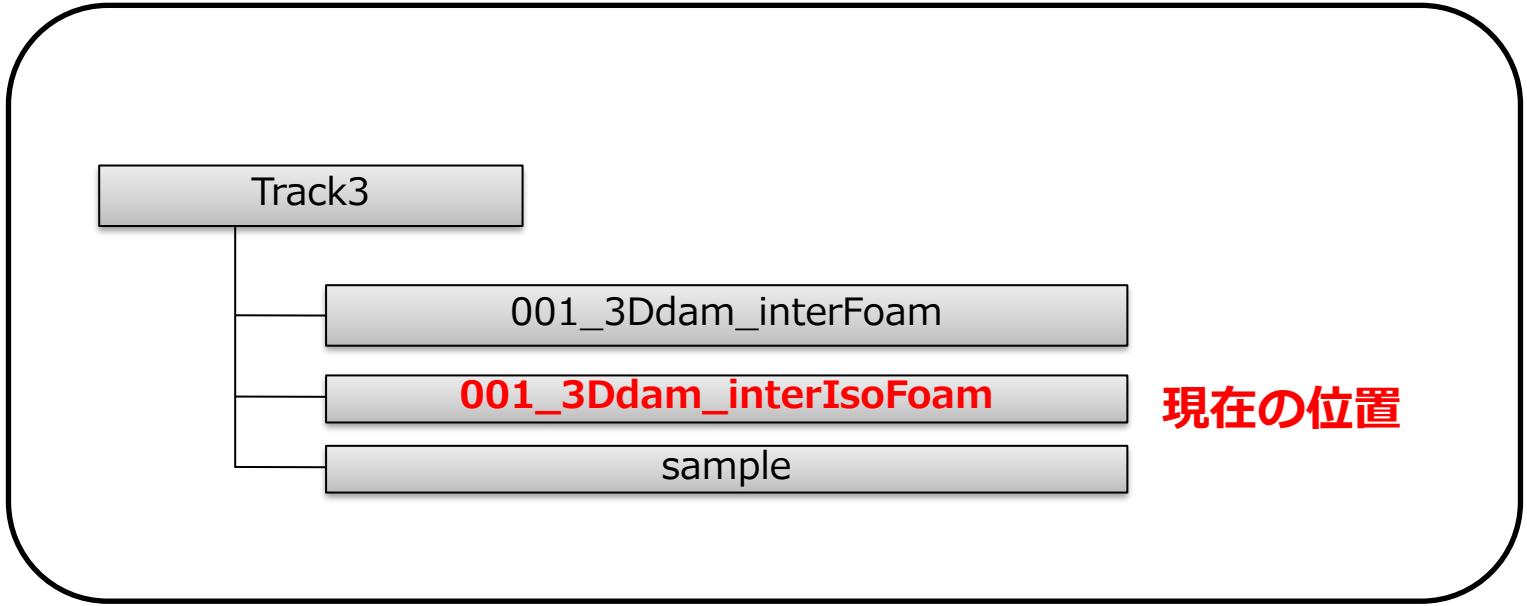


interIsoFoamのチュートリアルがコピーされた

以下のコマンドで「**001_3Ddam_interIsoFoam**」フォルダに移動

(3)フォルダを移動

```
$cd 001_3Ddam_interIsoFoam
```



ファイルの変更

(4) 001_3Ddam_interFoamから以下のフォルダとファイルをコピー

- 0.orig
- constant
- system
  ※fvSchemesとfvSolution以外をコピー
- exp_data
- Allrun
- Allclean

001_3Ddam_interIsoFoam
へコピー（上書き）

controlDict

(5)ファイル内変更する

```
application     interIsoFoam;

startFrom       startTime;

startTime       0;

stopAt          endTime;

endTime         8;

deltaT          0.001;

writeControl    adjustable;

writeInterval   0.1;

purgeWrite      0;

writeFormat     ascii;

writePrecision  6;

writeCompression off;

timeFormat      general;

timePrecision   6;

runTimeModifiable yes;

adjustTimeStep  yes;

maxCo           0.5;

maxAlphaCo      0.5;

maxDeltaT       1;

functions // 関数定義
{
    #includeFunc probes; // 壁面での圧力時刻歴
    #includeFunc interfaceHeight; // 水面高さの時刻歴

    #includeFunc residuals; // 残差
}
//
*****************************************************
***************** //
```

