

2018/9/23
for DEXCS2018

DEXCSランチャーの使い方

形状作成メニュー



- 1 「実行」ボタンを押すと、対象の解析フォルダのファイル
選択画面(次頁)が開きます。

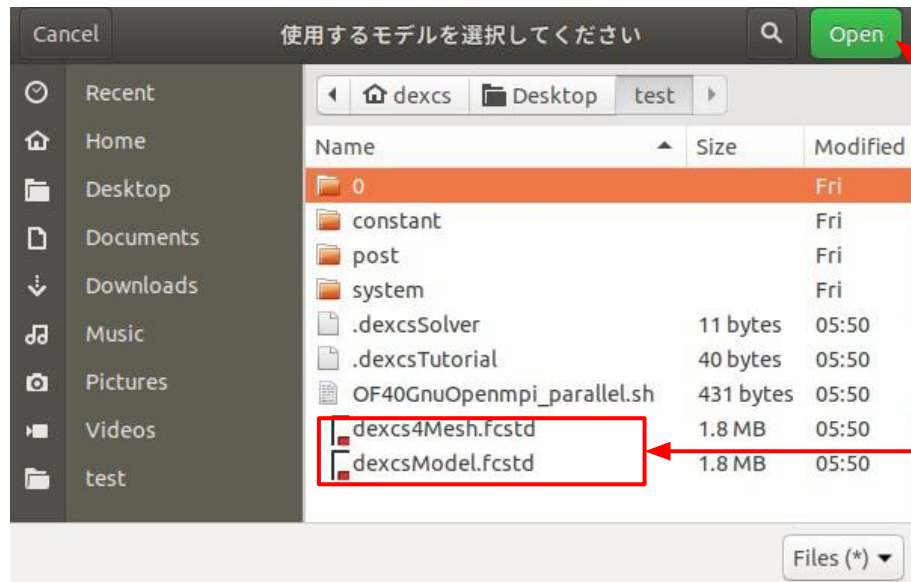


形状作成の詳細はこちらの
マニュアルも参照下さい

本マニュアル

モデル選択画面

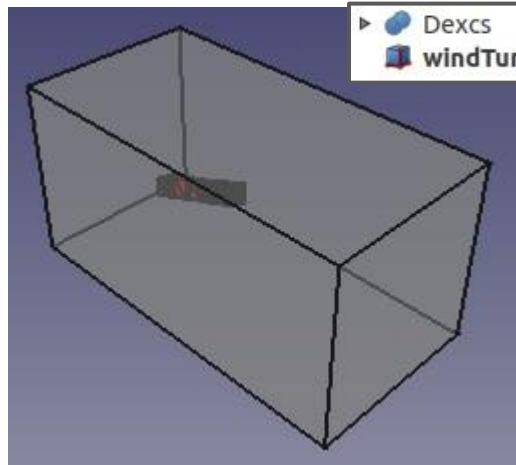
CADデータ(拡張子が**.fcstd**)を選択すれば、FreeCADが起動してCADデータの編集と、cfMesh用の設定ファイル作成(マクロの使用)が可能になります。



どちらかを選択して「Open」ボタンを押します

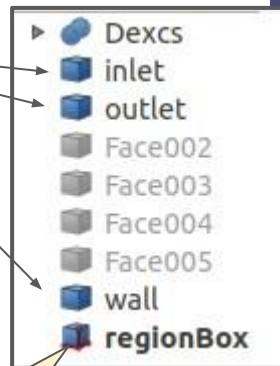
モデルの説明

dexcsModel.fcstd



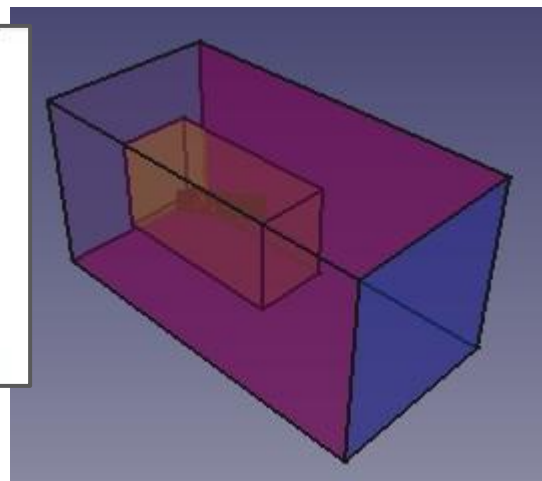
仮想風洞(windTunnel)や解析対象(Dexcs)を変更したい場合にはこれを使います
(このままでは解析に使える cfMesh 用設定ファイルを作成できません)

仮想風洞の境界毎に
面の性質に応じて名
前変更(p.6)



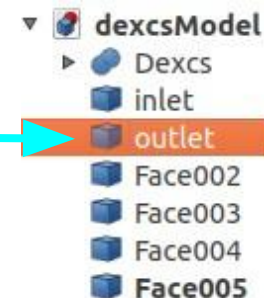
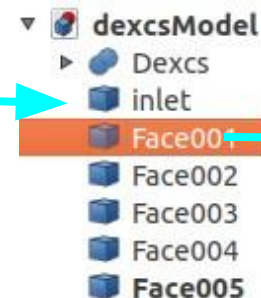
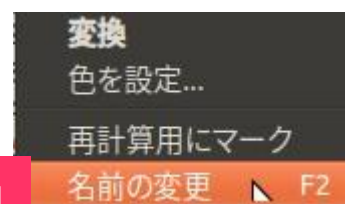
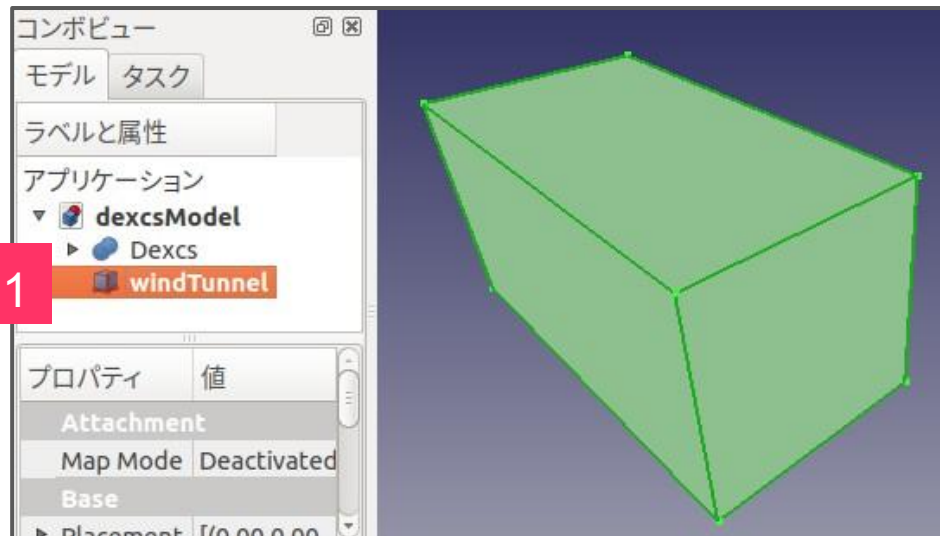
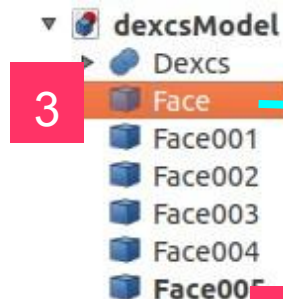
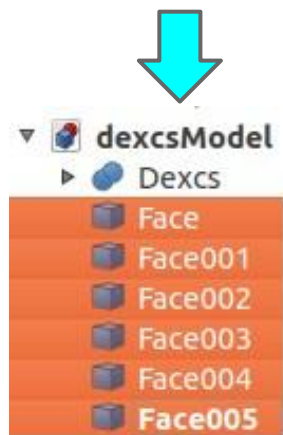
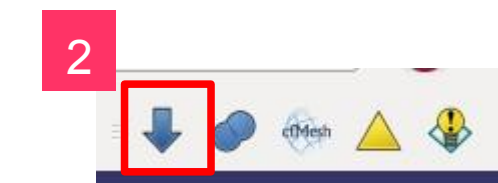
メッシュ
細分割領域を
追加(p.8)

dexcs4Mesh.fcstd

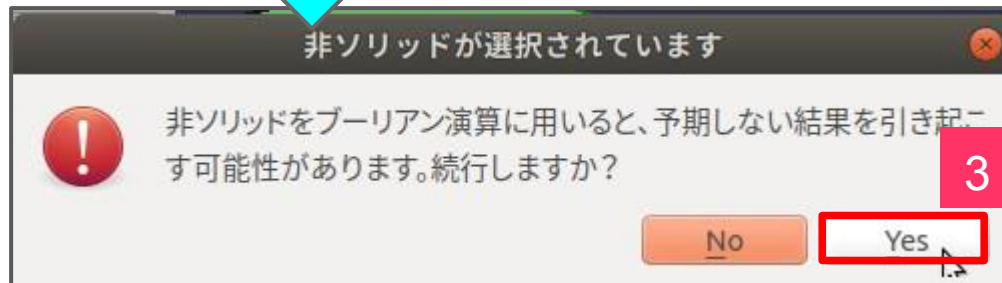


cfMesh用の設定ファイルを作成する
場合にこれをそのまま使えます

境界面の区分方法(1/3)



境界面の区分方法(2/3)



境界面の区分方法(3/3)

1

アプリケーション

- dexcsModel
 - Dexcs
 - inlet
 - outlet
 - Fusion001

2

変換
色を設定...
再計算用にマーク
名前の変更 F2

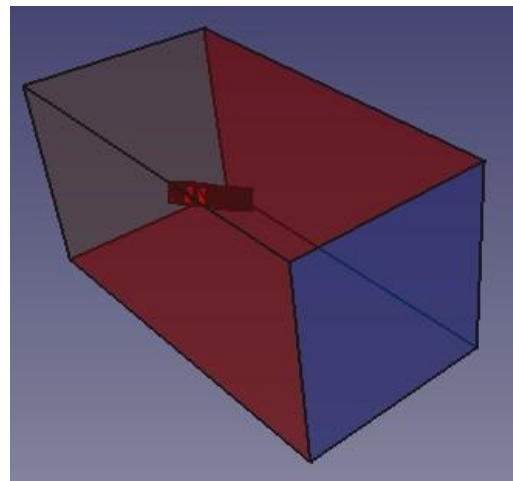
3

dexcsModel

- Dexcs
- inlet
- outlet
- walls

4

プロパティ	値
Selection Style	Shape
Shape Color	
Transparency	50
Visibility	true



メッシュ細分割領域(1/2)



メッシュ細分割領域 (2/2) (サイズと配置)

プロパティ	値
Attachment	
Map Mode	Deactivated
Base	
Placement	[(0.00 0.00 1.00);0.00 °;(...)]
Label	regionBox
Box	
Length	10.000000 mm
Width	10.000000 mm
Height	10.000000 mm

Box	
Length	4.000000 mm
Width	1.600000 mm
Height	2.000000 mm

コンボビュー

モデル タスク

配置

平行移動量:

X: 0.000000 mm

Y: 0.000000 mm

Z: 0.000000 mm

中心:

X: 0.000000 mm

Y: 0.000000 mm

Z: 0.000000 mm

☐ 重心を使用

回転:

回転軸と角度

軸: Z

角度: 0.000000 °

☐ オブジェクト位置にインクリメンタルな変更を適用

リセット

Apply

Cancel

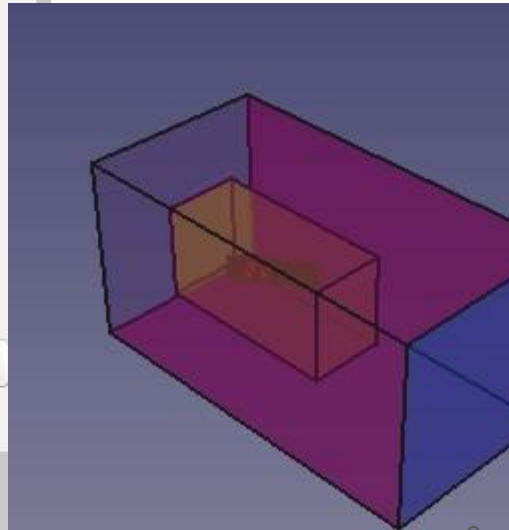
OK

平行移動量:

