

Case Study 1:- 2D Unsteady Conduction Equation

General eqⁿ:-

$$\frac{\partial}{\partial x} \left(k \frac{\partial T}{\partial x} \right) + \frac{\partial}{\partial y} \left(k \frac{\partial T}{\partial y} \right) + \frac{\partial}{\partial z} \left(k \frac{\partial T}{\partial z} \right) + q_v = \rho \cdot C_p \cdot \frac{\partial T}{\partial t}$$

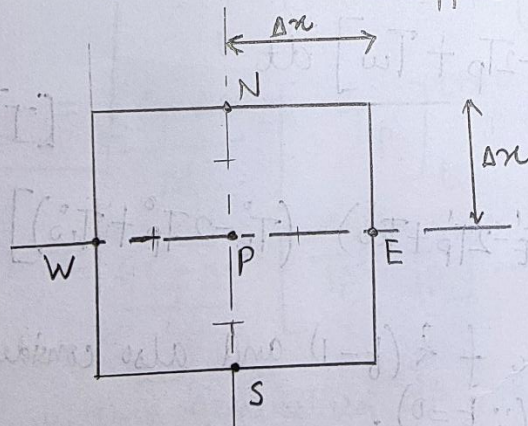
To get:- 2D, Unsteady, without heat generation.

$$q_v = 0$$

$$\frac{\partial}{\partial z} \left(k \frac{\partial T}{\partial z} \right) = 0$$

$$\therefore \frac{\partial}{\partial x} \left(k \frac{\partial T}{\partial x} \right) + \frac{\partial}{\partial y} \left(k \frac{\partial T}{\partial y} \right) = \rho C_p \frac{\partial T}{\partial t}$$

Figure:- Assuming square plate,
 \rightarrow Copper



Solving eqⁿ ①, $\frac{\partial}{\partial x} \left(k \frac{\partial T}{\partial x} \right)$

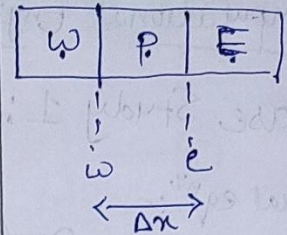
Double integrate it w.r.t time (dt) and space (dx).

$$\int \int \frac{\partial}{\partial x} \left(k \frac{\partial T}{\partial x} \right) dx dt$$

Using limits as,

Time $t = t$, $x = \omega$

$t = t + \Delta t$, $x = e$



$$\Rightarrow \int_t^{t+\Delta t} \int_{\omega}^e \frac{\partial}{\partial x} \left(k \frac{\partial T}{\partial x} \right) dx dt$$

$$= \int_t^{t+\Delta t} \left[k \frac{\partial T}{\partial x} \right]_{\omega}^e dt$$

$$= \int_t^{t+\Delta t} \left\{ (k_e - k_{\omega}) \cdot \left[\left(\frac{\partial T}{\partial x} \right)_e - \left(\frac{\partial T}{\partial x} \right)_{\omega} \right] \right\} dt$$

$$= \int_t^{t+\Delta t} k \cdot \left[\frac{T_E - T_P}{\Delta x} - \left(\frac{T_P - T_{\omega}}{\Delta x} \right) \right] dt$$

$$= \frac{k}{\Delta x} \int_t^{t+\Delta t} [T_E - 2T_P + T_{\omega}] dt$$

$$\Delta x = x_e - x_{\omega}$$

$$[T]_t^{t+\Delta t} = T' - T^{\circ}$$

$$= \frac{k}{\Delta x} [(T_E' - 2T_P' + T_{\omega}') - (T_E^{\circ} - 2T_P^{\circ} + T_{\omega}^{\circ})]$$

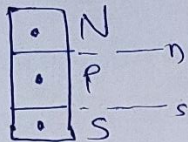
Using weighing factor f & $(f-1)$ and also considering explicit method. ($\therefore f=0$).

