Numerical Methods of Thermo-Fluid Dynamics I

Practical No. 1 - Group 9

Numerical Solution of 2D Heat Equation

Abhik Sarkar (23149662), Gokhul Shriram (23212226)

Table of Contents

List of Figures

1 Problem

Given the dimensionless 2D heat equation:

$$\frac{\partial w}{\partial t} = \frac{\partial^2 w}{\partial x^2} + \frac{\partial^2 w}{\partial y^2} \tag{1}$$

$$[x, y] \in [0, 1] \times [0, 1], \quad t \in [0, 0.16] \tag{2}$$

The initial boundary conditions are:

$$w(x, y, 0) = 0 \tag{3}$$

$$w(0, y, t) = 1 - y^3, \quad w(1, y, t) = 1 - \sin\left(\frac{\pi}{2}y\right), \quad w(x, 0, t) = 1, \quad w(x, 1, t) = 0 \tag{4}$$

2 Solutions: Discretization and Numerical Schemes

Discretization of the scheme using CDS and Crank-Nicolson method in time:
Let's denote $w_{i,j}^n$ as numerical approximation to $w(x_i, y_j, t_n)$ at timestep n,
where $x_i = i \cdot \Delta x$, $y_j = j \cdot \Delta y$ and $t_n = n \cdot \Delta t$.

Here, $\Delta x, \Delta y$ - spatial discretization steps, and Δt - temporal discretization step.

2.1 Forward Time Step and Spatial Derivative

$$\frac{\partial w}{\partial t} = \frac{w_{i,j}^{n+1} - w_{i,j}^n}{\Delta t} \tag{5}$$

The above equation represents the forward timestep of the variable w. By taking its spatial derivative with respect to x and y, we get,

$$\frac{\partial^2 w}{\partial x^2} = \frac{w_{i+1,j}^n - 2w_{i,j}^n + w_{i-1,j}^n}{(\Delta x)^2} \tag{6}$$

$$\frac{\partial^2 w}{\partial y^2} = \frac{w_{i,j+1}^n - 2w_{i,j}^n + w_{i,j-1}^n}{(\Delta y)^2} \tag{7}$$

2.2 Explicit Euler Method Derivation

The explicit Euler method is used to solve the partial differential equations. In the 2D heat equation, to approximate the solution at the next timestep $(n + 1)$, this method is used. Therefore, the equation for the explicit Euler method is given by:

$$\frac{w_{i,j}^{n+1} - w_{i,j}^n}{\Delta t} = \frac{w_{i+1,j}^n - 2w_{i,j}^n + w_{i-1,j}^n}{(\Delta x)^2} + \frac{w_{i,j+1}^n - 2w_{i,j}^n + w_{i,j-1}^n}{(\Delta y)^2} \tag{8}$$

The explicit Euler method can also be represented in the form of a matrix. If w is assumed to be the temperature field across a physical domain with dimensions $m \times n$, then w could be written as a column vector in $R^{m \times n}$. The explicit Euler-updated equation can be written as a matrix-vector multiplication:

$$w^{n+1} = A_{eu} w^n \tag{9}$$

Here, in the above equation, A_{eu} contains the coefficients that determine how each element in the solution element of w is updated based on its neighboring elements.

2.3 Crank-Nicolson method

The Crank-Nicolson method is a discretized form of the 2D heat equation, where the right-hand side of the equation 8 includes both the terms at the timestep $n+1$ and timestep n. The central difference approximation in both time and space is given by the following equation:

$$\begin{aligned}\frac{w_{i,j}^{n+1} - w_{i,j}^n}{\Delta t} = \frac{1}{2} \Bigg[&\frac{w_{i+1,j}^{n+1} - 2w_{i,j}^{n+1} + w_{i-1,j}^{n+1}}{(\Delta x)^2} + \frac{w_{i,j+1}^{n+1} - 2w_{i,j}^{n+1} + w_{i,j-1}^{n+1}}{(\Delta y)^2} \\ &+ \frac{w_{i+1,j}^n - 2w_{i,j}^n + w_{i-1,j}^n}{(\Delta x)^2} + \frac{w_{i,j+1}^n - 2w_{i,j}^n + w_{i,j-1}^n}{(\Delta y)^2} \Bigg]\end{aligned} \tag{10}$$