X: -6.000000 mm

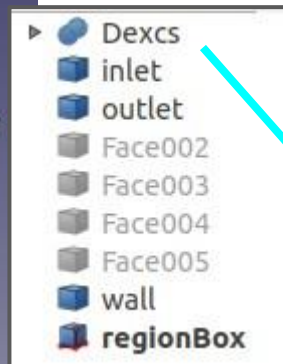
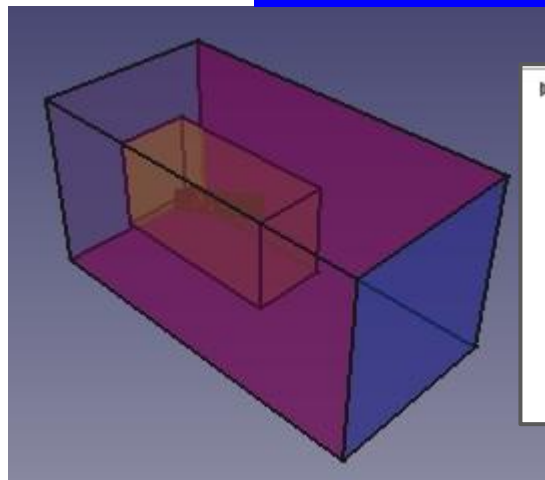
Y: -4.500000 mm

Z: -0.800000 mm

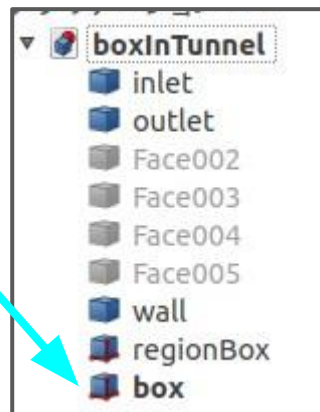
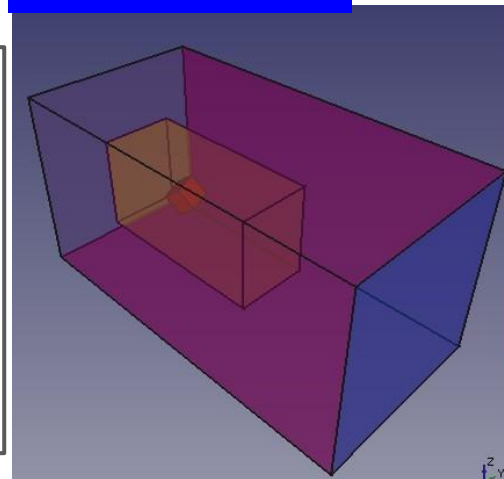


モデル変更方法例

dexcs4Mesh.fcstd

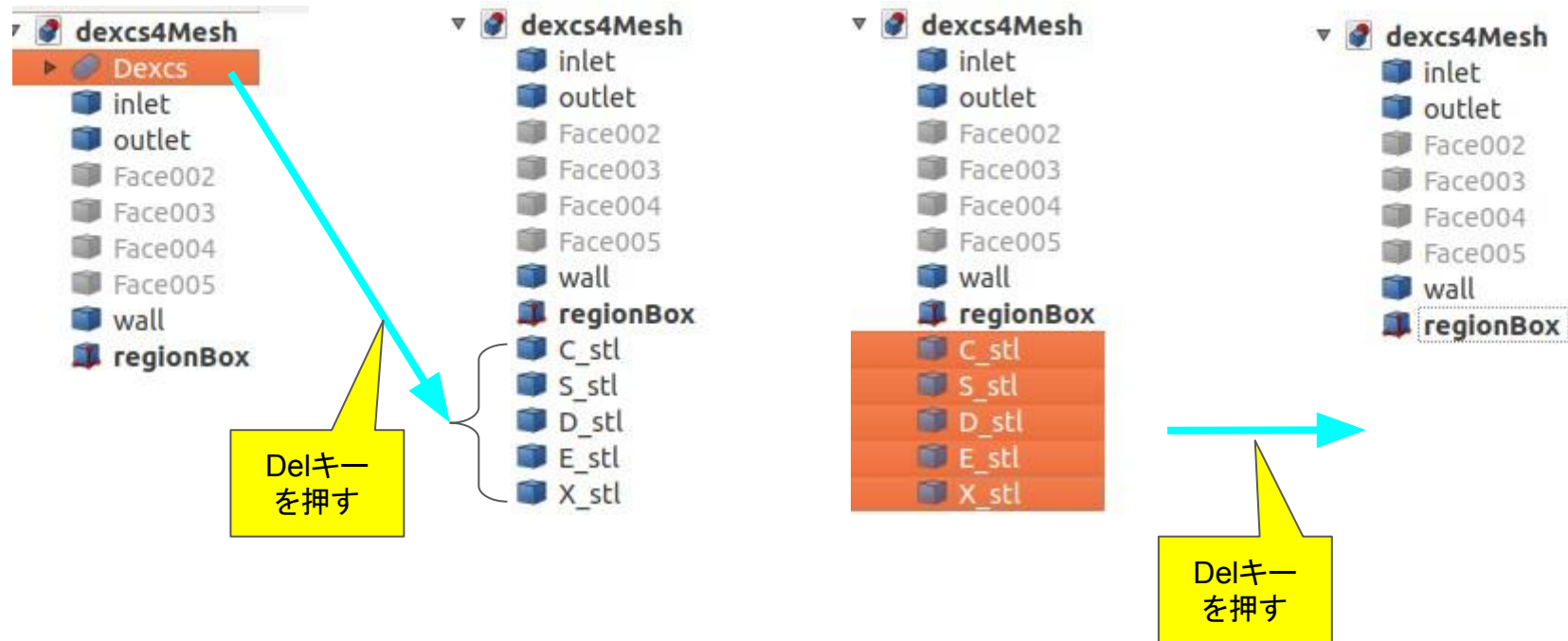


boxInTunnel.fcstd

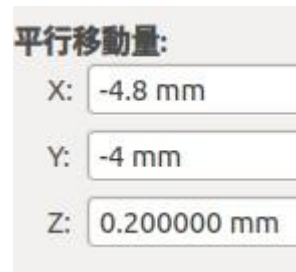
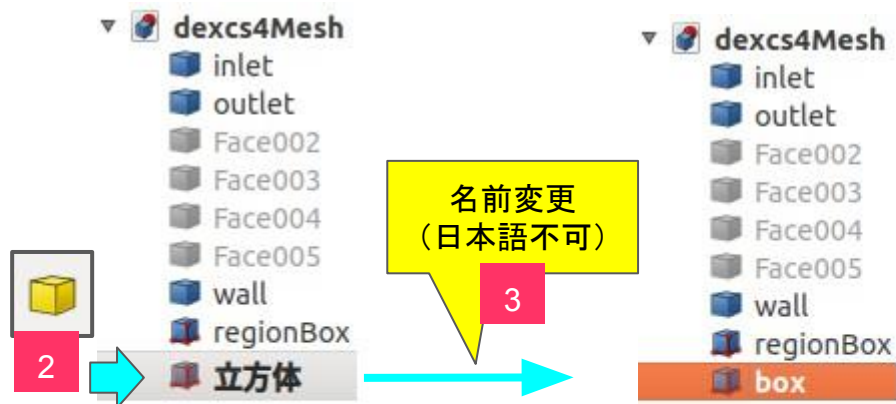


以下、本例を対象としてDEXCSランチャーの使い方を解説します

モデル変更方法例(既存モデル削除)



モデル変更方法例(box追加)

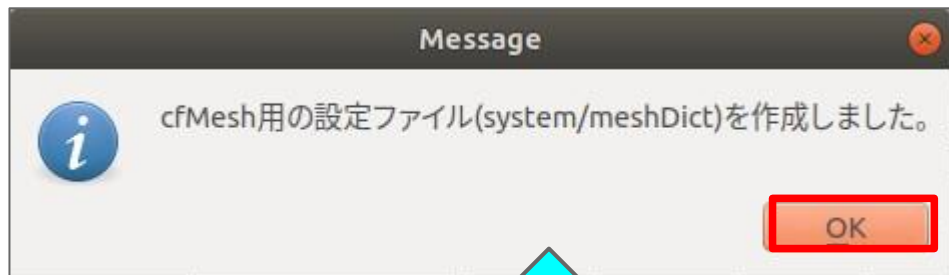


プロパティ	値
Label	box
Box	
Length	0.500000 mm
Width	0.500000 mm
Height	0.500000 mm

5

サイズ変更

cfMesh用設定ファイル作成マクロ



メッシュメニュー



1



テキストエディタ (gedit) が起動し、cfMesh設定ファイル (meshDict) を直接編集できます

2



cfMesh(cartesianMesh)が起動し、メッシュ作成が始まります

3



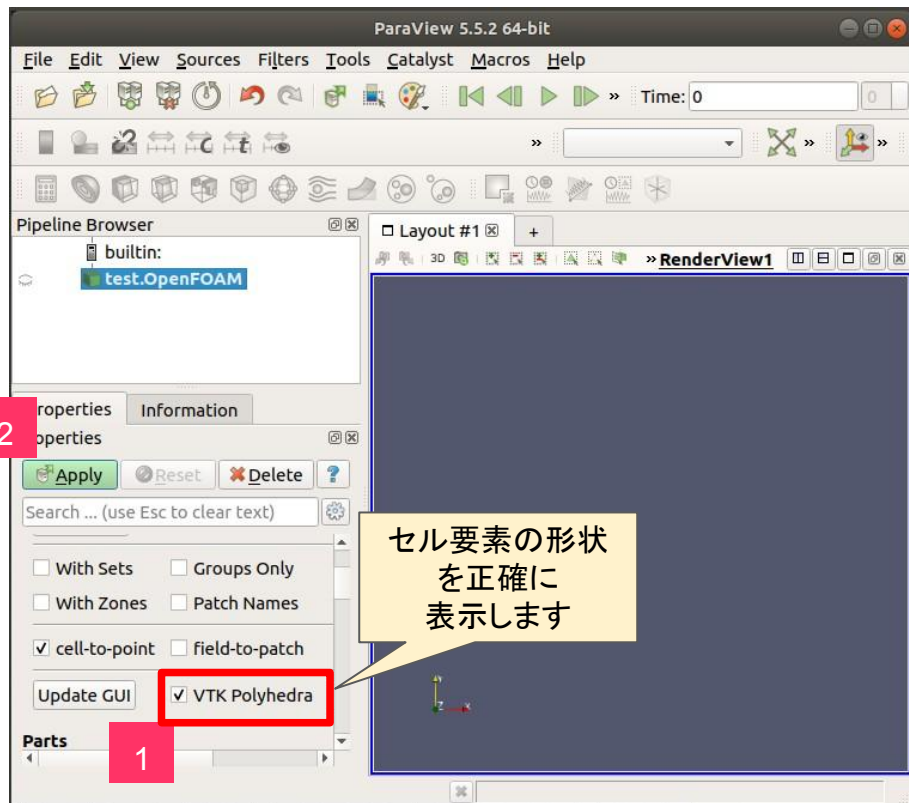
paraFoamが起動し、メッシュを可視化して確認できます

meshDict編集

- ❑ マクロで設定できるパラメタは主要なパラメタだけです。
- ❑ 自動生成される meshDict 中、マクロで設定できないパラメタは、マニュアルに記されたパラメタの説明文を含めてコメントアウトして出力されており(次々頁例参照)、パラメタ行頭の (//) を外すだけで有効にすることが出来ます。
- ❑ 狭いすき間でメッシュがうまく作成できない場合などは、これらのパラメタを調節して作成できるようになる場合が多くありますのでお試しください。

