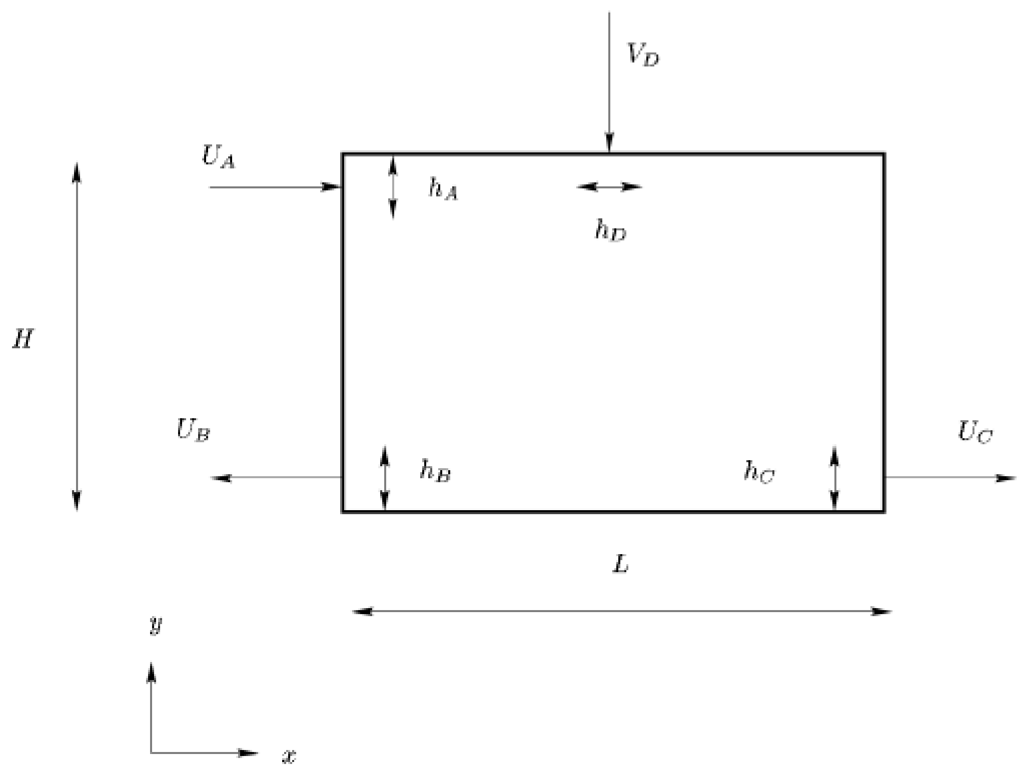
AM5630 Foundation of CFD
Assignment -3

AE25S051 MAYANK LONKAR

2D Steady Convection diffusion Equation

$$\frac{\partial}{\partial x}(\rho U T) + \frac{\partial}{\partial y}(\rho V T) = \frac{\partial}{\partial x}\left(\Gamma \frac{\partial T}{\partial x}\right) + \frac{\partial}{\partial y}\left(\Gamma \frac{\partial T}{\partial y}\right) + S$$

