In this paper, we will study the two dimensional diffusion problem on a rectangle $[0, L_x] \times [0, L_y]$:

$$\boxed{\frac{\partial u}{\partial t} = D(x,y)(\frac{\partial^2 u}{\partial x^2} + \frac{\partial^2 u}{\partial y^2}) + f(x,y)} \tag{1}$$

Where $D(x,y)$ denotes the diffusion coefficient. We will denote the simulation time by L_t.

To solve this problem numerically, we will use the explicit Euler scheme <u>for the time</u> as well as <u>for the space</u> integration. If we discretize our problem in time and in space and denote $U(t_n, x_i, y_j) := U^n_{i,j}$ as the solution of our discretized problem at time $t_n := n\Delta t$ (supposing a uniform time grid) and at position $(x_i := j\Delta x, y_j := j\Delta y)$ (supposing a uniform space grid), we obtain a discretized problem of the form :

$$\frac{U^{n+1}_{i,j} - U^n_{i,j}}{\Delta t} = D(i\Delta x, j\Delta y)\Big(\frac{U^n_{i+1,j} - 2U^n_{i,j} + U^n_{i-1,j}}{(\Delta x)^2} + \frac{U^n_{i,j+1} - 2U^n_{i,j} + U^n_{i,j-1}}{(\Delta y)^2}\Big) + f(i\Delta x, j\Delta y)$$

Now, since we have a purely explicit model, we can simply isolate $U^{n+1}_{i,j}$ from the above equality to find the numerical solution at iteration $n+1$. We obtain the following equality :

$$U^{n+1}_{i,j} = D(i\Delta x, j\Delta y)\underbrace{\frac{\Delta t}{\Delta x^2}}_{:=\alpha_x}(U^n_{i+1,j} - 2U^n_{i,j} + U^n_{i-1,j}) + D(i\Delta x, j\Delta y)\underbrace{\frac{\Delta t}{\Delta y^2}}_{:=\alpha_y}(U^n_{i,j+1} - 2U^n_{i,j} + U^n_{i,j-1}) + \Delta t f(i\Delta x, j\Delta y) + U^n_{i,j}$$

Numerically, we can easily implement it through a simple "for" loop if we know the initial and boundary conditions of our problem. However, the whole point will here be to take advantage of the **sparse matrix** behaviour of our model.

Putting the above expression under matrix form (by vectorizing the 2D problem) leads to the following explicit scheme :

$$\mathbf{U}_{k+1} = \mathbf{A}\mathbf{U}_k + \mathbf{b} \tag{2}$$

where $\mathbf{A}$ contains the coefficients of the discretized problem and $b_{(i-1)(nx+1)+j} = \Delta t f(i\Delta x, j\Delta y)$.

In a first time, we will analyze the numerical results obtained. In a second time, we will analyze the rate of convergence of the numerical solution as a function of n_t, n_x and n_y.

An important remark to mention is that we we need to choose our parameters in accordance with the stability condition of the scheme we used. For a more precise explanation, we could check the maximum allowable CFL number of this model. However, a clear analysis of the stability conditions will not be provided here.

1 Numerical results

In this section, we will analyze the numerical results we obtain at different time steps, for a freely chosen diffusion problem. To do this, we will define our parameters to get a concrete everyday life problem. Let us consider that we are in the framework of the heat equation :

- We consider a rectangular two dimensional room of dimension $[0, L_x] \times [0, L_y] = [0, 5m] \times [0, 3m]$ and we want to investigate the temperature profile in this room over $L_t = 10$ seconds (arbitrary value).
- Initially, our room is heated, but since we have a window at the extremity of the room (at $y = 0$), it is a bit colder ($10°C = 283.15K$) than at the other extremity ($20°C = 293.15K$). We will consider for simplicity that our temperature profile increases linearly, so that at $u(x,y) = 283.15 + \frac{10}{3}y$.
- The thermal diffusivity (or diffusion coefficient) $D(x,y)$ varies thus over the room, because it depends on the temperature. To have relatively fast diffusion result over the time, we will use a high diffusion coefficient to make the temperature vary quickly to have clear results. We will choose $D(x, y = 0m) = 20.36 \times 10^{-2}\frac{m^2}{s}$ and $D(x, y = 3m) = 21.70 \times 10^{-2}\frac{m^2}{s}$ (10^4 times bigger than the realistic values of air). Supposing it varies linearly, we have $D(x,y) = 4.47 \times 10^{-3}x + 20.36 \times 10^{-2}\frac{m^2}{s}$.

- Our boundary conditions are constant over time (except at the very initial step, by hypothesis). We consider that at $y = 0m$, we always have a uniform temperature of $283.15K$ and at $y = 3m$, we have a uniform temperature of $293.15K$, and in between, our temperature increases exponentially : $u(x = 0m, y) = u(x = 5m, y) = 0.5239 \exp x + 282.62$.
- We have an external source $f(x, y)$ over the whole wall at $x = 0m$ which is a radiator, heating during the whole simulation the room. We will thus consider an external source of the form $f(x, y) = -2x + 10$ (linear variation of the external source for simplicity).

