PROBLEM STATEMENT

A 2 D domain of size $2\pi \times 2\pi$ is given. The flow inside the domain is steady and is given by the following non dimensional velocity components in x & y

$$u = \sin(x) \cos(y)$$
$$v = -\cos(x) \sin(y)$$

Initial condition: The non dimensional temperature is given by
$$T(x, y, t=0) = 0$$

Boundary condition: At time $t = 0$, non dimensional temperatures of both left and right walls are increased to 1.0 while the other two walls remain at $T=0$. The boundary conditions stay steady subsequently.

Governing equation: The temperature evolution in the domain is given by the advection-diffusion equation given by

$$\frac{\partial T}{\partial t} = -u \frac{\partial T}{\partial x} - v \frac{\partial T}{\partial y} + \frac{1}{Pe}\left(\frac{\partial^2 T}{\partial x^2} + \frac{\partial^2 T}{\partial y^2}\right)$$

where Pe is the Peclet number. $Pe = 1$.

Objective: Apply finite difference method using a ~~f~~ uniform grid to obtain –

* Temperature field at $t = 2.5, 5, 7.5, 10$ s
* Behavior of temperature profile at $x = \frac{3\pi}{4}$, $y = \frac{\pi}{4}$
* Grid refinement study.

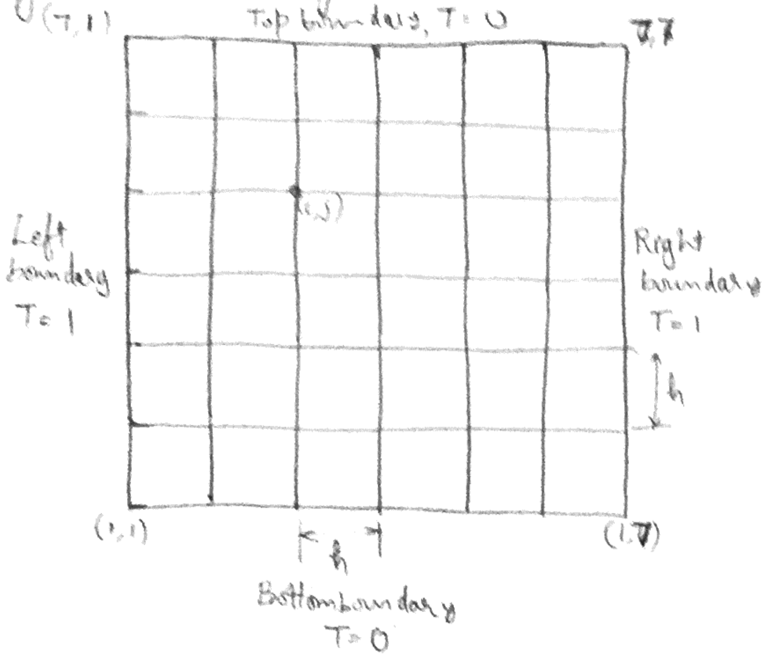
Numerical schemes used :

* Convection terms – Central difference scheme – 2^{nd} order
* Diffusion term – Central difference scheme – 2^{nd} order.
* $\frac{\partial T}{\partial t}$ term – AB2 method – 2^{nd} order
  – RK4 method – 4^{th} order.

Assumptions :

1. The flow inside the domain is viscous, incompressible and in steady state
2. The Peclet number is taken as 1.
3. The functions are continuous and differentiable in the given domain and hence Taylor series approximations are valid.
4. The discretization is carried out on uniform grid spacing with finite difference techniques.
5. Heat transfer modes ~~used~~ present are convection and conduction
6. All the nodal values of temperature on the left and right walls including the corner points have an initial value of 1 and stay the same independent of time.
7. All the nodal values of temperature on the upper & lower walls excluding the corner points have an initial value of 0 and stay the same independent of time.
8) Since the exact solution is unavailable, the highest grid resolution (80x80) that is available is assumed to be the exact solution for error calculations.