fvSchemes

interFoam

```
ddtSchemes
{
    default         Euler;
}

gradSchemes
{
    default         Gauss linear;
}

divSchemes
{
    div(rhoPhi,U)   Gauss upwind;
    div(phi,alpha)  Gauss vanLeer;
    div(phirb,alpha) Gauss linear;
    div(((rho*nuEff)*dev2(T(grad(U))))) Gauss linear;
}

laplacianSchemes
{
    default         Gauss linear corrected;
}

interpolationSchemes
{
    default         linear;
}

snGradSchemes
{
    default         corrected;
}
wallDist
{
    method meshWave;
}
```

interIsoFoam

```
ddtSchemes
{
    default          Euler;
}

gradSchemes
{
    default          Gauss linear;
}

divSchemes
{
    div(rhoPhi,U)    Gauss limitedLinearV 1;  線形補間にTVD制限をつけたもの（ベクトル版）
    div(phi,alpha)   Gauss vanLeer;
    div(phirb,alpha) Gauss interfaceCompression;
    div(((rho*nuEff)*dev2(T(grad(U))))) Gauss linear;
}

laplacianSchemes
{
    default          Gauss linear corrected;
}

interpolationSchemes
{
    default          linear;
}

snGradSchemes
{
    default          corrected;
}

fluxRequired    流束を計算させるかどうかの指定でソルバーが要求したときに設定
{
    default          no;
    p_rgh;
    pcorr;
    alpha.water;
}
```

fvSolution

interFoam

```
solvers
{
    "alpha.water.*"
    {
        nAlphaCorr      2;
        nAlphaSubCycles 1;
        cAlpha          1;

        MULESCorr       yes;
        nLimiterIter    5;

        solver          smoothSolver;
        smoother        symGaussSeidel;
        tolerance       1e-8;
        relTol          0;
    }

    "pcorr.*"
    {
        solver          PCG;
        preconditioner  DIC;
        tolerance       1e-5;
        relTol          0;
    }

    p_rgh
    {
        solver          PCG;
        preconditioner  DIC;
        tolerance       1e-07;
        relTol          0.05;
    }

    p_rghFinal
    {
        $p_rgh;
        relTol          0;
    }

    U
    {
        solver          smoothSolver;
        smoother        symGaussSeidel;
        tolerance       1e-06;
        relTol          0;
    }
}
PIMPLE
{
    momentumPredictor   no;
    nOuterCorrectors    1;
    nCorrectors         3;
    nNonOrthogonalCorrectors 0;
}

relaxationFactors
{
    equations
    {
        ".*" 1;
    }
}
```

interIsoFoam

```
solvers
{
    "alpha.water.*"
    {
        isoFaceTol      1e-6;
        surfCellTol     1e-6;
        nAlphaBounds    3;
        snapTol         1e-12;
        clip            true;
        reconstructionScheme isoAlpha;
        writeFields     true;

        nAlphaSubCycles 1;
        cAlpha          1; // Note: cAlpha is not used by isoAdvector but must
                           // be specified because interfacePropertes object
                           // reads it during construction.
    }

    "pcorr.*"
    {
        solver          PCG;
        preconditioner  DIC;
        tolerance       1e-10;
        relTol          0;
    }

    p_rgh
    {
        solver          GAMG;
        smoother        DICGaussSeidel;
        tolerance       1e-09;
        relTol          0.05;
    }

    p_rghFinal
    {
        $p_rgh;
        tolerance       1e-09;
        relTol          0;
    }

    U
    {
        solver          PBiCGStab;
        preconditioner  DILU;
        tolerance       1e-06;
        relTol          0;
    }
}


PIMPLE
{
    momentumPredictor no;
    nCorrectors     3;
    nOuterCorrectors 1;
    nNonOrthogonalCorrectors 0;
    pRefCell        0;
    pRefValue       0;
}
```

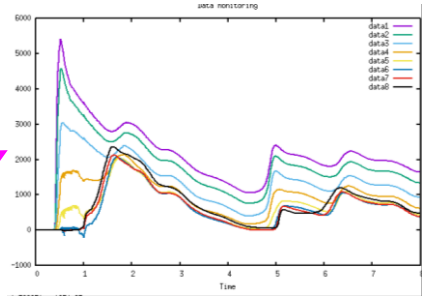
計算実行

(6)計算実行

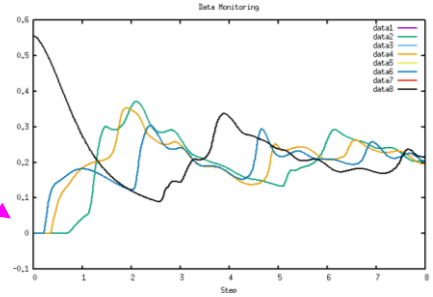
```
$./Allrun &
```

Run-Time Processing(復習)
圧力と水位の時刻歴をモニタリング

```
$foamMonitor postProcessing/probes/0/p
```




```
$foamMonitor postProcessing/interfaceHeight/0/height.dat
```