(注) cfMesh ではこれらパラメタ名を間違っても、エラーや警告はなく、無視されるだけです。

メッシュ確認



断面と交差するセル要素は切断しません

3



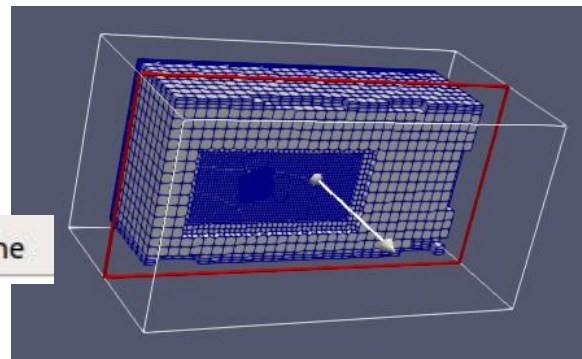
4

☒ Crinkle clip

5

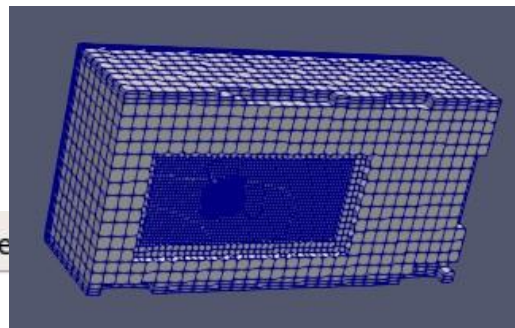
Surface With Edge

☒ Show Plane



6

☐ Show Plane

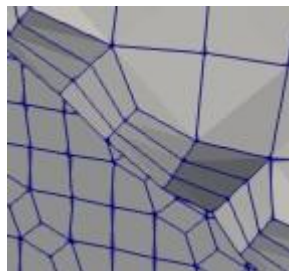
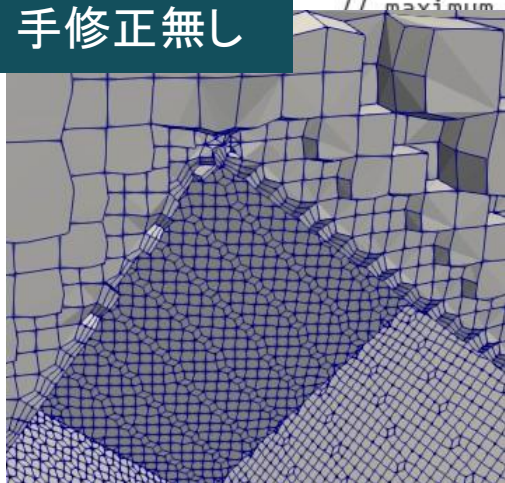


meshDict 手修正の例

```
meshDict
~/Desktop/test/system

42 // activates smoothing of boundary layers (optional)
43 // deactivated by default
44 // optimiseLayer 1; //
45
46 optimisationParameters
47 {
48     // number of iterations in the procedure
49     // for reducing normal variation (optional)
50     nSmoothNormals 3;
51     // maximum number of iterations
52     // for hole procedure (optional)
53     iterations 5;
```

手修正無し

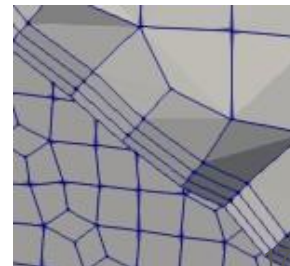
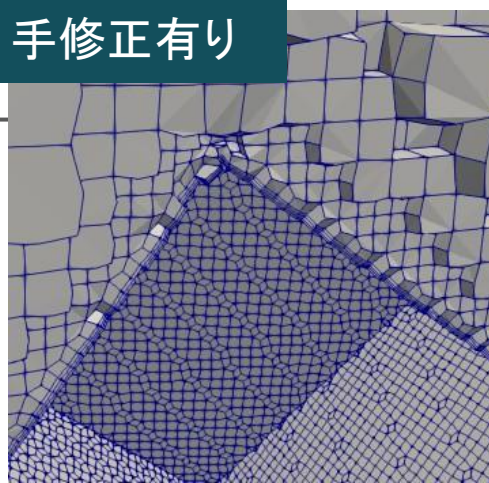


行頭の
// を削除

```
42 // activates smoothing of boundary layers (optional)
43 // deactivated by default
44 optimiseLayer 1; //
45
```

optimisationParameters/sSmoothNormals 等の
パラメタが有効になるので、指定数字は適宜変
更して使用

手修正有り



計算実行メニュー



1



計算結果をクリアします
(メッシュや計算条件はクリア
されません)

2



systemフォルダ下のファイルを一括編集(変更・確認)します
(p.19 - 28)

3



constantフォルダ下のファイルを一括編集(変更・確認)します
(p.29)

4



0フォルダ下のField変数ファイルと、境界名定義ファイル
(constant/polyMesh/bpimday)を一括編集(変更・確認)します
(p.30)

system

▼ system

【各ファイルの設定内容】

 controlDict

時間とデータの入出力制御方法 (p.20)

 decomposeParDict

並列計算の領域分割方法 (p.24)

 forceCoeffs

空力係数の計算方法 (p.28)

 forces


流体力の計算方法 (p.27)

 fvSchemes

導関数等の離散化に使用する数値スキーム (p.21)

 fvSolution

方程式を求解する際の公差やアルゴリズム (p.22)

 massFlow

境界面を指定した積算流量等の計算方法 (p.26)

 meshDict

cfMesh用制御パラメタ (p.17)

 probes

プローブ点を指定したField変数の出力方法 (p.25)

 sampleDict

Field変数サンプリング方法 (p.47)

青色: controlDictから参照されるファイル (p.20 参照)

system/controlDict

4.3 時間とデータの入出力制御 U-113

時間制御

startFrom
- firstTime
- startTime
- latestTime

解析の開始時刻の制御

存在する時刻ディレクトリのうちで最初の時刻

startTime の項目の入力により定める時刻

存在する時刻ディレクトリのうちで最新の時刻

startTime

startFrom の startTime を用いた解析の開始時刻

stopAt

解析の終了時刻の制御

- endTime
- writeNow
- noWriteNow
- nextWrite

endTime の項目の入力により定める時刻

現在の時間ステップで解析を止めデータを書き出す

現在の時間ステップで解析を止めデータは書き出さない

writeControl で指定した次のデータ書き出しの時間ステップで解析を止める

endTime

stopAt の endTime で指定した解析の終了時刻

deltaT

解析の時間ステップ

データの書き出し

writeControl
- timeStep

ファイルへのデータの書き出しのタイミングの制御

時間ステップの writeInterval ごとにデータを書き出す

```
controlDict
~/Desktop/test/system

controlDict x decomposeParDict x forceCoeffs x forces
17
18 application simpleFoam;
19
20 startFrom latestTime;
21
22 startTime 0;
23
24 stopAt endTime;
25
26 endTime 1000;
27
28 deltaT 1;
29
30 writeControl timeStep;
31
32 writeInterval 50;
33
34 purgeWrite 0;
35
36 writeFormat ascii;
37
38 writePrecision 6;
39
40 writeCompression u
41
42 timeFormat ge
43
44 timePrecision 6;
45
46 runTimeModifiable yes;
47
48 functions
49 {
50     #include "forces"
51     #include "forceCoeffs"
52     #include "probes"
53     #include "massFlow"
54     #include "sampleDict"
55 }
56
```

実行時に追加ライブラリを使用
して様々な処理を行う関数を、
別ファイルで定義している

system/fvSchemes

ヘルプ

ランチャの使い方

OpenFoam UsersGuide(v5.0)

同上 (和訳版)

OpenFoam ProgrammersGuid

4.4 数値スキーム

U-115

```

fvSchemes
~/Desktop/test/system
保存

forces × fvSchemes × fvSolution × massFlow × meshDict × probes ×
16
17 ddtSchemes
18 {
19     default      steadyState;
20 }
21
22 gradSchemes
23 {
24     default      Gauss linear;
25 }
26
27 divSchemes
28 {
29     default      none;
30 //    div(phi,U)    bounded Gauss linearUpwindV gra
31    div(phi,U)    bounded Gauss upwind;
32    div(phi,k)    bounded Gauss upwind;
33    div(phi,omega) bounded Gauss upwind;
34    div(phi,epsilon) bounded Gauss upwind;
35    div((nuEff*dev(T(grad(U)))) Gauss linear;
36    div((nuEff*dev2(T(grad(U)))) Gauss linear;
37 }
    
```

```

39 laplacianSchemes
40 {
41     default
42 //    default      Gauss linear limited 0.5;
43 //    default      Gauss linear uncorrected;
44 }
45
46 interpolationSchemes
47 {
48     default      linear;
49 }
50
51 snGradSchemes
52 {
53     default      corrected;
54 }
55
56 fluxRequired
57 {
58     default      no;
59    p;
60 }
61
    
```

キーワード	数学的タームのカテゴリ
interpolationSchemes	2点間の値の補間
snGradSchemes	格子界面の法線方向勾配の各要素
gradSchemes	勾配 ∇
divSchemes	発散 $\nabla \cdot$
laplacianSchemes	Laplacian ∇^2
timeScheme	1次と2次の時間導関数 $\partial/\partial t$, $\partial^2/\partial^2 t$
fluxRequired	フラックスの生成が必要な物理量

表 4.5 fvSchemes で使用する主なキーワード

system/fvSolution(1/2)

ヘルプ

ランチャの使い方

OpenFoam UsersGuide(v5.0)

同上 (和訳版)

OpenFoam ProgrammersGuid

```
16
17 solvers
18 {
19     p
20     {
21         solver      GAMG;
22         tolerance   1e-7;
23         relTol      0.1;
24         smoother    GaussSeidel;
25         nPreSweeps  0;
26         nPostSweeps 2;
27         cacheAgglomeration on;
28         agglomerator faceAreaPair;
29         nCellsInCoarsestLevel 10;
30         mergeLevels 1;
31     }
32
33     U
34     {
35         solver      smoothSolver;
36         smoother    GaussSeidel;
37         tolerance   1e-8;
38         relTol      0.1;
39         nSweeps     1;
40     }
41 }
```

```
42 k
43 {
44     solver
45     smoother
46     tolerance
47     relTol
48     nSweeps
49 }
50
51 omega
52 {
53     solver
54     smoother
55     tolerance
56     relTol
57     nSweeps
58 }
59
60 epsilon
61 {
62     solver
63     smoother
64     tolerance
65     relTol
66     nSweeps
67 }
68 }
```

smoothS

GaussSeidel

1e-8;

0.1;

1;

4.5 解法とアルゴリズム制御 U-122

4.5.1 線形ソルバ制御 U-123

ソルバ	キーワード
前処理付き(双)共役勾配	PCG/PBiCG [†]
スムーサを使ったソルバ	smoothSolver
汎用幾何学的代数マルチグリッド	GAMG
陽的な系のための対角ソルバ	diagonal

[†] PCG は対称用, PBiCG は非対称用

表 4.12 線形ソルバ

system/fvSolution(2/2)

4.5.3 PISOとSIMPLEアルゴリズム U-127

```
70 SIMPLE
71 {
72     nNonOrthogonalCorrectors 0;
73
74     residualControl
75     {
76         p          1e-3;
77         U          1e-3;
78         "(k|epsilon)" 1e-3;
79     }
80 }
81
82 potentialFlow
83 {
84     nNonOrthogonalCorrectors 10;
85 }
86
```

4.5.2 不足緩和解析

U-125

```
87 relaxationFactors
88 {
89     fields
90     {
91         p          0.5;
92     }
93     equations
94     {
95         U          0.7;
96         k          0.7;
97         omega      0.7;
98         epsilon    0.7;
99     }
100 }
101
102 cache
103 {
104     grad(U);
105 }
```

system/decomposeParDict

ヘルプ
ランチャの使い方
OpenFoam UsersGuide(v5.0)
同上 (和訳版)
OpenFoam ProgrammersGuid

```
decomposeParDict
~/Desktop/test/system

1 |
2 // * * * * * //
3 FoamFile
4 {
5     version 0.5;
6     format ascii;
7     root "ROOT";
8     case "CASE";
9     class dictionary;
10    object nix;
11 }
12 method scotch;
13 numberOfSubdomains 2;
14 scotchCoeffs
15 {
16 }
```