Domain -

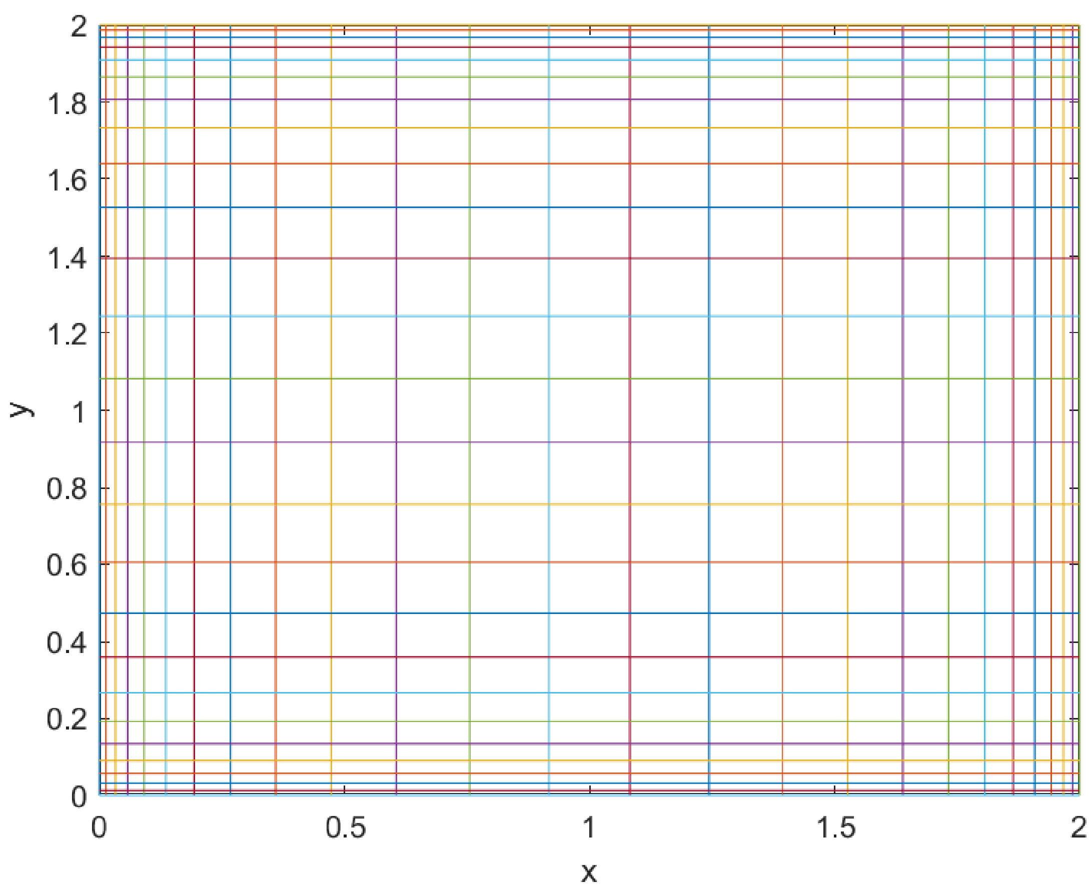


Boundary Conditions

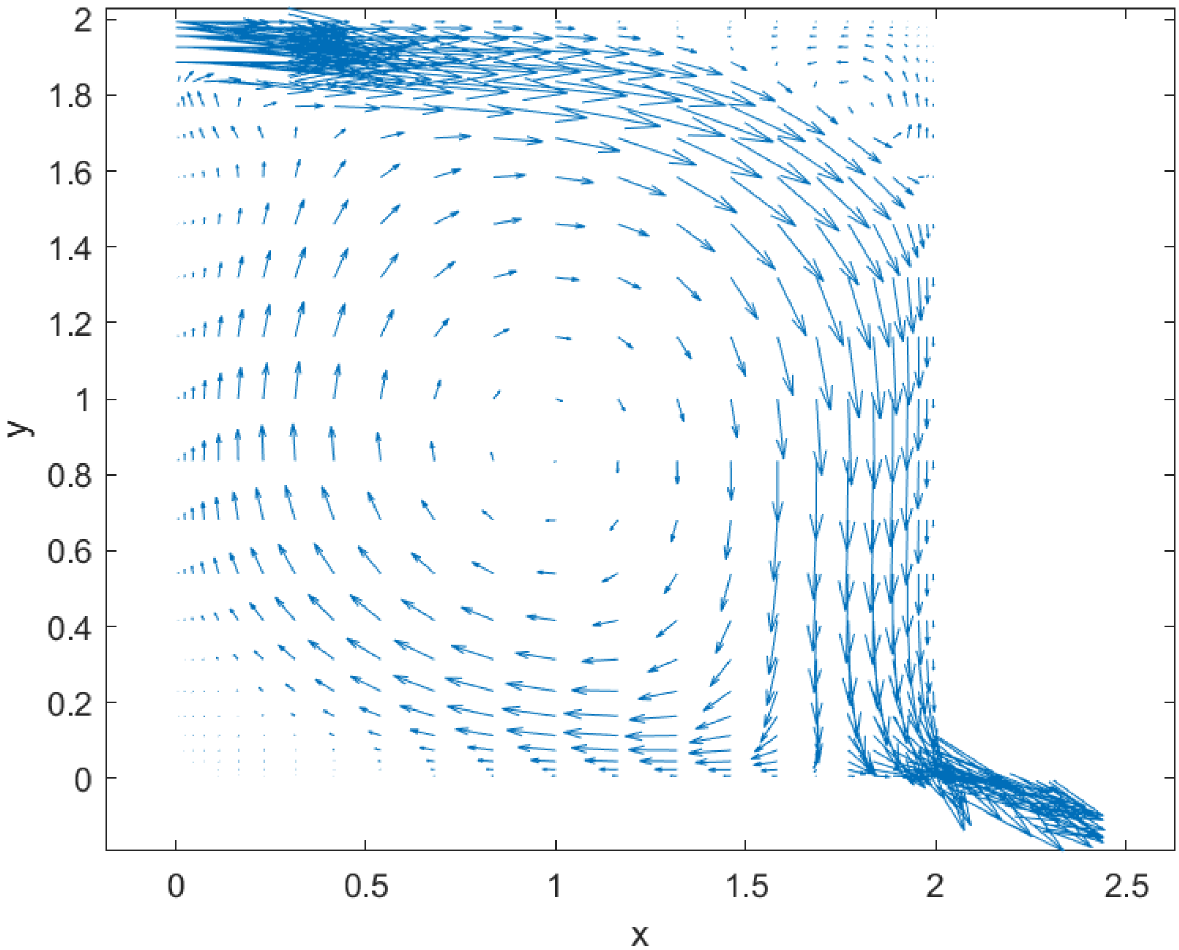
Bottom	∂T/∂η = 0
Top	∂T/∂η = 0
TA	20 C
Right (except TC)	50 C

∂T/∂η = 0 is also applied at right and left wall where Temperature condition is not mentioned.

Mesh



Velocity Field

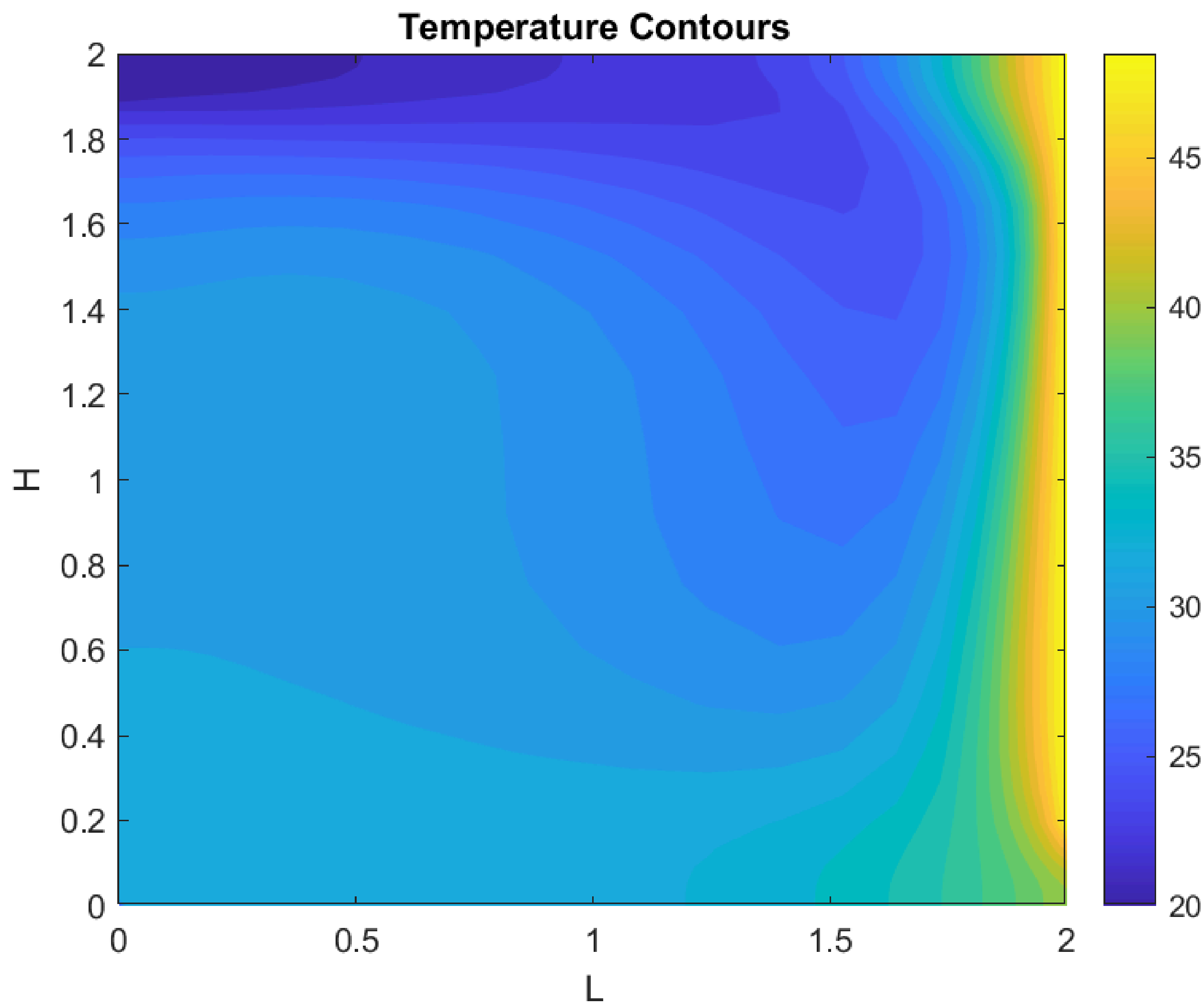


Velocity field is provided on the nodes and the boundary points (ni x nj matrix), but for the scheme we require velocity at cell faces which is found out using interpolation. From the velocity plot it can be seen that the inlet is A and the outlet is at C .

