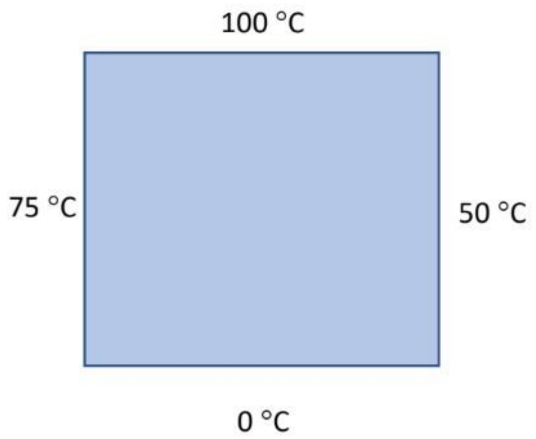
Problem T1*

Consider two-dimensional unsteady state heat conduction in a square aluminium plate. At t =0, assume that the temperature of the plate is zero and the boundary temperatures are instantaneously brought to the levels shown below. The thermal diffusivity of aluminium is 0.835 cm^2/s. Using explicit method and Alternating-Direction Implicit (ADI) method, obtain the temporal evolution of the temperature distribution of the plate. Compare the performance of the two methods.



Solution :

The *heat conduction equation* in 2-dimension is given as :

$$\frac{\partial T}{\partial t} = \alpha\left(\frac{\partial^2 T}{\partial x^2} + \frac{\partial^2 T}{\partial y^2}\right)$$

Here α is the thermal diffusivity $\alpha = 0.835\ cm^2/s$

Solution by explicit method :

To employ the explicit method, the second order partial differential can be approximated by a centered finite-divided difference

$$\frac{\partial^2 T}{\partial x^2} = \frac{T^n_{i+1,j} - 2T^n_{i,j} + T^n_{i-1,j}}{\Delta x^2} \quad \text{and} \quad \frac{\partial^2 T}{\partial y^2} = \frac{T^n_{i,j+1} - 2T^n_{i,j} + T^n_{i,j-1}}{\Delta y^2}$$

Substituting these in two heat conduction equation, we get

$$\frac{T^{n+1}_{i,j} - T^n_{i,j}}{\Delta t} = \alpha(\frac{T^n_{i+1,j} - 2T^n_{i,j} + T^n_{i-1,j}}{\Delta x^2} + \frac{T^n_{i,j+1} - 2T^n_{i,j} + T^n_{i,j-1}}{\Delta y^2})$$

$$T^{n+1}_{i,j} = T^n_{i,j} + \lambda(T^n_{i+1,j} + T^n_{i-1,j} + T^n_{i,j-1} + T^n_{i,j+1} - 4T^n_{i,j})$$

Where $\lambda = \frac{\alpha\Delta t}{(\Delta x)^2}$ note that $\Delta x = \Delta y$

Let the dimension of the square plate be 100cm. $\boldsymbol{L = 40cm}$.

The explicit method is implemented in a MATLAB program named T1_explicit.m.

The program solves for the temperature of plate for a **time interval** of 0s to 100s in time-step of $\boldsymbol{\Delta t = 10s}$ and the distance step of $\boldsymbol{\Delta x = 10cm.}$ The value of temperature of plate at $t = 100s$ is tabulated below :

100	100	100	100	100
75	61.87548	53.39645	53.67336	50
75	41.93041	27.45611	32.23881	50
75	29.067	14.63008	20.86488	50
0	0	0	0	0

The stability criterion for the explicit method, gives a lower bound on the value of the time and the distance step, used for calculations. The stability criterion for two-dimensional equation is

$$\Delta t \leq \frac{1}{8}\frac{(\Delta x)^2 + (\Delta y)^2}{k}$$

Where in this case $k = \frac{\alpha\Delta t}{\Delta x^2} = 0.0835$

For given time step of Δt=10s , and $\Delta x = \Delta y$, one can get $(\Delta x)^2 \geq 4k\Delta t = 3.328$

$$\therefore \Delta x \geq 1.83$$

This upper bound on Δx can be verified by putting $\Delta x = 1$ in the MATLAB program. The program will give negative values of temperatures, suggesting that the explicit method becomes unstable of smaller Δx.