3 Stability

3.1 Conditions under which Crank-Nicolson scheme is stable

Since the grid width is taken uniformly after discretization, we can substitute the values of $\Delta x = \Delta y = h$, where h is uniform grid spacing in Equation 10, we get,

$$\begin{aligned} w_{i,j}^{n+1} - w_{i,j}^{n} = \frac{\Delta t}{2h^2} \Big[& w_{i+1,j}^{n+1} - 2w_{i,j}^{n+1} + w_{i-1,j}^{n+1} + w_{i,j+1}^{n+1} - 2w_{i,j}^{n+1} + w_{i,j-1}^{n+1} \\ & + w_{i+1,j}^{n} - 2w_{i,j}^{n} + w_{i-1,j}^{n} + w_{i,j+1}^{n} - 2w_{i,j}^{n} + w_{i,j-1}^{n} \Big] \end{aligned} \tag{11}$$

Therefore, now we substitute $\alpha = \frac{\Delta t}{h^2}$ in the above Equation 11, where α is the Courant-Friedrichs-Lewy number.

$$\begin{aligned} w_{i,j}^{n+1} - w_{i,j}^{n} = \alpha \Big[& w_{i+1,j}^{n+1} - 2w_{i,j}^{n+1} + w_{i-1,j}^{n+1} + w_{i,j+1}^{n+1} - 2w_{i,j}^{n+1} + w_{i,j-1}^{n+1} \\ & + w_{i+1,j}^{n} - 2w_{i,j}^{n} + w_{i-1,j}^{n} + w_{i,j+1}^{n} - 2w_{i,j}^{n} + w_{i,j-1}^{n} \Big] \end{aligned}$$

$$\begin{aligned} w_{i,j}^{n+1} - w_{i,j}^{n} = & \alpha \left[w_{i+1,j}^{n+1} + w_{i-1,j}^{n+1} + w_{i,j+1}^{n+1} + w_{i,j-1}^{n+1} - 4w_{i,j}^{n+1} \right] \\ & + \alpha \left[w_{i+1,j}^{n} + w_{i-1,j}^{n} + w_{i,j+1}^{n} + w_{i,j-1}^{n} - 4w_{i,j}^{n} \right] \end{aligned} \tag{12}$$

Now, we will take the $(n+1)^{th}$ timestep on our left side and keep the n^{th} timestep on our right-hand side.

$$\begin{aligned} (1+4\alpha)w_{i,j}^{n+1} - \alpha \left(w_{i+1,j}^{n+1} + w_{i-1,j}^{n+1} + w_{i,j+1}^{n+1} + w_{i,j-1}^{n+1} \right) = & \\ (1-4\alpha)w_{i,j}^{n} + \alpha \left(w_{i+1,j}^{n} + w_{i-1,j}^{n} + w_{i,j+1}^{n} + w_{i,j-1}^{n} \right) & \end{aligned} \tag{13}$$

Now, we use the discrete perturbation theory to prove stability. We add a perturbation at time $(n)^th$ and this perturbation induces a change in the $(n+1)^th$ time step.

$$\begin{aligned} (1+4\alpha)w_{i,j}^{n+1} - \alpha \left(w_{i+1,j}^{n+1} + w_{i-1,j}^{n+1} + w_{i,j+1}^{n+1} + w_{i,j-1}^{n+1} \right) = & \\ (1-4\alpha)(w_{i,j}^{n} + \epsilon_{i,j}^{n}) + \alpha \left(w_{i+1,j}^{n} + w_{i-1,j}^{n} + w_{i,j+1}^{n} + w_{i,j-1}^{n} \right) & \end{aligned} \tag{14}$$

