Solver Development in OpenFOAM

OpenFOAM Course 2nd Edition

C. Fernandes, L.L. Ferrás, J.M.Nóbrega
i3N/IPC – Institute for Polymers and Composites, University of Minho, Campus de Azurém, Guimarães, Portugal

PROGRAMA COOPERACIÓN TRANSFRONTERIZA ESPAÑA – PORTUGAL COOPERAÇÃO TRANSFRONTEIRIÇA 2007 – 2013

0682_CLOUDPYME2_1_E

Unión Europea
FEDER

Invertimos en su futuro

El proyecto CloudPYME (id: 0682_CLOUDPYME2_1_E) está cofinanciado por la Comisión Europea a través de el Fondo Europeo de Desarrollo Regional (FEDER), dentro de la tercera convocatoria de proyectos del Programa Operativo de Cooperación Transfronteriza España-Portugal 2007-2013 (POCTEP).

Outline

- Modify ***simpleFoam*** solver to predict temperature evolution
- Compile ***utility*** to compute also any field average and standard deviation
- Mixing flow Case Study

Modify *simpleFoam* solver

- Create ***applications*** folder in ***$WM_PROJECT_USER_DIR*** folder
  - » `mkdir $WM_PROJECT_USER_DIR/applications`
- Copy ***simpleFoam*** solver from ***$FOAM_SOLVERS*** folder to *the new* folder
  - » `cp -r $FOAM_SOLVERS/incompressible/simpleFoam $WM_PROJECT_USER_DIR/applications`
- Rename new ***simpleFoam*** folder to ***simpleFoamTemp***
  - » `cd $WM_PROJECT_USER_DIR /applications`
  - » `mv simpleFoam simpleFoamTemp`

Modify *simpleFoam* solver

- Rename the source code file ***simpleFoam.C***

  » *cd simpleFoamTemp*

  » mv simpleFoam.C simpleFoamTemp.C

- Edit ***files*** file that is in ***Make*** folder to read

```
SimpleFoamTemp.C

EXE = $(FOAM_USER_APPBIN)/simpleFoamTemp
```

- *Return to **simpleFoamTemp** folder and compile the solver (to be sure everything was done properly)*

  » *wclean*

  » *wmake*

Modify *simpleFoam* solver

- Verify if the new solver was created in ***$FOAM_USER_APPBIN*** folder
  - » *ls $FOAM_USER_APPBIN*

- New transport equation to solve (viscous dissipation is neglected)

$$\underbrace{\nabla \cdot \left(\vec{u} T \right)}_{\text{advection}} - \underbrace{\alpha \nabla^2 T}_{\substack{\text{diffusion} \\ \text{(heat conduction)}}} = 0$$

- We have to <u>insert the new transport equation in the solver</u>, <u>add a new scalar field **T** (for temperature)</u> and <u>create the new scalar field ***DT***</u> (that will represent the **thermal diffusivity α**)

Modify *simpleFoam* solver

- Analyze and edit ***createFields.H*** file

  » `gedit createFields.H`

- Insert the following lines **after velocity vector field definition** to add the new **temperature field** (hint: start by copy and paste the pressure field definition lines)

```
Info<< "Reading field T\n" <<endl;
volScalarField T
(
    IOobject
    (
        "T",
        runTime.timeName(),
        mesh,
        IOobject::MUST_READ,
        IOobject::AUTO_WRITE
    ),
    mesh
);
```

Modify *simpleFoam* solver

- Insert the following lines after ***singlePhaseTransportModel*** definition to define the new ***DT*** field

```
Info<< "Reading transportProperties\n" << endl;
IOdictionary transportProperties
(
    IOobject
    (
        "transportProperties",
        runTime.constant(),
        mesh,
        IOobject::MUST_READ,
        IOobject::NO_WRITE
    )
);
dimensionedScalar DT
(
     transportProperties.lookup("DT")
);
```

Modify *simpleFoam* solver

- Analyze and edit ***simpleFoamTemp.C*** file
  - » `gedit simpleFoamTemp.C`
- Add the command to insert the resolution of Temperature equation after ***turbulence->correct();*** command
  - » `#include “TEqn.H”`
- Analyze ***UEqn.H*** and ***pEqn.H*** files
- Create ***TEqn.H*** file (hint: start by copying **UEqn.H**)
  - » `cp UEqn.H TEqn.H`
  - » `gedit TEqn.H`