分割方法

並列分割数

3.4 アプリケーションの並列実行 U-86

ユーザは、以下に述べる `method` キーワードにより指定できる四つの分割方法から選択します。

simple 簡単なジオメトリの分割：ドメインは x, y 方向に、例えば x 方向に二つに、 y 方向一つにというように、ピースが分割されます。

hierarchical 階層的なジオメトリの分割方法：基本的には `simple` と同じですが、ユーザが、最初に y 方向を、次に x 方向を、というように、各方向の分割する順番を指定する点が異なります。

scotch Scotch 分割はユーザからのジオメトリの入力を必要とせず、プロセッサの境界の数値を最小化するよう試みます。ユーザは、任意指定の `processorWeights` キーワードによりプロセッサ間の重み付けを行うことができるため、パフォーマンスの異なるマシン同士を有効に使うことができます。また、もう一つ `strategy` という任意のキーワードエントリがあり、複雑な文字列を Scotch に渡すことにより分割の戦略を制御できます。さらなる情報を得るには、ソースコードファイル `$FOAM_SRC/decompositionMethods/decompositionMethods/scotchDecomp/scotchDecomp.C` を読んでください。

manual マニュアルでの分割：個別のプロセッサに対して、各々のセルの割り当てを直接指定します。

DEXCSランチャーで実行する際には、本ファイルは自動作成されるので、手修正出来ません(しても無効になります)

system/probes

```
6 | \\\ M anipulation | web. www.openfoam.org
7 | *-----*/
8 | probes
9 | {
10 |     writeControl    timeStep;
11 |     writeInterval    1;
12 |
13 |     type probes; // Type of functionObject
14 |     // Where to load it from (if not already in solver)
15 |     functionObjectLibs ("libsampling.so");
16 |
17 |     log              yes;|
18 |
19 |     probeLocations // Locations to be probed. runTime modifiable!
20 |     (
21 |         (0.0 -3.5 0.2)
22 |     );
23 |     // Fields to be probed. runTime modifiable!
24 |     fields
25 |     (
26 |         p
27 |         k
28 |         U
29 |     );
30 | }
31 |
```

timeStep毎に出力

実行時ログにも出力

座標値

検査したいField変数

system/massFlow

```
massFlow
~/Desktop/test/system

forces x fvSchemes x fvSolution x massFlow x meshDict x prob

/*
8 surfaceData_type surfaceFieldValue; // for v1706+
9 //surfaceData_type surfaceRegion; // for 4.1
10
11 inletMassFlow
12 {
13     type                $surfaceData_type;
14     libs ("libfieldFunctionObjects.so");
15     writeControl        timeStep;
16     log                  true;
17     // Output field values as well
18     writeFields         false;
19     regionType          patch;
20     name                 inlet;
21     operation            sum;
22
23     fields
24     (
25         phi
26     );
27 }
28
```

inlet において
phi(流束)の
sum(総和)
を計算します

```
29 outletMassFlow
30 {
31     type                $surfaceData_type;
32     libs ("libfieldFunctionObjects.so");
33     writeControl        timeStep;
34     log                  true;
35     // Output field values as well
36     writeFields         false;
37     regionType          patch;
38     name                 outlet;
39     operation            sum;
40
41     fields
42     (
43         phi
44     );
45 }
46
```

outlet において
phi(流束)の
sum(総和)
を計算します

system/forces



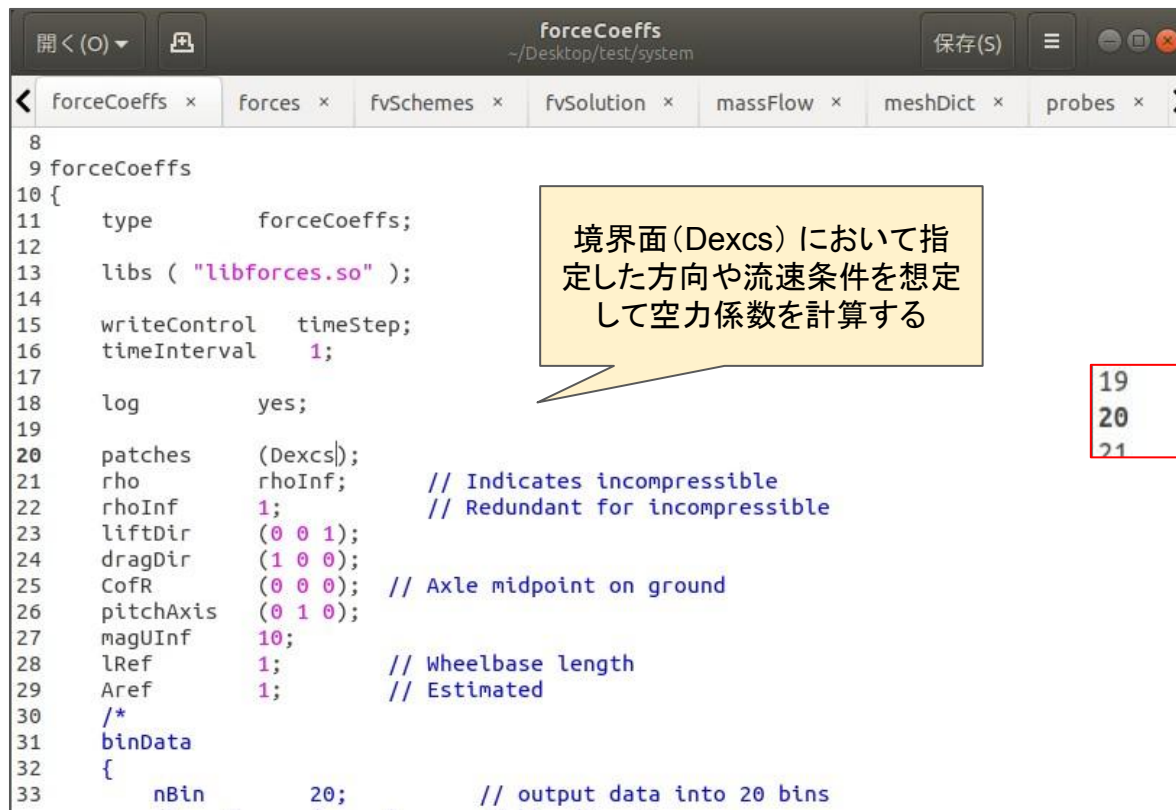
```
4 | \ \ / \ Operation | version: 4.0
5 | \ \ / \ And | Web: www.OpenFOAM.org
6 | \ \ / \ Manipulation |
7 | *-----*
8
9 forces
10 {
11     type          forces;
12
13     libs ( "libforces.so" );
14
15     writeControl   timeStep;
16     timeInterval   1;
17
18     log            yes;
19
20     patches        (Dexcs);
21         pName      p;
22         UName      U;
23         rhoName    rhoInf;
24     rho            rhoInf;      // Indicates incompressible
25     rhoInf         1.225;      // Redundant for incompressible
26     CofR           (0.25 0.007 0.125); // Axle midpoint on ground
27 }
28
29
30 // ***** //
```

境界面(Dexcs)において
pName,UName,rhoName
で指定した変数を使って流
体力を計算する

解析対象を変更したら対応の名
前に変更して使用する。

```
19
20 patches      (box);
21     pName      p;
```

system/forceCoeffs



```
8
9 forceCoeffs
10 {
11     type            forceCoeffs;
12     libs ( "libforces.so" );
13     writeControl     timeStep;
14     timeInterval     1;
15
16     log              yes;
17
18     patches          (Dexcs);
19     rho              rhoInf;          // Indicates incompressible
20     rhoInf           1;              // Redundant for incompressible
21     liftDir          (0 0 1);
22     dragDir          (1 0 0);
23     CofR             (0 0 0);        // Axle midpoint on ground
24     pitchAxis        (0 1 0);
25     magUInf          10;
26     lRef             1;              // Wheelbase length
27     Aref             1;              // Estimated
28     /*
29     binData
30     {
31         nBin          20;            // output data into 20 bins
32     }
```

境界面(Dexcs)において指定した方向や流速条件を想定して空力係数を計算する

解析対象を変更したら相応の名前に変更して使用する。

```
19
20 patches (box);
21 rho rhoInf;
```

constant

constant

polyMesh

transportProperties

turbulenceProperties

```
transportProperties
~/Desktop/test/constant

transportProperties x turbulenceProperties x
15 // * * * * *
16
17 transportModel Newtonian;
18
19 nu          nu [0 2 -1 0 0 0 0] 1.54e-05;
20
21 CrossPowerLawCoeffs
22 {
23     nu0          nu0 [0 2 -1 0 0 0 0] 1e-06;
24     nuInf        nuInf [0 2 -1 0 0 0 0]
25     m            m [0 0 1 0 0 0 0] 1;
26     n            n [0 0 0 0 0 0 0] 1;
27 }
28
```

動粘性係数

3.7 標準のライブラリ

U-100

非圧縮性流れ用輸送モデル — incompressibleTransportModels	
Newtonian	線形粘性流れモデル
CrossPowerLaw	Cross Power 則非線形粘性モデル
BirdCarreau	Bird-Carreau 非線形粘性モデル
HerschelBulkley	Herschel-Bulkley 非線形粘性モデル
powerLaw	べき乗則非線形粘性モデル
interfaceProperties	多相流解

```
turbulenceProperties
~/Desktop/test/constant

transportProperties x turbulence
15 // * * * * *
16
17 simulationType RAS;
18
19 RAS
20 {
21     RASModel      kEpsilon;
22     turbulence     on;
23     printCoeffs    on;
24     <RASModel>Coeffs
25
```

RASModel	RAS モデルの名前
turbulence	乱流モデルの on/off スイッチ
printCoeffs	シミュレーション開始時にモデル係数をターミナルに出力するスイッチ
<RASModel>Coeffs	各 RASModel における係数のディクショナリ (省略可能)

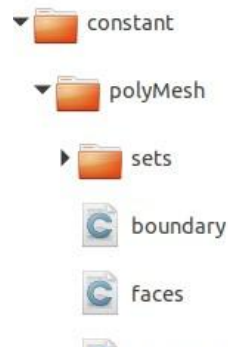
表 7.3 RASProperties ディクショナリにおけるキーワードエントリ

7.2 乱流モデル

U-187

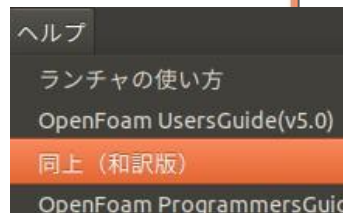
乱流モデルは k-ε

0/* , boundary



```
U x | epsilon x | k x | nut x | omega x | p x | boundary x
17 dimensions [ 0 1 -1 0 0 0 0 ];
18 internalField uniform (0 0 0);
19 boundaryField
20 {
21   wall
22   {
23     type slip;
24   }
25   inlet
26   {
27     type value;
28     fixedValue;
29     uniform (10 0 0);
30   }
31   Dexcs
32   {
33     type value;
34     fixedValue;
35     uniform (0 0 0);
36   }
37   outlet
38   {
39     type zeroGradient;
40   }
41 }
```

標準テンプレートでの
patch名はDexcsです



5.2.3 基本型

U-138

種類	物理量 ϕ に対して与える条件	与えるデータ
fixedValue	ϕ の値を指定	value
fixedGradient	ϕ の勾配を指定	gradient
zeroGradient	ϕ の勾配が 0	-
calculated	ϕ の境界条件が他の物理量から決まる	-
mixed	fixedValue と fixedGradient の組み合わせ. valueFraction に依存する条件	refValue, refGradient valueFraction
directionMixed	テンソル型の valueFraction を用いる mixed 条件. 例えば法線方向と接線方向で異なる取り扱いを行う場合	refValue, refGradient valueFraction

0/* , boundary



本例で作成したメッシュでのpatch名はboxです

```
19 4
20 (
21 box
22 {
23     type wall;
24     nFaces 8922;
25     startFace 342589;
26 }
27
28 inlet
29 {
30     type patch;
31     nFaces 400;
32     startFace 351511;
33 }
34
35 outlet
36 {
37     type patch;
38     nFaces 400;
39     startFace 351911;
40 }
41
42 wall
43 {
44     type wall;
45     nFaces 2400;
46     startFace 352311;
47 }
48
```

ヘルプ
ランチャの使い方
OpenFoam UsersGuide(v5.
同上 (和訳版)
OpenFoam ProgrammersC

5.2.2 基底型 U-136

種類	意味
patch	一般的なパッチ
symmetryPlane	対称面
empty	2次元形状の前後の面
wedge	軸対称形状のための、くさび型の前後
cyclic	周期境界面
wall	壁面（乱流の壁関数に使用）
processor	並列計算時のプロセッサ間の境界

表 5.2 基底型の境界の種類

C ▾ タブ幅: 8 ▾ (1行、1列) ▾ [挿入]

計算実行メニュー



5

標準テンプレートで使用している patch名と、作成したメッシュの patch名が異なる場合にこのメニューを使用します (p.33)

4

直接編集メニューを使用して、個別のファイル毎に変更することも可能ですが、手間がかかります

4

5

patch名変更(1/3)

