OpenFOAM® como herramienta de resolución de PDE's

Métodos Numéricos en Fluidos 2017

Federico Teruel

Ezequiel Fogliatto

Departamento de Mecánica Computacional - Centro Atómico Bariloche

Instituto Balseiro

7 de noviembre de 2017

Outline

Outline

Características principales

Field
Operation
And
Manipulation

Conjunto libre de bibliotecas de C++ para solución de ecuaciones diferenciales desarrollado por OpenCFD Ltd.

Aplicaciones

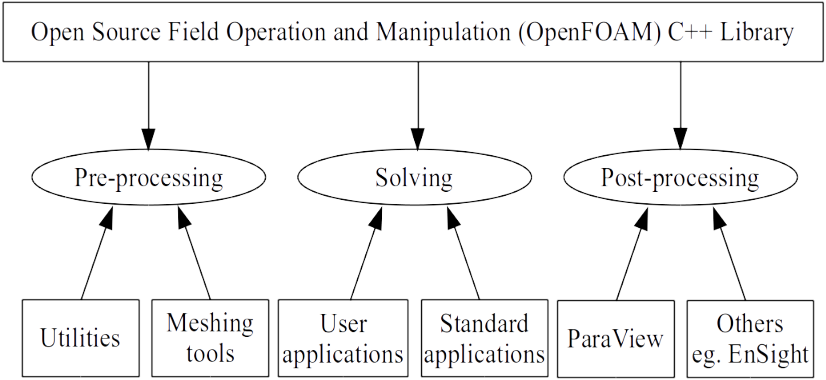
- Fluidos compresibles
- Fluidos incompresibles
- Flujos multifase
- Transferencia de calor
- Dinámica molecular
- Electromagnetismo

Método de volúmenes finitos en 3 dimensiones

Elementos poliédricos con cantidad ilimitada de caras

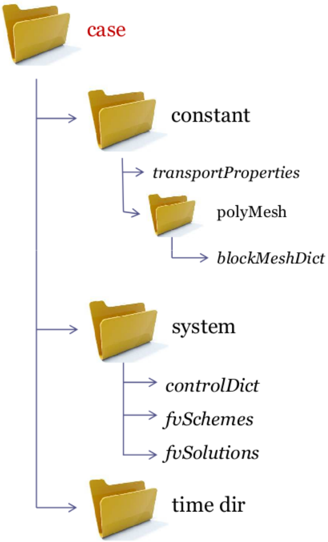
Gran flexibilidad en la descripción geométrica del dominio

Características principales



- Código libre y gratuito, bajo licencia GNU.
- Los usuarios pueden crear sus propias aplicaciones y utilidades.

Características principales




```
dimensions     [0 1 -1 0 0 0 0];
internalField  uniform (0 0 0);
boundaryField
{
   movingWall
     {
           type        fixedValue;
           value       uniform (1 0 0);
     }
   fixedWalls
     {
           type        fixedValue;
           value       uniform (0 0 0);
     }
   frontAndBack
     {
           type        empty;
     }
}
```

Outline

Origen

OpenFOAM® comenzó como parte de un trabajo de doctorado

H. Jasak. *Error Analysis and Estimation for Finite Volume Method with Applications to Fluid Flow*. PhD thesis, Imperial College, University of London, 1996.

Método de volúmenes finitos

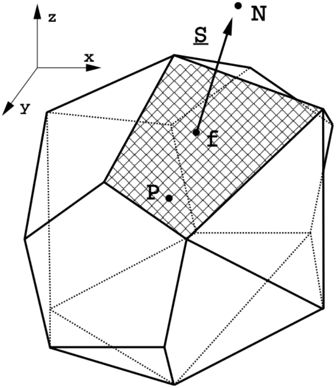
Discretización

El proceso de discretización está compuesto por dos etapas

1. Discretización del dominio
   - El espacio es dividido en un número finito de regiones discretas, llamadas volúmenes de control o celdas.
   - Para simulaciones transitorias, el tiempo también se descompone en un número finito de intervalos.
2. Discretización de las ecuaciones
   - Forma integral de las ecuaciones gobernantes sobre cada volumen de control.
   - Las ecuaciones se resuelven en un sistema de coordenadas cartesianas fijo a la malla, que no cambia con el tiempo.
   - Los volúmenes de control pueden adoptar cualquier forma polihédrica, lo que permite crear mallas no estructuradas. Todas las variables dependientes comparten los mismos volúmenes de control (arreglo **superpuesto** o **no desplazado**).
   - Los sistemas de ecuaciones diferenciales en derivadas parciales se resuelven en forma **segregada**.

