FVM implementation of NS Equation discretization in OpenFoam

Ajit Kumar
Shiv Nadar University
Email: *ajit.kumar@snu.edu.in*

May 15, 2018

1 Incompressible NS Equation

$$\nabla \cdot \mathbf{U} = 0,$$
$$\frac{\partial \mathbf{U}}{\partial t} + \nabla \cdot (\mathbf{U}\mathbf{U}) - \nabla \cdot (\nu \nabla \mathbf{U}) = -\nabla p$$

2 Finite volume form of continuity equation

With a bit of abuse of notation, from now let us assume that $\mathbf{U}$ and $\mathbf{p}$ denote the discretized variables.

To mimick the OpenFoam implementation, lets assume that discretized form of the left hand side of the momentum equation is

$$\mathbf{A}\mathbf{U} - \mathbf{H}(U)$$

which is implemented in foam as

```
fvVectorMatrix UEqn
(
     fvm::ddt(U)
   + fvm::div(phi, U)
   + turbulence->divDevReff()
);
```

The matrix A is a diagonal matrix. The `turbulence->divDevReff()` is actually implementing

```
-fvm::laplacian(nuEff(), U) -fvc::div(nuEff()*dev(T(fvc::grad(U))))
```

which `nuEff()` represents effective viscosity which is calculated as fluid viscosity by turbulence viscosity, $\nu_{\text{eff}} = \nu + \nu_T$. The second term I believe represents explicit treatment of devioteric stress tensor which should have been vanished in incompressible flows, right?

Lets say the finite volume discretization of ∇p takes the shape $D\mathbf{p}$. This is never explicitly implemented in OpenFoam. Now say the discretized momentum equation looks like

$$\mathbf{U} = \frac{\mathbf{H}(\mathbf{U})}{\mathbf{A}} - \frac{\mathbf{D}}{\mathbf{A}}\mathbf{p}$$

Let $\mathbf{U}_f$ be the interpolation vector of velocities at cell faces and lets say the connection between $\mathbf{U}$ and $\mathbf{U}_f$ is through a mapping matrix M

$$\mathbf{U}_f = M\mathbf{U}$$

Discretized continuity equation is

$$\sum_{\text{faces in cell}} \mathbf{U}_f \cdot \mathbf{S}_f = 0, \text{ for all cells}$$

which in matrix form, suppose, looks like

$$C(M\mathbf{U}) = 0$$

So, the algebraic equation we intend to solve here is

$$\mathbf{U} = \frac{\mathbf{H}(\mathbf{U})}{\mathbf{A}} - \frac{\mathbf{D}}{\mathbf{A}}\mathbf{p}$$
$$C(M\mathbf{U}) = \mathbf{0}$$

3 PISO loop

- Presently known values of velocity and pressure: $\mathbf{U}^n$, $\mathbf{p}^n$.
- **Momentum predictor** Using $\mathbf{U}^n$, construct the linearization matrix $\mathbf{L}(\mathbf{U^n})$ and constant term $\mathbf{X}(\mathbf{U}^n)$and solve for a momentum predictor $\mathbf{U}^*$ from the equation $\mathbf{U}^*$ from the equation

  $$\mathbf{A}\mathbf{U}^* - \underbrace{[\mathbf{L}(\mathbf{U}^n)\mathbf{U}^* + \mathbf{X}(\mathbf{U}^n)]}_{=H(\mathbf{U}^*)} = -D\mathbf{p}^n$$

  For momentum prediction the above equation is rephrased in the form

  $$\mathbf{U}^* = \frac{H(\mathbf{U}^*)}{A} - \frac{D}{\mathbf{A}}\mathbf{p}^n$$

  The momentum predictor is not expected to be conservative, and is used only to construct $H(\mathbf{U}^*)$.
- **PISO** correction till desired conservation reached
  - Assume $\mathbf{U} = \frac{H(\mathbf{U}^*)}{A} - \frac{D}{A}\mathbf{p}$
  - Substitute in the continuity equation

    $$\frac{CMD}{A}\mathbf{p} = \frac{CM}{A}\mathbf{H}(\mathbf{U}^*)$$

  - Solve for $\mathbf{p}$