$$= \frac{k}{\Delta x} [f(T_E' - 2T_P' + T_{\omega}') - (f-1)(T_E^{\circ} - 2T_P^{\circ} + T_{\omega}^{\circ})]$$

$$= \frac{k}{\Delta x} [T_E^{\circ} - 2T_P^{\circ} + T_{\omega}^{\circ}]$$

(4)

for



$$\left\{ \frac{\partial}{\partial y} \left(k \frac{\partial T}{\partial y} \right) dy dt \right\} \quad dy = dx \text{ here.}$$

we get,

$$\frac{k}{\Delta x} [T_N^0 - 2T_P^0 + T_S^0] \quad \text{--- (5)}$$

Solving eqⁿ (3), $\int_t^{t+\Delta t} \int_e^w \rho C_p \cdot \frac{\partial T}{\partial t} dx dt$

$$= \rho C_p \int_t^{t+\Delta t} \frac{\partial T}{\partial t} dt \cdot \int_e^w dx$$

$$= \rho C_p \cdot \int_t^{t+\Delta t} \partial T dt \cdot (x_w - x_e)$$

$$\int_t^{t+\Delta t} \partial t dt \longrightarrow [t]_t^{t+\Delta t}$$

$$= \frac{\rho C_p \Delta x}{\Delta t} [T_P]_t^{t+\Delta t}$$

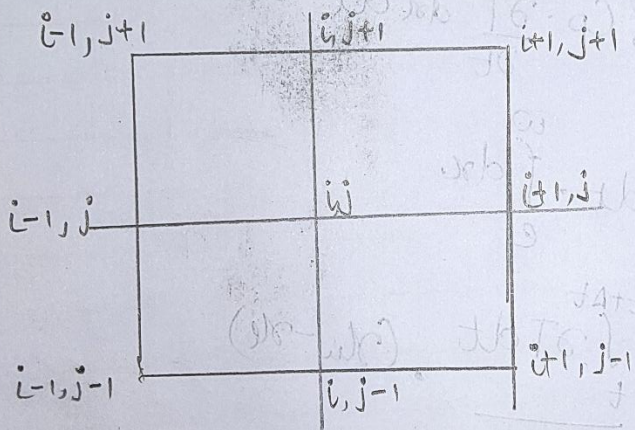
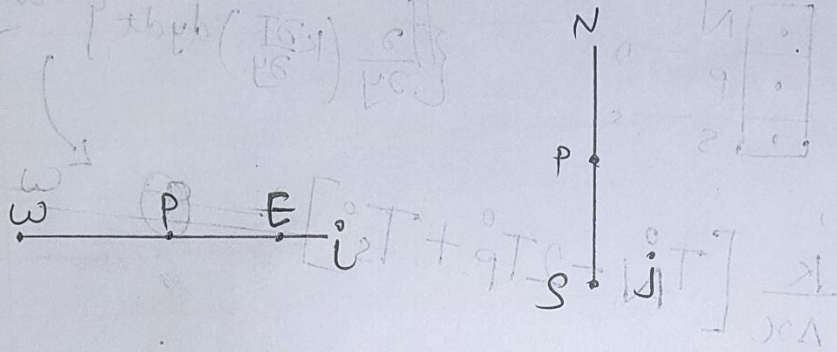
$$= \frac{\rho C_p \Delta x}{\Delta t} [T_P' - T_P^0] \quad \text{--- (6)}$$

By rewriting generalized discretised equation as,

$$\frac{k}{\Delta x} [T_E^0 + T_W^0 - 2T_P^0] + \frac{k}{\Delta x} [T_N^0 + T_S^0 - 2T_P^0] = \frac{\rho C_p \Delta x}{\Delta t} [T_P' - T_P^0]$$

$$T_P' - T_P^0 = \frac{\kappa \Delta t}{(\Delta x)^2} [T_E^0 + T_W^0 + T_N^0 + T_S^0 - 4T_P^0]$$

$$\Rightarrow T_P' = T_P^0 + \frac{\kappa \Delta t}{(\Delta x)^2} [T_E^0 + T_W^0 + T_N^0 + T_S^0 - 4T_P^0]$$



