Project 1: MECH 6371
Computational Fluid Mechanics

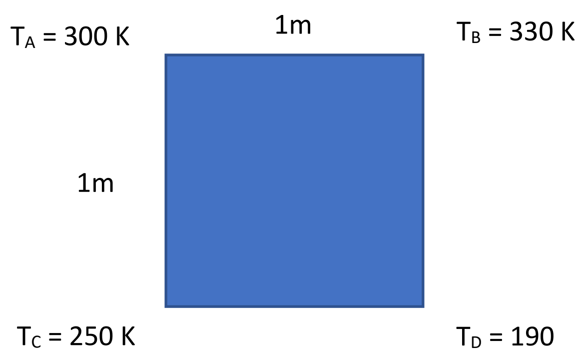
Yajat Pandya
UTD ID: 2021440761

1. Introduction

The problem given is a 2-D transient conduction problem of an Aluminum plate with dimensions of 1m x 1m. The corners of the plate are externally forced to be at a certain temperature with the edges having a linear gradient between the corners.

Initial condition: T (x, y, t=0) = 200 K;

At steady-state:



Linear temperature increase along the edges of the plate.

Required:

1. Steady-state temperature distribution
2. Temperate distribution at t = 10, 50 and 100 s
3. Isolines of temperature
4. T (x=0.5, y) – steady state
5. Grid dependence study

2. Methodology

Governing equation for the given 2-D plate:

$$\frac{1}{\alpha}\frac{\partial T}{\partial t} = \frac{\partial^2 T}{\partial x^2} + \frac{\partial^2 T}{\partial y^2}$$

Let T (x, y, t) – temperature at any (x, y, t) spatio-temporal coordinate on the plate. For the material Aluminum, the thermal diffusivity is $\alpha = 9.7 \times 10^{-5}$ m^2/s. Also, the given equation is a second order PDE which can be formulated in terms of an explicit **first order finite difference method**. That is,

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

As this is a conditionally stable method, we need to choose the grid size and time-step size with the following condition: (equal grid size in both directions)

$$r = \frac{2\alpha\Delta t}{(\Delta x)^2} \leq \frac{1}{2}$$

Because of the low thermal diffusivity of Aluminum, it is possible to have initial simulations with a coarse grid before making it fine for the grid dependence study. Hence, a coarse grid was chosen initially and the steady-state time was noted. For the grid dependence study, the time-step was constant and only the grid size was varied in the following manner:

Δt (s)	**Δx** (m)	**r**
0.25	0.01	0.485
0.25	0.02	0.121
0.25	0.05	0.019
0.25	0.1	0.004

Table 1: Grid size variation

From the linear gradient boundary condition at the edges, it is important to note that for all t > 0,

$$T(x,0,t) = T_C + (T_D - T_C)x$$

$$T(x,1,t) = T_A + (T_B - T_A)x$$

$$T(0,y,t) = T_C + (T_A - T_C)y$$

$$T(1,y,t) = T_D + (T_B - T_D)y$$

In order to start the solving process, we set the initial parameters of the number of time-steps, time-step size and spatial grid size (we use same grid size in both the directions due to shape symmetry). We the develop a 3-D node array of the variable Temperature T with the third dimension being the time. Thereafter, we initialize T (x, y, 1) = 200 K. Next, we use the above linear relations to find the temperature at all the nodes along the edges. Hence, we use the above finite difference equation only at the nodes *inside* the plate. So, we continue by creating a three for-loop algorithm with the temporal dimension in the most external loop and (x, y) dimension in the inner two loops.

Every n^{th} iteration of the outermost loop will successfully create temperature data for the n^{th} timestep. This will continue for all timesteps. Note that the number of timesteps is chosen by the user; in this case, as mentioned above, the steady-state time was noted using initial simulations of coarse grid.

3. Results

In order to know the steady state time, we compute the mean residual from all nodes on the plate i.e. the spatial average of the difference of temperature at a node at n^{th} timestep and at $(n - 1)^{th}$ timestep.

$$Residual = Spatial\ Mean(T(:,:n) - T(:,:,n-1))$$

Considering computing limitations, we take the threshold of 10^{-4} for computing the steady state solution time from the residual plot. Fig. 1. shows the variation of residual with time for the highest grid resolution. It is clear after t = 3000 s the average residual drops below 10^{-4}. Hence, we take t = 3000 s as the steady state time. This is of course for only one value or r – stability parameter, but in order to get an estimate of the steady state time, computing the steady state variation with r seemed unnecessary.