Método de volúmenes finitos
Discretización del dominio

- Tiempo: la solución se obtiene marchando en el tiempo, a partir de la condición inicial prescripta.
- Espacio



El punto computacional **P** se encuentra en el centroide del volumen de control, tal que

$$\int_{V_p} (\mathbf{x} - \mathbf{x_p}) dV = 0$$

No importa la topología del VC.

Método de volúmenes finitos

Discretización de la ecuación de transporte

Ecuación de transporte prototipo

$$\underbrace{\frac{\partial \rho\phi}{\partial t}}_{1} + \underbrace{\nabla \cdot (\rho \mathbf{U} \phi)}_{2} - \underbrace{\nabla \cdot (\rho \Gamma_\phi \nabla \phi)}_{3} = \underbrace{S_\phi(\phi)}_{4}$$

Forma Integral

$$\underbrace{\frac{\partial}{\partial t} \int_{V_P} \rho\phi dV}_{1} + \underbrace{\int_{V_P} \nabla \cdot (\rho \mathbf{U} \phi) dV}_{2} - \underbrace{\int_{V_P} \nabla \cdot (\rho \Gamma_\phi \nabla \phi) dV}_{3} = \underbrace{\int_{V_P} S_\phi(\phi) dV}_{4}$$

1. Derivada temporal
2. Término convectivo
3. Término difusivo
4. Término fuente

Método de volúmenes finitos
Discretización de la ecuación de transporte

Aproximación de segundo orden

$$\phi(\mathbf{x}) = \phi_P + (\mathbf{x} - \mathbf{x}_P) \cdot (\nabla\phi)_P$$

$$\phi(t + \Delta t) = \phi^t + \Delta t (\frac{\partial \phi}{\partial t})^t$$

Se puede verificar mediante el desarrollo de Taylor de ϕ

$$\phi(\mathbf{x}) = \phi_P + (\mathbf{x} - \mathbf{x_p}) \cdot (\nabla\phi)_P + \frac{1}{2}(\mathbf{x} - \mathbf{x_P})^2 : (\nabla\nabla\phi)_P$$
$$+ \frac{1}{3!}(\mathbf{x} - \mathbf{x_P})^3 :: (\nabla\nabla\nabla\phi)_P$$
$$+ \ldots + \frac{1}{n!}(\mathbf{x} - \mathbf{x_P})^n \underbrace{::}_{n} (\underbrace{\nabla\nabla\ldots\nabla}_{n}\phi)_P$$

Método de volúmenes finitos

Discretización de la ecuación de transporte

Discretización de los términos espaciales usando identidades de Gauss

$$\int_V \nabla \cdot \mathbf{a} dV = \oint_{\partial V} d\mathbf{S} \cdot \mathbf{a} \qquad \int_V \nabla \cdot \phi dV = \oint_{\partial V} d\mathbf{S} \phi$$

$$\int_V \nabla \mathbf{a} dV = \oint_{\partial V} d\mathbf{S} \mathbf{a}$$

$$\begin{aligned}
\int_{V_P} \phi(\mathbf{x})\, dV &= \int_{V_P} \left[\phi_P + (\mathbf{x} - \mathbf{x_P}) \cdot (\nabla \phi)_P\right] dV \\
&= \phi_P \int_{V_P} dV + \left[\int_{V_P} (\mathbf{x} - \mathbf{x_P}) dV\right] \cdot (\nabla \phi)_P \\
&= \phi_P V_P
\end{aligned}$$

Método de volúmenes finitos
Discretización de la ecuación de transporte

Discretización de los términos espaciales usando identidades de Gauss

$$\int_V \nabla \cdot \mathbf{a} dV = \oint_{\partial V} d\mathbf{S} \cdot \mathbf{a} \qquad \int_V \nabla \cdot \phi dV = \oint_{\partial V} d\mathbf{S}\phi$$

$$\int_V \nabla \mathbf{a} dV = \oint_{\partial V} d\mathbf{S}\mathbf{a}$$

$$\int_{V_P} \nabla \cdot \mathbf{a}\, dV = \oint_{\partial V} d\mathbf{S} \cdot \mathbf{a} = \sum_f \left(\int_f d\mathbf{S} \cdot \mathbf{a} \right)$$

donde

$$\int_f d\mathbf{S} \cdot \mathbf{a} = \left(\int_f d\mathbf{S} \right) \cdot \mathbf{a_f} + \left[\int_f d\mathbf{S}(\mathbf{x} - \mathbf{x_f}) \right] : (\nabla \mathbf{a})_f = \mathbf{S} \cdot \mathbf{a}_f$$