MATLAB Code: -

```
%CASE STUDY on 2-D Unsteady Conduction in Square Copper Plate.
clear all
clc

%% Defining the Variables
L = input('L = ') ; %Length of Plate
n = input('n = '); %Nodes
dx = input('dx = '); %L/(n-1); %Grid spacing in x      %Assuming that our
plate is square.
x = linspace(0,L,n); %Grid points in x
y = linspace(0,L,n); %Grid points in y
alpha = 113/1000000; %Thermal Diffusivity
dt = input('dt = '); %Time Step

%% Maximum Time to be Iterated
T_max = input('T_max = ');

%% Initial Temperature Matrix Values
T = zeros(n,n);

%% Boundary Conditions
T(:,1) = input(' Left Side Node Temp Value = ');
T(:,n) = input(' Right Side Node Temp Value = ');
T(1,:) = input(' Upper Side Node Temp Value = ');
T(n,:) = input(' Bottom Side Node Temp Value = ');

%% Y;X;Time Looping
for t = 0:T_max
    for i = 2:n-1
        for j = 2:n-1
            T_w = T(i-1,j);
            T_p = T(i,j);
            T_e = T(i+1,j);
            T_n = T(i,j-1);
            T_s = T(i,j+1);
            T(i,j) = T_p+(alpha*dt/(dx^2))*(T_e-4*T_p+T_w+T_s+T_n);
        end
    end
end

%% Plot
%plot(x,T)
%surf(x,y,T)
%contour(x,y,T)
contourf(T,1000,'edgecolor','none')
colormap jet
colorbar
xlabel('Distance in X')
ylabel('Distance in X')
zlabel('Temperature')
title('Temperature Profile of a Square Plate')
```

Output [Command Window]:-

```
L = 1
n = 21
dx = 0.05
dt = 1
T_max = 500
Left Side Node Temp Value = 800
Right Side Node Temp Value = 100
Upper Side Node Temp Value = 100
Bottom Side Node Temp Value = 100
>>
```

Results:-

I compared my Code with *Korosh Agha Mohammad Ghasemi Chemical Engineering at Shiraz University*. Though it was the comparison among Numerical Methods, I was confident enough to know my code is providing correct results. Also, my code is short compared to that of online MATLAB Code.

I used explicit method to discretise my “*Two-Dimensional Heat Conduction equation without Heat Generation*” so the results are showing lower accuracy. Instead of explicit method one can also use Central Difference Scheme or Implicit Method, etc.

And the results are provided below:

Graph:-

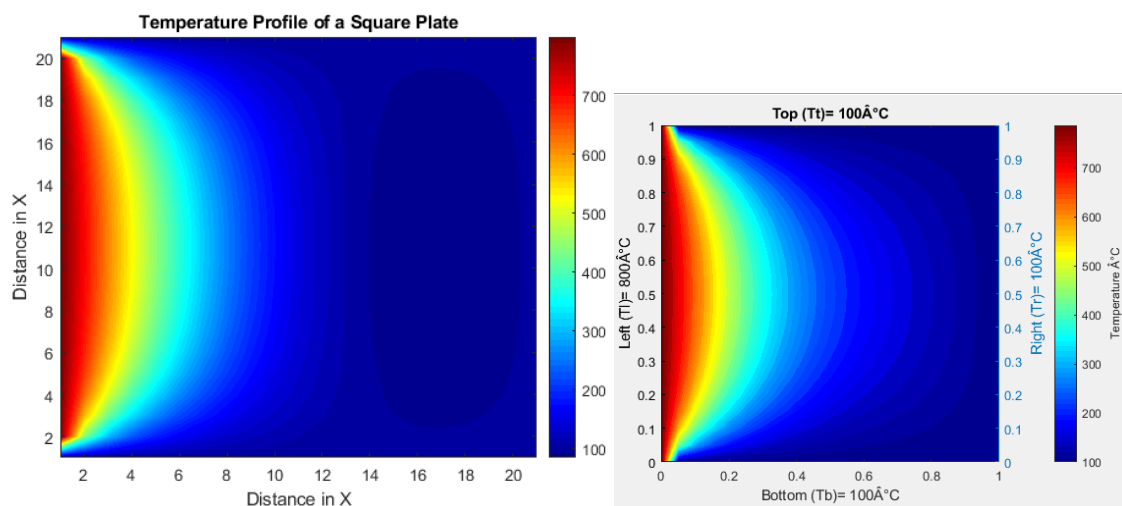


Figure 1 Student Plot

Figure 2 ONLINE MATLAB Coded Plot