

snappyHexMeshを使った 2次元モデル作成

2013年5月25日

オープンCAE勉強会@富山

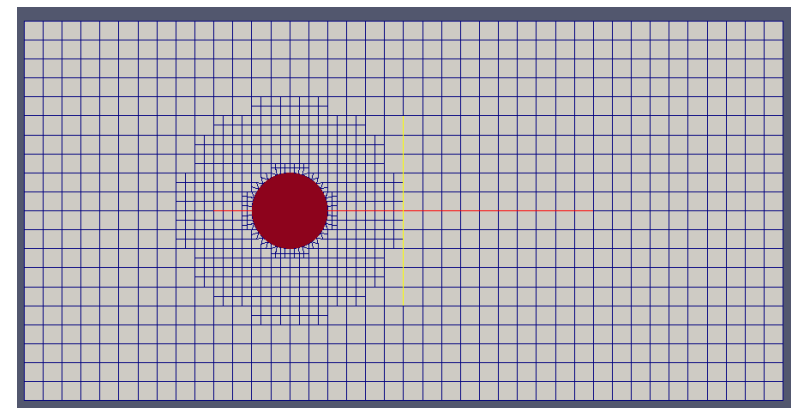
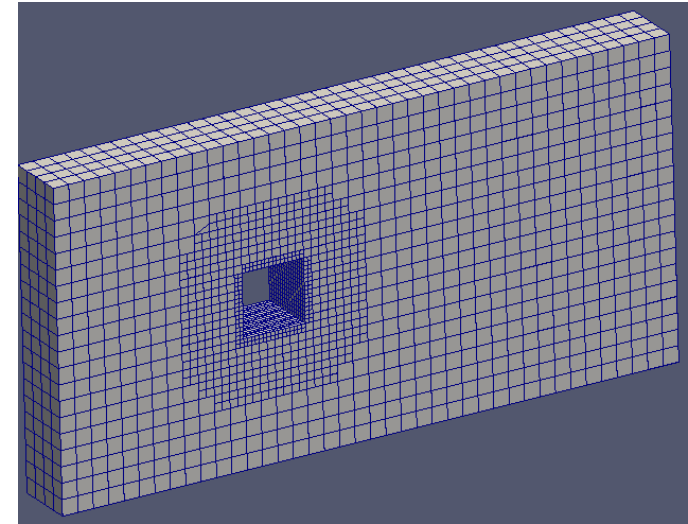
中川慎二

背景

- 2013年3月23日オープンCAE勉強会@富山でのsnappyHexMeshミニ講習会を開催した。
- 単純なモデル(平行平板間流路に角柱)を作成した。
- snappyHexMeshでは、3次元モデルを作成することになる。
- これをベースにして、2次元モデルを作成したい。

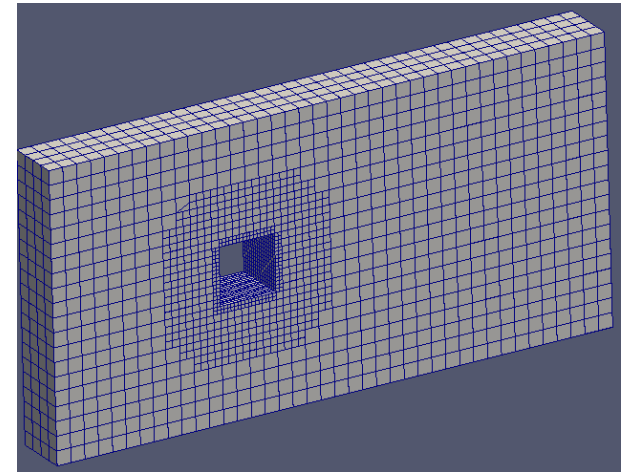
例題

- snappyTestRect
 - 計算領域: 直方体
 - 内部に四角柱を設置
 - 四角柱の周りにセルを生成
- snappyTestCyl
 - 計算領域: 直方体
 - 内部に円柱を設置
 - 円柱の周りにセルを生成