TDMA

Following is the plot obtained from TDMA, the residuals were 10e-3, TDMA proved to have faster convergence as compared to Gauss seidel.

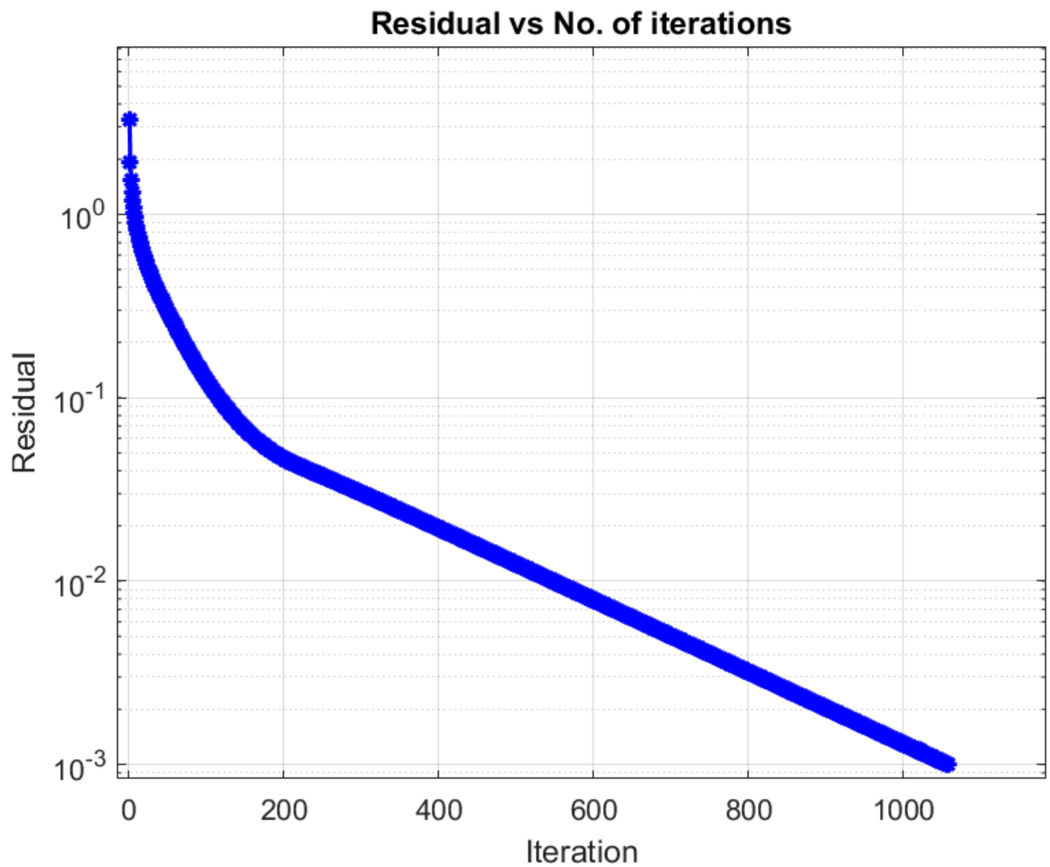
The following figure shows the Temperature contour obtained -



The right wall has a temperature of 50 C which causes the gradient of temperature near it due to the diffusion as well as the strong convective heat flow . The dominant direction of convection can be assessed from the velocity contour .

The Interior has a gradient of temperature between 20C to 45C showing heat transfer from right to left. The top region is cooler than the bottom region.
The diffusion patterns are visible from the left corner coming towards the right wall. The left wall has zero flux condition except the velocity inlet.

The residual plot shows a monotonic decay, indicating convergence for both TDMA and gauss seidel loop .