Substituting $n = n-1$ in 13, subtracting with 14 and simplifying the equation, we get

$$(1+4\alpha)(w_{i,j}^{n+1} - w_{i,j}^{n}) = (1-4\alpha)\epsilon_{i,j}^{n} \tag{15}$$

Simplifying further, we get:

$$\frac{\epsilon_{i,j}^{n+1}}{\epsilon_{i,j}^{n}} = \frac{1-4\alpha}{1+4\alpha} \qquad \text{where} \quad (w_{i,j}^{n+1} - w_{i,j}^{n}) = \epsilon_{i,j}^{n+1} \tag{16}$$

As per discrete perturbation theory, if this ratio is less than 1, we can say that the numerical scheme is stable. $\frac{1-4\alpha}{1+4\alpha}$ is always less than 1, as the numerator is always less than the denominator. Therefore, we can say that the above numerical scheme is stable.

4 Convergence

A numerical solution is said to be convergent only if the model returns the same solution for consecutively finer grid sizes. Here, in this case, the temperature at points $x = 0.5$ and $y = 0.5$ at $t = 0.16$ is used as a parameter and plotted against varying grid sizes, the plot for which is given below. From the plot, we can infer that the values of temperature at the points $x = 0.5$ and $y = 0.5$ for grid sizes $\Delta h = \frac{1}{20}$ and $\Delta h = \frac{1}{30}$ are almost the same, inferring that the numerical results converge on refining the grid size.

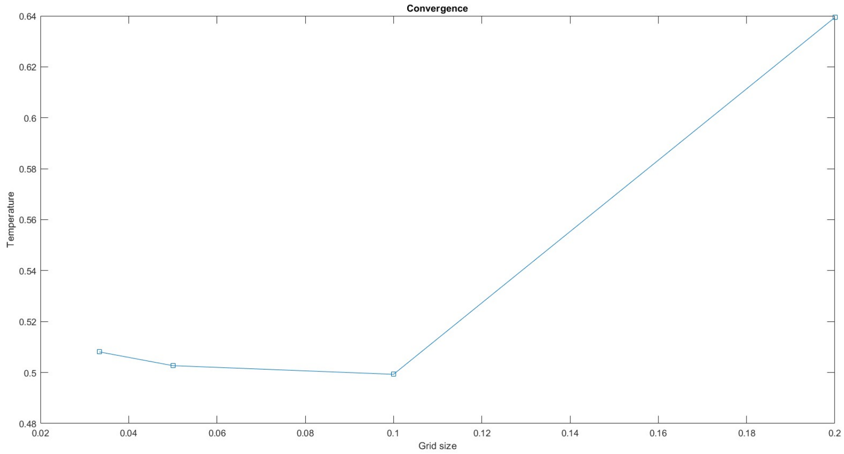


Fig. 1: Convergence

5 MATLAB codes/functions

This section discusses the implementation of the two methods: the explicit Euler and Crank-Nicolson methods. In general, both methods have

- a **while loop** for time tracking; runs from **0 to 0.16** with increments of 0.01, 0.001, or 0.0001 as the case requires
- **2 for loops**; one for traversing through the rows and another for traversing through the columns

Here, it is important to note that $\Delta x = \Delta y = \Delta h = \frac{1}{40}$ in this case. The codes are run for different Δt values, ranging from 0.01 to 0.0001 for both the Crank-Nicolson and explicit Euler methods. These values are appropriately substituted in 10 and 8, respectively, to get the final discretized equations.

6 Discussion of Results

In this section, we discuss the results and the plots obtained by running the MATLAB codes.

6.1 Time evolution of the temperature at x=y=0.4 for stable Δt

The time evolution plots for both Crank Nicolson and Explicit Euler look similar, but there is a small difference in the values obtained in the Explicit Euler method as compared to the values obtained in the Crank Nicolson method.