A contour plot of temperature, at $t = 100s$ is shown below :

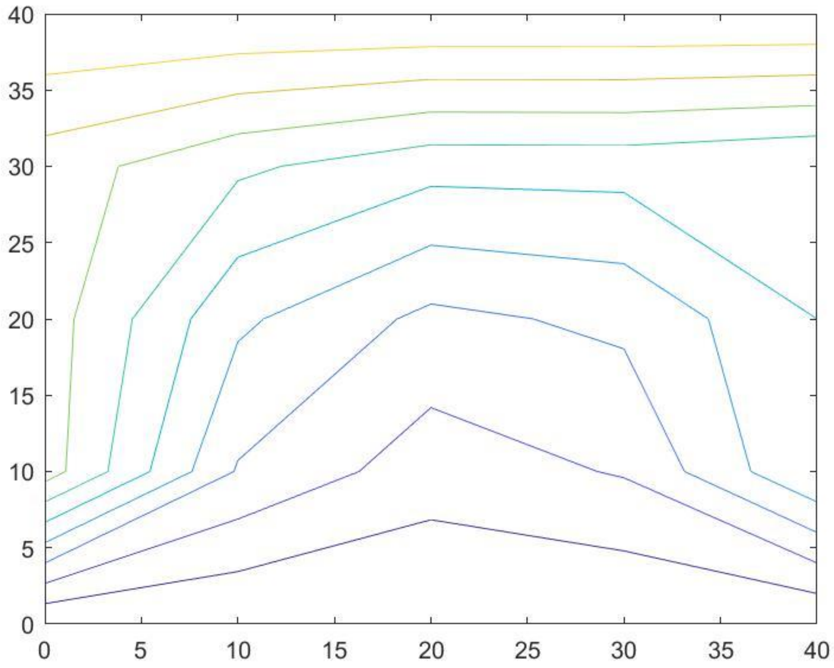


Figure 1 Contour plot of temperature at t=100s

Another type of contour plot of temperature at t=100s, which is more colourful is shown below :

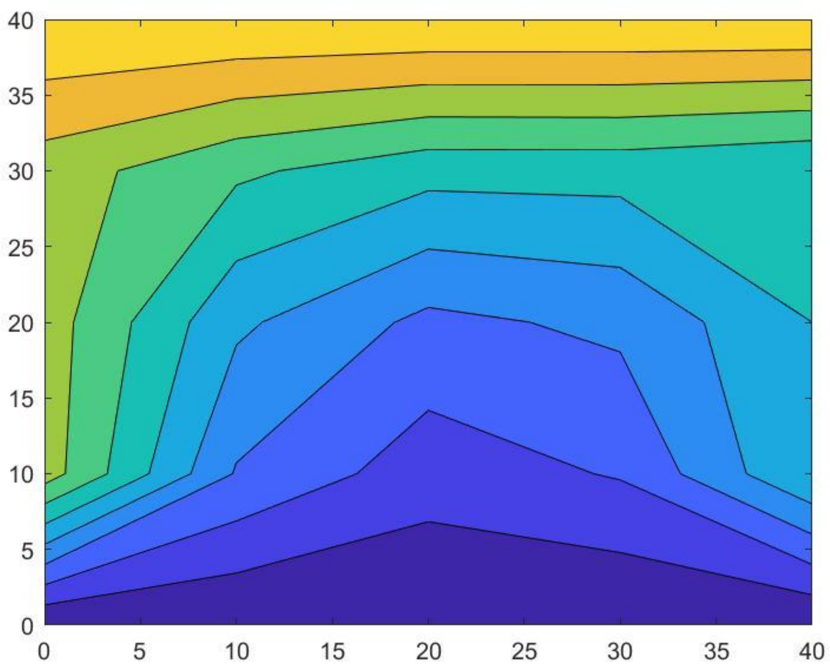


Figure 2 Colourful contour plot of temperature at t=100s

A mesh plot of temperature at time $t = 100s$ is also plotted using MATLAB, the plot is shown below :

