

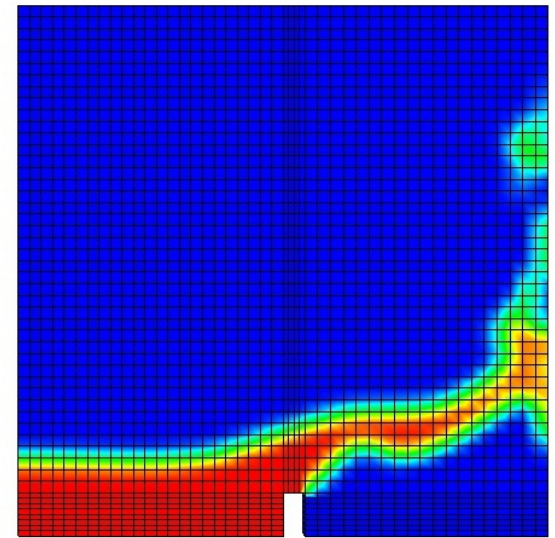
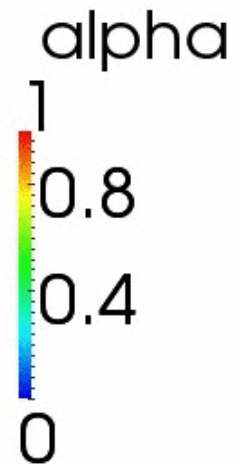
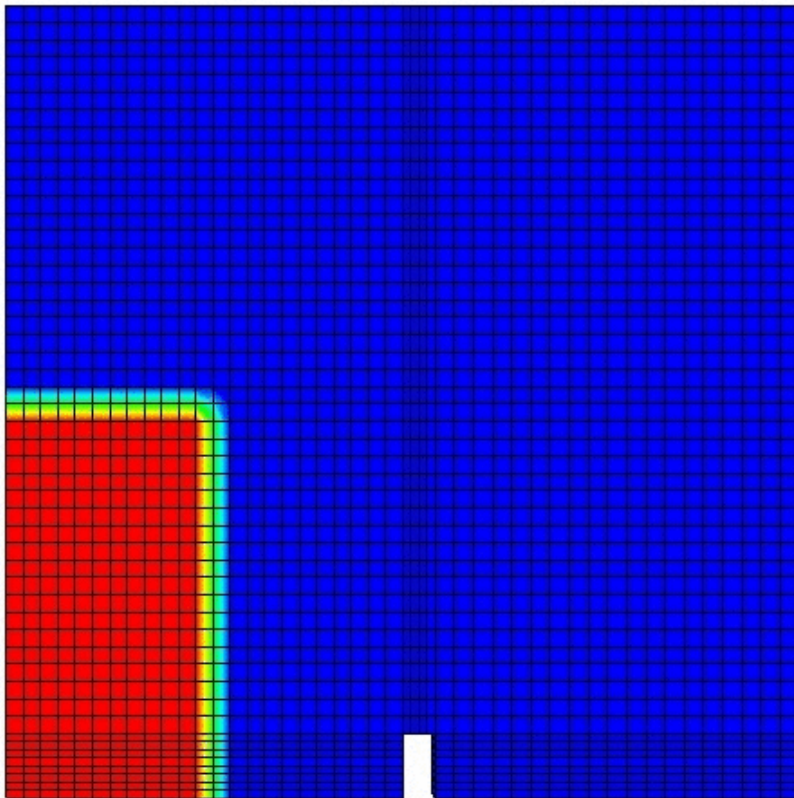


interFoamのパラ メータ設定方法

大阪大学大学院基礎工学研究科
修士2年 山本 卓也

interFoamとは？

Dam Break (Tutorial)



混相流(水/気体等)のシミュレーションを行うsolver



InterFoamを使う際に用いられているパラメータ

InterFoam内で用いられているパラメータ

System/fvSolution中

```
PISO
{
  nCorrectors      3;
  nNonOrthogonalCorrectors 0;
  nAlphaCorr      1;
  nAlphaSubCycles 2;
  cAlpha          1;
}
```

PISO法に用いられる修正パラメータ

格子歪み等がひどいときに用いる圧力場を安定的に解くための反復計算用のパラメータ

3つのパラメータ

InterFoamに用いられている

これらはどういう意味か??