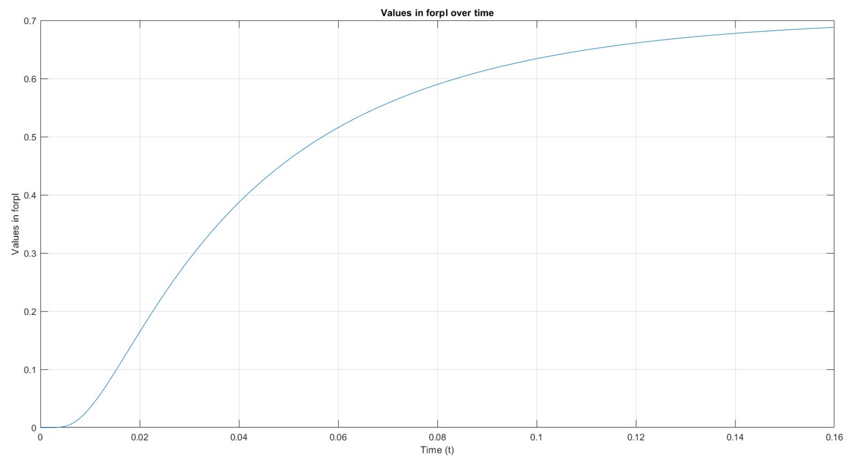


Fig. 2: Crank Nicholson: Time evolution of the temperature at x=y=0.4

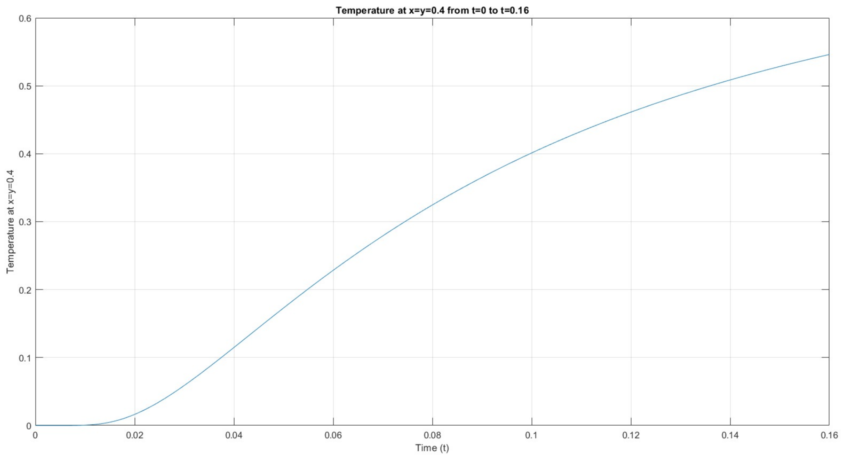


Fig. 3: Explicit Euler: Time evolution of the temperature at x=y=0.4

Both plots have similar characteristics with regard to temperature evolution: explicit Euler seems to have a gradual increase, while in the Crank-Nicolson method, the increase is rather quick.

6.2 Vertical temperature profile at t = 0.16 and x = 0.4

The vertical temperature profiles for both explicit Euler and Crank Nicholson are plotted. Both the plots look almost identical, for $\Delta t = 0.0001$. From the plots, we can infer that the temperature is maximum at $x = 0$ and $y = 1$ and minimum at $x = 1$ and $y = 0$, and has an almost linear temperature gradient along y.