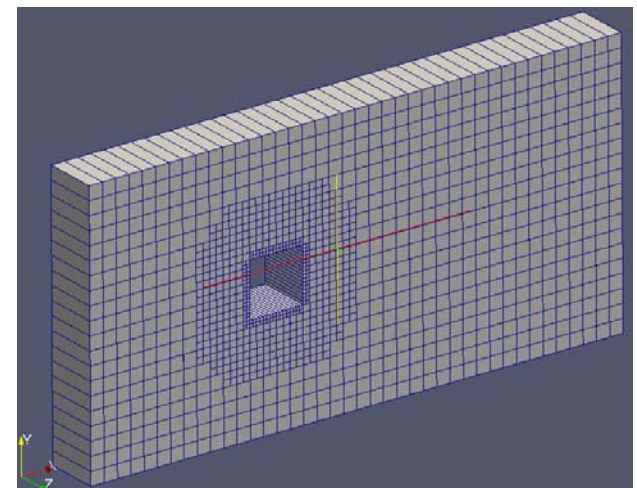


作業手順

- snappyHexMeshで3次元モデルを作成する。
 - snappyHexMesh ユーティリティ



- 境界面にできたセルを押し出し、2次元モデルを作成する。
 - extrudeMesh ユーティリティ



参考にする例題

`$FOAM_TUTORIALS/incompressible/pimpleDyMFoam/wingMotion/wingMotion2D_simpleFoam`

- 下記ファイルを元に修正する
`/system/extrudeMeshDict`

extrudeMesh -help

Usage: extrudeMesh [OPTIONS]

options:

- case <dir> specify alternate case directory, default is the cwd
- noFunctionObjects
do not execute functionObjects
- parallel run in parallel
- region <name> specify alternative mesh region
- roots <(dir1 .. dirN)>
slave root directories for distributed running
- srcDoc display source code in browser
- doc display application documentation in browser
- help print the usage

Using: OpenFOAM-2.2.0 (see www.OpenFOAM.org)

Build: 2.2.0-5be49240882f

wingMotion2D_simpleFoam/system/ extrudeMeshDict

```
// What to extrude:
//   patch : from patch of another case           expansionRatio  1.0;
('sourceCase')
//   mesh  : as above but with original case     linearNormalCoeffs
included                                           {
//   surface : from externally read surface       thickness  0.05;
                                                    }

constructFrom patch;
sourceCase "../wingMotion_snappyHexMesh";
sourcePatches (front);

// If construct from patch: patch to use for back (can
// be same as sourcePatch)
exposedPatchName back;

// Flip surface normals before usage. Valid only for
// extrude from surface or
// patch.
flipNormals false;

//- Linear extrusion in point-normal direction
extrudeModel  linearNormal;

nLayers      1;
```

wingMotion2D_simpleFoam/system/ extrudeMeshDict

```
// What to extrude:
// patch : from patch of another case
('sourceCase')
// mesh : as above but with original case
included
// surface : from external
expansionRatio 1.0;
linearNormalCoeffs
{
  thickness 0.05;
}
constructFrom patch;
sourceCase "../wingMotion_snappyHexMesh";
sourcePatches (front);
// If construct from patch: patch to use for back (can
be same as sourcePatch)
// Do front and back need to be merged? Usually
only makes sense for 360
// degree wedges.
mergeFaces false; //true;
exposedPatchName back;
Merge small edges. Fraction of bounding box.
mergeTol 0;
// Flip surface normals before usage. Valid only for
extrude from surface or
// patch.
flipNormals false;
//- Linear extrusion in point-normal direction
extrudeModel linearNormal;
nLayers 1;
```

どのケースから作るか。

全体の厚み。

どの面から作るか。

どの面まで作るか。

修正内容

- ソースケース
 - 現在のケースのメッシュを修正することとする。
 - そのため、sourceCase を、現在のディレクトリ“.”に変更する。
- 押出したい面(patch)の指定
 - 今回は、zMin 面を押出してセルをつくる。
 - そのため、sourcePatches に (zMin) を指定した。
 - さらに、どこまで押し出すかの設定として、exposedPatchName を zMax に指定した。
- セルの厚さ
 - 2次元モデルなので、厚さはいくらでも良いのだが、元のモデルと同じとするために、thickness を 0.01 とする。