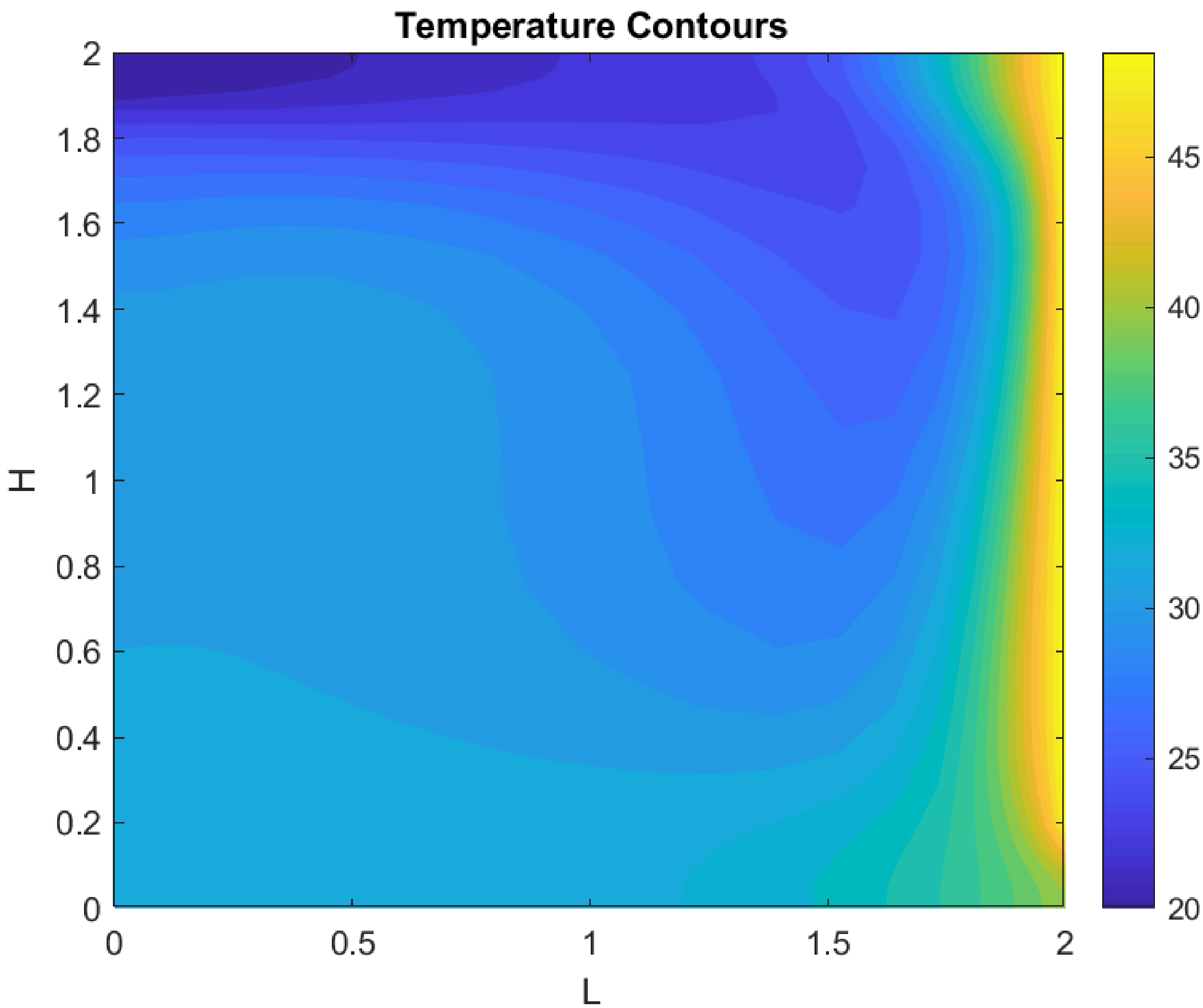


TDMA algorithm for j th row

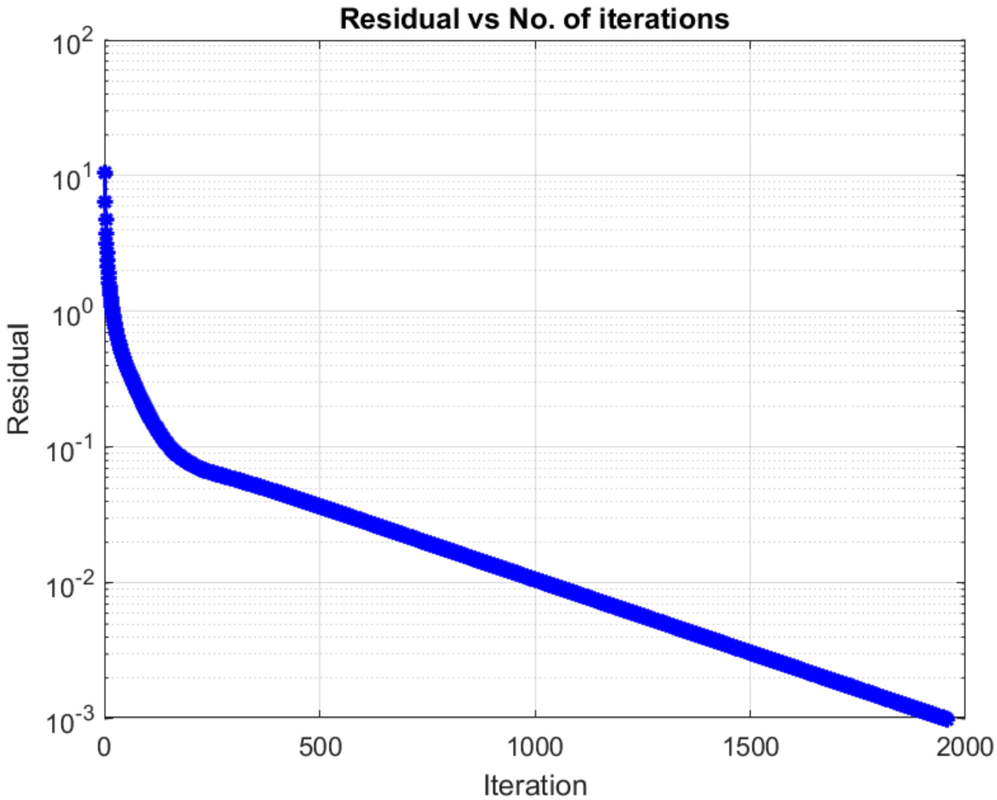
```
for k = 3:nim1
P(k) = Ae(k,j) / (Ap(k,j)-Aw(k,j)*P(k-1));
Q(k) = (Aw(k,j)*Q(k-1) + d(k) ) / ( Ap(k,j) -Aw(k,j)*P(k-1));
end
for n = nim1:-1:2
    Temp(n,j)= P(n)*Temp(n+1,j) + Q(n);
End
```

X sweep algorithm was implemented sweeping row 1 to row nim1 and it converged close to 1100 iterations for E> 10e-3

Gauss Sciedal



The solution using gauss seidel algorithm also shows a similar temperature as obtained by TDMA, however it is slower in convergence than TDMA



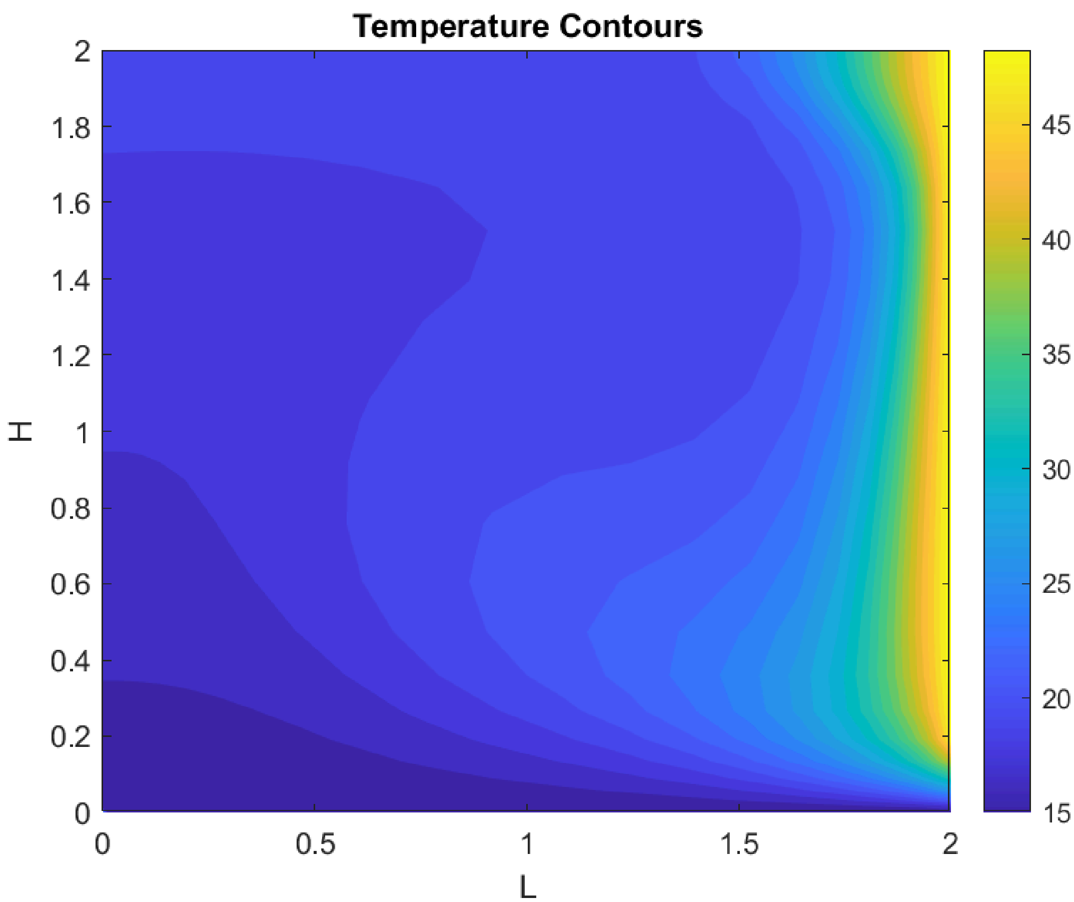
Gauss Sciedal implementation

```
    for m = 2:nim1
        for n = 2:njm1
        Temp(m,n)= (Ae(m,n)*Temp(m+1,n)+ Aw(m,n)*Temp(m-1,n) +
An(m,n)*Temp(m,n+1) + As(m,n)*Temp(m,n-1))/Ap(m,n);
        end
    end
```