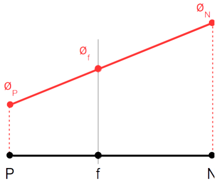
Método de volúmenes finitos

Discretización del término convectivo

Término convectivo

$$\frac{\partial}{\partial t}\int_{V_P}\rho\phi dV + \int_{V_P}\nabla\cdot(\rho\mathbf{U}\phi)dV - \int_{V_P}\nabla\cdot(\rho\Gamma_\phi\nabla\phi)dV = \int_{V_P}S_\phi(\phi)dV$$

$$\int_{V_P}\nabla\cdot(\rho\mathbf{U}\phi)dV = \sum_f \mathbf{S}\cdot(\rho U\phi)_f = \sum_f \mathbf{S}\cdot(\rho U)_f\phi_f = \sum_f F\phi_f$$



Esquemas de interpolación

$$\phi_f = \frac{\overline{fN}}{\overline{PN}}\phi_P + (1-\frac{\overline{fN}}{\overline{PN}})\phi_N$$ Diferencias centradas

$$\phi_f = \begin{cases} \phi_f = \phi_P & \text{si } F \geq 0 \\ \phi_f = \phi_N & \text{si } F \leq 0 \end{cases}$$ Upwind

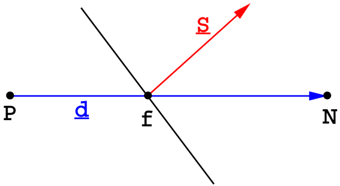
Método de volúmenes finitos

Discretización del término difusivo

Término difusivo

$$\frac{\partial}{\partial t}\int_{V_P} \rho\phi dV + \int_{V_P} \nabla\cdot(\rho\mathbf{U}\phi)dV - \int_{V_P} \nabla\cdot(\rho\Gamma_\phi\nabla\phi)dV = \int_{V_P} S_\phi(\phi)dV$$

$$\int_{V_P} \nabla\cdot(\rho\Gamma_\phi\nabla\phi)dV = \sum_f \mathbf{S}\cdot(\rho\Gamma_\phi\nabla\phi)_f = \sum_f (\rho\Gamma_\phi)_f\mathbf{S}\cdot(\nabla\phi)_f$$



Si la malla es ortogonal ($\mathbf{d} \parallel \mathbf{S}$)

$$\mathbf{S}\cdot(\nabla\phi)_f = \frac{|\mathbf{S}|}{|\mathbf{d}|}(\phi_N - \phi_P)$$

o bien

$$(\nabla\phi)_P = \frac{1}{V_P}\sum_f \mathbf{S}\phi_f$$

$$(\nabla\phi)_f = f_x(\nabla\phi)_P + (1-f_x)(\nabla\phi)_N$$

Método de volúmenes finitos
Discretización del término difusivo

Desafortunadamente, una malla ortogonal es más excepción que regla

Se divide el operador en dos partes

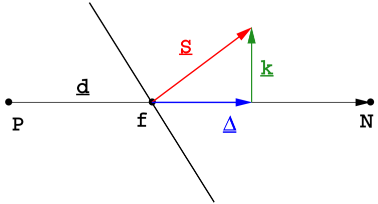
$$\mathbf{S}_f \cdot (\nabla\phi)_f = \underbrace{\mathbf{\Delta} \cdot (\nabla\phi)_f}_{\text{Contribución ortogonal}} + \underbrace{\mathbf{k} \cdot (\nabla\phi)_f}_{\text{Contribución no ortogonal}}$$

$$\mathbf{S} = \mathbf{\Delta} + \mathbf{k}$$

Con esta aproximación se elije $\mathbf{S} \parallel \mathbf{\Delta}$ de forma de usar la representación ortogonal en el primer término. Se plantean 3 formas para representar la contribución no ortogonal

Método de volúmenes finitos
Discretización del término difusivo

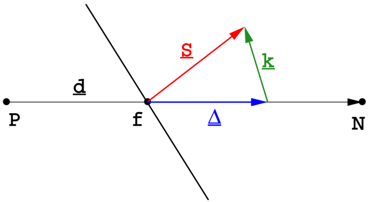
Corrección mínima



Minimiza la contribución no ortogonal usando $\boldsymbol{\Delta} \perp \mathbf{k}$

$$\boldsymbol{\Delta} = \frac{\mathbf{d} \cdot \mathbf{S}_f}{\mathbf{d} \cdot \mathbf{d}} \mathbf{d}$$

