

## Práctica 2. Análisis térmico de una habitación

E. Martín<sup>1</sup>, M. Meis<sup>1,2</sup> y F. Varas<sup>1</sup>

<sup>1</sup>Univ. de Vigo, <sup>2</sup>Vicus Desarrollos Tecnológicos

Dinámica de fluidos computacional con OpenFOAM  
18–20 de Junio de 2014



Unión Europea  
FEDER



Invertimos en su futuro

# Proyecto CloudPYME

El proyecto CloudPYME (ID 0682\_CLOUDPYME2\_1\_E) está cofinanciado por la Comisión Europea a través del Fondo Europeo de Desarrollo Regional (FEDER), dentro de la tercera convocatoria de proyectos del Programa Operativo de Cooperación Transfronteriza España–Portugal 2007–2013 (POCTEP).



Unión Europea  
FEDER



# Outline

- 1 **Formulación del problema y del modelo**
  - Formulación del problema
  - Formulación del modelo
  
- 2 **Resolución con OpenFOAM**
  - Generación de mallado
  - Implementación en OpenFOAM
  - Resolución y postprocesado en OpenFOAM

# Plan

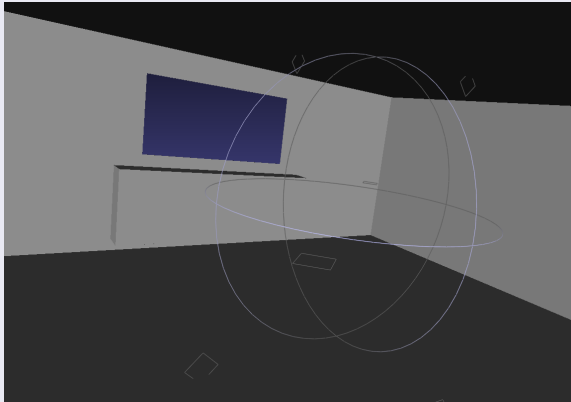
- 1 **Formulación del problema y del modelo**
  - Formulación del problema
  - Formulación del modelo
- 2 **Resolución con OpenFOAM**
  - Generación de mallado
  - Implementación en OpenFOAM
  - Resolución y postprocesado en OpenFOAM

# Plan

- 1 **Formulación del problema y del modelo**
  - **Formulación del problema**
  - Formulación del modelo
- 2 **Resolución con OpenFOAM**
  - Generación de mallado
  - Implementación en OpenFOAM
  - Resolución y postprocesado en OpenFOAM

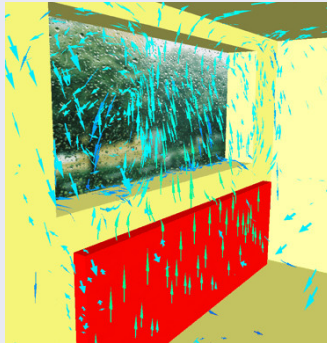
# Análisis térmico de habitación

## Formulación del problema



# Análisis térmico de habitación

## Flujos de calor en habitación



<http://blogs.mentor.com/robinbornoff>

# Análisis térmico de habitación

## Hipótesis de cálculo

- habitación cerrada sin forzamiento de flujo de aire
- temperatura de  $290K$  en vidrio interior de ventana
- paredes, suelo y techo adiabáticos
- temperatura en radiador  $310K$
- aire inicialmente en reposo y a  $300K$



# Plan

- 1 **Formulación del problema y del modelo**
  - Formulación del problema
  - **Formulación del modelo**
- 2 **Resolución con OpenFOAM**
  - Generación de mallado
  - Implementación en OpenFOAM
  - Resolución y postprocesado en OpenFOAM

# Modelado del flujo

## Estimación de velocidades

- Problema de convección natural
- Orden de magnitud a partir de equilibrio:

$$\rho \vec{v} \cdot \vec{\nabla} \vec{v} \sim \Delta \rho g \vec{e}_g$$

Con  $\Delta \rho = \rho_0 \beta \Delta T$ :

$$U_c = \sqrt{\beta \Delta T g H}$$

Para aire a 20C con  $\Delta T = 10K$  y  $H = 1m$ :

$$U_c \simeq 0.5m/s$$

# Modelado del flujo

## Régimen laminar o turbulento

$U_c = 0.5m/s$ ,  $H = 2.5m$  y  $\nu = 1.5 \times 10^{-5}m^2/s$  (aire a 20C):

$$Re_H = \frac{U_c H}{\nu} \simeq 8.3 \times 10^4$$

## Flujo compresible o incompresible

$\Delta T = 10K$ ,  $\beta = 3 \times 10^{-3}K^{-1}$  y  $\rho_0 = 1.2kg/m^3$  (aire a 20C):

$$\frac{\Delta\rho}{\rho_0} \simeq \beta\Delta T = 0.03$$

# Análisis térmico de habitación

## Flujo turbulento

- número de Reynolds en torno a  $1.5 \times 10^5$
- modelo RANS (Reynolds Averaged Navier–Stokes) estacionario/evolutivo
- modelo de turbulencia  $k - \epsilon$