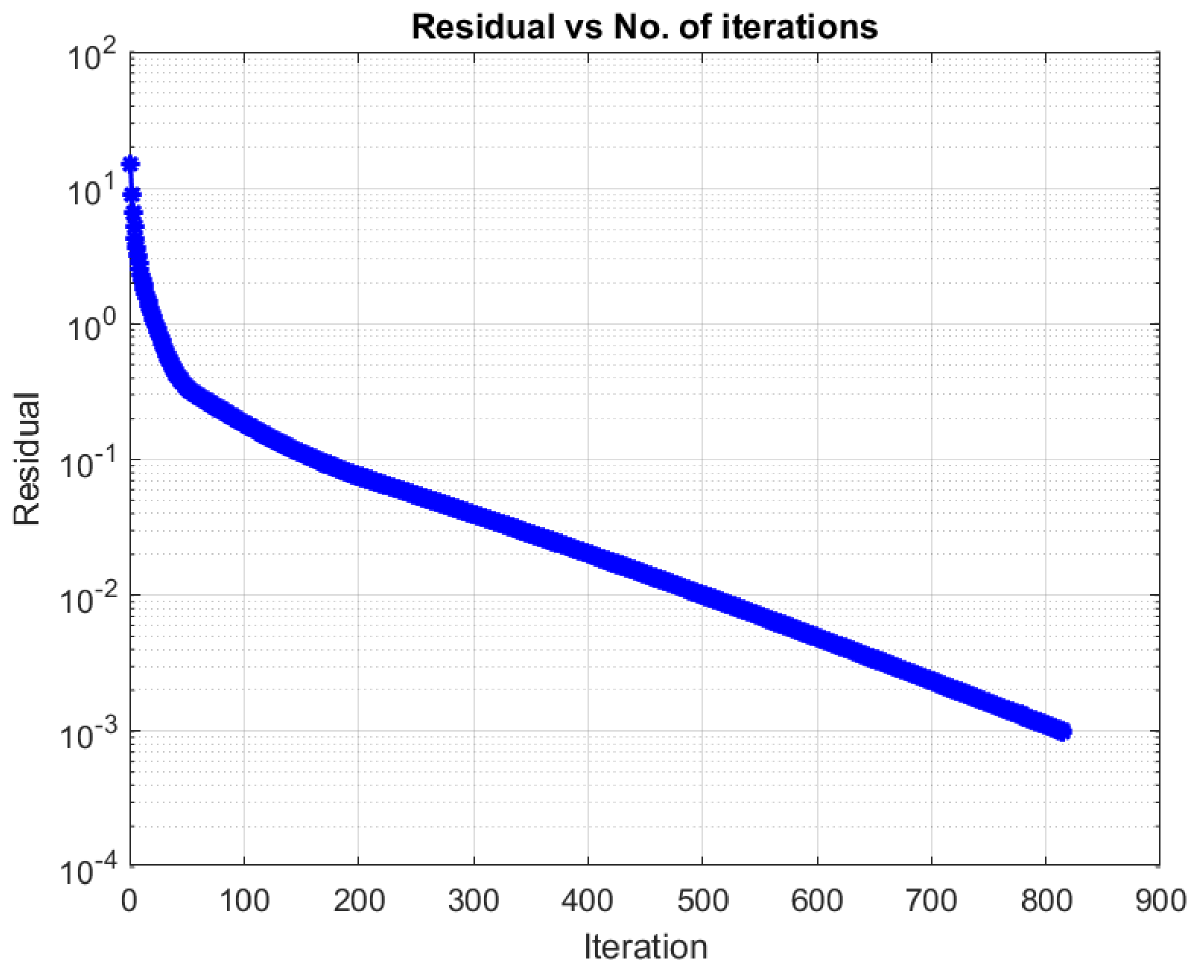
Changing Boundary condition -

Bottom boundary is changed from Neumann to Direchelt
T bottom - 15 C

Results -



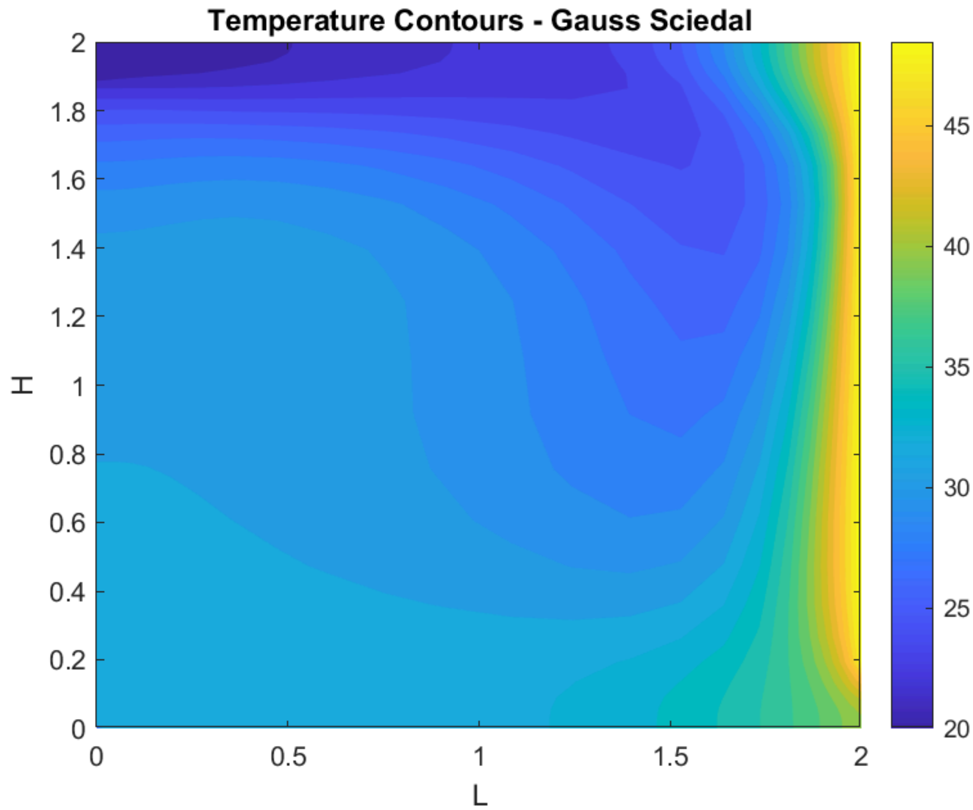
Changing the boundary condition of the bottom boundary shows change in the temperature contour. A gradient of temperature is visible normal to the bottom wall especially near the right wall. There is diffusion from the right as well as the left boundary towards the bottom wall.