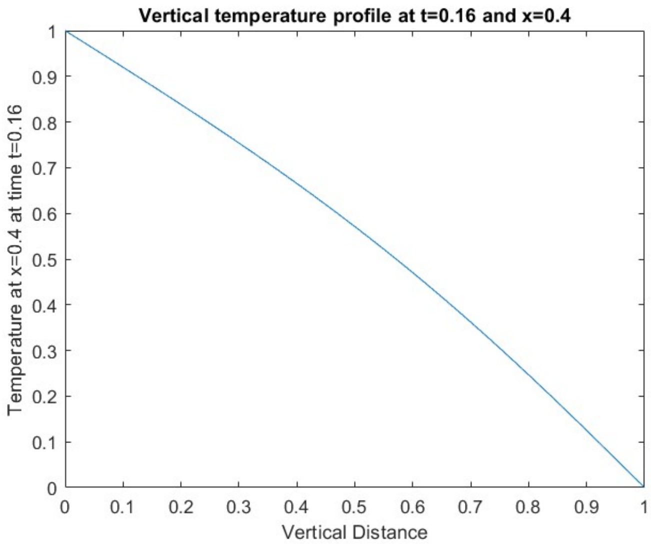


Fig. 4: Crank Nicholson: Vertical temperature profile at t = 0.16 and x = 0.4

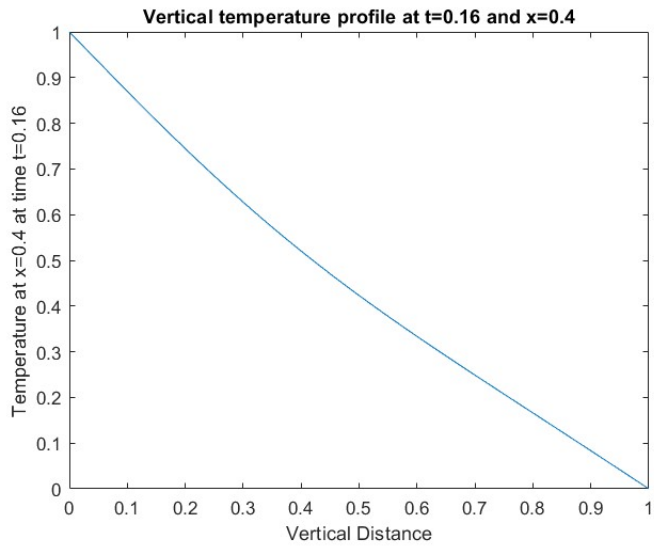


Fig. 5: Explicit Euler: Vertical temperature profile at t = 0.16 and x = 0.4

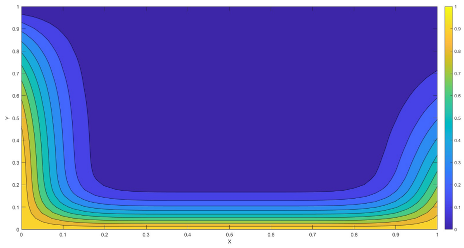
6.3 Performance of the 2 methods

Both Crank Nicholson and Explicit Euler perform well at $\Delta t = 0.0001$ and $\Delta h = \frac{1}{40}$. However, the Crank-Nicolson method can be preferred over explicit Euler, as it is the average of both implicit Euler and explicit Euler and is therefore preferable.

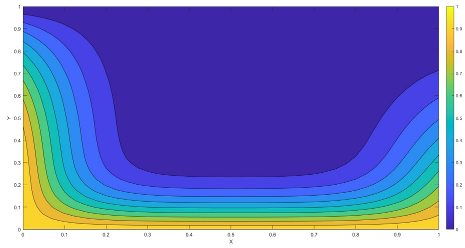
6.4 Numerical Solution for the whole domain

The numerical solution for the whole domain from $t = 0.01$, $t = 0.02$, $t = 0.04$, $t = 0.08$ and $t = 0.16$ is shown. From the plots, we can see how the temperature variation over the entire domain changes with time. This part focuses on the numerical solutions obtained from both the explicit Euler and Crank-Nicolson methods, the plots for which are shown below.