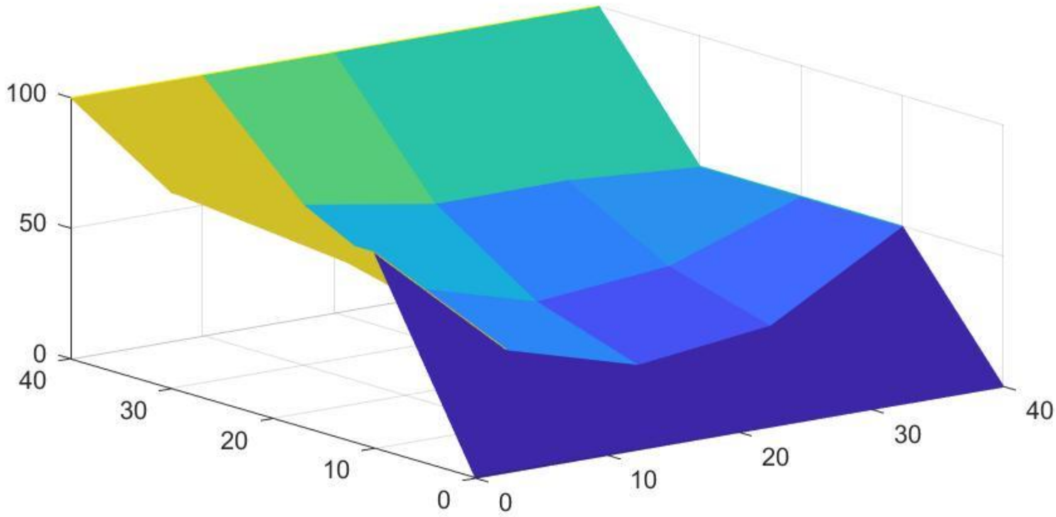


Figure 3 Mesh plot of temperature at t=100s

Solution by ADI method :

Direct substitution of the implicit differentials into the heat conduction equation, results in a simultaneous system of $m \times n$ equation. These equations cannot be solved efficiently.

The Alternating Direction Implicit method, the time step is incremented in two step. In the first step, the approximation for $\partial^2 T/\partial x^2$ is written explicitly whereas $\partial^2 T/\partial y^2$ is written implicitly, as

$$\frac{T_{i,j}^{n+1/2} - T_{i,j}^{n}}{\Delta t/2} = \alpha(\frac{T_{i+1,j}^{n} - 2T_{i,j}^{n} + T_{i-1,j}^{n}}{\Delta x^2} + \frac{T_{i,j+1}^{n+1/2} - 2T_{i,j}^{n+1/2} + T_{i,j-1}^{n+1/2}}{\Delta y^2})$$

Also, $\Delta x = \Delta y$ and $\lambda = \alpha \Delta t/(\Delta x^2)$ On rearranging we get,

$$-\lambda\, T_{i,j-1}^{n+1/2} + 2(1+\lambda)T_{i,j}^{n+1/2} - \lambda\, T_{i,j+1}^{n+1/2} = \lambda T_{i-1,j}^{n} + 2(1-\lambda)\, T_{i,j}^{n} + \lambda\, T_{i+1,j}^{n} \qquad (\mathbf{1})$$

The step in equation-**(1)** is popularly known as *column sweep*, as after this step temperature at $(n + 1/2)th$ time will be determined for ith column.

The above equation can be solved by using efficient algorithm of *Thomas*. In the next step the approximation for $\partial^2 T/\partial y^2$ is written explicitly whereas $\partial^2 T/\partial x^2$ is written implicitly, as shown below :

$$\frac{T_{i,j}^{n+1} - T_{i,j}^{n+1/2}}{\Delta t/2} = \alpha(\frac{T_{i+1,j}^{n+1} - 2T_{i,j}^{n+1} + T_{i-1,j}^{n+1}}{\Delta x^2} + \frac{T_{i,j+1}^{n+1/2} - 2T_{i,j}^{n+1/2} + T_{i,j-1}^{n+1/2}}{\Delta y^2})$$

On rearranging, we get

$$-\lambda\, T_{i-1,j}^{n+1} + 2(1+\lambda)T_{i,j}^{n+1} - \lambda\, T_{i+1,j}^{n+1} = \lambda T_{i,j-1}^{n+1/2} + 2(1-\lambda)T_{i,j}^{n+1/2} + \lambda T_{i,j+1}^{n+1/2} \qquad (\mathbf{2})$$

This step is known as *row sweep*, as at the end of this step temperature would have been solved for one row.