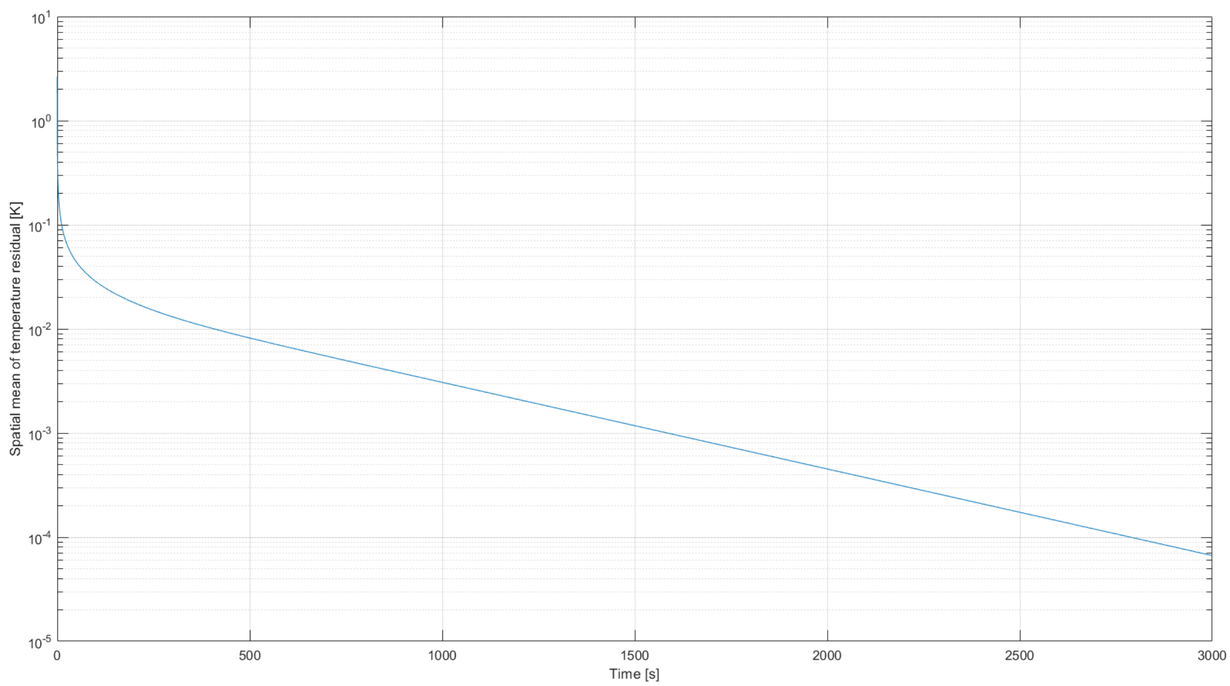


Fig. 1: Average residual [K] variation with time [s]

For the highest grid resolution, the steady state profile is shown in Fig. 2. The overall profile has achieved convergence in accordance with the forced boundary conditions. This makes the profile look extremely polarized from point D to B. Due to this, the rightmost edge, DB, of the plate has maximum notable temperature gradient.