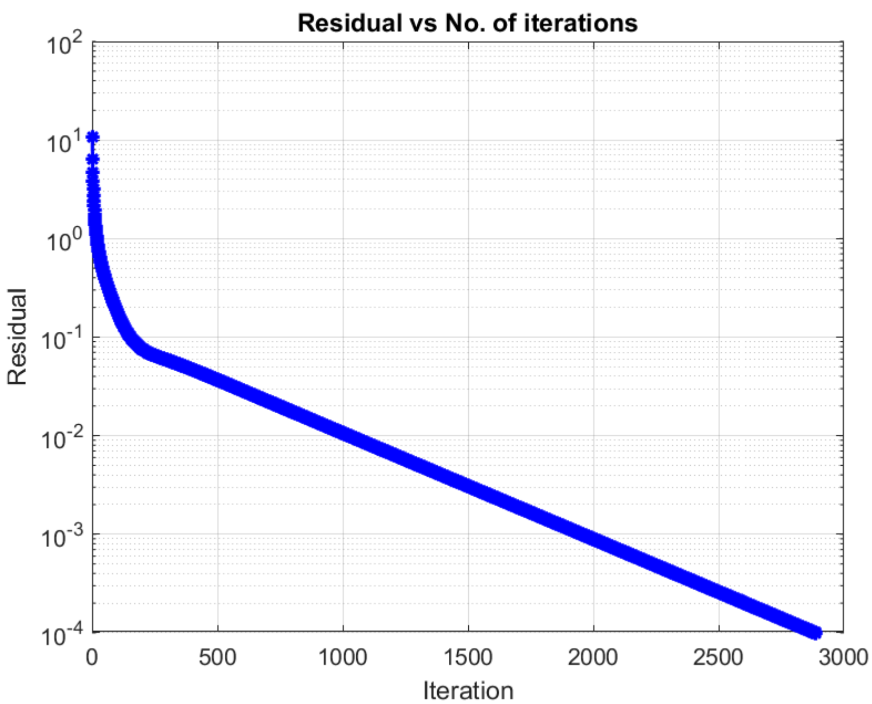


Residual vs No. of iterations
Residual
Iteration
10^{2}
10^{1}
10^{0}
10^{-1}
10^{-2}
10^{-3}
10^{-4}
0
100
200
300
400
500
600
700
800
900

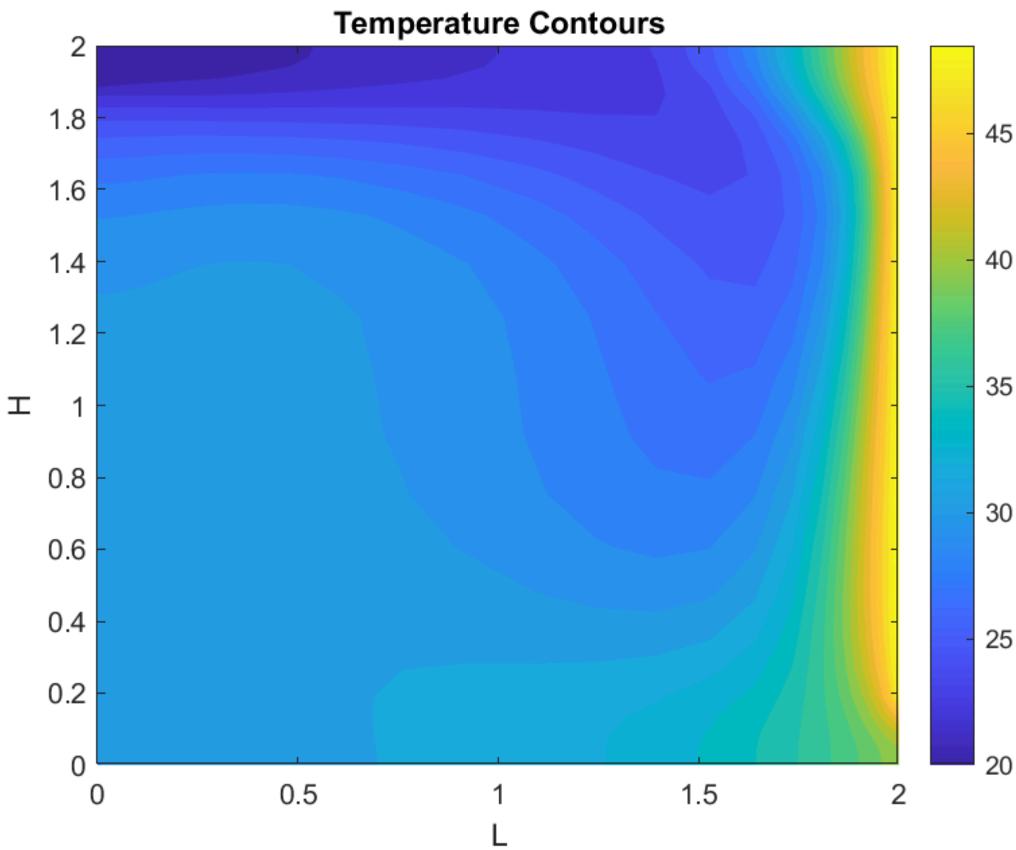
Changing Residual values :