Corrección ortogonal

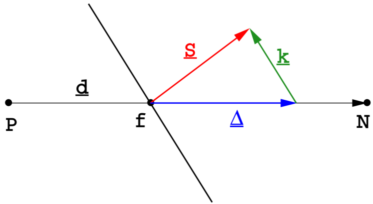


Conserva la contribución de ϕ_P y ϕ_N respecto a la contribución ortogonal

$$\boldsymbol{\Delta} = \frac{\mathbf{d}}{|\mathbf{d}|} |\mathbf{S}|$$

Método de volúmenes finitos
Discretización del término difusivo

Corrección sobre-relajada



La contribución de ϕ_P y ϕ_N crece con la no ortogonalidad

$$\mathbf{\Delta} = \frac{\mathbf{d}}{\mathbf{d}\cdot\mathbf{S}}|\mathbf{S}|^2$$

En definitiva

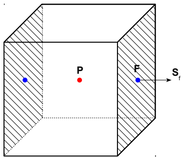
$$\mathbf{S}\cdot(\nabla\phi)_f = |\mathbf{\Delta}|\frac{\phi_N - \phi_P}{|\mathbf{d}|} + \mathbf{k}\cdot(\nabla\phi)_f$$

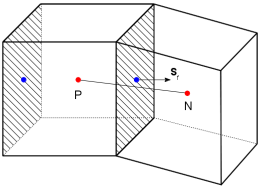
con

$$(\nabla\phi)_f = f_x(\nabla\phi)_P + (1-f_x)(\nabla\phi)_N \qquad (\nabla\phi)_P = \frac{1}{V_P}\sum_f \mathbf{S}\phi_f$$

Método de volúmenes finitos
Discretización del término fuente

$$\frac{\partial}{\partial t}\int_{V_P}\rho\phi dV + \int_{V_P}\nabla\cdot(\rho\mathbf{U}\phi)dV - \int_{V_P}\nabla\cdot(\rho\Gamma_\phi\nabla\phi)dV = \int_{V_P} S_\phi(\phi)dV$$





Término fuente

$$S_\phi(\phi) = Su + Sp\phi$$

$$\int_{V_P} S_\phi(\phi)dV = Su\ V_P + Sp\ V_P\phi_P$$

Se realiza una linealización del término fuente, previa a la discretización.

Método de volúmenes finitos

Discretización temporal

Semi-discretización temporal

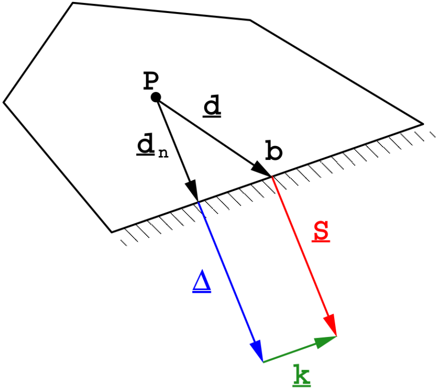
$$\int_t^{t+\Delta t}\left[\left(\frac{\partial\rho\phi}{\partial t}\right)_P V_P + \sum_f \mathbf{S}_f\cdot(\rho U)_f\phi_f - \sum_f(\rho\Gamma_\phi)_f\,\mathbf{S}_f\cdot(\nabla\phi)_f\right]dt = \int_t^{t+\Delta t}(Su\,V_P + Sp\,V_P\phi_P)\,dt$$

Crank-Nicolson

$$\frac{\rho_P\phi_P^n - \rho_P\phi_P^0}{\Delta t}V_P + \frac{1}{2}\sum_f \mathbf{S}_f\cdot(\rho U)_f\phi_f^n - \frac{1}{2}\sum_f(\rho\Gamma_\phi)_f\,\mathbf{S}_f\cdot(\nabla\phi)_f^n + \frac{1}{2}\sum_f \mathbf{S}_f\cdot(\rho U)_f\phi_f^0$$
$$-\frac{1}{2}\sum_f(\rho\Gamma_\phi)_f\,\mathbf{S}_f\cdot(\nabla\phi)_f^0 = Su\,V_P + \frac{1}{2}Sp\,V_P\phi_P^n + \frac{1}{2}Sp\,V_P\phi_P^0$$

$$a_P\phi_P^n + \sum_N a_N\phi_N^n = R_P \rightarrow [A][\phi] = [R]$$

Método de volúmenes finitos
Condiciones de borde



La condición se extiende a toda la cara. La componente ortogonal es igual a **S**, pero no se encuentra en el centro de la cara. El vector entre el centro de celda y la cara es normal a la frontera

$$\mathbf{d_n} = \frac{\mathbf{S}}{|\mathbf{S}|}\frac{\mathbf{d}\cdot\mathbf{S}}{|\mathbf{S}|}$$

y ya no se emplea la corrección **k**

Método de volúmenes finitos
Condiciones de borde

Valor fijo (Dirichlet)
El valor de ϕ en la cara b es ϕ_b.