InterFoamのソースコード解説

Ver. 1.6.x

• 支配方程式

Navier-Stokes 式

$$\frac{\partial \mathbf{v}}{\partial t} + \mathbf{v} \cdot \nabla \mathbf{v} = -\nabla P + \nu \nabla^2 \mathbf{v} + \mathbf{F}_\sigma + \rho \mathbf{g}$$

$$\mathbf{F}_\sigma = \sigma \kappa \mathbf{n} \delta_s$$

流体率 α の移流方程式

$$\frac{\partial \alpha}{\partial t} + \nabla \cdot (\alpha \mathbf{v}) = 0$$

$$\alpha = 1 \quad :: \text{liquid phase}$$

$$0 < \alpha < 1 \quad :: \text{interface}$$

$$\alpha = 0 \quad :: \text{gas phase}$$

$$\rho = \alpha \rho_g + (1 - \alpha) \rho_l$$

$$\mu = \alpha \mu_g + (1 - \alpha) \mu_l$$

$$\frac{\partial \alpha}{\partial t} + \nabla \cdot (\alpha \mathbf{v}) = 0$$



液相領域 $\frac{\partial \alpha}{\partial t} + \nabla \cdot (\alpha \mathbf{v}_l) = 0$

固相領域 $\frac{\partial \alpha}{\partial t} + \nabla \cdot ((1 - \alpha) \mathbf{v}_g) = 0$

小文字 l, g はそれぞれ液相、気相を表す。

再定義

$$\mathbf{v} = \alpha \mathbf{v}_l + (1 - \alpha) \mathbf{v}_g$$

$$\mathbf{v}_r = \mathbf{v}_l - \mathbf{v}_g$$

\mathbf{v}_r : 相関速度 (圧縮速度)

InterFoamのソースコード解説

Ver. 1.6.x

- 支配方程式

Navier-Stokes 式

$$\frac{\partial \mathbf{v}}{\partial t} + \mathbf{v} \cdot \nabla \mathbf{v} = -\nabla P + \nu \nabla^2 \mathbf{v} + \mathbf{F}_\sigma + \rho \mathbf{g}$$

$$\mathbf{F}_\sigma = \sigma \kappa \mathbf{n} \delta_s$$

流体率 α の移流方程式

$$\frac{\partial \alpha}{\partial t} + \nabla \cdot (\alpha \mathbf{v}) = 0$$

$\alpha = 1$:: liquid phase

$0 < \alpha < 1$:: interface

$\alpha = 0$:: gas phase

$$\rho = \alpha \rho_g + (1 - \alpha) \rho_l$$

$$\mu = \alpha \mu_g + (1 - \alpha) \mu_l$$

最終形

$$\frac{\partial \alpha}{\partial t} + \nabla \cdot (\alpha \mathbf{v}) + \nabla \cdot ((1 - \alpha) \alpha \mathbf{v}_r) = 0$$

alphaEqn.H中で設定

この項は界面上に働くもの
(1- α) α が入っているため



実際の物理現象では界面厚みがないため数値計算のために用いられる仮想的なもの

InterFoamのソースコード解説

Ver. 1.6.x

$$\frac{\partial \alpha}{\partial t} + \nabla \cdot (\alpha \mathbf{v}) + \nabla \cdot ((1 - \alpha) \alpha \mathbf{v}_r) = 0$$



離散化 (program中で解く形に変換)

α 式の設定

$$\frac{d}{dt} \int_{\Delta V} \alpha dV + \int_S \alpha \mathbf{v} \cdot \mathbf{n} dS + \int_S (1 - \alpha) \alpha \mathbf{v}_r \cdot \mathbf{n} dS$$