Grid Schematic for n=6



Equally spaced grid

i : 1 → 7

j : 1 → 7

Discretization

$$\frac{\partial T}{\partial t} = -u\frac{\partial T}{\partial x} - v\frac{\partial T}{\partial y} + \frac{1}{Pe}\left(\frac{\partial^2 T}{\partial x^2} + \frac{\partial^2 T}{\partial y^2}\right)$$

* Convection term

→ $u = \sin(x)\cdot\cos(y)$, $v = -\cos(x)\sin(y)$ } steady

→ $$\left.\frac{\partial T}{\partial x}\right|_{i,j} = \left[\frac{T_{i+1} - T_{i-1}}{2h}\right]_j$$

→ $$\left.\frac{\partial T}{\partial y}\right|_{i,j} = \left[\frac{T_{j+1} - T_{j-1}}{2h}\right]_i$$

$T_{i,j} = 1$ if $i = 1$ or $i = N+1$

$= 0$ if $j = 1$ or $j = N+1$ except at corner points

* Diffusion term

→ $$\left.\frac{\partial^2 T}{\partial x^2}\right|_{i,j} = \left[\frac{T_{i+1} - 2T_i + T_{i-1}}{h^2}\right]_j$$

→ $$\left.\frac{\partial^2 T}{\partial y^2}\right|_{ij} = \left[\frac{T_{j+1} - 2T_j + T_{j-1}}{h^2}\right]$$

Time integration

* Adam Bashforth 2nd order. (Explicit method)

$$T^{n+1} = T^{n} = \frac{1}{2}\,\Delta t\left[3\,\frac{\partial T}{\partial t}^{n} - \frac{\partial T}{\partial t}^{n-1}\right]$$

T^{n+1} → Temperature at next time step

T^{n} → Temperature at present time step

$3\frac{\partial T}{\partial t}^{n}$ → At current time step

$\frac{\partial T}{\partial t}^{n-1}$ → At previous time step

AB2 is multistep method which requires past value of $\frac{\partial T}{\partial t}$ to predict future value. Hence it does not work at initial value of $t = 0$. So in order to start the loop, we use forward Euler method.

$$T^{1} = T^{0} + \left.\frac{\partial T}{\partial t}\right|_{t=0}^{\Delta t} \cdot \Delta t$$

* Runge Kutta IV order. (Explicit method)

$$T^{n+1} = T^{n} + \frac{\Delta t}{6}\left[F_1 + 2F_2 + 2F_3 + F_4\right]$$

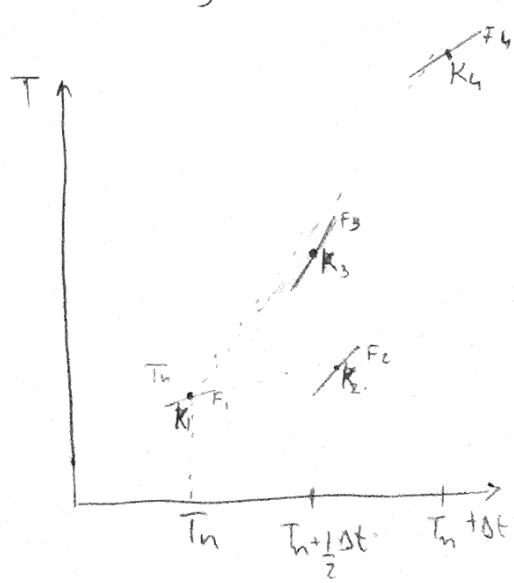
where (F → slope)

$$F_1 = f(K_1, t^n)$$

$$F_2 = f(K_2, t^n + \tfrac{1}{2}\Delta t)$$

$$F_3 = f(K_3, t^n + \tfrac{1}{2}\Delta t)$$

$$F_4 = f(K_4, t^n + \Delta t)$$



and (K → Predicted value.)

$$K_1 = T^n$$

$$K_2 = T^n + \tfrac{1}{2}\Delta t \cdot F_1$$

$$K_3 = T^n + \tfrac{1}{2}\Delta t \cdot F_2$$

$$K_4 = T^n + \Delta t \cdot F_3$$

Methodology - Algorithm used for implementation of numerical scheme.

The code for implementing the numerical scheme was written in MATLAB. The code can be broadly split into three parts as discussed below :

a) Numerical computation of temperature.

Each of the grid sizes are used to initialize the necessary temperature and dT/dt values of the corresponding sized matrices. Further, the spatial loops are used to select each T_{ij} and calculate the corresponding $\frac{\partial T}{\partial x}$, $\frac{\partial T}{\partial y}$, $\frac{\partial^2 T}{\partial y^2}$, $\frac{\partial^2 T}{\partial x^2}$ at each of the interior nodes. The values of T at boundaries stay steady and hence evaluation of these at boundaries are not necessary. The $\frac{\partial T}{\partial t}$ value can be evaluated by substituting above obtained values in the given governing equation. Once $\frac{\partial T}{\partial t}$ value is obtained two schemes are used to evaluate the T value at next time step..