The screenshot shows a window titled "boundaryPatchNameConverterGUI". It contains a table with columns for patch names and boundary conditions. The table is as follows:

patchName(From)	patchName(To)	k	epsilon	p	nut	omega	U
wall		sli	slip;	sli	slip;	slip;	sli
inlet		tu	turbulen	ze	calcu	turbuler	fix
Dexcs		kq	epsilonW	ze	nutk	omegaV	fix
outlet		ze	zeroGra	fix	calcu	zeroGra	ze

Annotations:

- 標準テンプレートの (0フォルダ下Field変数で定義されている) patch名リスト
- Field変数毎の境界条件type (表示のみで変更は出来ない)
- 新しく作成したメッシュのpatch名リスト

patch名変更(2/3)

Dexcs を box に変更

boundaryPatchNameConverterGUI

patchName(From)	patchName(To)	k	epsilon	p	nut	omega	U
wall		sli	slip;	sli	slip;	slip;	sli
inlet		tu	turbulen	ze	calcu	turbuler	fix
Dexcs	box	kq	epsilonW	ze	nutk	omegaV	fix
outlet		ze	zeroGra	fix	calcu	zeroGra	ze

Convert... Exit

```
ファイル(F) 編集(E) 表示(V) 検索(S) 端末(T) ヘルプ(H)

From function void Foam::GeometricField<Type, Pa
<TypeR, GeoMesh>&, const Foam::dictionary&) [with Ty
= Foam::volMesh]
in file /home/dexcs/OpenFOAM/OpenFOAM-v1806/src/

## EXEC ... /opt/OpenFOAM/OpenFOAM-v1806/etc/bas
## EXEC ... xed ./0/* ./constant/polyMesh/boundary
tcFilesConverter completed --> to tmp/*
```

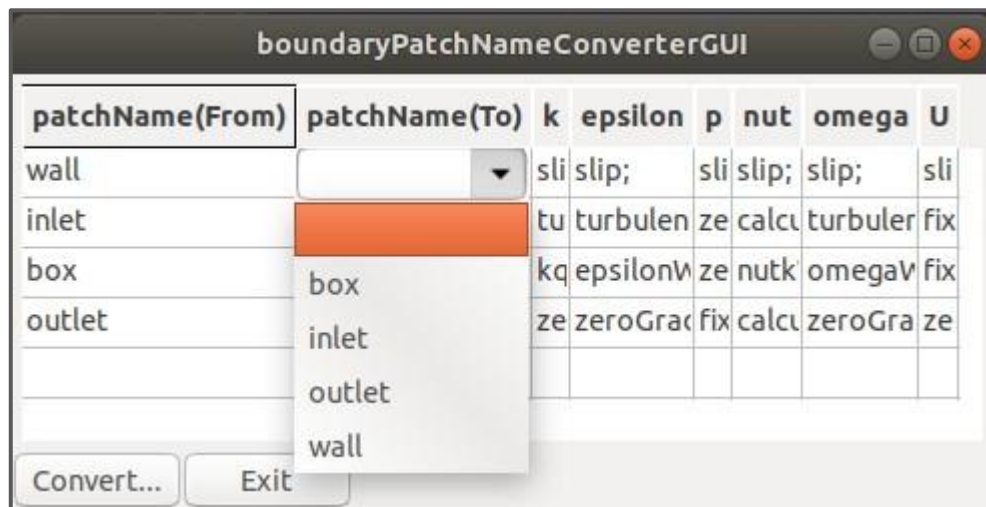
確認

Question

0/*を tmp/*で書き換えますか

No Yes

patch名変更(3/3) (確認)



patch名リストが合致 ⇒ 計算実行可能

但し本例では、wallの基底typeを変更した(注)ので、境界条件も変更する必要があります(次頁参照)。

(注)p.13にて、patch⇒ wallに変更
(標準チュートリアルでは変更していない)

0/* の適合

```
開く(O) ▾ 保存(S)
*U x epsilon x k x nut x
16 //
17 dimensions [ 0 1 -1 0 0 0 0 ];
18 internalField uniform (0 0 0);
19 boundaryField
20 {
21   wall
22   {
23     type
24     value
25   }
26   inlet
27   {
28     type
29     value
30   }
31   box
32   {
33     type
34     value
35   }
36   outlet
37   {
38     type zeroGradient;
39   }
40 }
```

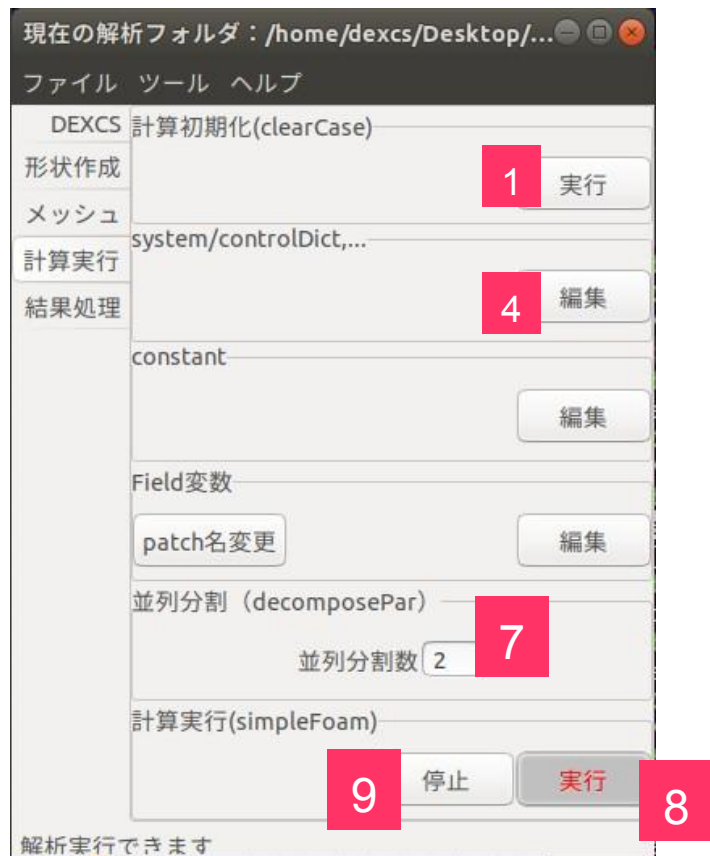
```
*epsilon
~/Desktop/test/0
開く(O) ▾ 保存(S)
U x *epsilon x k x nut x omega x p x bou
17 dimensions [ 0 2 -3 0 0 0 0 ];
18 internalField uniform 14.855;
19 //internalField uniform 1;
20 boundaryField
21 {
22   wall
23   {
24     type
25     value
26   }
27   inlet
28   {
29     //
30     type
31     type
32     mixingLength
33     value
34     //
35   }
36   box
37   {
38     type
39     value
40   }
41   outlet
42 }
```

他のField変数についても同様に実施

コピペ

コピペ

計算実行メニュー



7

並列計算の際の領域分割数を指定できます

8

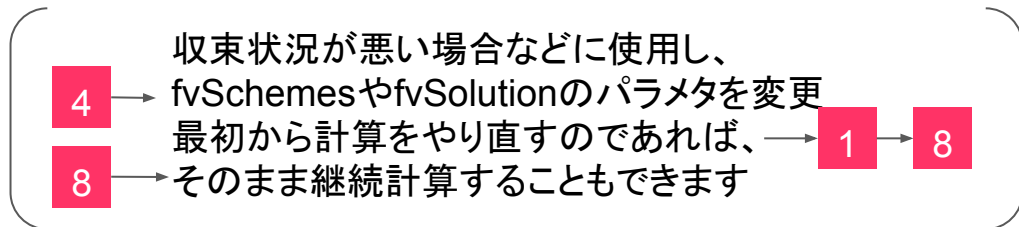


計算が開始されます

9



計算を途中で停止します



計算実行

実行コマンド

```
ファイル(F) 編集(E) 表示(V) 検索(S) 端末(T) ヘルプ(H)
py : Process 26880 was already dead
## EXEC ... xed ./system/* & ... Completed.
## EXEC ... ./opt/OpenFOAM/OpenFOAM-v1806/etc/bashrc;pyFoamClearCase.py . ; rm processor* -rf ... Completed.
./opt/OpenFOAM/OpenFOAM-v1806/etc/bashrc;pyFoamDecompose.py --method=scotch . 2;pyFoamPlotRunner.py --procn=2 simpleFoam &
## EXEC ... xed ./0/* /constant/polyMesh/boundary & ... Completed
./opt/OpenFOAM/OpenFOAM-v1806/etc/bashrc;pyFoamDecompose.py --method=scotch . 2;pyFoamPlotRunner.py --procn=2 simpleFoam &
PyFoam WARNING on line 42 of file /usr/local/lib/python2.7/dist-packages/PyFoam-0.9.10-py2.7.egg/PyFoam/Infrastructure/ServerBase.py: No certificate file /home/dexcs/.pyFoam/foamServer possible
```

```
To generate a private key:
openssl genrsa -out /home/dexcs/.pyFoam/foamServer.key
Then generate the certificate that is valid for 3 years:
openssl req -new -x509 -key /home/dexcs/.pyFoam/foamServer.key -out /home/dexcs/.pyFoam/foamServer.crt -days 1095
```

```
-----
|=====|
| \\      / F ield      | OpenFOAM: The Open Source CFD toolbox |
| \\      / O peration   | Version: v1806                      |
| \\      / A nd         | Web: www.OpenFOAM.org                |
| \\      / M anipulation|                                     |
|=====|
|*|
```

```
Build : v1806
Arch : "LSB;label=32;scalar=64"
Exec : decomposePar -case /home/dexcs/Desktop/test
Date : Sep 22 2018
Time : 07:44:03
Host : "dexcs2017Test-virtual-machine"
PID : 27604
```

ファイル(F) 編集(E) 表示(V) 検索(S) 端末(T) ヘルプ(H)

```
Sum of forces
Total : (22.5213 -4.8279 -0.808555)
Pressure : (22.3664 -4.82867 -0.806692)
Viscous : (0.154886 0.000765529 -0.00186258)
Sum of moments
Total : (3.54475 -1.76761 106.219)
Pressure : (3.54067 -1.78356 105.651)
Viscous : (0.00408033 0.0159501 0.56745)
```

forceCoeffs forceCoeffs execute:

Coefficients		
Cm	: 0.0204031	(pressure: 0.019819 viscous: 0.000584107)
Cd	: 0.367695	(pressure: 0.365166 viscous: 0.00252876)
Cl	: -0.0132009	(pressure: -0.0131705 viscous: -3.04095e-05)
Cl(f)	: 0.0138027	
Cl(r)	: -0.0270036	

```
surfaceFieldValue inletMassFlow write:
sum(inlet) of phi = -160
```

```
surfaceFieldValue outletMassFlow write:
sum(outlet) of phi = 160
```

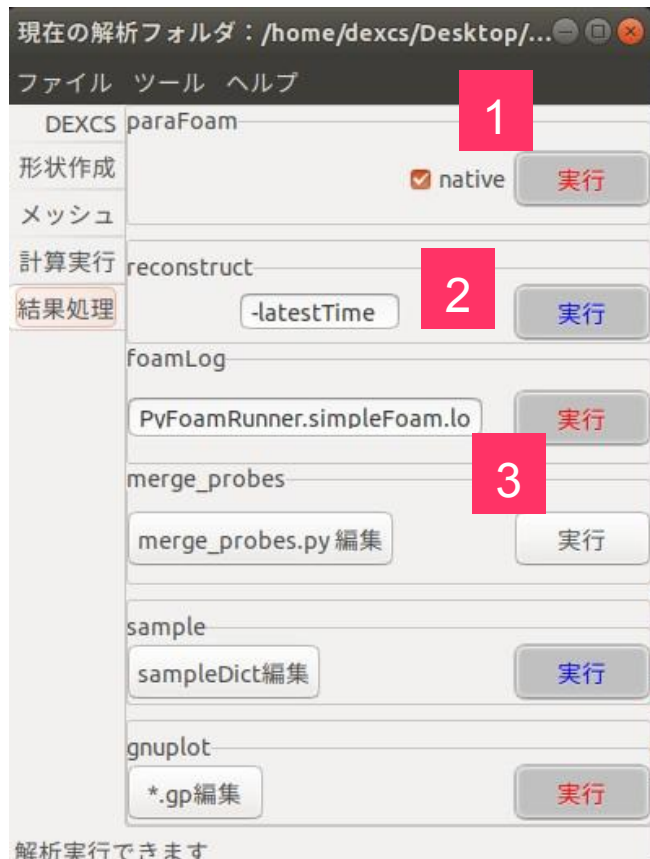
End

Finalising parallel run

計算終了

スクロールバックして計算時間情報など読み取ることができます

結果処理メニュー



1



流れ場を可視化します
(並列計算結果の場合)