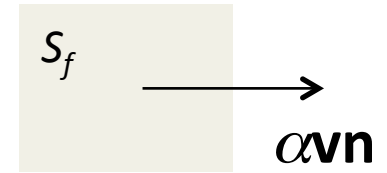
離散化

中点公式により近似

$$(\alpha \mathbf{v} \cdot \mathbf{n})_f \cdot S_f$$

$$((1 - \alpha) \alpha \mathbf{v}_r \cdot \mathbf{n})_f \cdot S_f$$

イメージ



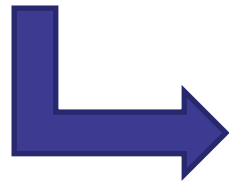
ここで、 f はセル界面上を表す。
 S_f は表面積

InterFoamのソースコード解説

Ver. 1.6.x

人工的に界面幅を圧縮する項

$$\left((1 - \alpha) \alpha \underline{v}_r \cdot \underline{n} \right)_f \cdot S_f$$



OpenFOAMのコード内

$$n_f \min \left[C_\alpha \frac{|u_f \cdot S_f|}{|S_f|}, \max \left(\frac{|u_f \cdot S_f|}{|S_f|} \right) \right]$$

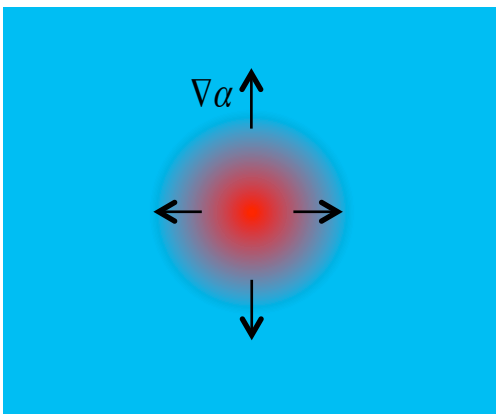
α 場 (赤:流体, 青:気体)

$$n_f = n_{fv} \cdot S_f$$

$$n_{fv} = \frac{(\nabla \cdot \alpha)_f}{|(\nabla \cdot \alpha)_f + \delta_N|}$$

$$\delta_N = \frac{1.0e^{-8}}{\left(\sum_N V_i / N \right)^{1/3}}$$

解を安定化するもの
(n_{fv} が無限大になるのを防ぐ)





InterFoamを使う際に 用いられているパラメータ

InterFoam内で用いられているパラメータ

System/fvSolution中

PISO

{

nCorrectors 3;

nNonOrthogonalCorrectors 0;

nAlphaCorr 1;

nAlphaSubCycles 2;

cAlpha 1;

}

nAlphaCorr; ??

nAlphaSubCycles; ??

cAlpha;

先程のスライドで出てきた C_α の値を設定
(値0~2)

$$n_f \min \left[C_\alpha \frac{|u_f \cdot S_f|}{|S_f|}, \max \left(\frac{|u_f \cdot S_f|}{|S_f|} \right) \right]$$

0; 界面圧縮が無い

2; 界面圧縮が強い



nAlphaSubCycles

alphaEqnSubCycle.H 中

```
if (nAlphaSubCycles > 1)
{
    dimensionedScalar totalDeltaT = runTime.deltaT();
    surfaceScalarField rhoPhiSum(0.0*rhoPhi);
    for
    (
        subCycle<volScalarField> alphaSubCycle(alpha1, nAlphaSubCycles);
        !(++alphaSubCycle).end();
    )
    {
        #include "alphaEqn.H"
        rhoPhiSum += (runTime.deltaT()/totalDeltaT)*rhoPhi;
    }
    rhoPhi = rhoPhiSum;
}
else
{
    #include "alphaEqn.H"
}
```

subCycleとは??

subCycle.Hの中をしてみる



subCycle

src/OpenFOAM/algorithms/subCycle/subCycle.H 中

元々 alphaSubCycles

元々 alpha1

```
subCycle(GeometricField& gf, const label nSubCycles)
:
  subCycleTime(const_cast<Time&>(gf.time()), nSubCycles),
  gf_(gf),
  gf0_(gf.oldTime())
{ }
```