## Modelo $k - \epsilon$

- modelo de turbulencia de dos ecuaciones:
  - $k$ : energía cinética turbulenta
  - $\epsilon$ : disipación turbulenta
- modelo propuesto para flujos no desprendidos
- amplio uso (no siempre justificado)

# Modelado del flujo

- aproximación de Boussinesq
- no se retiene variación de otras propiedades

## Conservación de materia y momentos

$$\operatorname{div} \vec{U} = 0$$

$$\frac{\partial \vec{U}}{\partial t} + (\vec{U} \cdot \vec{\nabla}) \vec{U} - \nu \Delta \vec{U} + \frac{1}{\rho_0} \vec{\nabla} P + \frac{1}{\rho_0} \operatorname{div} \mathbb{T}' = -(1 - \beta(T - T_{ref})) g \vec{e}_3$$

## Conservación de energía

$$\rho_0 c_p \frac{\partial T}{\partial t} + \rho_0 c_p \vec{U} \cdot \vec{\nabla} T - \operatorname{div}(k \vec{\nabla} T) = 0$$

# Modelado del flujo

## Coeficientes en ecuación de energía

$$\frac{\partial T}{\partial t} + \vec{U} \cdot \vec{\nabla} T - \text{div}(\alpha_{tot} \vec{\nabla} T) = 0$$

$\alpha_{tot}$  contiene contribución turbulenta:

$$\alpha_{tot} = \frac{k}{\rho c_p} + \frac{k_t}{\rho c_p} = \frac{\mu}{\rho} \frac{k}{\mu c_p} + \nu_t \frac{k_t}{\mu_t c_p} = \frac{\nu}{Pr} + \frac{\nu_t}{Pr_t}$$

## Descomposición de la presión

$$p = p_{rgh} - \rho g z$$

de modo que:

$$\vec{\nabla} p = \vec{\nabla} p_{rgh} - \rho g e_z$$

# Modelado del flujo

## Modelado de turbulencia

$$\rho \frac{\partial k}{\partial t} + \rho \vec{U} \cdot \vec{\nabla} k - \vec{\nabla} \cdot \left( \frac{\mu_t}{\sigma_k} \vec{\nabla} k \right) = 2\mu_t S_{ij} S_{ij} - \rho \epsilon$$

$$\rho \frac{\partial \epsilon}{\partial t} + \rho \vec{U} \cdot \vec{\nabla} \epsilon - \vec{\nabla} \cdot \left( \frac{\mu_t}{\sigma_\epsilon} \vec{\nabla} \epsilon \right) = C_{1\epsilon} \frac{\epsilon}{k} 2\mu_t S_{ij} S_{ij} - C_{2\epsilon} \rho \frac{\epsilon^2}{k}$$

## Viscosidad turbulenta

$$\mu_t = \rho C_\mu \frac{k^2}{\epsilon}$$

# Condiciones de contorno

## Espesor de capa límite

$$\frac{\delta}{L} \sim \frac{1}{\sqrt{\text{Re}}}$$

## Resolución de capa límite

Con  $h \sim \delta$

$$N \sim (L/h)^3 \sim \text{Re}^{3/2}$$

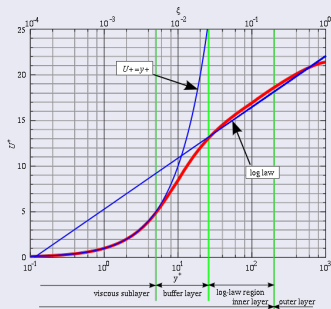
implica coste de cálculo inabordable



# Condiciones de contorno

## Ley de pared

### Análisis de capa límite



$$y^+ = \frac{y u_\tau}{\nu} \quad u^+ = \frac{u}{u_\tau} \quad u_\tau = \sqrt{\frac{\tau_w}{\rho}}$$

# Plan

- 1 **Formulación del problema y del modelo**
  - Formulación del problema
  - Formulación del modelo
- 2 **Resolución con OpenFOAM**
  - Generación de malla
  - Implementación en OpenFOAM
  - Resolución y postprocesado en OpenFOAM

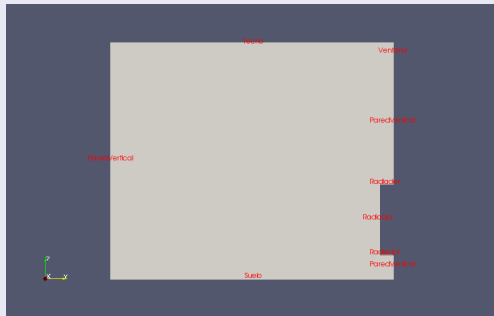
# Plan

- 1 **Formulación del problema y del modelo**
  - Formulación del problema
  - Formulación del modelo
  