- **Término convectivo**

$$\int_{V_P} \nabla \cdot (\rho \mathbf{U} \phi) dV = \sum_f F \phi_f$$

- **Término difusivo**

$$\int_{V_P} \nabla \cdot (\rho \Gamma_\phi \nabla \phi) dV = \sum_f (\rho \Gamma_\phi)_f \mathbf{S} \cdot (\nabla \phi)_f = \sum_f (\rho \Gamma_\phi)_f |\mathbf{S}| \frac{\phi_b - \phi_P}{|\mathbf{d_n}|}$$

Método de volúmenes finitos

Condiciones de borde

Gradiente fijo (Newmann)

Queda determinado el producto interno entre el gradiente y la normal

$$\left(\frac{\mathbf{S}}{|\mathbf{S}|}\cdot\nabla\phi\right)_b = g_b$$

- **Término convectivo**

$$\phi_b = \phi_P + \mathbf{d_n}\cdot(\nabla\phi)_b = \phi_P + |\mathbf{d_n}|g_b$$

- **Término difusivo**

  Para el término correspondiente se emplea

$$(\rho\Gamma_\phi)_b|\mathbf{S}|g_b$$

Método de volúmenes finitos
Errores inducidos por la malla

Además de los errores de truncamiento presentes por considerar discretizaciones espaciales y temporales de segundo orden, surgen términos de error debido a la elección de la malla.

- **No ortogonalidad**: si no se considera la componente no ortogonal del término difusivo (para acotar la solución), la discretización del término difusivo suma un término de error

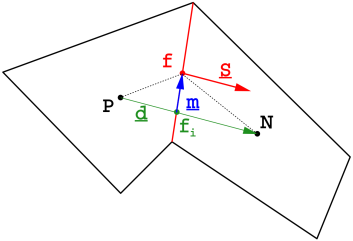
$$E_d = \sum_f \mathbf{S} \cdot [(\rho \mathbf{U})_f \mathbf{k} (\nabla \phi)_f] = \nabla \cdot ((\rho \mathbf{U})_f \mathbf{k} \cdot \nabla \phi) = \nabla \cdot (\Gamma_D \cdot \nabla \phi)$$

La magnitud del error depende del ángulo de no ortogonalidad y de la aproximación. Es mínimo para la corrección mínima.

Método de volúmenes finitos

Errores inducidos por la malla

- **Oblicuidad (skewness)**:



El cálculo de integrales sobre las caras requiere el valor de la variable en el centro de la misma

$$\int_f d\mathbf{S}\phi = \mathbf{S}\phi_f$$

Pero si ϕ_f se obtiene por interpolación lineal usando P y N, en realidad se calcula ϕ en el punto f_i

El error puede estimarse como

$$E_s = \sum_f \mathbf{S} \cdot (\rho \mathbf{U} \delta\phi)_f = \sum_f \mathbf{S} \cdot [(\rho\mathbf{U})_f \mathbf{m} \cdot (\nabla\phi)_f]$$

con $\mathbf{m} = \mathbf{x}_f - \mathbf{x}_{f_i}$

Discretización de Navier-Stokes

Flujo incompresible

$$\nabla \cdot \mathbf{U} = 0$$

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

$$\begin{aligned} \nabla \cdot (\mathbf{U}\mathbf{U}) &= \sum_f \mathbf{S} \cdot (\mathbf{U})_f (\mathbf{U})_f \\ &= \sum_f F(\mathbf{U})_f \\ &= a_P \mathbf{U}_P + \sum_N a_N \mathbf{U}_N \end{aligned}$$

Los coeficientes a_N y a_P dependen de $\mathbf{U}$. Dos aspectos requieren especial atención:

1. No linealidad en la ecuación de momento
2. Acoplamiento de velocidad y presión
   - PISO para problemas dependientes del tiempo
   - SIMPLE para casos estacionarios