* Adam Bashforth method - 2nd order

The scheme is not self starting. Hence at $t=0$, forward Euler method is used to find temperature at next time step. For further time steps, matrices are used to save value of $\frac{\partial T}{\partial t}$ at present time step and previous time step. These values along with present value of temperature are used to evaluate T at next time step for the entire grid

* RK 4 method

This scheme is self starting and hence does not require other schemes at start up. As explained in the discretization, T^n and $\left.\frac{\partial T}{\partial t}\right|^n$ are used to evaluate F_1, F_2, F_3, F_4, and K_1, K_2, K_3, K_4. Applying RK4 formula, this gives the values of T^{n+1} on the entire grid for next time step.

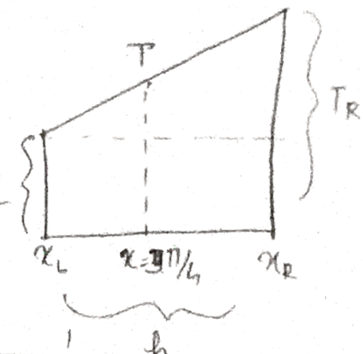
The matrix of temperature values are used to plot the temperature distribution at four time values $t = 2.5, 5, 7.5, 10$.

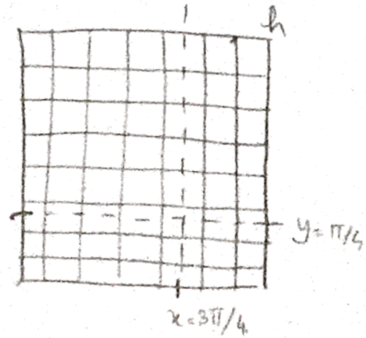
b) Interpolation of temperature along $x = \frac{3\pi}{4}$ and $y = \frac{\pi}{4}$.

The temperature distribution along the line was obtained for the highest grid refinement ~~along these~~ at $t = 2.5, 5, 7.5, 10$. Since the given x and y value do not necessarily fall on the exact node, it is necessary to do linear interpolation. First the indices of the upper and lower (or right & left node) nodes are obtained by calculation. Then the temperature at these nodes are used to interpolate the value of T

$$T = T_L + (T_R - T_L)\left[\left(\frac{3\pi}{4} - x_L\right)/h\right]$$

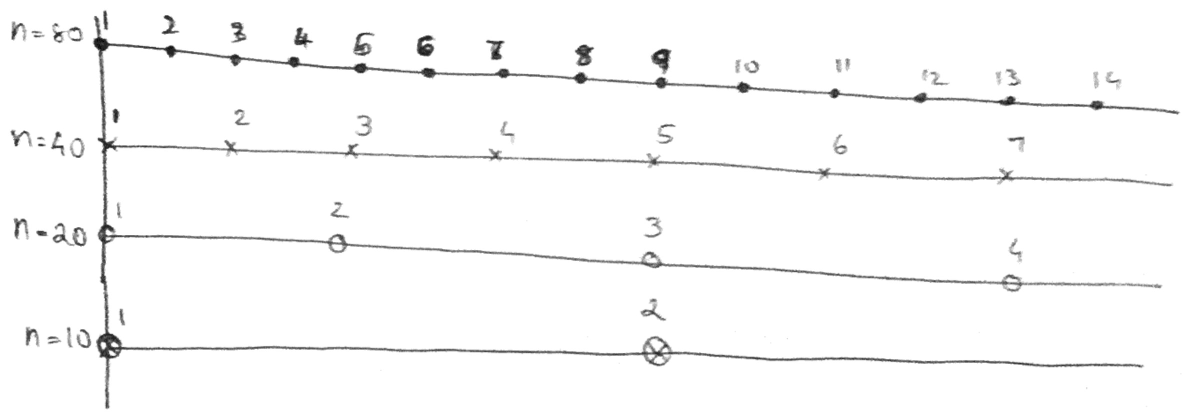
$$T = T_L + (T_R - T_L)\left[\left(\frac{\pi}{4} - y_L\right)/h\right]$$





c) Error Evaluation:

Error evaluation of the schemes is carried out for each of the grid refinements at the final time of $t = 10\,s$. For this, the value of ~~spatial dis~~ maximum resolution grid is taken as exact value and the nodes of individual meshes are matched to the nodes on high resolution mesh in order to find the corresponding error matrices.



After evaluating the error matrix, L_2 norm of error is given by

$$E_2 = \sqrt{\frac{\sum_i \sum_j e_{ij}^2}{\text{Total elements}}} = \sqrt{\frac{(\text{norm}(e))^2}{N^2}}$$

$(\text{norm}(e))^2 \longrightarrow$ matlab fn that gives $\sqrt{\sum_i \sum_j e_{ij}^2}$

$$= \underline{\underline{\frac{\text{norm}(e)}{N}}}$$