- 2 **Resolución con OpenFOAM**
  - **Generación de malla**
  - Implementación en OpenFOAM
  - Resolución y postprocesado en OpenFOAM

# Un caso simplificado 2D

## Importación de mado de Gmsh

```
$ gmshToFoam MallaHabitacion.msh
```



Página principal código Gmsh

<http://geuz.org/gmsh/>

# Plan

- 1 **Formulación del problema y del modelo**
  - Formulación del problema
  - Formulación del modelo
- 2 **Resolución con OpenFOAM**
  - Generación de mallado
  - **Implementación en OpenFOAM**
  - Resolución y postprocesado en OpenFOAM

# Implementación de modelo físico

## Descripción del modelo físico

- modelo de turbulencia
- propiedades del fluido
- condiciones de contorno (e iniciales)

# Implementación de modelo físico

## Directorio `constant`

- subdirectorio `polyMesh` (geometría/mallado)
- Diccionario `RASProperties` (turbulencia)
- Diccionario `transportProperties` (fluido)
- Diccionario `g` (gravedad)

# Implementación de modelo físico

## Diccionario RASProperties

```
RASModel      kEpsilon;  
turbulence     on;  
printCoeffs   on;
```

Asume parámetros estándar para modelo  $k - \epsilon$



# Implementación de modelo físico

## Diccionario transportProperties

```
transportModel      Newtonian;  
rho      rho [ 1 -3 0 0 0 0 0 ] 1;  
nu      nu [0 2 -1 0 0 0 0] 1.5e-05;  
beta      beta [0 0 0 -1 0 0 0] 3e-03;  
TRef      TRef [0 0 0 1 0 0 0] 300;  
Pr      Pr [0 0 0 0 0 0 0] 0.9;  
Prt      Prt [0 0 0 0 0 0 0] 0.7;
```

## Diccionario g

```
dimensions      [0 1 -2 0 0 0 0];  
value      ( 0 0 -9.81 );
```

# Implementación de modelo físico

## Condiciones de contorno (e iniciales)

velocidad:      archivo 0/U  
presión:        archivo 0/p  
variable  $p_{rgh}$ :    archivo 0/p\_rgh  
temperatura:    archivo 0/T  
variable  $k$ :        archivo 0/k  
variable  $\epsilon$ :    archivo 0/epsilon

## Otros archivos

variable  $\nu_t$ :      archivo 0/nut  
variable  $\alpha_t$ :    archivo 0/alphat

# Plan

- 1 **Formulación del problema y del modelo**
  - Formulación del problema
  - Formulación del modelo
- 2 **Resolución con OpenFOAM**
  - Generación de malla
  - Implementación en OpenFOAM
  - Resolución y postprocesado en OpenFOAM

# Selección de solver

## Solvers en OpenFOAM

- `buoyantBoussinesqPimpleFoam`:  
Transient solver for buoyant, turbulent flow of incompressible fluids
- `buoyantBoussinesqSimpleFoam`:  
Steady-state solver for buoyant, turbulent flow of incompressible fluids

# Resolución del caso

## Ejecución del caso

Ejecución y almacenamiento de salida en archivo:

```
$ buoyantBoussinesqSimpleFoam > solver.log
```

## Algunas comprobaciones elementales

- Evolución de los residuos en resolución
  - consultar archivo `solver.log`
- Estimación de  $y_+$  en celdas con ley de pared
  - uso de utilidad `yPlusRAS`

# Visualización

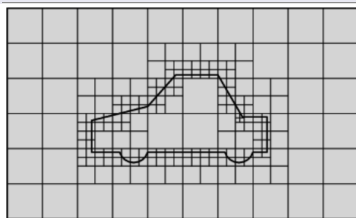
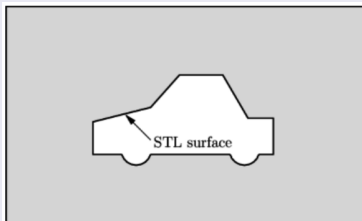
## Representación con `Paraview`

- Imágenes de mapas de velocidades
- Imágenes de mapas de temperaturas
- Líneas de corriente

# Mallado no estructurado en OpenFOAM

## Herramienta snappyHexMesh

- definición de geometría (formato STL/OBJ)
- generación de malla estructurada
- refinamiento iterativo de malla



# Mallado no estructurado en OpenFOAM

## Algoritmo de mallado con `snappyHexMesh`

- generación malla base (`blockMesh`)
- refinamiento y selección (`castellatedMeshControls`)
- ajuste a superficie (`snapControls`)
- adición de capas (`addLayerControls`)

## Información en:

*A comprehensive tour of snappyHexMesh*, E. de Villiers

<http://www.openfoamworkshop.org/2012/OFW7.html>



# Mallado no estructurado en OpenFOAM

## Generación de malla no estructurada

```
$ BlockMesh  
$ snappyHexMesh -overwrite
```

## Opciones de generación de malla

```
Archivo snappyHexMeshDict
```