Discretización de Navier-Stokes

Derivación de una ecuación para la presión

Uso de una forma semi-discretizada de la ecuación de momento, siguiendo el "espíritu" de Rhie y Chow

$$a_P \mathbf{U_P} = \mathbf{H}(\mathbf{U}) - \nabla p$$

El término $\mathbf{H}(\mathbf{U})$ tiene dos partes:

1. Transporte
2. Fuente. Términos de la discretización temporal y otras fuentes (además de ∇p)

$$\mathbf{H}(\mathbf{U}) = -\sum_N a_N \mathbf{U_N} + \frac{\mathbf{U}^0}{\Delta t}$$

Discretización de Navier-Stokes

Derivación de una ecuación para la presión

Discretizando la ecuación de continuidad

$$\nabla \cdot \mathbf{U} = \sum_f \mathbf{S}\mathbf{U}_f = 0$$

y usando

$$\mathbf{U_P} = \frac{\mathbf{H}(\mathbf{U})}{a_P} - \frac{\nabla p}{a_P}$$

se obtiene una expresión para el flujo en las caras

$$\mathbf{U_f} = \left(\frac{\mathbf{H}(\mathbf{U})}{a_P}\right)_f - \left(\frac{\nabla p}{a_P}\right)_f$$

Discretización de Navier-Stokes

Derivación de una ecuación para la presión

Reemplazando la expresión para velocidad en las caras en la ecuación de continuidad, puede obtenerse una ecuación para la presión

$$\nabla \cdot \left(\frac{\nabla p}{a_P} \right) = \nabla \cdot \left(\frac{\mathbf{H}(\mathbf{U})}{a_P} \right) = \sum_f \mathbf{S} \cdot \left(\frac{\mathbf{H}(\mathbf{U})}{a_P} \right)_f$$

Discretizando el operador laplaciano en la forma estándar, se obtienen las ecuaciones de Navier-Stokes discretas

$$a_P \mathbf{U_P} = \mathbf{H}(\mathbf{U}) - \sum_f \mathbf{S}(p)_f$$

$$\sum_f \mathbf{S} \cdot \left[\left(\frac{1}{a_P} \right)_f (\nabla p)_f \right] = \sum_f \mathbf{S} \cdot \left(\frac{\mathbf{H}(\mathbf{U})}{a_P} \right)_f$$

Discretización de Navier-Stokes

Acoplamiento presión-velocidad

Algoritmo PISO

- Se resuelve la ecuación de momento, usando el campo de presión del paso anterior (**momentum predictor**).
- Conocido el campo de velocidades, se construye $\mathbf{H}(\mathbf{U})$ y resuelve la ecuación de la presión (**pressure solution**).
- Se calculan flujos conservativos consistentes con la nueva presión, y se corrige la velocidad en forma explícita (**explicit velocity correction**).
- Se repiten los pasos **pressure solution** y **explicit velocity correction** hasta alcanzar la convergencia.

Discretización de Navier-Stokes

Acoplamiento presión-velocidad

Algoritmo SIMPLE

No es necesario resolver en forma completa el acoplamiento, Se hace uso de:

- Velocidades obtenidas a través de la ecuación de momento, calculando el gradiente de presión a partir de un paso anterior. La ecuación se relaja usando un factor de sobre-relajación.
- Se resuelve la ecuación para la presión.
- Se sobre-relaja la ecuación para la presión

$$p^{new} = p^{old} + \alpha_p(p^p - p^{old})$$

Outline

Esquemas de discretización
Término temporal

4 esquemas

- steadyState
- Euler
- backward
- CrankNicolson (θ)

Esquemas de discretización
Término convectivo

Cerca de 60 esquemas (depende de la versión)

- upwind
- linear
- linearUpwind
- cubic
- QUICK
- downwind
- SuperBee
- MUSCL
- Gamma
- Blended
- VanLeer

Sistemas lineales

Solvers

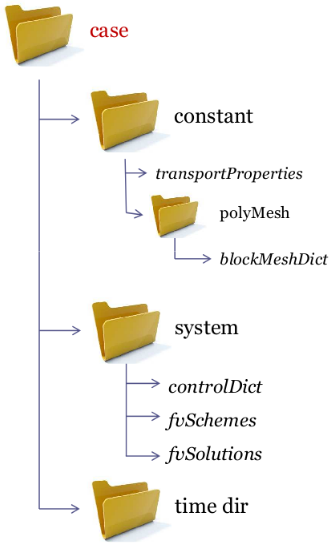
- PCG
- PBiCG
- smoothSolver
- GAMG
- diagonal
- ICCG
- BICCG