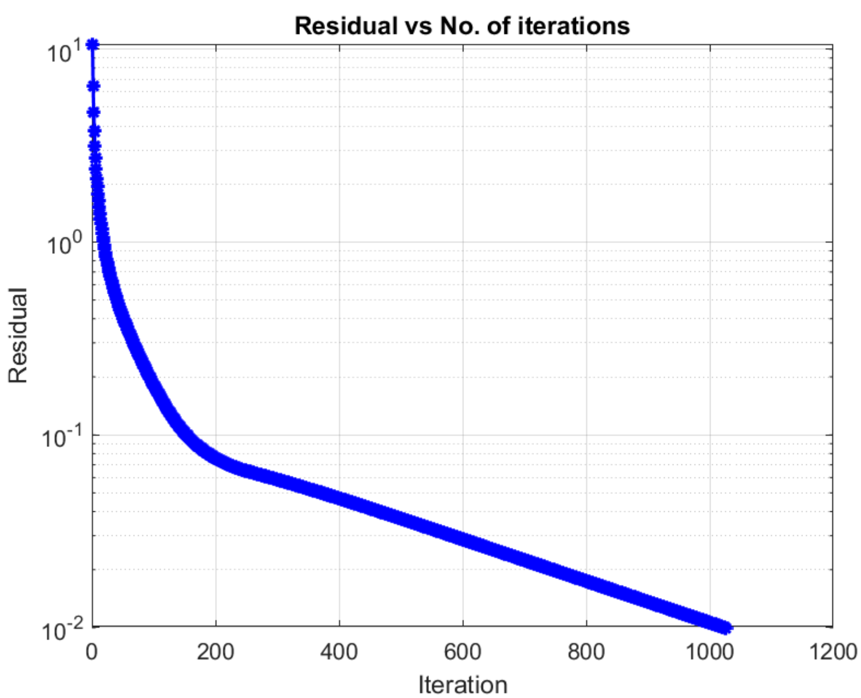
Error - 0.0001 (Gauss Sciedal)