改良 extrudeMeshDict

```
// What to extrude:
// patch : from patch of another case
('sourceCase')
// mesh : as above but with original case
included
// surface : from externally read surface

constructFrom patch;
sourceCase "."; // "../wingMotion_snappyHexMesh";
sourcePatches (zMin); //(front);

// If construct from patch: patch to use for back (can
be same as sourcePatch)
exposedPatchName zMax; //back;

// Flip surface normals before usage. Valid only for
extrude from surface or
// patch.
flipNormals false;

//- Linear extrusion in point-normal direction
extrudeModel linearNormal;

nLayers 1;

expansionRatio 1.0;

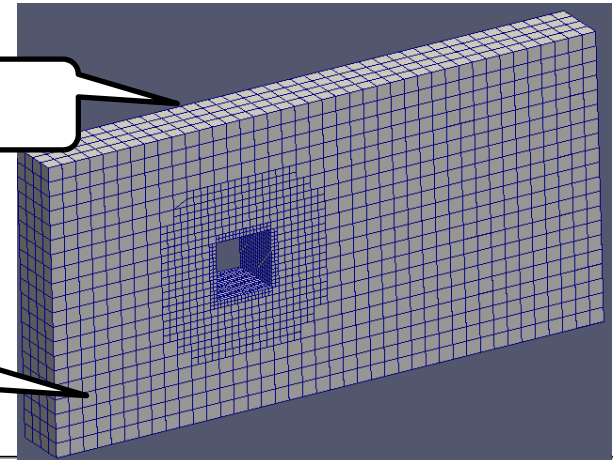
linearNormalCoeffs
{
    thickness 0.01;
}

// Do front and back need to be merged? Usually
only makes sense for 360
// degree wedges.
mergeFaces false; //true;

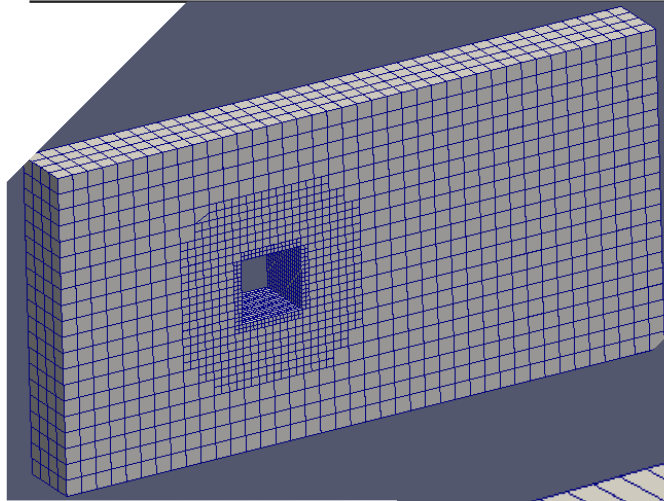
// Merge small edges. Fraction of bounding box.
mergeTol 0;
```