With all those parameters, we can plot our numerical solution at different time steps :

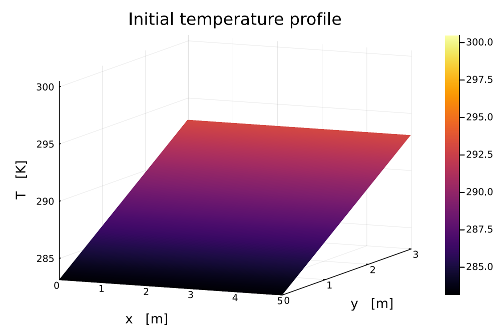

Figure 1: Initial solution

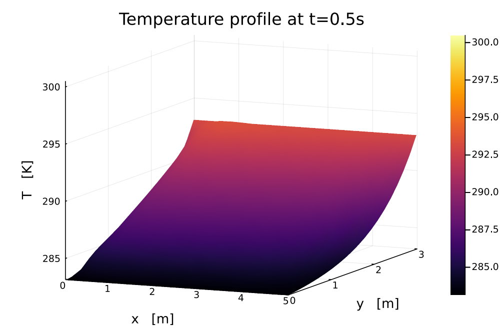

Figure 2: Solution at $t = 0.5s$

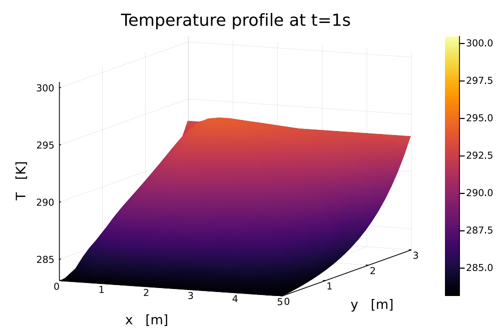

Figure 3: Solution at $t = 1s$

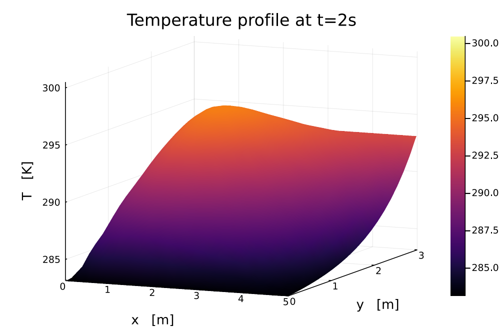

Figure 4: Solution at $t = 2s$

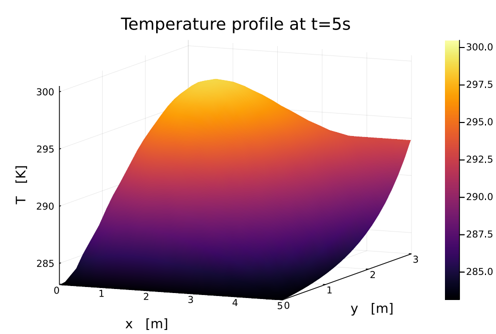

Figure 5: Solution at $t = 5s$

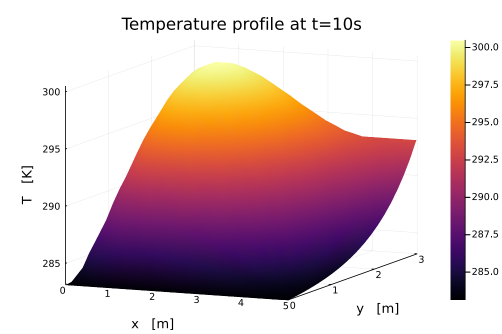

Figure 6: Solution at $t = 10s$

We can make the following observations :

- We can check that our numerical solution is stable and seems plausible compared to the reality. The temperature of the heated room globally increases due ton the presence of the radiator.

- During the evolution of the simulation, we can check that the temperature profile becomes asymmetric. This is because we have chosen our external source such that it is bigger close to $x = 0m$ than to $x = 5m$. Our temperature profile is thus bigger close to the x-axis (physically due to the presence of a radiator).
- The most important thing to see here is that our numerical solution tends to a **steady-state** over time, meaning that it is stabilizing. This is physically coherent because in a heat equation, the dissipation tends to be balanced by the external source over a certain time. If we had put a zero external source, we would just have observed a steady-state "flattened" solution.
- The diffusion coefficient $D(x, y)$ plays here an important role : it represents the rate at which the solution will converge to the steady state. This is because this parameter governs the rate of dissipation of the model. The initial and boundary conditions are also indeed respected.

2 Experimental analysis of the rate of convergence

Now that we have built relatively coherent diffusion problem, we move on to the analysis of the rate of convergence of our numerical method and analyze it using the chosen problem.