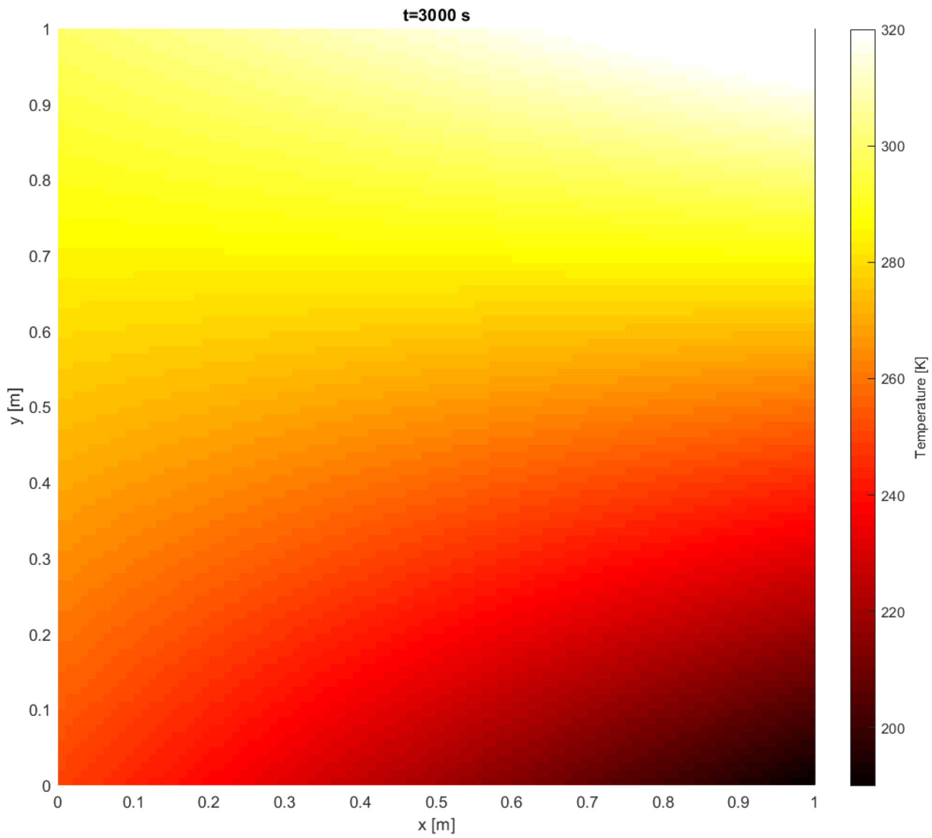


Fig. 2: Steady state temperature profile for Δx = 0.01 m

Similarly, the profiles at t = 10, 50 and 100 s are shown in Fig 3. At t = 10 s, the conduction has barely started to affect the majority area of the plate. Whereas at t = 100 s, the temperature shift is noted to happen with the low temperature shifted towards point D and high temperature shifted towards point B.

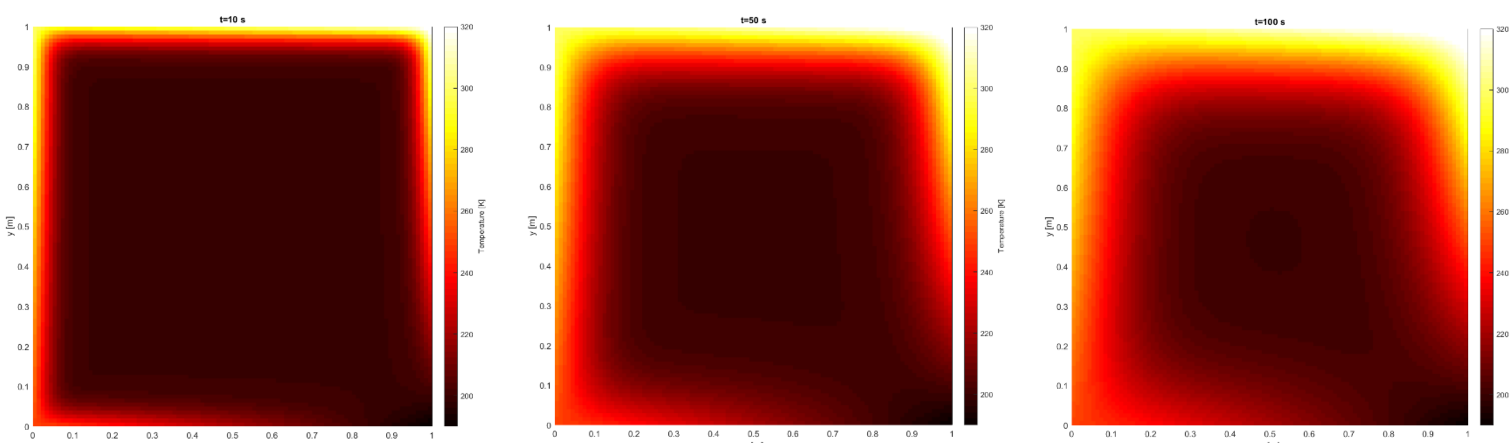


Fig. 3: Instantaneous temperature profiles at t = 10, 50 and 100 s

The temperature isolines are shown in Fig. 4 for t = 10, 50, 100 and 3000 s. As expected at steady state, most of the profiles originate on the DB edge of the plate and end on other edges. Moreover, moving from D to B increases the temperature of the isolines. Whereas the heat conduction is still in the process of reaching the center of the plate at other timesteps.