Modify *simpleFoam* solver

- After editing the ***TEqn.H*** file should read (review the new equation to be solved)

```
fvScalarMatrix TEqn
(
        fvm::div(phi, T)
        - fvm::laplacian(DT, T)
);
TEqn.relax();
TEqn.solve();
```

- Compile the new solver

  » *wclean*

  » *wmake*

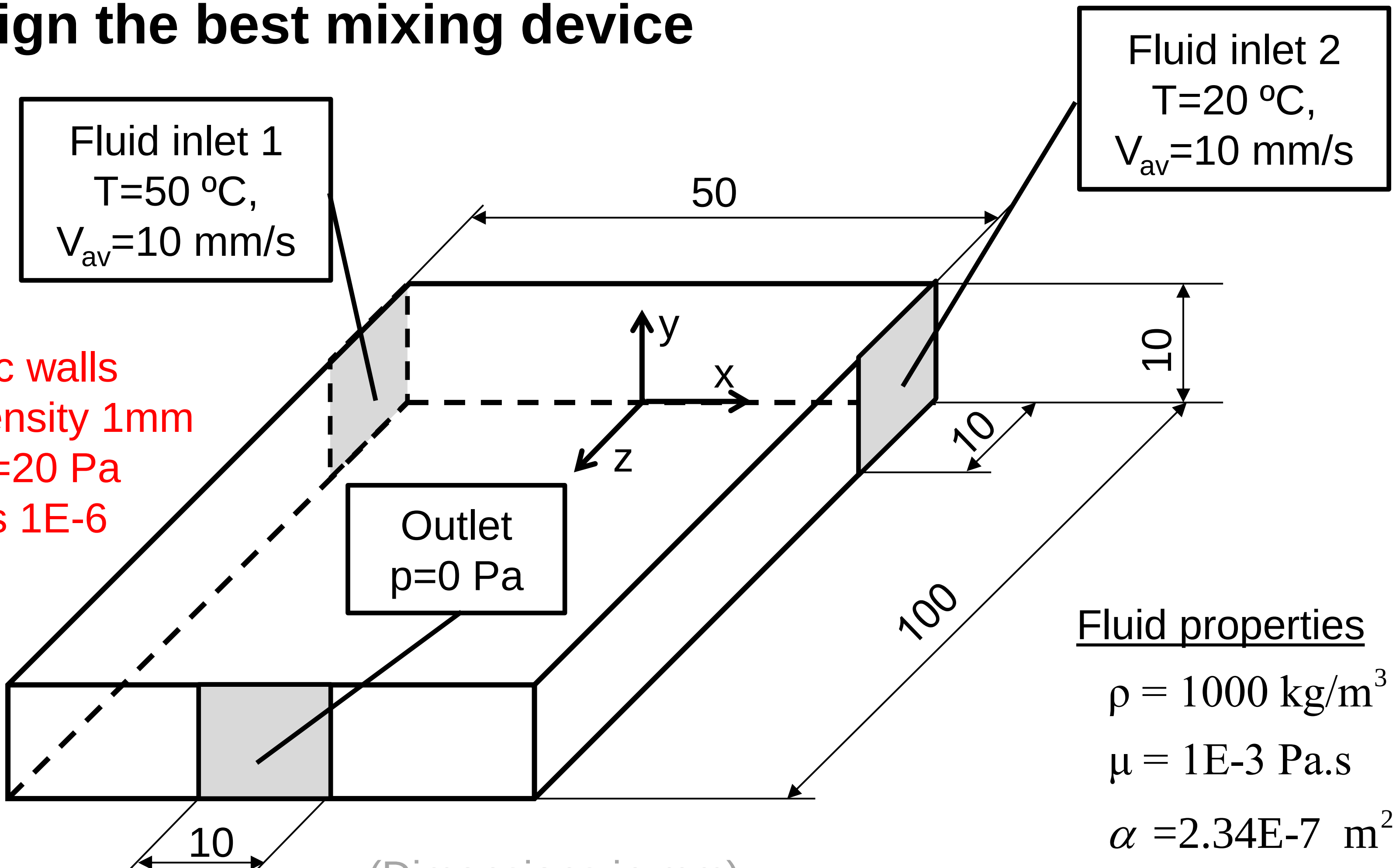
New utility

- Study ***patchAverageStd*** utility located in ***SHARED*** solver
- Compile ***patchAverageStd*** utility

$$\sigma_T = \sqrt{\frac{\sum_{i=1}^{nfaces} \left(T_i - \overline{T}\right)^2 \times A_i}{A_T}}$$