(7)フォルダ移動

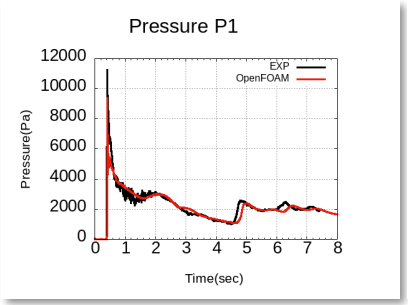
```
$cd exp_data/001_gnuplot
```

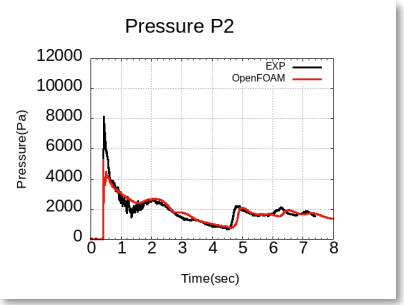
結果

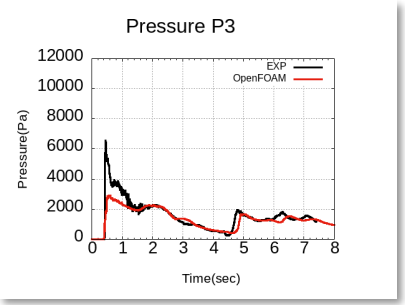
(8)グラフ化

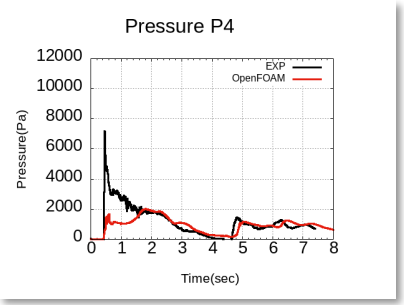
```
$gnuplot "gnuplot_run_P.plt"
```

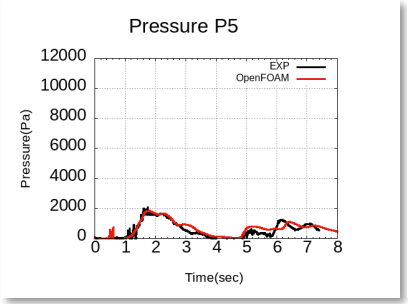
P1の圧力のピークが改善

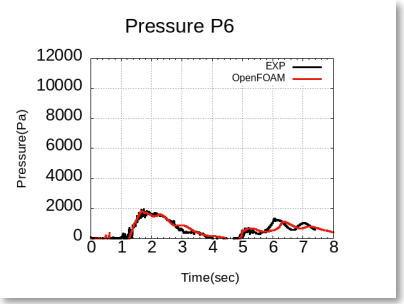


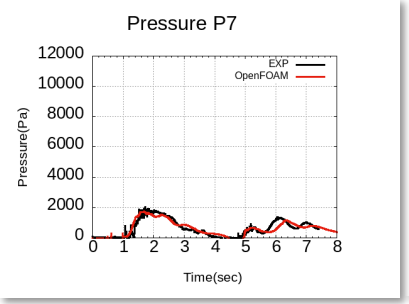


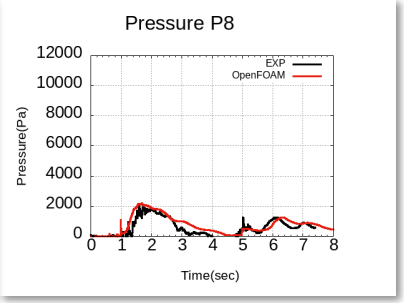








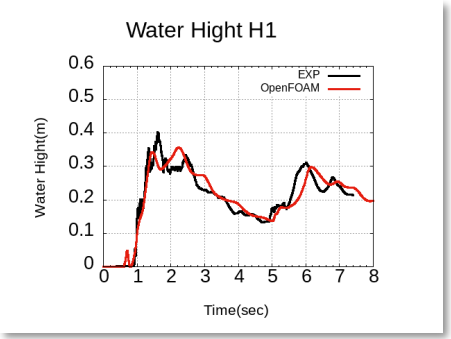


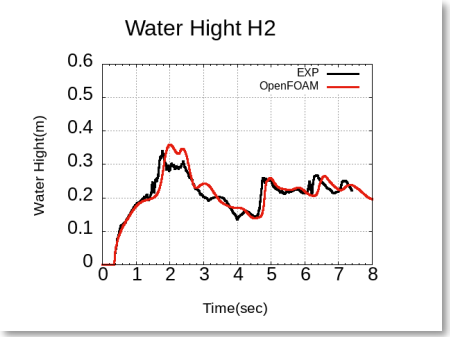


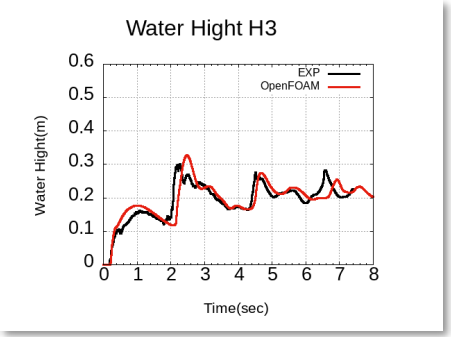
結果

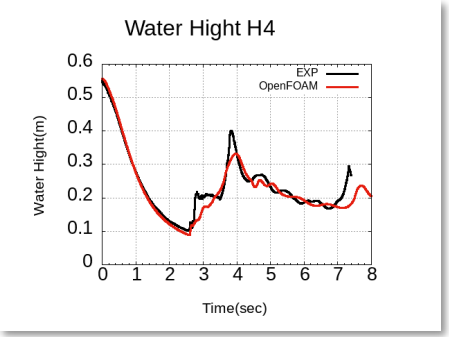
(9)グラフ化

```
$gnuplot "gnuplot_run_H.plt"
```









モデルの修正

Track2で学んだことを活用して以下のことにチャレンジしてみてください

- interIsoFoamソルバの使用（資料内で実施）
- 界面圧縮項Cα
- メッシュ修正

  再分割

  dynamicMesh
- スキーム変更

モデルの修正(例1)

https://drive.google.com/drive/folders/1wFxCQo6ZbK5E712ck8RMc3Gezv0n7PnG

- Track3_run
  - 001_3Ddam_interFoam
  - 001_3Ddam_interIsoFoam
  - sample
    - **001_3Ddam_interFoam**
    - **001_3Ddam_interIsoFoam**
    - **002_3Ddam_interFoam_refinemesh**
    - **002_3Ddam_interIsoFoam_refinemesh**
    - 003_3Ddam_interFoam_dynamicMesh

モデルの修正(例1)

◻ 構造物まわりのメッシュ再分割

system/snappyHexMeshDict

```
geometry
{
  walls_mm_tr.stl
  {
    type triSurfaceMesh;
    name walls;
  }
}
```

構造物stlファイル

```
castellatedMeshControls
{
  features (
    {
        file "walls_mm_tr.eMesh";
        level 0;
    }
  );
```

surfaceFeatureExtractで作成した特徴線を指定(constant/triSurface)

```
refinementSurfaces
 {
   walls
   {
     level (1 2);
     patchInf
     {
        type wall;
     }
   }
 }
 refinementRegions
 {
 }
 locationInMesh (2.0 0.5 0.5);
```