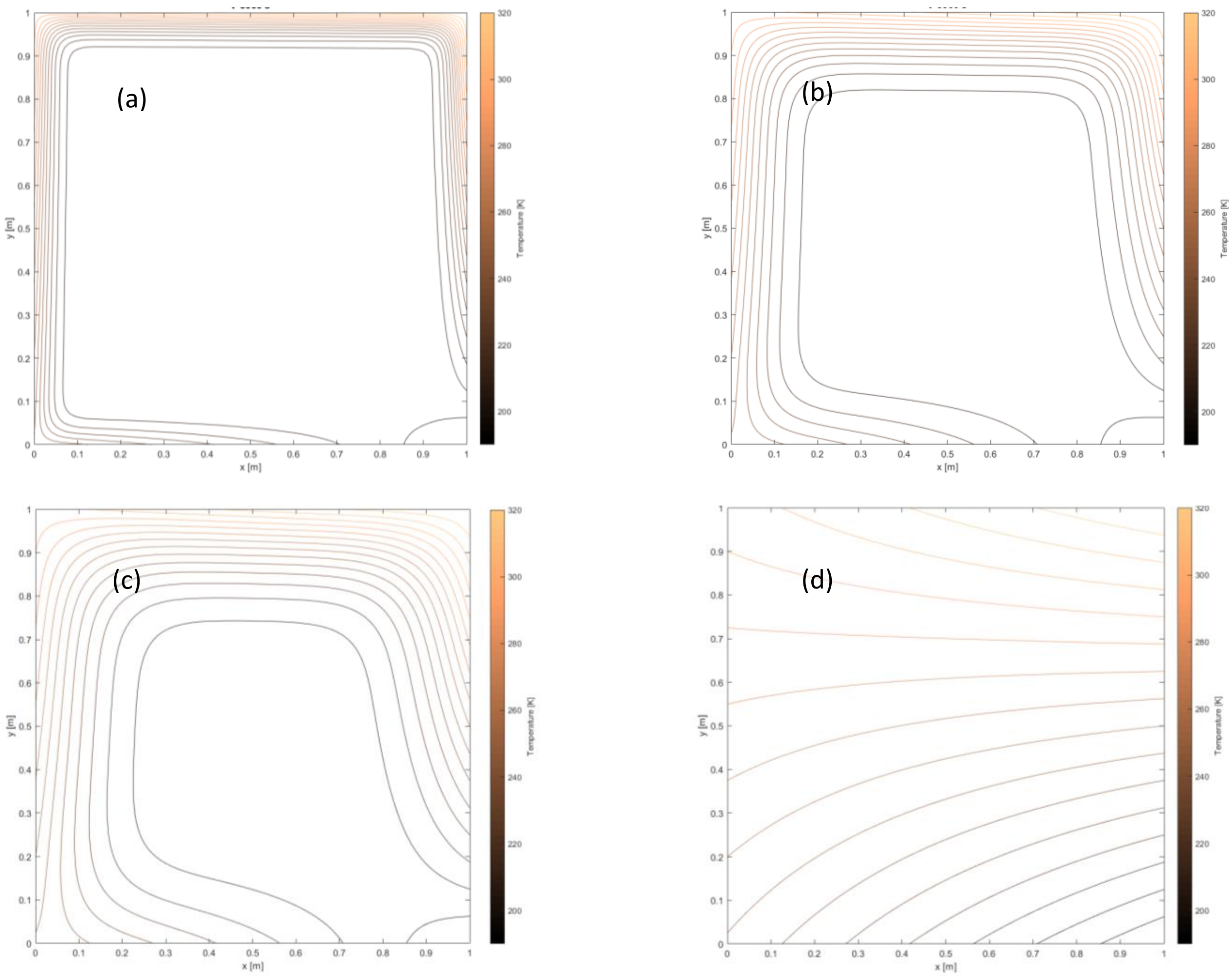


Fig. 4: (a), (b), (c) and (d) – Temperature isolines at t = 10, 50, 100 s and steady state

Temperature at x = 0.5 m for the grid size Δx = 0.01 m is shown in Fig. 5. The temperature variation is linear. This is because at steady state, the linear boundary conditions will force the plate to have linear gradients along the perpendicular directions.