Mixing Flow Case Study

Design the best mixing device



Adiabatic walls
Mesh density 1mm
Max ΔP=20 Pa
Max Res 1E-6

(Dimensions in mm)

<u>Fluid properties</u>

$\rho = 1000\ kg/m^3$

$\mu = 1E\text{-}3\ Pa.s$

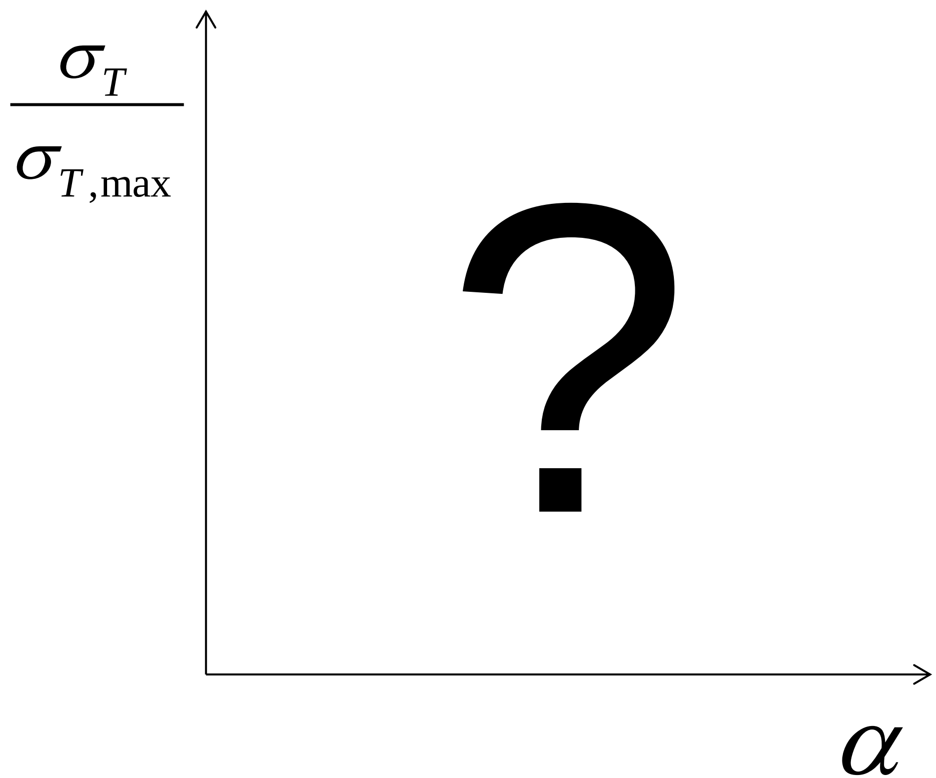
$\alpha = 2.34E\text{-}7\ m^2/s$

To-do:

- Use simpleFoam/pitzdaily tutorial as base study;
- Define laminar flow (**constant/RASProperties**)
- Insert DT definition in ***transportProperties***
- Create new ***blockMeshDict*** file
- Generate mesh and check it (***checkMesh*** and ***paraFoam***)
- Remove utility calculations from ***controlDict***
- Define missing discretization schemes (try to run code to identify what is missing)
- Define System of Equation Solver for Energy Equation (copy pressure solver)

Mixing Flow Case Study

- For the previous problem study the effect of the Thermal diffusivity on the mixing efficiency



PROGRAMA
COOPERACIÓN TRANSFRONTERIZA
ESPAÑA – PORTUGAL
COOPERAÇÃO TRANSFRONTEIRIÇA
2007 – 2013

0682_CLOUDPYME2_1_E

Unión Europea
FEDER

Invertimos en su futuro