同上
(結合結果のある場合)

2



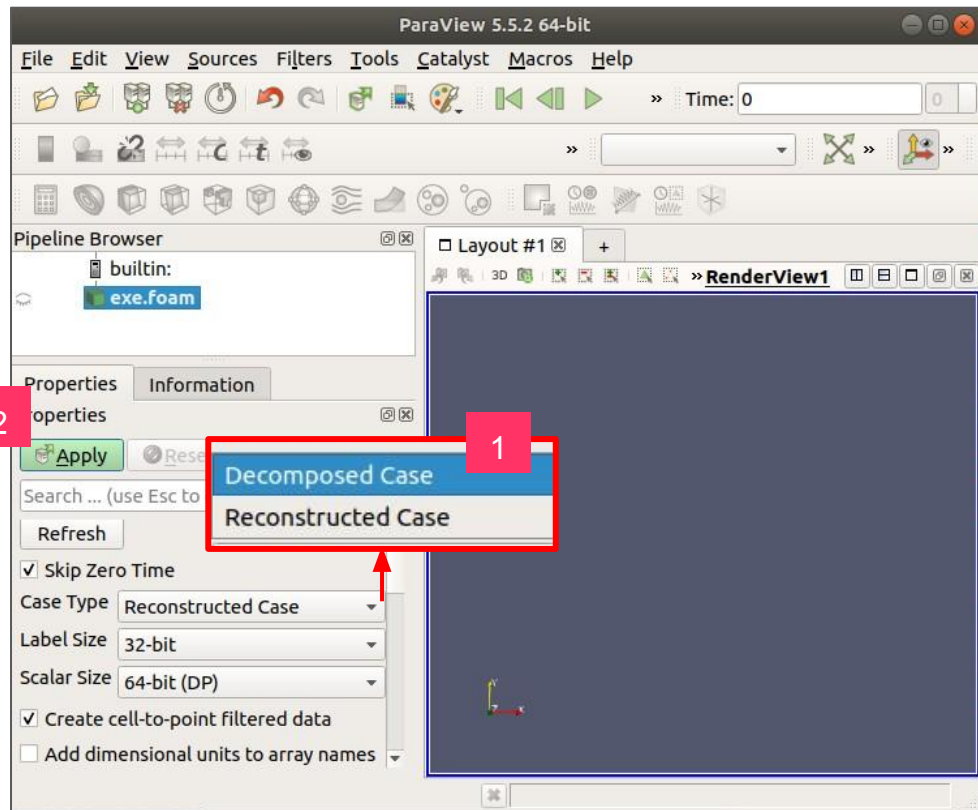
並列計算結果を結合します
(p.43)

3

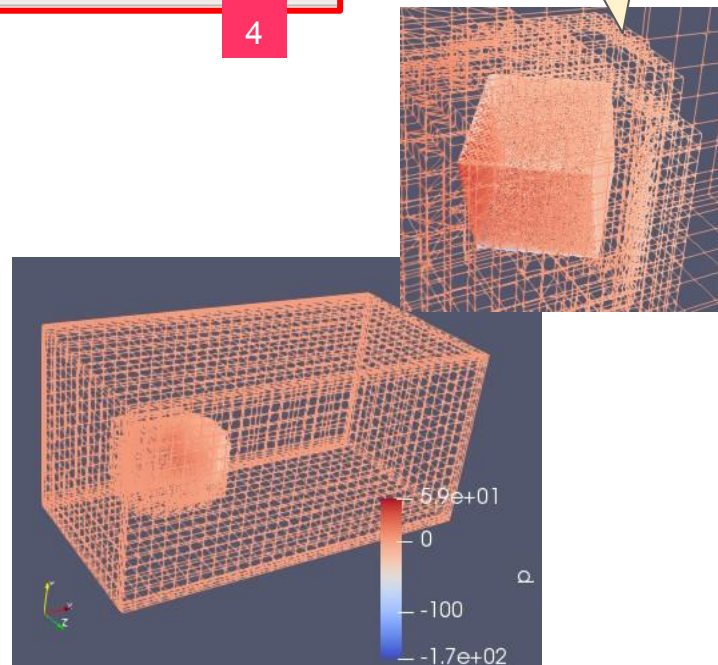


計算実行結果のログを解析します

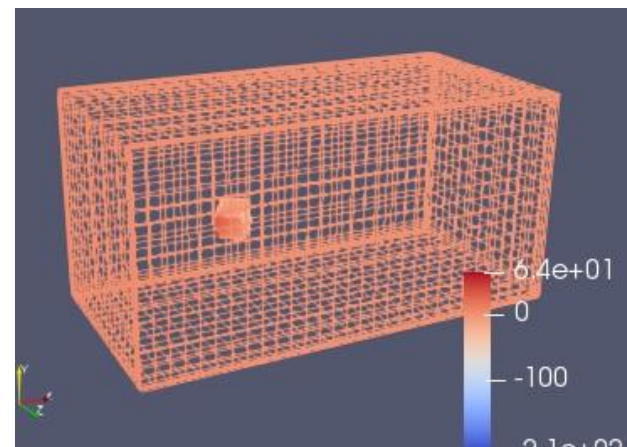
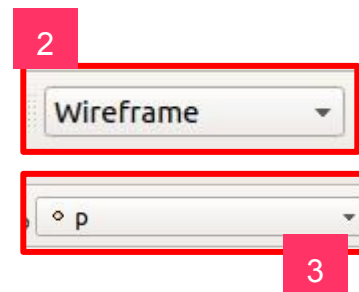
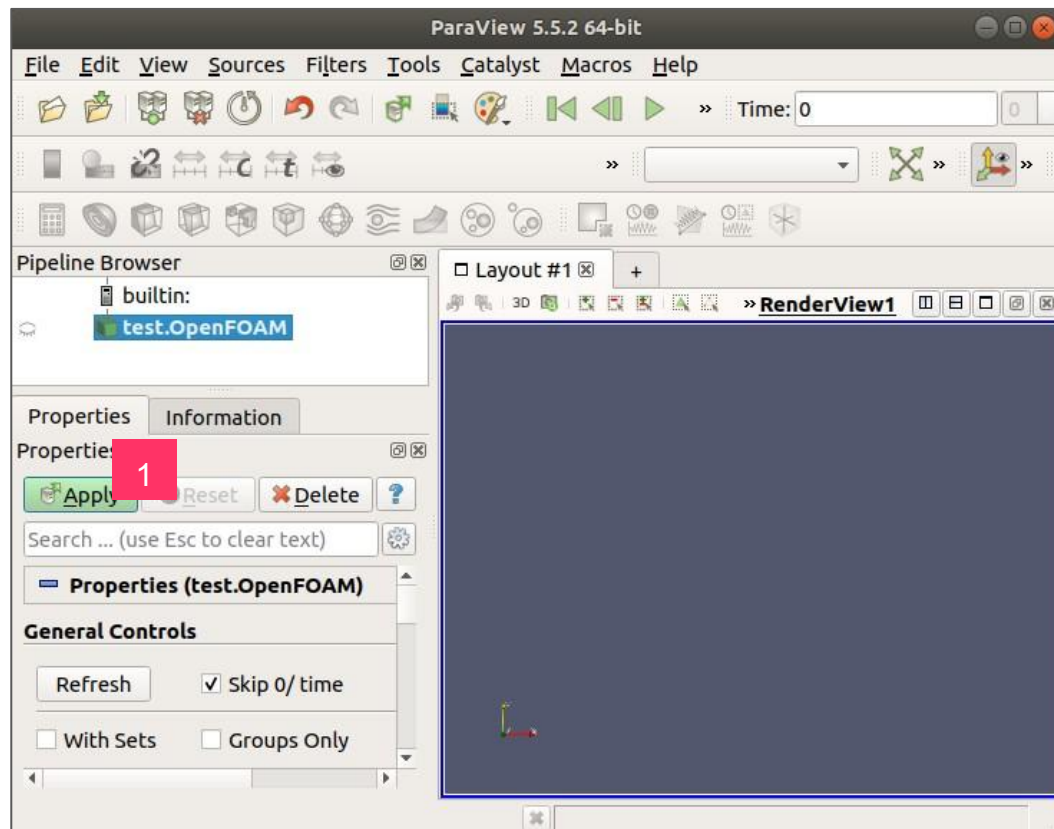
paraView(native)



並列領域分割
境界

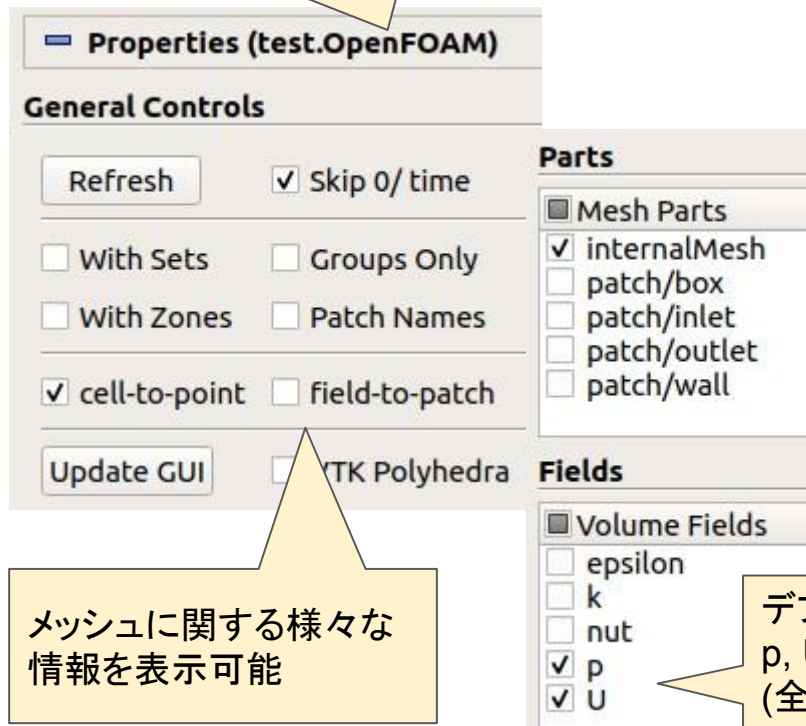


paraFoam



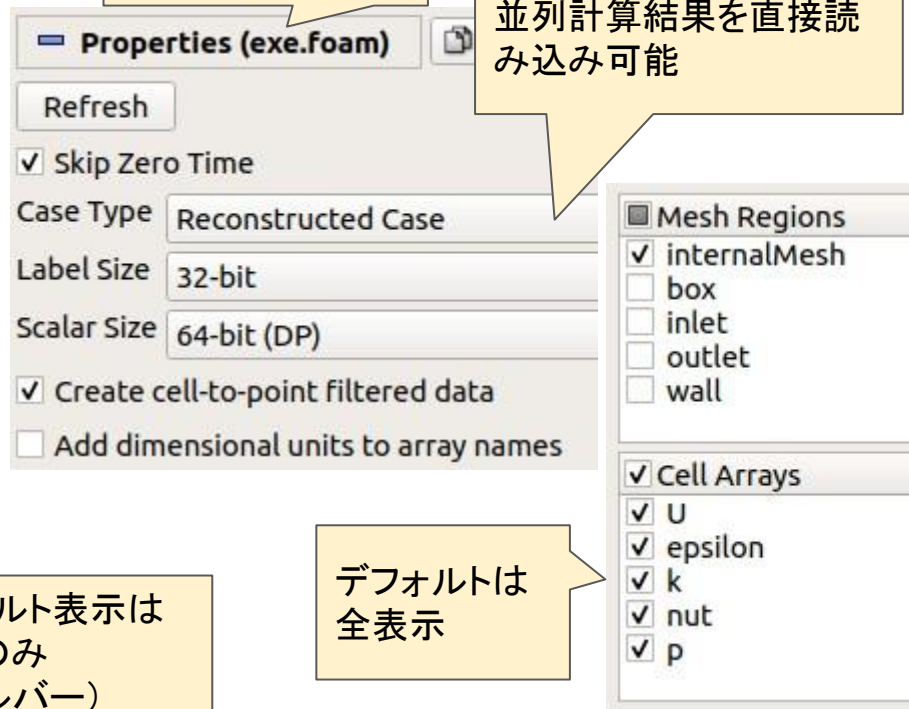
paraFoam vs paraview(native)