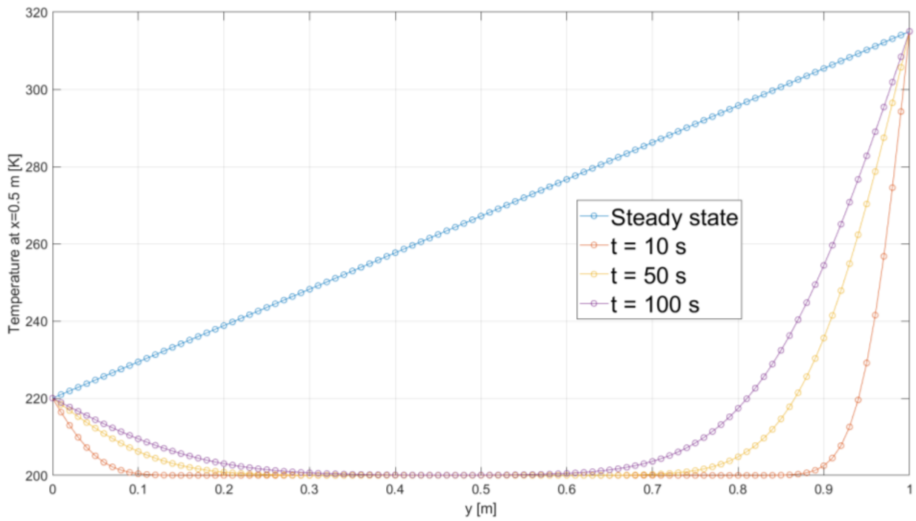


Fig. 5: Temperature at x = 0.5 m as a function of y; Steady state is t = 3000 s

Grid dependence study was conducted for four different grid sizes as abovementioned in Table 1.