Precondicionadores

- DIC
- FDIC
- GAMG
- diagonal

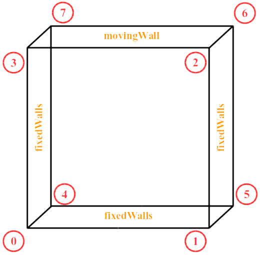
Outline

Estructura de archivos



1. **0**
   Directorio de tiempo. Se almancenan las variables dependientes del problema.
   - p
   - U
2. **constant**
   Constantes del problema. Datos de la malla y propiedades físicas.
   - transportProperties: definición de la viscosidad laminar
   - blockMeshDict: diccionario para blockMesh (malla).
3. **system**
   Discretización, algoritmo y control de ejecución
   - controlDict: paso de tiempo, intervalo de escritura, etc.
   - fvSchemes: esquemas de discretización.
   - fvSolution: sistemas lineales, constantes para PISO/SIMPLE.

Estructura de archivos
blockMeshDict




```
vertices
(
    (0 0 0)
    (1 0 0)
    (1 1 0)
    (0 1 0)
    (0 0 0.1)
    (1 0 0.1)
    (1 1 0.1)
    (0 1 0.1)
);

blocks
(
    hex (0 1 2 3 4 5 6 7) (20 20 1) simpleGrading (1 1 1)
);
```

```
boundary
(
    movingWall
    {
        type wall;
        faces
        (
            (3 7 6 2)
        );
    }
    fixedWalls
    {
        type wall;
        faces
        (
            (0 4 7 3)
            (2 6 5 1)
            (1 5 4 0)
        );
    }
    frontAndBack
    {
        type empty;
        faces
        (
            (0 3 2 1)
            (4 5 6 7)
        );
    }
);
```

Estructura de archivos
Variables dependientes

p

```
dimensions      [0 2 -2 0 0 0 0];

internalField   uniform 0;

boundaryField
{
    movingWall
    {
        type            zeroGradient;
    }

    fixedWalls
    {
        type            zeroGradient;
    }

    frontAndBack
    {
        type            empty;
    }
}
```

U

```
dimensions      [0 1 -1 0 0 0 0];

internalField   uniform (0 0 0);

boundaryField
{
    movingWall
    {
        type            fixedValue;
        value           uniform (1 0 0);
    }

    fixedWalls
    {
        type            fixedValue;
        value           uniform (0 0 0);
    }

    frontAndBack
    {
        type            empty;
    }
}
```

Estructura de archivos
fvSchemes

temporal

```
ddtSchemes
{
    default          Euler;
}
```

gradiente

```
gradSchemes
{
    default          Gauss linear;
    grad(p)          Gauss linear;
}
```

convectivo

```
divSchemes
{
    default          none;
    div(phi,U)       Gauss linear;
}
```

difusivo

```
laplacianSchemes
{
    default    Gauss linear orthogonal;
}
```

Estructura de archivos

fvSolution

Sistemas lineales

```
solvers
{
    p
    {
        solver          PCG;
        preconditioner  DIC;
        tolerance       1e-06;
        relTol          0;
    }

    U
    {
        solver          smothSolver;
        smoother        symGaussSeidel;
        tolerance       1e-05;
        relTol          0;
    }
}
```

Algoritmo para N-S

```
PISO
{
    nCorrectors      2;
    nNonOrthogonalCorrectors 0;
    pRefCell         0;
    pRefValue        0;
}
```

Estructura de archivos

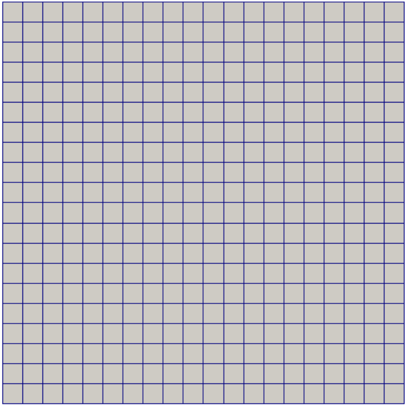
controlDict

```
application     icoFoam;

startFrom       startTime;

startTime       0;

stopAt          endTime;

endTime         0.5;

deltaT          0.005;

writeControl    timeStep;

writeInterval   20;
```

```
purgeWrite      0;

writeFormat     ascii;

writePrecision  6;

writeCompression off;

timeFormat      general;

timePrecision   6;

runTimeModifiable true;
```