拡張子は.OpenFOAM



メッシュに関する様々な
情報を表示可能

拡張子は.foam

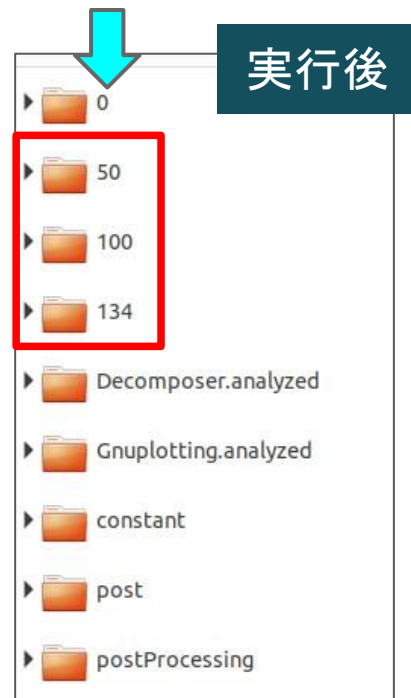
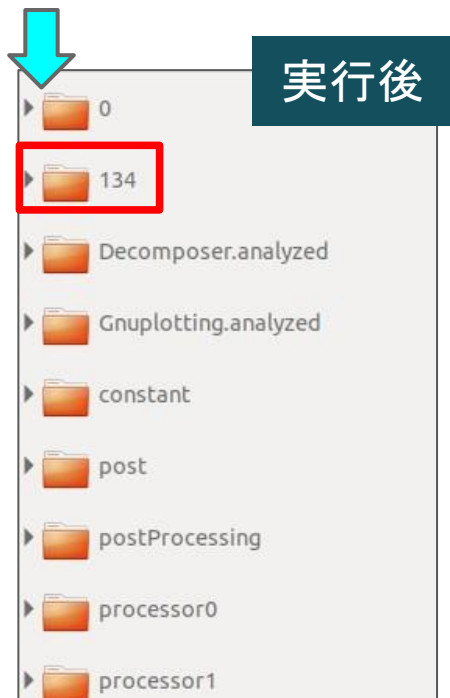
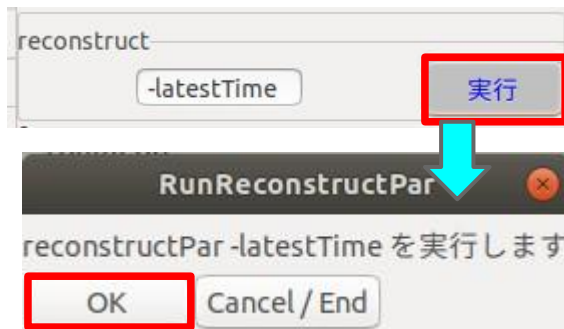
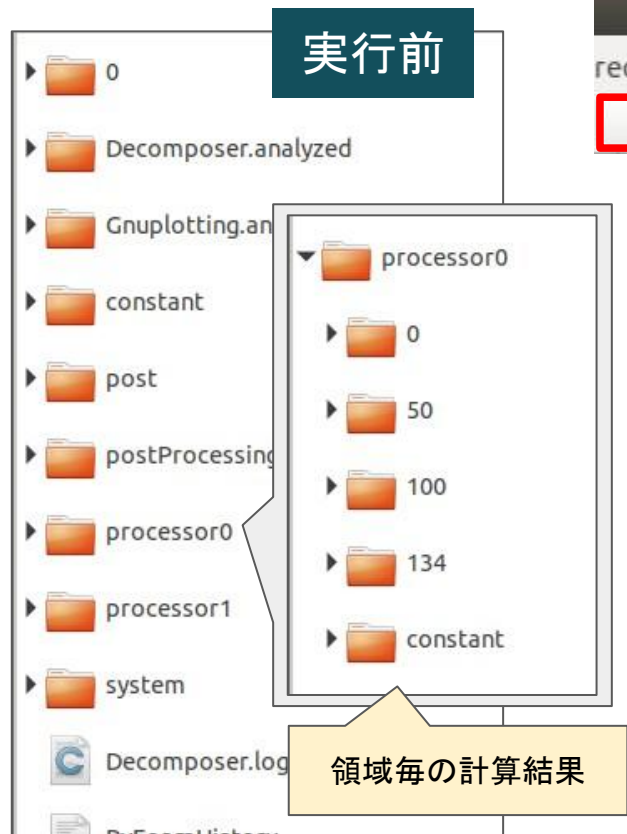


並列計算結果を直接読み込み可能

デフォルト表示は
p, U のみ
(全ソルバー)

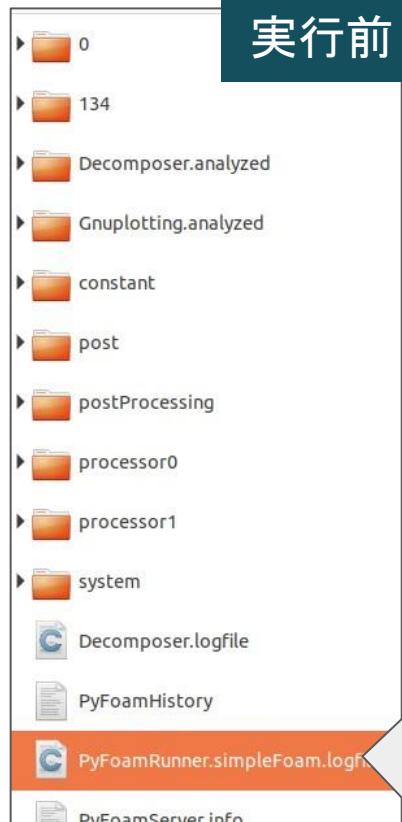
デフォルトは
全表示

reconstruct



foamLog

実行前



実行後

logs

UxFinalRes_0

UxIlters_0

Ux_0

UyFinalRes_0

UyIlters_0

Uy_0

UzFinalRes_0

UzIlters_0

Uz_0

	Ux_0
1 1	1
2 2	0.0907587
3 3	0.395163
4 4	0.0973121
5 5	0.256757
6 6	0.0877036
7 7	0.131683
8 8	0.0553431
9 9	0.0804026
10 10	0.0393708
11 11	0.0481312
12 12	0.0307301
13 13	0.0275244
14 14	0.0255606
15 15	0.019192
16 16	0.0195072
17 17	0.015607
18 18	0.016052
19 19	0.0153191
20 20	0.0139925
21 21	0.0130234
22 22	0.0139566
23 23	0.01352

PyFoamRunner.simpleFoam.logfile

~/Desktop/test

保存(S)

184 surfaceTetrahedra outletMassFlow write:

185 sum(outlet) of phi = 159.666

186

187 Time = 2

188

189 smoothSolver: Solving for Ux, Initial residual = 0.0907587, Final residual = 0.00561893, No Iterations 4

190 smoothSolver: Solving for Uy, Initial residual = 0.421121, Final residual = 0.0359183, No Iterations 4

191 smoothSolver: Solving for Uz, Initial residual = 0.45551, Final residual = 0.0335442, No Iterations 4

192 GAMG: Solving for p, Initial residual = 0.830919, Final residual = 0.033839, No Iterations 6

193 time step continuity errors : sum local = 0.0457377, global = -0.00519228, cumulative = -0.00780176

194 smoothSolver: Solving for epsilon, Initial residual = 0.514166, Final residual = 0.0281726, No Iterations 4

結果処理メニュー



4



計算を途中停止後に継続計算を実施した場合、結果処理データを結合します

4'

後処理方法を変更したら、応じて変更する必要があります (p.46)

5



計算結果データからのサンプリング処理を実行します

5'

サンプリング方法を変更したい場合に使います (p.47)

6



後処理用に出力した様々なデータをプロットグラフ化します

6'

後処理方法を変更した場合や、プロット方法を変更したい場合に使います (p.48 - 52)

merge_probes.py

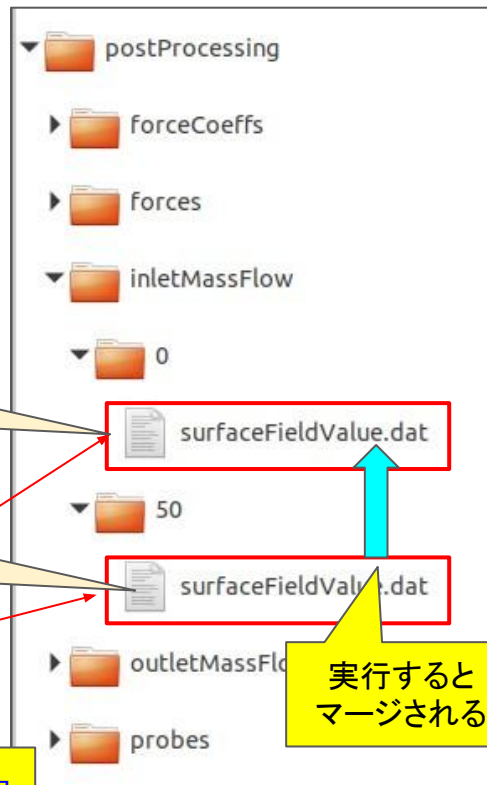
```
merge_probes.py
~/Desktop/test/post

14 if os.path.exists(probe_dir):
15     if os.path.exists(probe_dir + 'all'):
16         shutil.rmtree(probe_dir + 'all')
17     a = os.listdir(probe_dir)
18     a.sort()
19     os.mkdir(probe_dir + 'all')
20     for prop in probe_prop:
21         outfile = file(probe_dir + 'all' + os.sep + prop, 'a')
22         for f in a:
23             infile = probe_dir + f + os.sep + prop
24             print "merging " + infile + " to all/" + prop + "..."
25             f_in = open(infile, "r")
26             data = f_in.read()
27             outfile.write(data)
28             f_in.close()
29         outfile.close()
30     if os.path.exists(probe_dir + '0'):
31         shutil.rmtree(probe_dir + '0')
32     shutil.move(probe_dir + 'all', probe_dir + '0')
33     return 1
34 else:
35     print probe_dir + " does not exists"
36     return False
37
38 merge_probes( '../postProcessing/probes', 'p', 'U', 'k')
39 merge_probes( '../postProcessing/forces', 'force.dat', 'moment.dat')
40 merge_probes( '../postProcessing/inletMassFlow', 'surfaceFieldValue.dat')
41 merge_probes( '../postProcessing/outletMassFlow', 'surfaceFieldValue.dat')
42
```

Time=50を過ぎて
途中停止した際のデータ

Time=50からリスタート計
算した際のデータ

probesなどポスト処理方法を変更した場
合にはこの部分を適合させる



実行すると
マージされる

sampleDict

```
sampleDict
~/Desktop/test/system

50 // 1] vertex values determined from neighbouring cell centre values
51 // 2] face values determined using the current face interpolation scheme
52 //   for the field (linear, gamma, etc.)
53 interpolationScheme cellPoint;
54
55 // Fields to sample.
56 fields
57 (
58   p
59   U
60 ),
61
62
63 // Set sampling definition: choice of
64 //   uniform          evenly distributed points
65 //   face              one point per face
66 //   midPoint          one point per cell
67 //   midPointAndFace   combination of face and cell
68 //
69 //   curve             specified points, no tracking
70 //   cloud             specified points, uses findCell
71 //
72 //
73 // axis: how to write point coordinate. Choice of
74 // - x/y/z: x/y/z coordinate only
75 // - xyz: three columns
76 // (probably does not make sense for anything but raw)
77 // - distance: distance from start of sampling line (if uses line) or
78 //             distance from first specified sampling point
```

pとU
をサンプリング

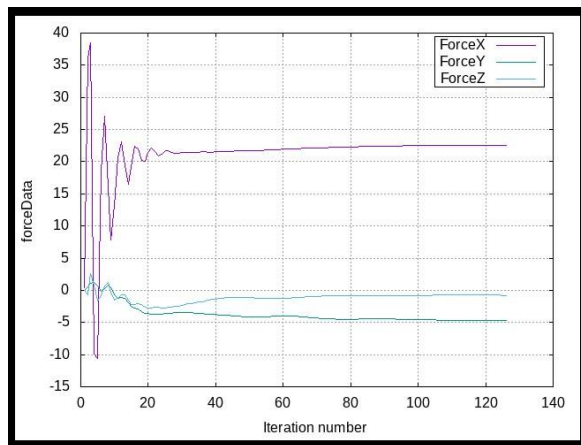
直線の、
・始点座標
・終点座標
・分割数
を指定

その他、様々なサンプリング方法が例
示されているので、必要に応じて行頭
のコメント(//)を削除し、パラメタを書き
換えて使用可能にできる

force.gp

```
force.gp  x  massFlowRate.gp  x  probes.gp  x  res.gp  x  sample.gp  x
1 set terminal png
2 set output "forces.png"
3 set xlabel "Iteration number"
4 set ylabel "forceData"
5 set grid
6 set key box
7 set style data line
8 plot '< sed -e "s/(//g" -e "s/)//g" "../postProcessing/forces/0/force.dat" ' using 1:2 title "ForceX",\
9      '< sed -e "s/(//g" -e "s/)//g" "../postProcessing/forces/0/force.dat" ' using 1:3 title "ForceY",\
10     '< sed -e "s/(//g" -e "s/)//g" "../postProcessing/forces/0/force.dat" ' using 1:4 title "ForceZ"
11 # EOF
```