境界メッシュの細分化

再分割の段階設定
1段から2段の間で境界メッシュを細分化

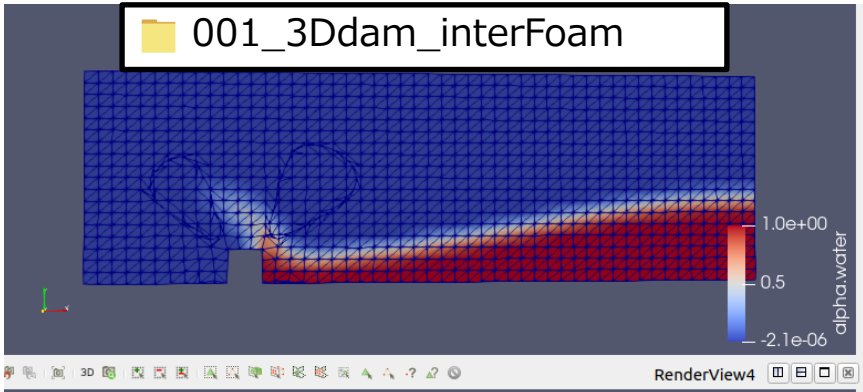
メッシュ内部の座標を指定する

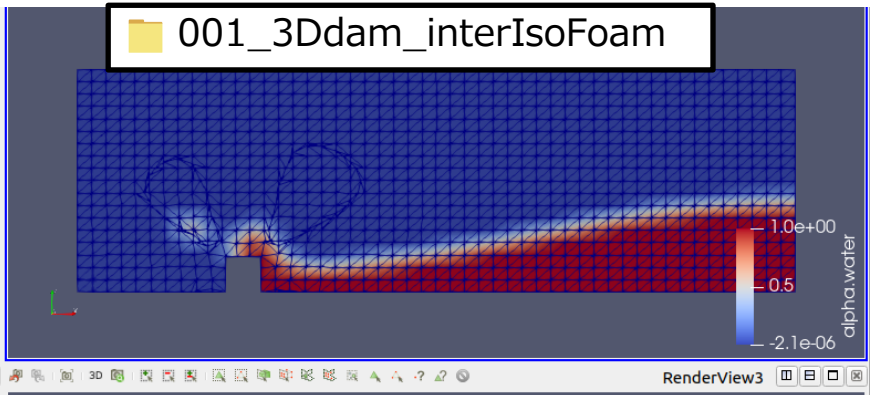
モデルの修正(例1)

□ 構造物まわりのメッシュ再分割

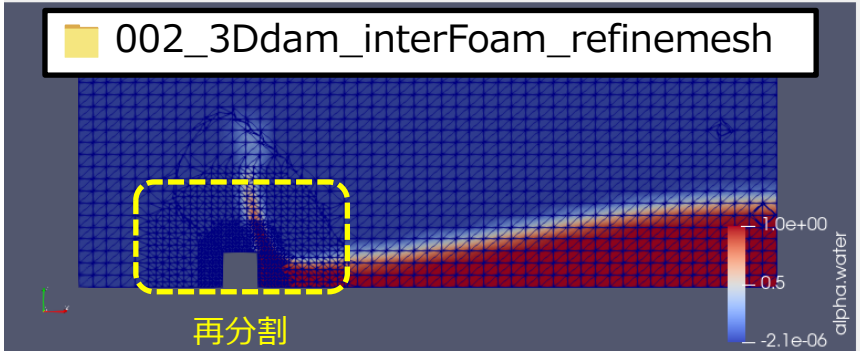
interFoam | **interIsoFoam**

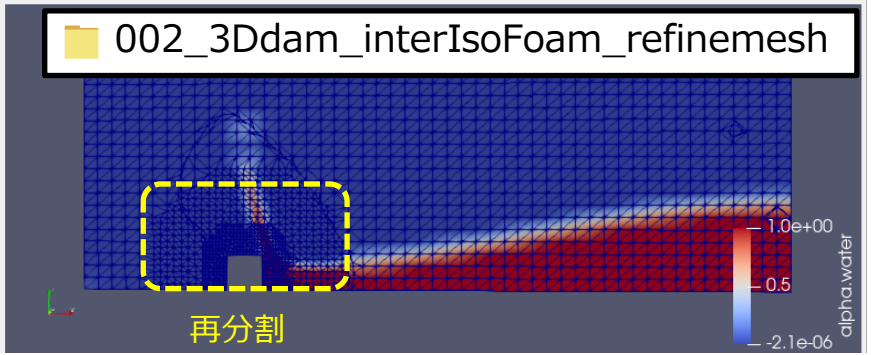
ベース





再分割





モデルの修正(例2)

https://drive.google.com/drive/folders/1wFxCQo6ZbK5E712ck8RMc3Gezv0n7PnG

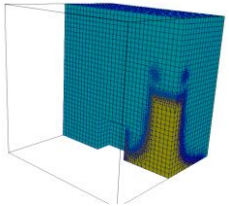
- Track3_run
  - 001_3Ddam_interFoam
  - 001_3Ddam_interIsoFoam
  - sample
    - 001_3Ddam_interFoam
    - 001_3Ddam_interIsoFoam
    - 002_3Ddam_interFoam_refinemesh
    - 002_3Ddam_interIsoFoam_refinemesh
    - **003_3Ddam_interFoam_dynamicMesh**

