

本日の内容

- ポテンシャル流れについて教えてください。

流体の方程式

- スカラー場 $p(x,y,z)$ とベクトル場
 $U(x,y,z)=(u(x,y,z),v(x,y,z),w(x,y,z))$ が質量保存の式(次のスライド)と運動量保存の式(その次のスライド)を満たすとき定常非圧縮流れという

質量保存の式(定常非圧縮)

- 次の等式(質量の時間微分/体積)

$$\rho \nabla \cdot U = 0$$

- ただし ρ は密度、 $\nabla \cdot U$ は次のこと

$$\frac{\partial U_i}{\partial x_i} = u_x + v_y + w_z = \frac{\partial u}{\partial x} + \frac{\partial v}{\partial y} + \frac{\partial w}{\partial z}$$

運動量保存の式(定常非圧縮)

- 次の等式(運動量の時間微分/体積)

$$\rho(U \cdot \nabla)U = -\nabla p + R + F$$

- ここで ρ は密度(定数)、 R は粘性、 F は外力の項
- simpleFoamでいえば、 R は乱流モデル、 F はsourcesとかを使えばいじれる

- ただし $(U \cdot \nabla)U$ は次のこと

$$U_i \frac{\partial U_j}{\partial x_i} = \begin{pmatrix} uu_x + vu_y + wu_z \\ uv_x + vv_y + wv_z \\ uw_x + vw_y + ww_z \end{pmatrix}$$

流れ

- 以上2つの式を満たす (U, p) を定常非圧縮流れ(の流速と圧力)という

ポテンシャル流れ

- 非圧縮定常流れがさらに次を満たすときポテンシャル流れという

- $R=0, F=0$

- あるスカラー場 ϕ があつて $\nabla\phi = U$

- 結果的には (U, p, ϕ) に関する次の3つの式になる

$$\nabla\phi = U, \quad \rho\nabla \cdot U = 0, \quad \rho(U \cdot \nabla)U = -\nabla p$$

- ✓ 下線部の認識でOKでしょうか？

ポテンシャル流れの式変形

- 運動量保存の式は次のようになる

$$\frac{\rho}{2} \nabla |\nabla \phi|^2 = -\nabla p$$

- 導出: $U_i \frac{\partial U_j}{\partial x_i} = \frac{\partial \phi}{\partial x_i} \frac{\partial \phi}{\partial x_i \partial x_j} = \frac{1}{2} \frac{\partial}{\partial x_j} \left(\frac{\partial \phi}{\partial x_i} \frac{\partial \phi}{\partial x_i} \right) = \frac{1}{2} \frac{\partial}{\partial x_j} |\nabla \phi|^2$

- よってCを定数として $p = -\frac{\rho}{2} |\nabla \phi|^2 + C$ となる
- 質量保存の式は次のようになる

$$\Delta \phi = 0$$

- ✓ 以上の認識でOKでしょうか？

potentialFoamの特徴

(初心者にはこの説明ではわからないかもしれませんが許してください)

- ポテンシャル流れを解くソルバ
- チュートリアルではmotorBike等で使われている
- writepやinitialiseUBCsのオプションがある
- 場所は
applications/solvers/basic/potentialFoam

potentialFoamでは

```
00065 fvScalarMatrix pEqn
00066 (
00067     fvm::laplacian
00068     (
00069         dimensionedScalar
00070         (
00071             "1",
00072             dimTime/p.dimensions()*dimensionSet(0, 2, -2, 0,
00073             1
00074         ),
00075         p
00076     )
00077 ==
00078     fvc::div(phi)
00079 );
```

potentialFoam.C

```
00042 template<class Type>
00043 void surfaceIntegrate
00044 (
00045     Field<Type>& ivf,
00046     const GeometricField<Type, fvsPatchField, surfaceMesh>& ssf
00047 )
00048 {
00049     const fvMesh& mesh = ssf.mesh();
00050
00051     const labelUList& owner = mesh.owner();
00052     const labelUList& neighbour = mesh.neighbour();
00053
00054     const Field<Type>& issf = ssf;
00055
00056     forAll(owner, facei)
00057     {
00058         ivf[owner[facei]] += issf[facei];
00059         ivf[neighbour[facei]] -= issf[facei];
00060     }
00061
00062     forAll(mesh.boundary(), patchi)
00063     {
00064         const labelUList& pFaceCells =
00065             mesh.boundary()[patchi].faceCells();
00066
00067         const fvsPatchField<Type>& pssf = ssf.boundaryField()[patchi];
00068
00069         forAll(mesh.boundary()[patchi], facei)
00070         {
00071             ivf[pFaceCells[facei]] += pssf[facei];
00072         }
00073     }
00074
00075     ivf /= mesh.V();
00076 }
```

surfaceIntegrate.C

$\Delta p = \nabla \cdot U$ を解いているのではないか？
境界条件はどうなる？
pは圧力じゃないのか？
⇒次回(?)