To do this, we will compare the steady state numerical solution with the steady state analytical solution using different discretizations. To model the analytical solution, we will consider a sufficiently fine mesh ($n_x = n_y = 80, n_t = 10000$). To compare both solution, we remember that we will need to fit the fine mesh with the other numerical meshes in order to compare the solutions at the same grid points. We will calculate the relative error in % in the following way :

$$E_{rel} = [\frac{1}{(n_x+1)(n_y+1)} \sum_{i=0}^{n_x} \sum_{j=0}^{n_y} \frac{u_{numerical}(x = i\Delta x, y = j\Delta y, t = t_{steady}) - u_{analytical}(x, y, t = t_{steady})}{u_{analytical}(x, y, t = t_{steady})}] \times 100$$

It means that we will take the mean relative error over the whole domain between numerical and analytical solution.

2.1 Variation n_x and n_y

We start off by varying the number of space mesh points. Experimentally, we obtain the following graph :

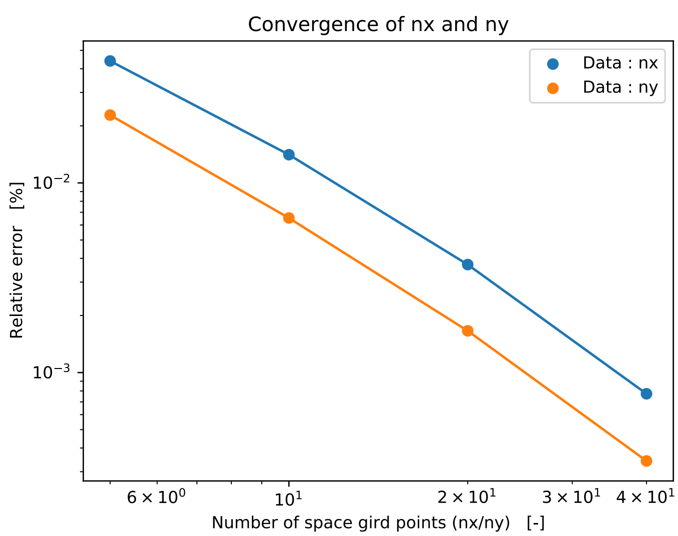

Figure 7: Space convergence

From this log-log diagram, we can observe some interesting points :

- The order of convergence appears to be the same for both n_x and n_y (the slope of the linear curves is the same). This is because we have used the same space scheme for both cases.
- In a log-log diagram, the slope of the linear curve appears to be the order of convergence of the method. Here, the slope is approximately two, meaning that this method has a **quadratic** order of convergence in space, with respect to n_x as well as with respect to n_y. This is coherent, since we used here a centered scheme of order 2.
- We can observe that both plots are translated one above another. This is because we have decided to use a different L_x and L_y in our problem. Since $L_x > L_y$, it is coherent that for a $n_x = n_y$, the error will be bigger for n_x than for n_y.
- We can finally see that the relative error is always really small, even for the "bad" discretizations. This is because we have a relatively small size of domain here. Using a much larger domain would have given a bigger relative error, but still the same curve as obtained here.

2.2 Variation of n_t

In this last section, we will vary the number of time steps n_t. To assure convergence for sufficiently small values of n_t (i.e $n_t = 200$), we will exceptionally use $L_t = 0.05s$ here. We obtain the following graph :

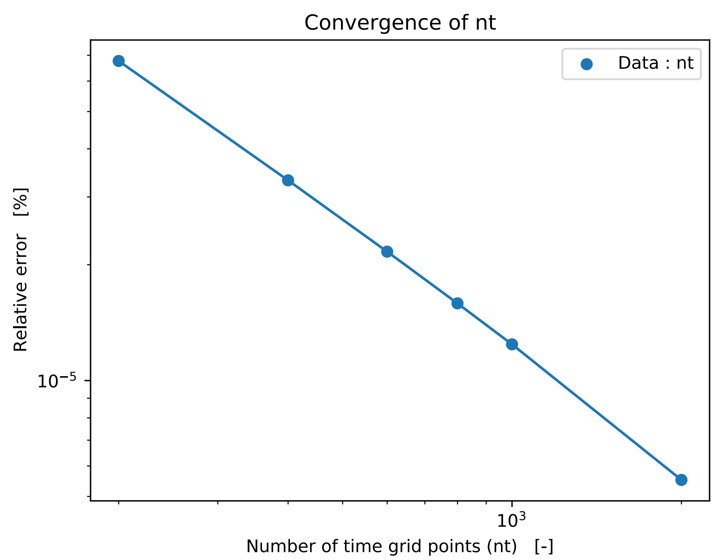

Figure 8: Time convergence

We can here observe an relatively intuitive result : we also have a linear plot in our log-log diagram, decreasing as n_t increases. More precisely, this is a graph of slope of approximately one. It means thus that we have a **linear** order of convergence in time, contrary to the space order of convergence that was quadratic.

If we think about it, this is coherent, since we used here an explicit Euler scheme of order one. All the theoretical results are thus matching with the experimental one.

N.B : In this section, we have used a fine numerical resolution to approximate our analytical one, but it could also have been possible to find analytically the solution of simple problems (for example the sine function, that has a simple mode decomposition). The use of an explicit analytical solution as reference would have lead to slightly different results.