モデルの修正(例2)

ロ アダプティブメッシュ

解析結果に基づいてメッシュを細分化

$FOAM_TUTORIALS/multiphase/interFoam/laminar/damBreakWithObstacle/



constant/dynamicMeshDict

staticFvMesh(移動メッシュを用いない)からアダプティブメッシュに変更

```
dynamicFvMesh   dynamicRefineFvMesh;
// How often to refine
refineInterval  1;

// Field to be refinement on
field           alpha.water;

// Refine field inbetween lower..upper
lowerRefineLevel 0.001;
upperRefineLevel 0.999;
```

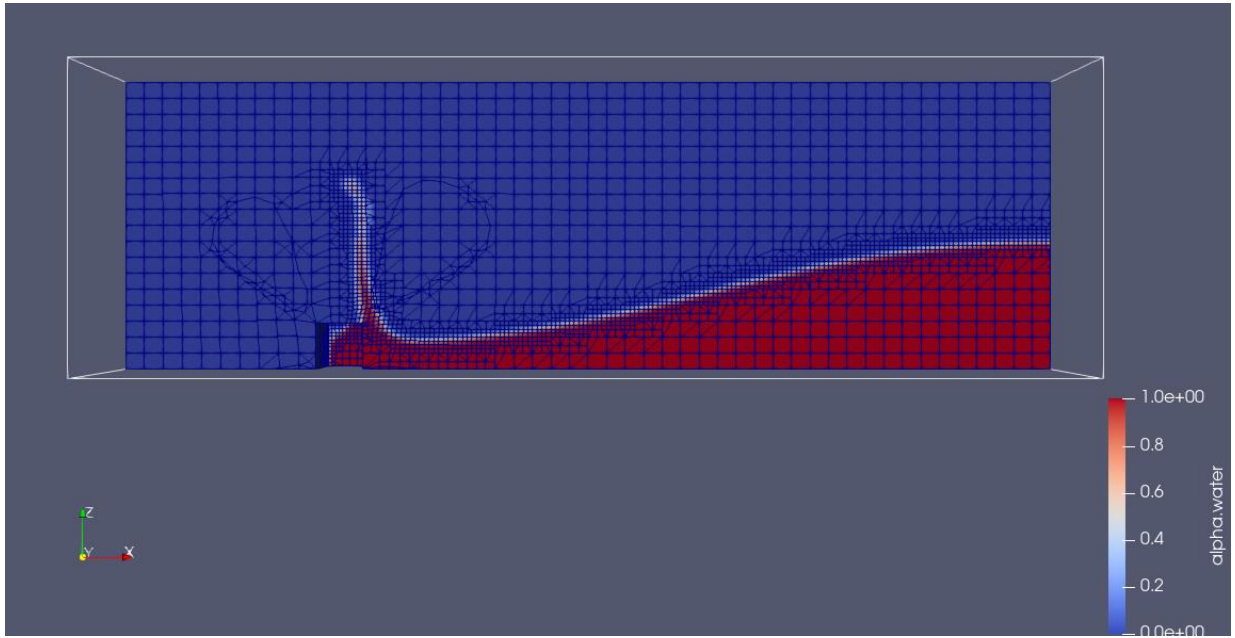
体積分率が0.001から0.999 ある相界面の格子のみ格子の細分割レベルを2(格子幅1/4)に細分割する.

```
// If value < unrefineLevel unrefine
unrefineLevel   10;
// Have slower than 2:1 refinement
nBufferLayers   1;
// Refine cells only up to maxRefinement levels
maxRefinement   2;

// Stop refinement if maxCells reached
maxCells        200000;

// Flux field and corresponding velocity field. Fluxes on changed
// faces get recalculated by interpolating the velocity. Use 'none'
// on surfaceScalarFields that do not need to be reinterpolated.
correctFluxes
(
    (phi none)
    (nHatf none)
    (rhoPhi none)
    (alphaPhi0.water none)
    (ghf none)
    (alphaPhiUn none)
);
// Write the refinement level as a volScalarField
dumpLevel       true;
```

モデルの修正(例2)

ロ アダプティブメッシュ

解析結果に基づいてメッシュを細分化



- 並列数4でも1秒までの計算で88分
- 1秒後の計算はさらに時間がかかる