INSA Toulouse, Applied Mathematics department
Finite Element Methods

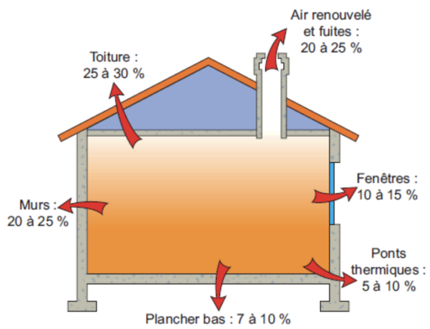
Programming Practical based on FEniCS

By Jerome.Monnier@insa-toulouse.fr
Other information and all files are available on the INSA Moodle page.

This PP will be marked.

1 The engineering problem

A technical analysis of a public mountain hut at high altitude indicates the following approximated thermal loss, Fig. 1.



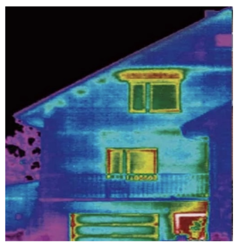


FIGURE 1 – (Left) Approximated thermal loss of the hut (2D vertical geometry). The fireplace is here not indicated. (Right) Thermography of the hut.

The hut is uninhabited for long. You enter the hut for long winter vocations :), it is very cold inside, it is at a uniform temperature u_0. At $t = 0$, you light on a fire in its lovely fireplace.

The goal aims here at simulating the temperature (in space and time), given the initial temperature u_0 (supposed to be uniform for simplicity) and given a "fire power policy" (supposed to be time-dependent or not) .

2 The model equations

For sake of simplicity, the model geometry is 2D : $\Omega \subset \mathbb{R}^2$.
The thermal transfer model consists to solve the following equations :

$$\partial_t u(x,t) - \mathrm{d}iv(\mu \ \nabla u(x,t)) \ + \ (\mathbf{w}^u \cdot \nabla u)(x,t) = \ f(x,t) \text{ in } \Omega \times [0,T]$$
$$\text{Plus I.C. } u(x,0) = u_0 \ \forall x \in \Omega, \text{ plus B.C. on } \partial\Omega.$$

where μ denotes the thermal diffusivity.

Natural convection is here an important phenomena (see e ;G ; the Wikipedia page for a short description). Then, the air velocity field $\mathbf{w}$ is solution of the Navier-Stokes equation with a source term depending on the temperature u. As a consequence, the thermal model becomes a fully coupled fluid dynamics-thermal system. Its resolution is far to be easy.
For a sake of simplicity, it is here supposed that $\mathbf{w}$ is given.

The domain boundary $\partial\Omega$ is decomposed as follows : $\partial\Omega = \Gamma_{flux} \cup \Gamma_{fire} \cup \Gamma_{diri}$ where Γ_{flux} represents the different walls, floor, ceiling etc where flux conditions apply to model the heat loss through the considered boundary, Γ_{fire} represents the fireplace boundary and Γ_{diri} denotes a part of boundary where the temperature is known.

The flux-type BCs read as below.

$$-\mu \ \nabla u(x,t) \cdot n(x) = c(x) u_f(x,t) \text{ on } \Gamma_{flux} \tag{1}$$

where $c(x)$ $(Wm^{-1}K^{-1})$ depends on the heat loss amplitude, in other words on the boundary insulation. ($c = 0$ corresponds to a perfectly insulated wall). The function $u_f(x,t)$ may equal the unknown $u(x,t)$ therefore defining Fourier type condition, or not.

On Γ_{fire}, it will be assumed that radiative exchanges apply. The latter are assumed to be black body type (Stefan-Boltzmann's law) at a given temperature $u_{fire}(x,t)$.

$$-\mu \ \nabla u(x,t) \cdot n(x) = \ \sigma \ \ (u^4 - u_{fire}^4)(x,t) \text{ on } \Gamma_{fire} \tag{2}$$

where σ is the Stefan-Boltzmann's constant, $\sigma \approx 5,67 \ 10^{-8} Wm^{-1}K^{-4}$.