奥: zMin 面

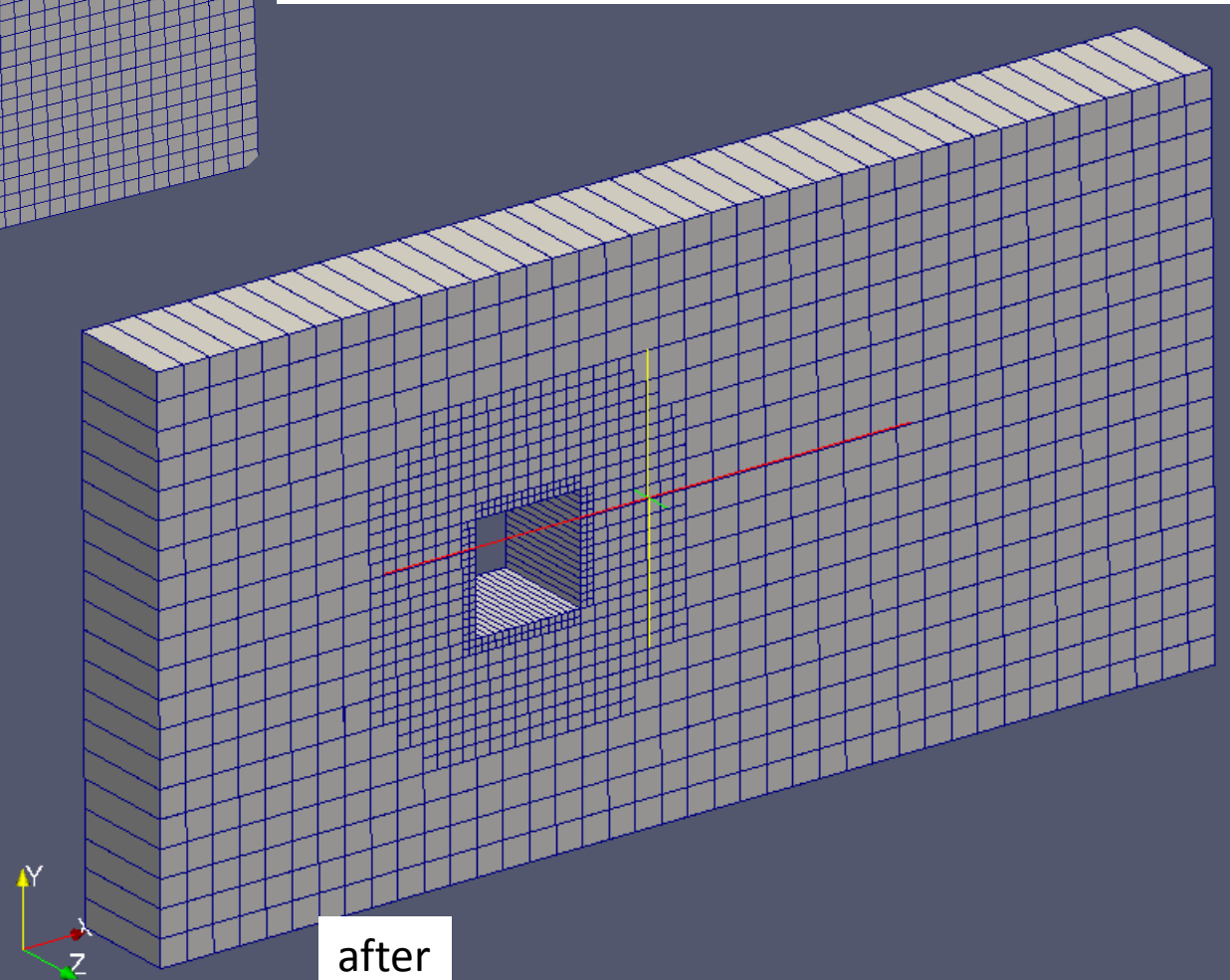
手前: zMax 面



2次元化したメッシュ



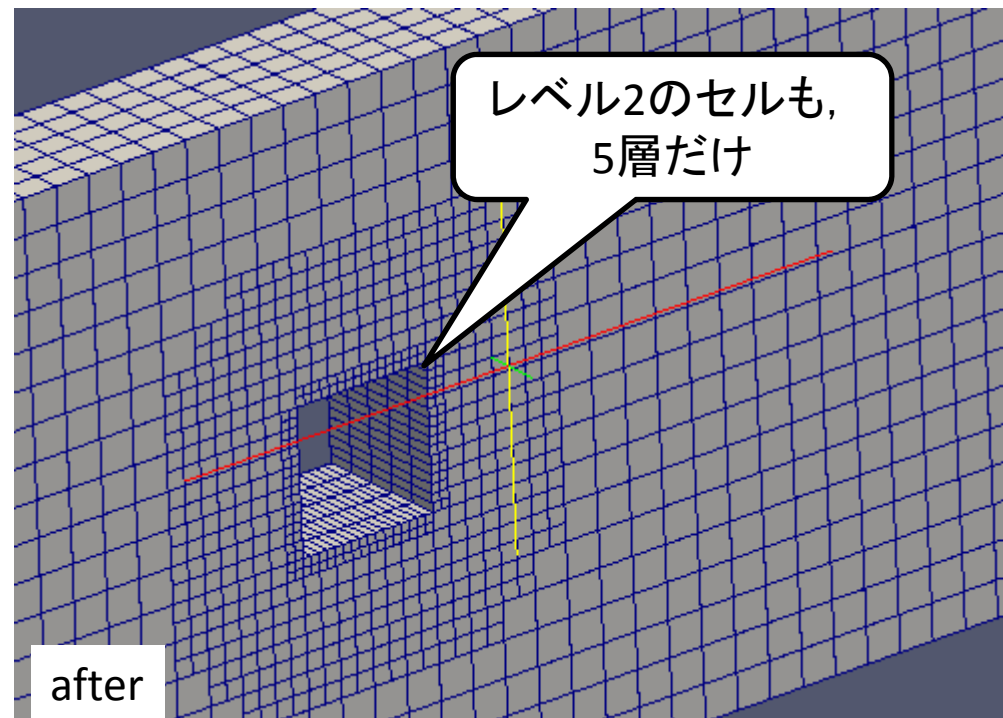
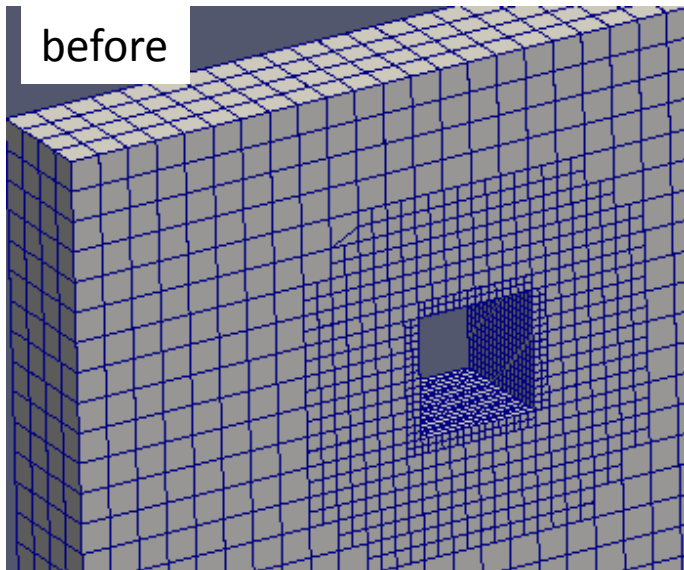
before



after

奥行き方向のセル数設定

- 先と同じメッシュで、奥行き方向のセル数を5にする。
- nLayers 5;



再改良 extrudeMeshDict

```
// What to extrude:
//   patch : from patch of another case
//   ('sourceCase')
//   mesh : as above but with original case
//   included
//   surface : from externally read surface

constructFrom patch;
sourceCase ".";
sourcePatches (zMin);

// If construct from patch: patch to use for back (can
// be same as sourcePatch)
exposedPatchName zMax;

// Flip surface normals before usage. Valid only for
// extrude from surface or
// patch.
flipNormals false;

//- Linear extrusion in point-normal direction
extrudeModel linearNormal;

nLayers 5;

expansionRatio 1.0;

linearNormalCoeffs
{
    thickness 0.01;
}

// Do front and back need to be merged? Usually
// only makes sense for 360
// degree wedges.
mergeFaces false; //true;

// Merge small edges. Fraction of bounding box.
mergeTol 0;
```

実行手順

各種ディクショナリを用意して、次のコマンドを順に実行する

blockMesh

surfaceFeatureExtract

snappyHexMesh

extrudeMeshDict