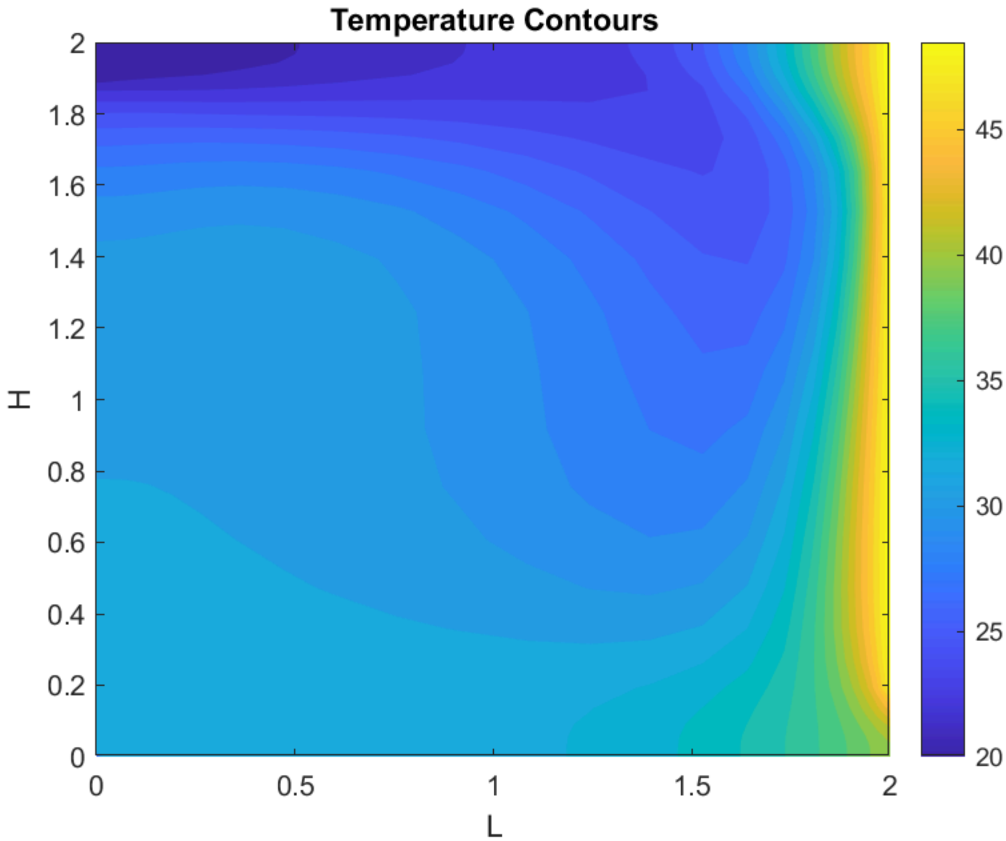


E > 0.01 - Gauss Seidel

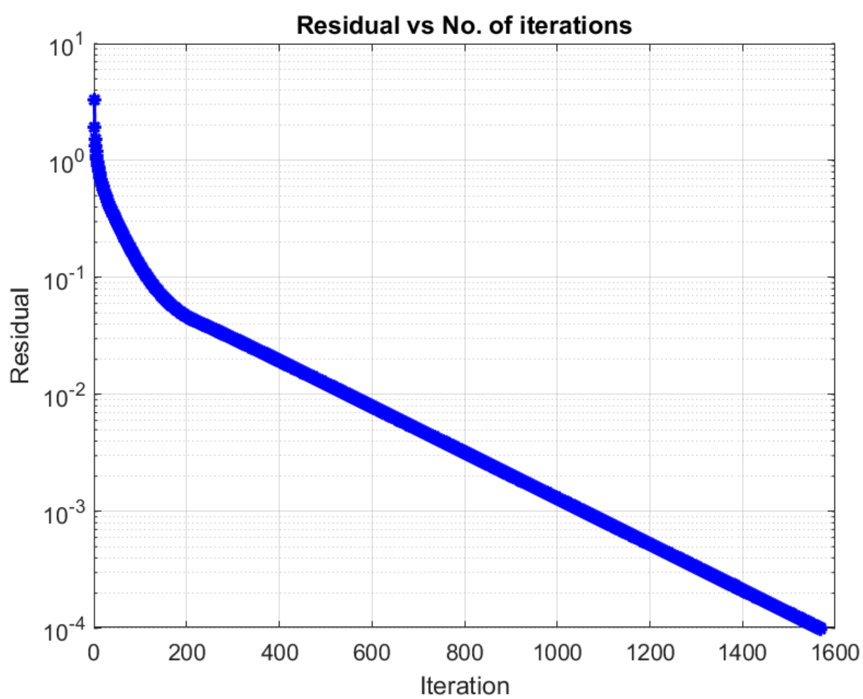




E > 0.0001 TDMA

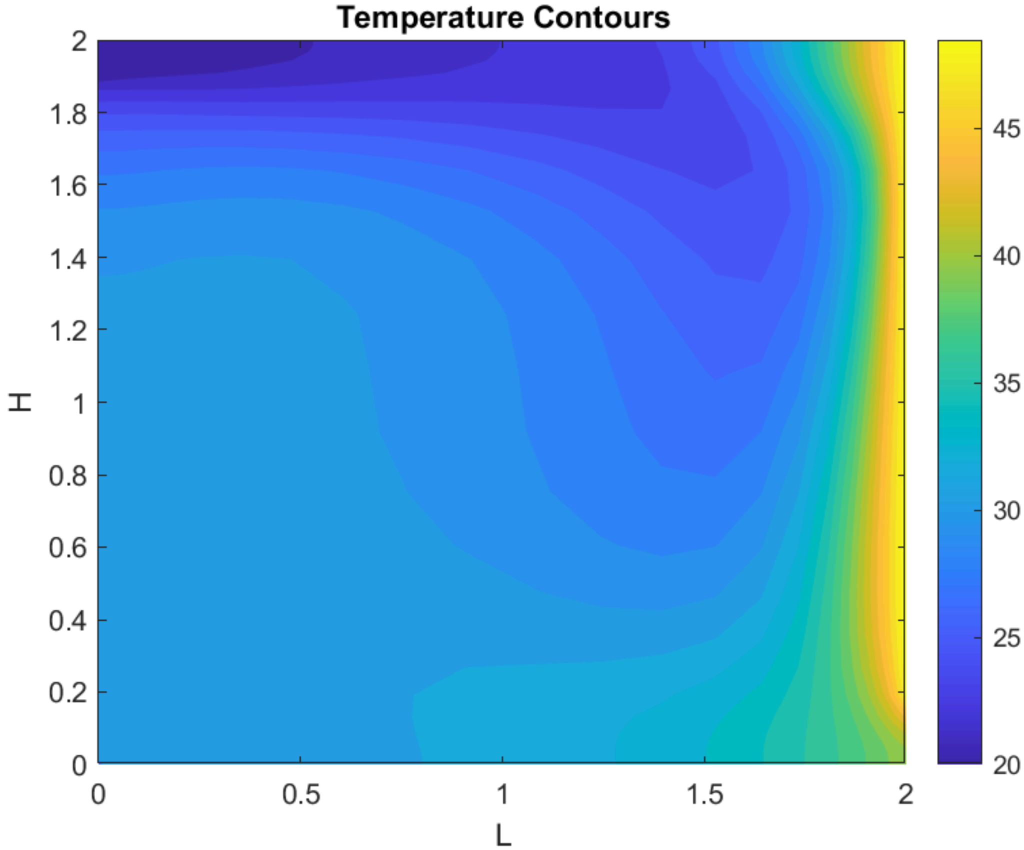


Temperature Contours
H
L
0
0.2
0.4
0.6
0.8
1
1.2
1.4
1.6
1.8
2
0
0.5
1
1.5
2
20
25
30
35
40
45

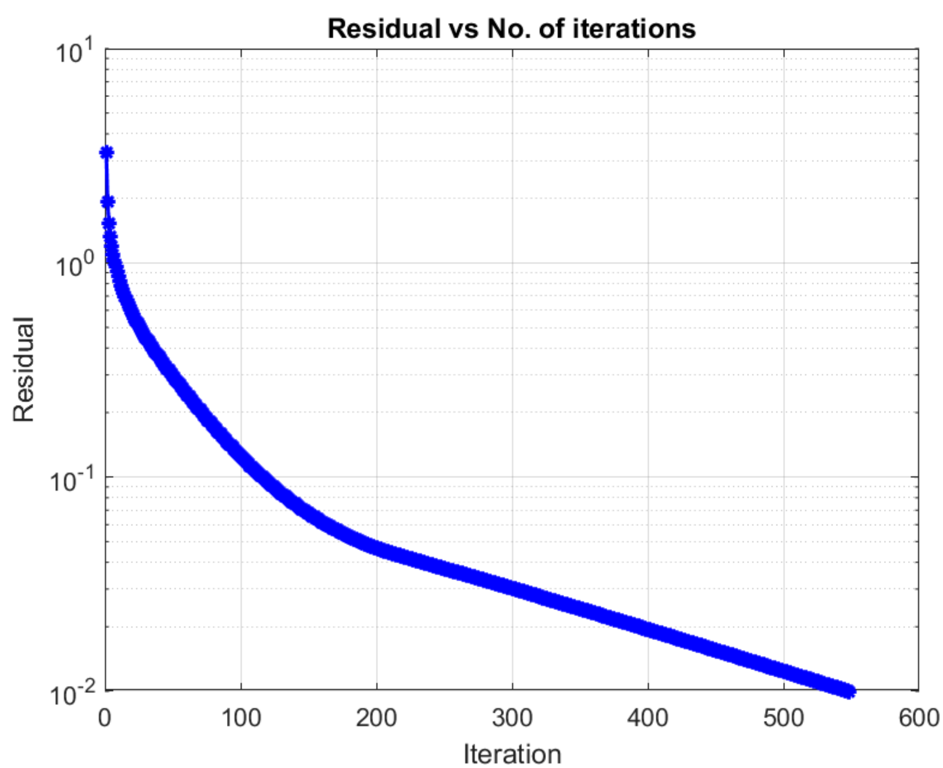


Residual vs No. of iterations
Residual
Iteration
10^{1}
10^{0}
10^{-1}
10^{-2}
10^{-3}
10^{-4}
0
200
400
600
800
1000
1200
1400
1600

E > 0.01 TDMA



Temperature Contours
H
L
0
0.2
0.4
0.6
0.8
1
1.2
1.4
1.6
1.8
2
0.5
1.5
20
25
30
35
40
45



Residual vs No. of iterations
Residual
Iteration
10^{1}
10^{0}
10^{-1}
10^{-2}
0
100
200
300
400
500
600

Temperature value at (15,15) grid point can be used to compare the if temperature varies with residual change

TDMA 0.01	27.8785 C
TDMA 0.001	28.1739 C
TDMA 0.0001	28.2033 C

This shows that changing the residual values can change the temperature, and thus temperature is sensitive to convergence.

Checking continuity

Calculation for top and bottom -

```
Qb = sum ((1/50)*(Temp(2:end-1,1)-Temp(2:end-1,2))/d_y(1));
Qt = sum ((1/50)*(Temp(:,nj)-Temp(:,nj-1))/d_y(end));
```

Calculation for right and left boundaries

Both convection and diffusion flux is calculated on the inlet and outlet at the right and left sides .

```
for x = 22:nj-1
Qlcond= (1/50)* (Temp(2,x) - Temp(1,x)) / d_x(1);
Qlconv = (0.136 * rho *d_y(x-1)* UA * Temp(1,x));
Ql = Ql + Qlconv + Qlcond;
end
for x = 2:6

Qrcond= (1/50) * (Temp(ni,x) - Temp(nim1,x)) / d_x(end);
Qrconv = (0.136 * d_y(x-1) *rho * UC * Temp(ni,x));
Qr =Qr+ Qrconv + Qrcond;
End
```

The other points on the right and left wall will have zero flux due to its boundary condition. dT / dx = 0

Flux results -

TDMA -
Top - 0
Bottom - 0

Right inlet - 0.6487
Left outlet - 0.6891

Net flux = 0.0404

Gauss Seidel -

Bottom heat flux = 0
Top heat flux = 0

Right heat flux = 0.64751
Left heat flux = 0.67312
Net Flux = 0.0256

Thus approximately the net flux is zero, inlet flux is equal to the outlet flux and continuity is maintained.