Note that the imposed function $u_{fire}(x,t)$ may be seen as a command of the temperature inside the room Ω.

The source term $f(x,t)$ may represent another heat source inside the domain.

Numerical values.
The thermal diffusivity μ in the air at ambiant temperature equals ≈ 19 (mm^2/s) (see values provided in any reliable reference).

3 Tasks

3.1 Stages

1) Problem setup
a) Setup your own $2D$ geometry and scenario.
As a first step, you may simply use the FEniCS routine to consider a square geometry (of a reasonable size in terms of IS units...).
To go further you may consider a more realistic 2D vertical plan of a hut as indicated on the figure by using Gmsh and the provided Python reading routine.
b) Detail the model equations, B.C., etc of your own test case.

2) Coding the model using the implicit Euler scheme

a) **Semi-linearized model**. You will first consider a semi-linearized model, by approximating the non linear term on the black body boundary.

Question : is your semi-linearized problem well -posed ? Detail your arguments.

b) **Implement the non-linear model**.
Of course as always, you will first write in detail the equations and algorithms to be coded...
Moreover, you won't use here the nonlinear FEniCS solver (the "black-box"), or if so it will be to compare it to your own home-developed non-linear solver.

3) Simulations - Starting from the I.C., you will compute the steady-state solution.

- Plot out the temperature evolution until steady-state is reached.

You will detail all the numerical parameters : the stopping criteria, Δt, etc.

3.2 Increasing model complexity...

— As a first step, you will neglect the convective transfers that is $\mathbf{w} = 0$. In this case, diffusive plus radiative transfers only are taken into account.

— Next, you will consider the convective transfers with $\mathbf{w}$ given : this is a forced convection context since $\mathbf{w}$ is not dependent on the temperature u as it is in natural convection.

— Moreover, as a first step, the time scheme will be the Euler implicit one, next the the Crank-Nicholson one.

3.3 A simple model to define the fluid velocity w

You here compute the fluid velocity $\mathbf{w}$ by solving a basic fluid flow model, namely the potential flow model. The fluid velocity is assumed to derive from a potential V : $\mathbf{w} = \nabla V$.
Assuming a steady-state flow and using the incompressibility assumption $div(\mathbf{w}) = 0$, it follows the equations :

$$-\Delta V(x,t) \;=\; 0 \text{ in } \Omega, \text{ for } t \tag{3}$$

with the BCs

$$\nabla V(x,t) \cdot n(x) = 0 \text{ on } \Gamma_{wall} \tag{4}$$

$$V = V_{in} \text{ on } \Gamma_{in} \;; \quad V = V_{out} = 0 \text{ on } \Gamma_{out} \tag{5}$$

In the case of this weakly coupled fluid-thermal model, you have to first solve the fluid model providing $\mathbf{w}(x)$ for all t, next the thermal model providing $u(x,t)$ given $\mathbf{w}(x)$.

4) To go further

-) Actual order of your computational code You will demonstrate the actual order of your computational code in a steady-state case only.
To do so, you will have to consider an exact solution of a similar system.
For example , the chosen exact solution can be of the form :

$$u_{ex}(x_1, x_2) = \cos(\omega_1 x_1) \sin(\omega_2 x_2)$$

with $\omega_\Box$ and α well chosen.

- Given $u_{ex}(x)$, calculate the corresponding source terms of the equations, both in Ω and on $\partial\Omega$.

- Solve numerically the system. Compute the error $\|u_{ex} - u_h\|/\|u_{ex}\|$.

- Plot the convergence curves in $\log - \log$ scale, like presented in your course manuscript.

- In the end, at what order is your code ? Detail your answers.

Evaluating the non-linear solver accuracy may involve a little more complex calculations. If so, start by evaluating the order of converge,ce of your code in a linear case, next in the non linear case (with the term u^4 on the boundary).

-) Mass condensation effects Explain briefly the pros and cons of the mass condensation technique in your context. Highlight your comments with numerical experiments.