The above equations results in a system of tridiagonal matrix, which can be solved using *Thomas algorithm* (Tridiagonal Matrix Algorithm TDMA).

For every time-interval, the ADI method involves in *alternate* sweep between columns and rows, the method got its name as *Alternating Direction Implicit.*

After solving the equation, **(2)** the temperature distribution at the jth column can be determined, this process has to be repeated for all the columns.

Note that the index i, j appearing in $T_{i,j}$ denotes the temperature at the jth row and ith column in the program.

For the same dimension of square plate as $L = 40cm$, and the same time step of $\boldsymbol{\Delta t = 10s}$ and the upper bound of time as $t = 100s$, with the distance step of $\boldsymbol{\Delta x = 10cm.}$ A MATLAB program named T1_ADI_18110174.m was developed. The results of the temperature calculated for the last time of $t = 100s$ is tabulated below :

100	100	100	100	100
75	76.54491	73.29287	67.6753	50
75	60.29949	52.2519	49.66919	50
75	40.825	30.42844	31.95622	50
0	0	0	0	0

Since, the ADI uses the implicit approach to evaluate the partial differentials, it is *unconditionally* stable. Thus to get more accurate temperature measurements, the distance step Δx can be reduced without any constraints.

The contour plot of temperature at $t = 300s$, calculated using $\Delta x = 2cm$ for a $40cm$ long plate is shown below :

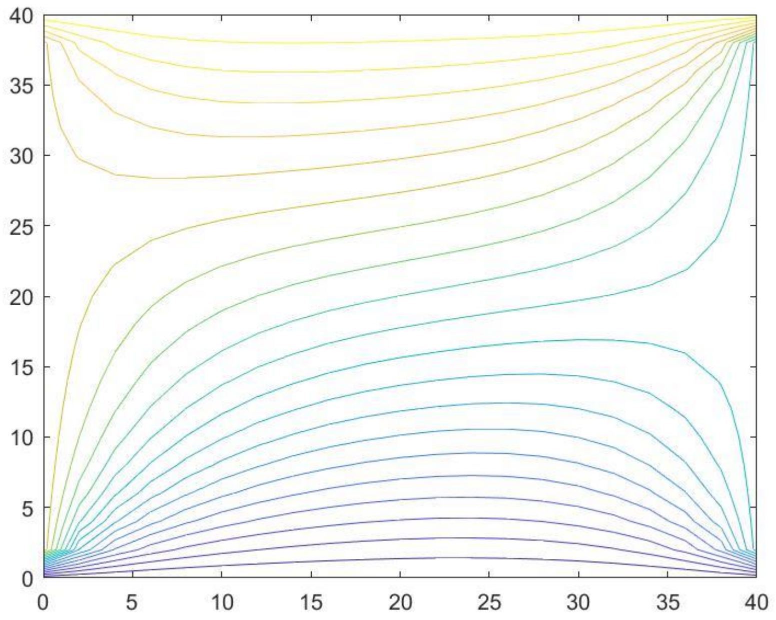


Figure 4 Contour plot of temperature at t=300s

Another type of contour plot of temperature at t=300s for the same time-step and distance-step is shown below :