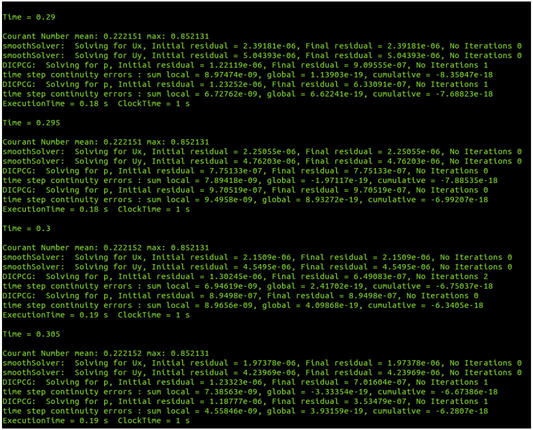
Ejecución

1. blockMesh



Ejecución

2. icoFoam



```
Time = 0.29

Courant Number mean: 0.222151 max: 0.852131
smoothSolver:  Solving for Ux, Initial residual = 2.39181e-06, Final residual = 2.39181e-06, No Iterations 0
smoothSolver:  Solving for Uy, Initial residual = 5.04393e-06, Final residual = 5.04393e-06, No Iterations 0
DICPCG:  Solving for p, Initial residual = 1.22119e-06, Final residual = 9.09555e-07, No Iterations 1
time step continuity errors : sum local = 8.97474e-09, global = 1.13903e-19, cumulative = -8.35047e-18
DICPCG:  Solving for p, Initial residual = 1.23252e-06, Final residual = 6.33091e-07, No Iterations 1
time step continuity errors : sum local = 6.72762e-09, global = 6.62241e-19, cumulative = -7.68823e-18
ExecutionTime = 0.18 s  ClockTime = 1 s

Time = 0.295

Courant Number mean: 0.222151 max: 0.852131
smoothSolver:  Solving for Ux, Initial residual = 2.25055e-06, Final residual = 2.25055e-06, No Iterations 0
smoothSolver:  Solving for Uy, Initial residual = 4.76203e-06, Final residual = 4.76203e-06, No Iterations 0
DICPCG:  Solving for p, Initial residual = 7.75133e-07, Final residual = 7.75133e-07, No Iterations 0
time step continuity errors : sum local = 7.89418e-09, global = -1.97117e-19, cumulative = -7.88535e-18
DICPCG:  Solving for p, Initial residual = 9.70519e-07, Final residual = 9.70519e-07, No Iterations 0
time step continuity errors : sum local = 9.4958e-09, global = 8.93272e-19, cumulative = -6.99207e-18
ExecutionTime = 0.18 s  ClockTime = 1 s

Time = 0.3

Courant Number mean: 0.222152 max: 0.852131
smoothSolver:  Solving for Ux, Initial residual = 2.1509e-06, Final residual = 2.1509e-06, No Iterations 0
smoothSolver:  Solving for Uy, Initial residual = 4.5495e-06, Final residual = 4.5495e-06, No Iterations 0
DICPCG:  Solving for p, Initial residual = 1.30245e-06, Final residual = 6.49083e-07, No Iterations 2
time step continuity errors : sum local = 6.94619e-09, global = 2.41702e-19, cumulative = -6.75037e-18
DICPCG:  Solving for p, Initial residual = 8.9498e-07, Final residual = 8.9498e-07, No Iterations 0
time step continuity errors : sum local = 8.9656e-09, global = 4.09868e-19, cumulative = -6.3405e-18
ExecutionTime = 0.19 s  ClockTime = 1 s

Time = 0.305

Courant Number mean: 0.222152 max: 0.852131
smoothSolver:  Solving for Ux, Initial residual = 1.97378e-06, Final residual = 1.97378e-06, No Iterations 0
smoothSolver:  Solving for Uy, Initial residual = 4.23969e-06, Final residual = 4.23969e-06, No Iterations 0
DICPCG:  Solving for p, Initial residual = 1.23323e-06, Final residual = 7.01604e-07, No Iterations 1
time step continuity errors : sum local = 7.38563e-09, global = -3.33354e-19, cumulative = -6.67386e-18
DICPCG:  Solving for p, Initial residual = 1.18777e-06, Final residual = 3.53479e-07, No Iterations 1
time step continuity errors : sum local = 4.55846e-09, global = 3.93159e-19, cumulative = -6.2807e-18
ExecutionTime = 0.19 s  ClockTime = 1 s
```

Ejecución

3. paraFoam