データファイルから
かっこ()を除去



```
force.dat
~/Desktop/test/postProcessing/F

1 # Force
2 # CoFR      : (2.500000e-01 7.000000e-03 1.250000e-01)|
3 #
4 # Time      (total_x total_y total_z)      (pressure_x pressure_y pressure_z)
5 1          (3.329804e-01 1.129154e-02 -1.096486e-03)      (3.328736e-01 1.1287
6 2          (3.619900e+01 5.130349e-01 -6.408403e-01)      (3.611602e+01 5.1153
7 3          (3.850320e+01 1.118731e+00 2.612676e+00)      (3.841067e+01 1.1165
8 4          (-9.981202e+00 1.291975e+00 8.647341e-01)      (-1.005926e+01 1.288
9 5          (-1.065930e+01 6.456328e-01 -1.607499e+00)      (-1.069914e+01 6.421
10 6         (1.855690e+01 -2.075249e-01 -8.232481e-01)      (1.851981e+01 -2.091
11 7         (2.711810e+01 2.270055e-01 7.226954e-01)      (2.706735e+01 2.2551
12 8         (1.591482e+01 8.210717e-01 1.156706e+00)      (1.585777e+01 8.1882
13 9         (7.778153e+00 3.397491e-01 -3.906749e-01)      (7.722977e+00 3.3759
14 10        (1.320551e+01 -7.637901e-01 -1.564317e+00)      (1.315006e+01 -7.649
15 11        (2.063414e+01 -1.195344e+00 -1.309980e+00)      (2.057314e+01 -1.195
          (2.296625e+01 -1.0889
```

Iteration number

ForceX

ForceY

ForceZ

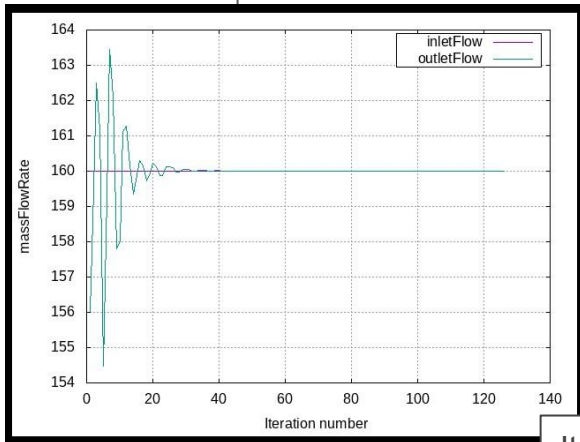
massFlowRate.gp

```
massFlowRate.gp
~/Desktop/test/post

force.gp x massFlowRate.gp x probes.gp x res.gp x sample.gp x

1 set terminal png
2 set output "massFlowRate.png"
3 set xlabel "Iteration number"
4 set ylabel "massFlowRate"
5 set grid
6 set key box
7 set style data line
8 plot "../postProcessing/inletMassFlow/0/surfaceFieldValue.dat" using 1:(-$2) title "inletFlow", \
9      "../postProcessing/outletMassFlow/0/surfaceFieldValue.dat" using 1:2 title "outletFlow"
10 # EOF
```

inletFlowは
符号を反転して表示



```
surfaceFieldValue.dat
~/Desktop/test/postProcessing/inletMassFlow/0

1 # Region type : patch inlet
2 # Faces      : 324
3 # Area       : 1.600000e+01
4 # Scale factor: 1.000000e+00
5 # Time       sum(phi)
6 1 1.560128e+02
7 2 1.594750e+02
8 3 1.625122e+02
9 4 1.611656e+02
10 5 1.544430e+02
11 6 1.594750e+02
12 7 1.634430e+02
13 8 1.621090e+02
14 9 1.678015e+02
15 10 1.680301e+02
16 11 1.600000e+02
17 12 1.600000e+02
18 13 1.600000e+02
19 14 1.600000e+02
```

inletFlow

outletFlow

Iteration number

Iteration number

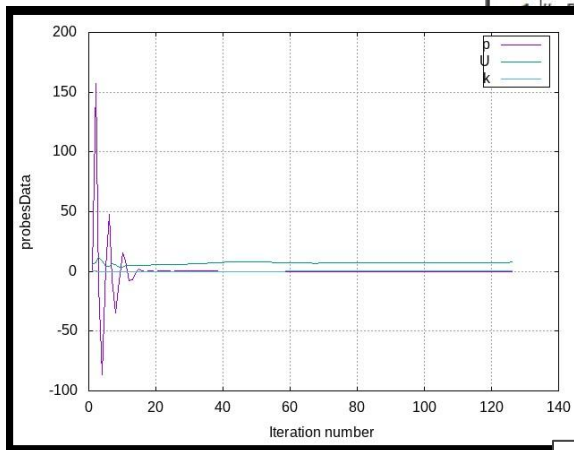
probes.gp

```

1 set terminal png
2 set output "probes.png"
3 set xlabel "Iteration number"
4 set ylabel "probesData"
5 set grid
6 set key box
7 set style data line
8 plot "../postProcessing/probes/0/p" title "p", \
9       '< sed -e "s/(//g" -e "s/)//g" "../postProcessing/probes/0/U" ' using 1:2 title "U", \
10      "../postProcessing/probes/0/k" title "k"
11 # EOF

```

データファイル(U)から
かっこ()を除去



Probe 0 (0 -3.5 0.2)

Time	0
1	0.589197
2	156.967
3	-12.472
4	-86.1278
5	3.85426
6	47.7662
7	-3.15365
8	-3.15365
9	-7.83717
10	15.5092
11	8.39877
12	-7.83717
13	-6.81535
14	0.0834249

p

Probe 0 (0 -3.5 0.2)

Time	0
1	0.227409
2	0.149952
3	0.103127
4	0.0721912
5	0.0540637
6	0.0408282
7	0.0315365
8	0.0215403
9	0.0202898
10	0.0184044
11	0.0170713
12	0.0161739
13	0.0154037
14	0.0154037

k

Probe 0 (0 -3.5 0.2)

Time	0
1	(5.97147 -0.136139 -0.0762368)
2	(7.46276 -0.262042 -0.0762368)
3	(11.9802 -0.271144 -0.0762368)
4	(8.99671 -0.509939 -0.0762368)
5	(4.04776 -0.65699 -0.0762368)
6	(4.44377 -0.267149 -0.0762368)
7	(6.51021 -0.0242591 -0.0762368)
8	(5.51021 -0.0762368 -0.0762368)
9	(0.0746158 -0.0762368 -0.0762368)
10	(3.61724 0.0406978 -0.0762368)
11	(4.76185 0.0215996 -0.0762368)
12	(5.11039 0.0557947 -0.0762368)

U

Iteration number

Iteration number

Iteration number

res.gp



UxFinalRes_0

UxItrs_0

Ux_0

UyFinalRes_0

UyItrs_0

Uy_0

UzFinalRes_0

UzItrs_0

Uz_0

epsilonFinalRes_0

epsilonItrs_0

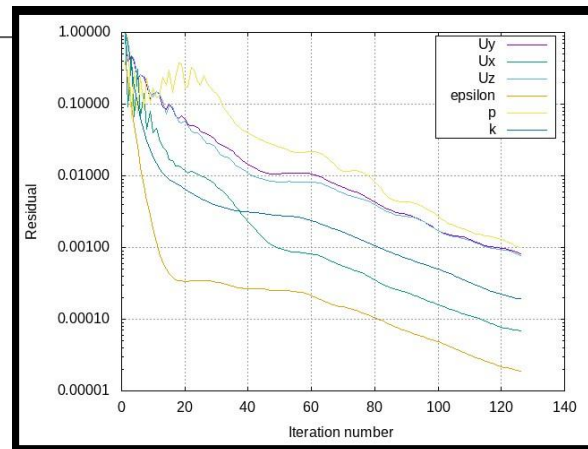
Iteration number

開<(O) ▾			Uy_0			保存(S)			≡		
			~/Des...								
1	1		1								
2	2		0.400187								
3	3		0.459478								
4	4		0.391224								
5	5		0.223119								
6	6		0.255343								
7	7		0.237549								
8	8		0.183244								
9	9		0								
10	10		0								
11	11		0.148448								
12	12		0.137133								
			0.0917529								
			0.0830714								
			0.0999555								
			0.0870401								

```
res.gp
~/Desktop/test/post

force.gp x
massFlowRate.gp x
probes.gp x
res.gp x

1 set terminal png
2 set output "res.png"
3 set logscale y 10
4 set xlabel "Iteration number"
5 set ylabel "Residual"
6 set grid
7 set key box
8 set style data line
9 plot "../logs/Uy_0" title "Uy", \
10      "../logs/Ux_0" title "Ux", \
11      "../logs/Uz_0" title "Uz", \
12      "../logs/epsilon_0" title "epsilon", \
13      "../logs/p_0" title "p", \
14      "../logs/k_0" title "k"
15 # EOF
```

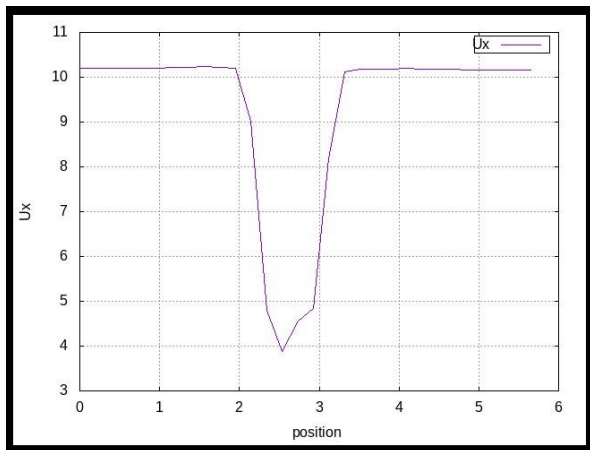


sample.gp

```
sample.gp
~/Desktop/test/post

force.gp x massFlowRate.gp x probes.gp x res.gp x sample.gp x

1 set terminal png
2 set output "sample.png"
3 set xlabel "position"
4 set ylabel "Ux"
5 set grid
6 set key box
7 set style data line
8 plot "../postProcessing/sampleDict/latestTime/lineX1_U.xy" using 1:2 title "Ux"
9 # EOF
```



lineX1_U.xy
~/Desktop/test/postProcessing/sampleDict/126

1	0	10.1907	-0.000150038	0.00141857
2	0.195064	10.1919	-0.00149217	0.0069926
3	0.390128	10.1936	-0.00217053	0.0133465
4	0.585192	10.1972	-0.0018755	0.0221088
5	0.780256	10.2023	-0.000972554	0.033152
6	0.97532	10.2089	0.00063807	0.0464684
7	1.17038	10.2147	0.0103772	0.0648698
8	1.36545	10.2163	0.0163263	0.0934167
9	1.56051	10.2172	0.0255676	0.136695
10	1.75558	10.2203	0.0239728	0.204849
11	1.95064	10.2013	-0.00618128	0.320215
12	2.1457	9.01688	-0.101841	0.496567
13	2.34077	4.80057	-0.200879	0.307666
		3.99845	-0.0073106	0.350349

position

Let's smart OpenCAE

powered by DEXCS presented by OCSE^2