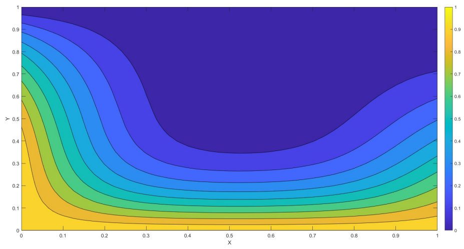
The numerical solutions for explicit Euler for the aforementioned time steps are also shown below.
From the plots, we can see the heat transfer along the surface of the domain. The bottom and left walls have a higher temperature in the beginning, and this heat is then transferred along the surface of the domain as depicted by 1. This heat slowly transfers from regions of high temperature to regions of lower temperature, i.e., from the left and bottom walls towards the top and right walls.



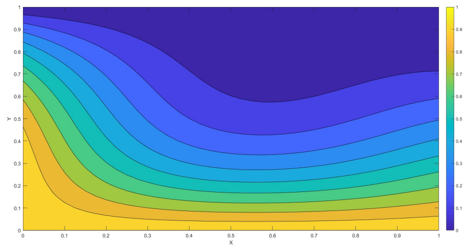
(a) Solution at t=0.01



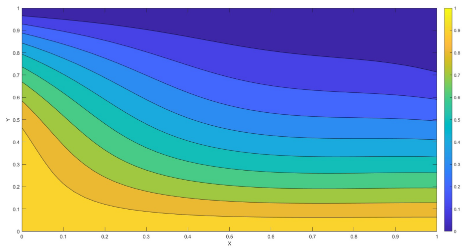
(b) Solution at t=0.02



(c) Solution at t=0.04

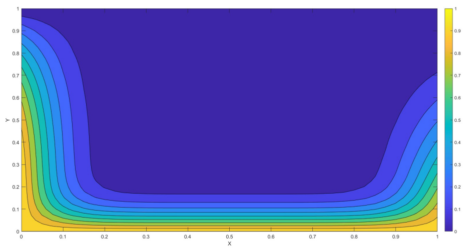


(d) Solution at t=0.08

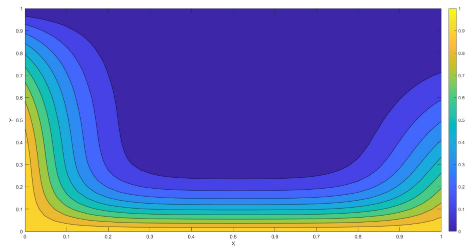


(e) Solution at t=0.16

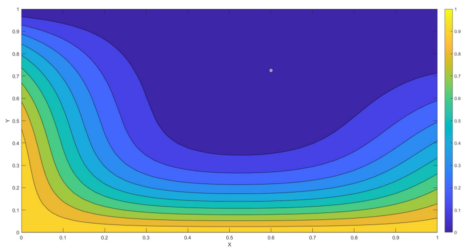
Fig. 6: Crank Nicholson: Numerical Solution over the entire domain



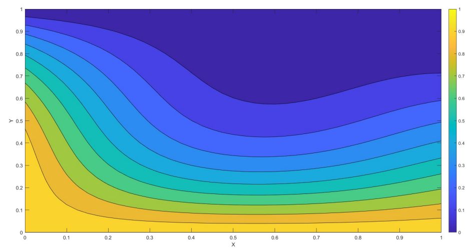
(a) Solution at t=0.01



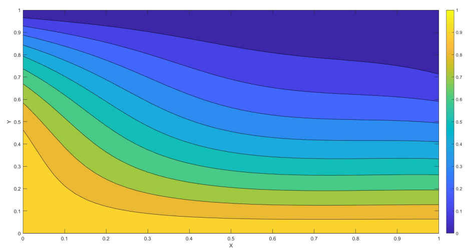
(b) Solution at t=0.02



(c) Solution at t=0.04



(d) Solution at t=0.08



(e) Solution at t=0.16

Fig. 7: Explicit Euler: Numerical Solution over the entire domain