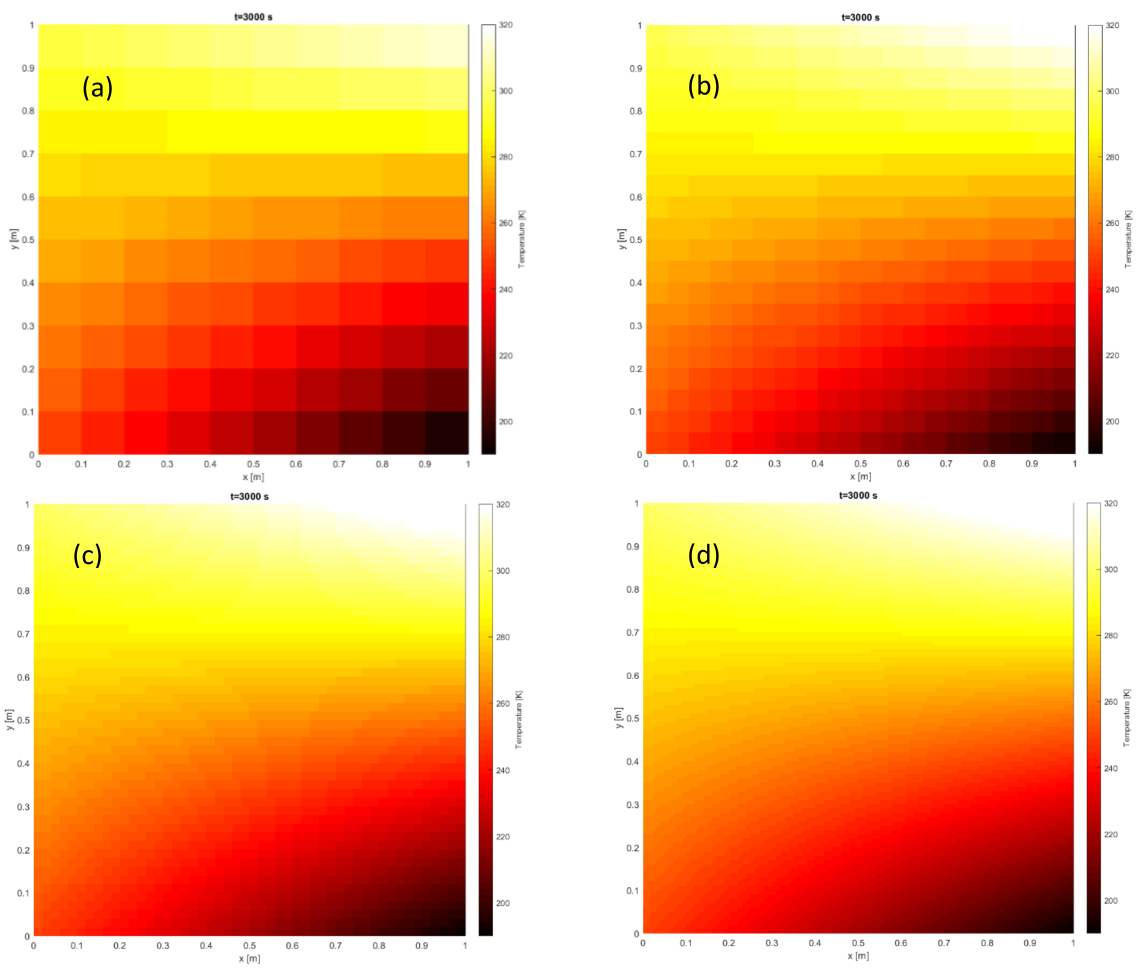


Fig. 6: (a), (b), (c) and (d) – Temperature profiles for Δx = 0.1, 0.05, 0.02 and 0.01 m respectively

Running a grid dependence test of the vertical temperature profiles is redundant as the steady state profiles are linear at x = 0.5 m. All profiles would fall onto each other as they are linear. So, we choose the temperature of the midpoint (center), i.e. T (0.5, 0.5, 3000 s) of the plate to run the grid dependence test. It is important to note that as we increase the number of grid points, the stability factor r increases to more than 0.5, so in some of the higher grid sizes, the parameter Δt has been chosen in a way that the scheme is always stable for all grid sizes. Fig. 7 shows the variation of T (0.5, 0.5, 3000 s) with the total number of grid points (n_x x n_y).

It is noted that the temperature at the midpoint of the plate stops varying significantly more than a grid size of Δx = 0.0135 for Δt = 0.15 s. This confirms that the solution is steady and spatially convergent enough in terms of the temperature at the midpoint of the plate.

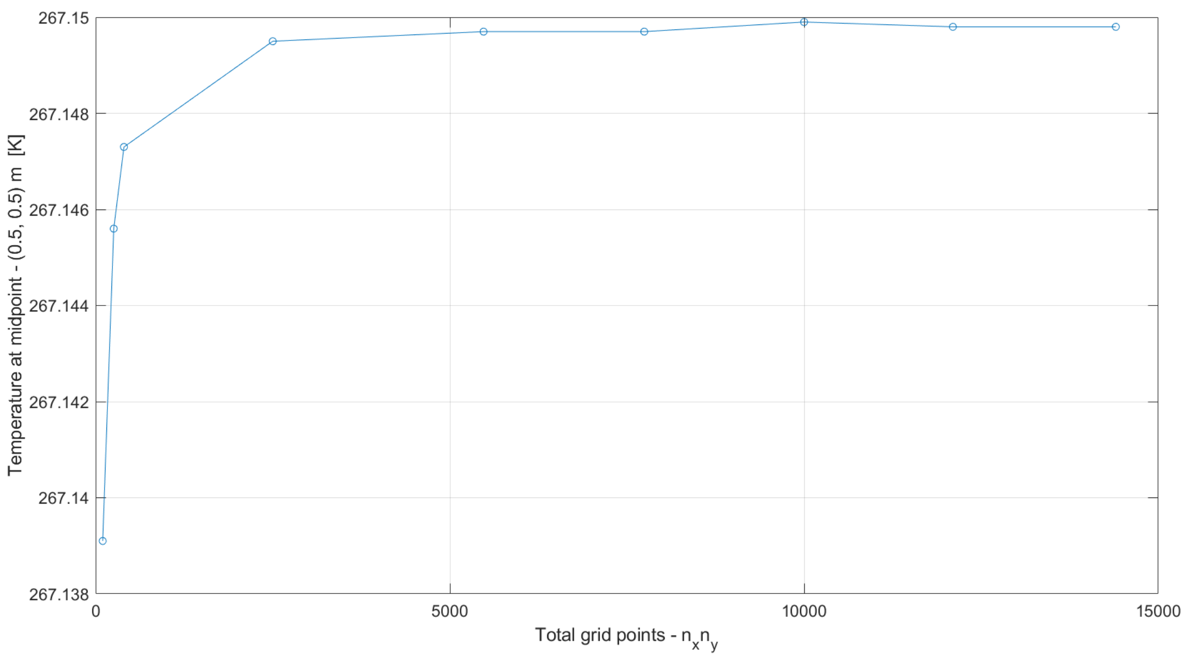


Fig. 7: Temperature at (x,y) = (0.5, 0.5) m as a function number of grid points – $n_x n_y$

4. Conclusions

This project consisted of formulating, designing and implementing a numerical solver for a 2D conductive plate with prescribed initial and boundary conditions. The problem algorithm was developed using MATLAB with an explicit finite difference scheme for the conductive heat equation. Timestep size and grid size were chosen considering the conditional stability of the scheme. Four cases were simulated based on the grid size.