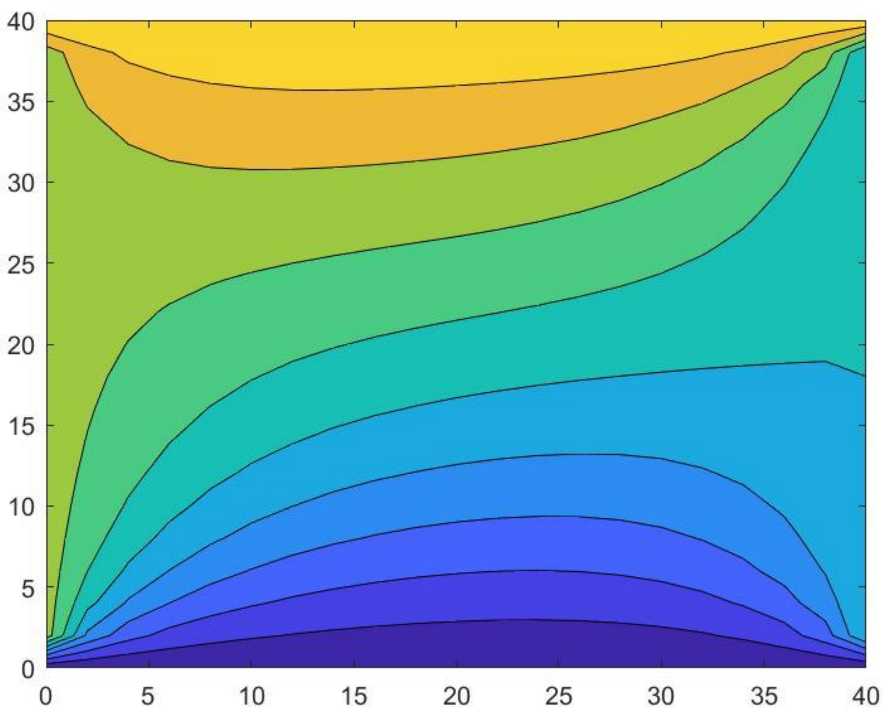


Figure 5 Colourful contour plot of temperature at t=300s, by ADI method

A mesh plot of temperature at time $t = 300s$, using ADI method for same time and distance step is plotted using MATLAB, the plot is shown below :

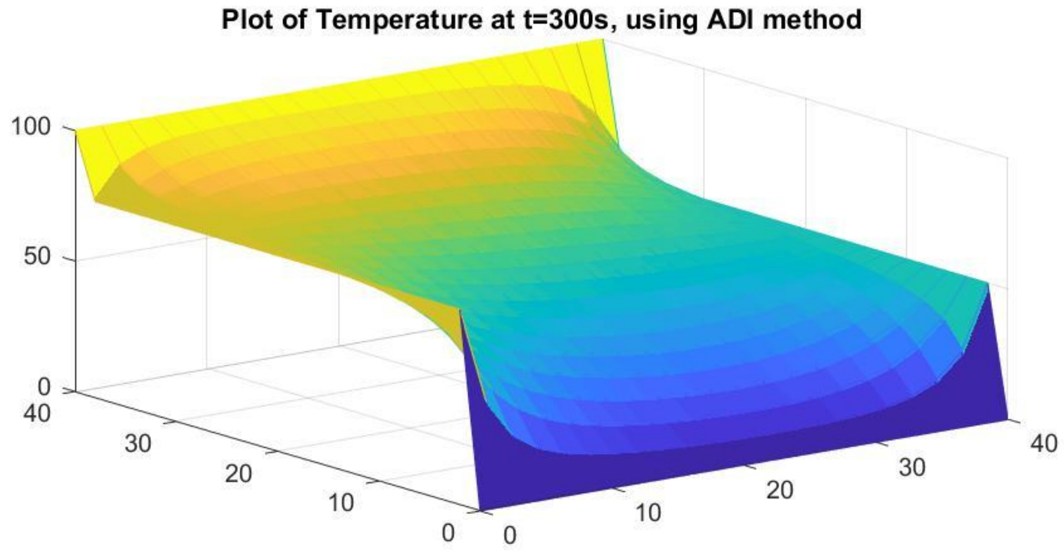


Figure 6 Mesh plot of temperature at t=300s, using ADI method

Comparison and Discussion :

The explicit method suffers from stability issues as distance step is reduced. However, the ADI method is unconditionally stable, it will remain stable even when the distance step is made too small. The ADI takes advantage of the TDMA, to efficiently solve the system of equation.

Since, ADI is an iterative method, it has a faster convergence as compared with the explicit method.