次にsubCycleTimeとは？

subCycleTime.Cを次に見てみる



subCycleTime

src/OpenFOAM/db/Time/subCycleTime.C 中

```
Foam::subCycleTime::subCycleTime(Time& t, const
label nSubCycles)
:
    time_(t),
    nSubCycles_(nSubCycles),
    subCycleIndex_(0)
{
    time_.subCycle(nSubCycles_);
}
```

subCycle??
Time.Cをしてみる



subCycleTime

src/OpenFOAM/db/Time/Time.C 中

602行目から

```
Foam::TimeState Foam::Time::subCycle(const label nSubCycles)
{
    subCycling_ = true;
    prevTimeState_.set(new TimeState(*this));
    setTime(*this - deltaT(), (timeIndex() - 1)*nSubCycles);
    deltaT_ /= nSubCycles;
    deltaT0_ /= nSubCycles;
    deltaTSave_ = deltaT0_;

    return prevTimeState();
}
```

nSubCycles; 元々はalphaSubCycles

これよりnSubCyclesで時間刻みを割っている

要するにalpha式を解くときのみ時間刻みを小さくできる



InterFoamを使う際に用いられているパラメータ

InterFoam内で用いられているパラメータ

System/fvSolution中

PISO

```

{
  nCorrectors 3;
  nNonOrthogonalCorrectors 0;
  nAlphaCorr 1;
  nAlphaSubCycles 2;
  cAlpha 1;
}

```

nAlphaCorr; ??

nAlphaSubCycles;

α の式を解く際に時間刻み Δt を変更して解くことができる。

$\Delta t/nAlphaSubCycles$

cAlpha;

先程のスライドで出てきた C_α の値を設定
(値0~2)

$$n_f \min \left[C_\alpha \frac{|u_f \cdot S_f|}{|S_f|}, \max \left(\frac{|u_f \cdot S_f|}{|S_f|} \right) \right]$$

0; 界面圧縮が無い

2; 界面圧縮が強い



nAlphaCorr

alphaEqn.H中

```
for (int aCorr=0; aCorr<nAlphaCorr; aCorr++)  
{  
    surfaceScalarField phiAlpha  
    (  
        fvc::flux  
        (  
            phi,  
            alpha1,  
            alphaScheme  
        )  
    + fvc::flux  
    (  
        -fvc::flux(-phi, scalar(1) - alpha1, alphaScheme),  
        alpha1,  
        alphaScheme  
    )  
    );  
  
    MULES::explicitSolve(alpha1, phi, phiAlpha, 1, 0);  
  
    rhoPhi = phiAlpha*(rho1 - rho2) + phi*rho2;  
}
```

alpha式を解く反復回数

ちなみにfvc::flux(1,2,3)

1; surfaceScalarField

2; GeometricField

3; word & name

のとき、1 * 2のfluxを返す

詳しくはsrc/finiteVolume/finiteVolume/
fvc/fvcFlux.C

MULES?? 今度時間があれば説明します



InterFoamを使う際に用いられているパラメータ

InterFoam内で用いられているパラメータ

System/fvSolution中

PISO

{

```

nCorrectors 3;
nNonOrthogonalCorrectors 0;
nAlphaCorr 1;
nAlphaSubCycles 2;
cAlpha 1;

```

}

nAlphaCorr;

α の式を解く際の反復回数
(安定的に解くため)

nAlphaSubCycles;

α の式を解く際に時間刻み Δt を変更して解くことができる。

$\Delta t/nAlphaSubCycles$

cAlpha;

先程のスライドで出てきた C_α の値を設定
(値0~2)

$$n_f \min \left[C_\alpha \frac{|u_f \cdot S_f|}{|S_f|}, \max \left(\frac{|u_f \cdot S_f|}{|S_f|} \right) \right]$$

0; 界面圧縮が無い

2; 界面圧縮が強い



まとめ

- interFoamの変数の設定方法をソースコードからまとめました。
- MULESについては次の機会に説明したいと思います。

質問・ご指摘等あればよろしく願います。