Using spatial average of the residuals, steady state convergence criteria was decided, and steady state results were analyzed for all four cases. It was found that the linear nature of the boundary conditions is homogenized throughout the 2D plate in the perpendicular directions at steady state, although with different slope values. Moreover, it is important to note that for a 1 m plate, it takes approximately 3000 s to reach steady state. This is because Aluminum has a very low thermal diffusivity – meaning the heat effects would take significant time to affect the whole body compared to a highly conductive material like *Silver* or *Copper*.

It was thus possible to visualize the evolution of heat transfer in the 2D plate at every instant, the duration of which would be from the user input. The steady state profiles for different grid sizes were resolved along with the temperature isolines. It was noted that at steady state, the whole plate observes a linear gradient along the x and y directions. This is due to the forcing from the boundary conditions along the edges of the plate.

The following MATLAB code was developed for this project.

```
clear
close;

%% Initialization
dt=0.15; %time step in sec
tmax=3000; %simulation time limit
nx=88; ny=88; %number of grid points
dx=1/nx;dy=1/ny;
al=9.7*10^(-5); %aluminum thermal diffusivity
r=2*al*dt/(dx*dx);
T(1:1:nx+1,1:1:ny+1,1:1:tmax/dt+1)=200; %Temperature spatio-temporal array
initialization

%% CFD
sx=al*dt/(dx)^2;sy=al*dt/(dy)^2;
res(1)=0;
for nt=2:1:tmax/dt+1
    % boundary conditions
    T(:,1,nt)=250+(190-250).*(0:1:nx).*dx;
    T(:,ny+1,nt)=300+(330-300).*(0:1:nx).*dx;
    T(1,:,nt)=250+(300-250).*(0:1:ny).*dy;
    T(nx+1,:,nt)=190+(330-190).*(0:1:ny).*dy;

    % finite diff scheme
    for y=2:1:ny
        for x=2:1:nx
            T(x,y,nt)=T(x,y,nt-1)+sx*(T(x,y+1,nt-1)+T(x,y-1,nt-1)-2*T(x,y,nt-
1))...
                +sy*(T(x+1,y,nt-1)+T(x-1,y,nt-1)-2*T(x,y,nt-1));
        end
    end
%     res(nt)=mean(mean(T(:,:,nt)-T(:,:,nt-1))); %residual avg.
%     if res(nt)<10^(-4)
%         tim=(nt-1)*dt;
%         break
%     end
end

%% POST PROCESSING
for nt=1:10/dt:tmax/dt+1
    figure;set(gcf, 'Position', get(0, 'Screensize'));
    pcolor((0:1:nx).*dx,(0:1:ny).*dy,T(:,:,nt)');shading flat;axis
equal;caxis([190, 320]);
    colormap(gca,'hot');c = colorbar;c.Label.String = 'Temperature [K]';
    xlim([0,1]);ylim([0,1]);
    ylabel('y [m]');xlabel('x [m]');title(['t=',num2str((nt-1)*dt),' s']);
    w = waitforbuttonpress;
    close
end
```

```
%%
figure;set(gcf, 'Position', get(0, 'Screensize'));
contour((0:1:nx).*dx,(0:1:ny).*dy,T(:,:,end)',15);shading flat;axis
equal;caxis([190, 320]);
colormap(gca,'copper');c = colorbar;c.Label.String = 'Temperature [K]';
xlim([0,1]);ylim([0,1]);
ylabel('y [m]');xlabel('x [m]');title(['t=',num2str((nt-1)*dt),' s']);
%%
figure;set(gcf, 'Position', get(0, 'Screensize'));
plot([100,256,400,2500,74*74,88*88,10000,12100,14400],...

[267.1391,267.1456,267.1473,267.1495,267.1497,267.1497,267.1499,267.1498,267.
1498],'-o');grid on;
ylabel('Temperature at midpoint - (0.5, 0.5) m  [K]');xlabel('Total grid
points - n_xn_y');
```