Indian Institute of Technology, Madras

Foundations of Computational Fluid Dynamics (AM5650)

Assignment 2

Charukhesh B.R (AE22B028)
Bhadra S (AE22B010)

Contents

List of Figures

1 Theory: Steady-2D Diffusion

$$0 = \frac{\partial}{\partial x}\left(k\frac{\partial T}{\partial x}\right) + \frac{\partial}{\partial y}\left(k\frac{\partial T}{\partial y}\right) + S \tag{1}$$

Integrating over the control volume and simplifying we get,

$$\left(\frac{\Gamma_w A_w}{\delta x_{WP}} + \frac{\Gamma_e A_e}{\delta x_{PE}} + \frac{\Gamma_s A_s}{\delta y_{SP}} + \frac{\Gamma_n A_n}{\delta y_{PN}} - S_p\right) T_P = \left(\frac{\Gamma_w A_w}{\delta x_{WP}}\right) T_W + \left(\frac{\Gamma_e A_e}{\delta x_{PE}}\right) T_E + \left(\frac{\Gamma_s A_s}{\delta y_{SP}}\right) T_S + \left(\frac{\Gamma_n A_n}{\delta y_{PN}}\right) T_N + S_u$$

Source term is represented in linearised form as:

$$\overline{S}\Delta V = S_u + S_p\phi_P,$$

General discretised equation for interior nodes is as follows:

$$a_P T_P = a_W T_W + a_E T_E + a_S T_S + a_N T_N + S_u \tag{2}$$

a_W	a_E	a_S	a_N	a_P
$\frac{\Gamma_w A_w}{\delta x_{WP}}$	$\frac{\Gamma_e A_e}{\delta x_{PE}}$	$\frac{\Gamma_s A_s}{\delta y_{SP}}$	$\frac{\Gamma_n A_n}{\delta y_{PN}}$	$a_W + a_E + a_S + a_N - S_p$

where the face areas in a two-dimensional case are $A_w = A_e = \Delta y$ & $A_s = A_n = \Delta x$

2 Gauss-Seidel Iterative Solver

We have implemented Gauss-Seidel Solver in MATLAB to **Case-4** of the assignment. The boundary conditions applied to the computational domain given in (1) are as follows:

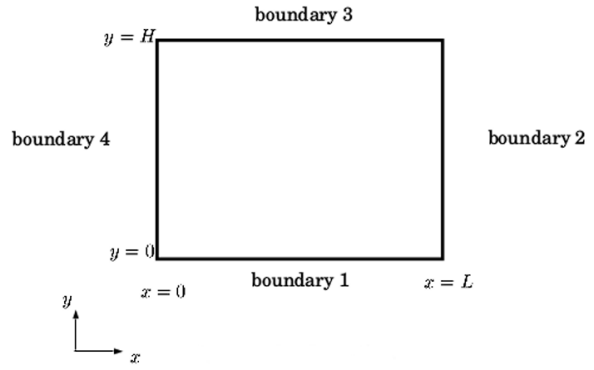


Figure 1: Computational Domain

T_1	T_2	T_3	T_4	S	k
15	$5(1-\frac{y}{H}) + 15\sin(\frac{\pi y}{H})$	10	$q = -5000$	-1.5	$16(1+\frac{y}{H})$

Table 1: Boundary Conditions for Case-4

3 Results & Insights

The solution of the steady 2D diffusion for a 10 × 10 equidistant mesh is given in figure (2):

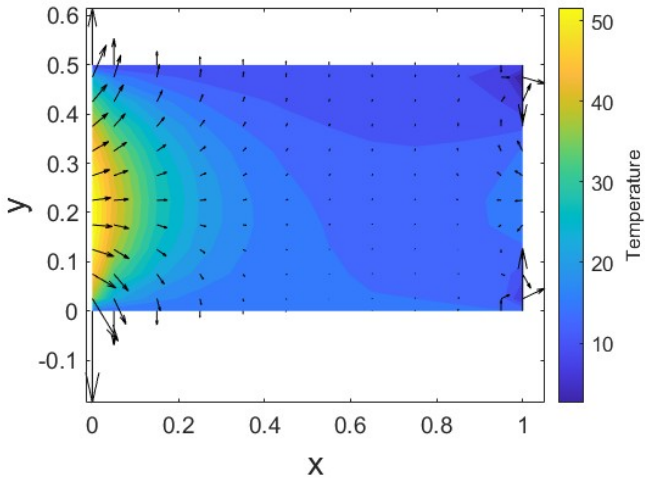


Figure 2: Vector plot of heat flux with temperature variation contour plot for a 10 ×10 mesh

Comparing the solutions for different equidistant meshes (**10x10, 20x20, 40x40, 80x80**). The methodology for comparison is as follows:

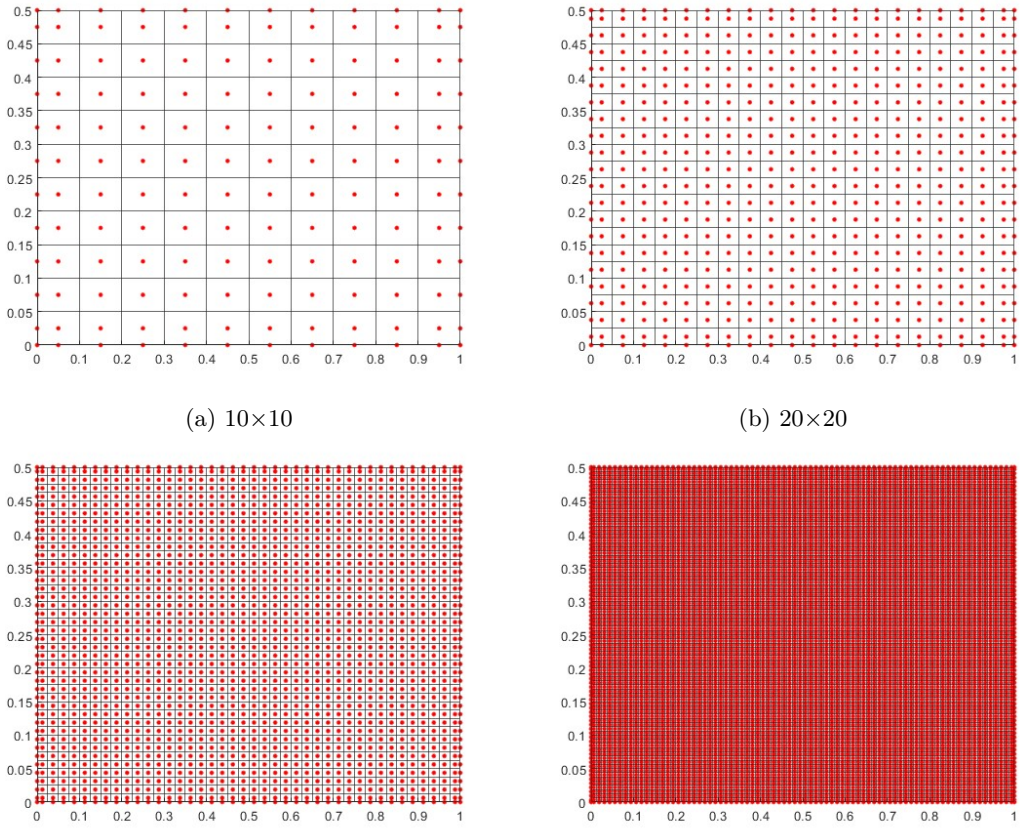
- The average of temperatures of the left section, right section and middle section is calculated for the different meshes mentioned above.
- The standard deviation of the average temperature data of the left section, right section and middle section is also calculated.
- The standard deviations for each of the sides are then compared.

3.1 Proving mesh independence

The solutions to the standard deviations of left section, right section and middle section for the different meshes **10x10, 20x20, 40x40, 80x80** is given below.

- Left section: **1.20**; Essentially, average Temperature is $(25.40 \pm 1.20$ K)
- Middle section: **0.06**; Essentially, average Temperature is $(13.90 \pm 0.06$ K)
- Right section: **0.10**; Essentially, average Temperature is $(12.91 \pm 0.10$ K)

As we can see, the standard deviation of all the three sections are minor (<1.2K). The variation in middle section is much lesser than left and right section. This is because of the enforced boundary condition being Neumann on left (heat source with an incoming flux) and Dirichlet with temperature variation in y-direction on the right, inducing more variation as compared to middle of the domain.



(a) 10×10

(b) 20×20

(c) 40×40

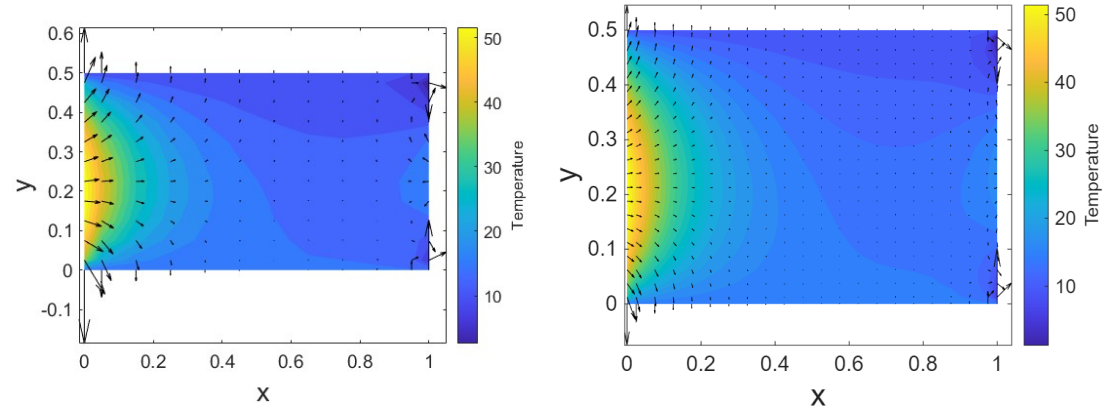
(d) 80×80

Figure 3: Equidistant meshes with different cell counts

The heat flux plots for each of the cases in figure (3) is given in figure (4).

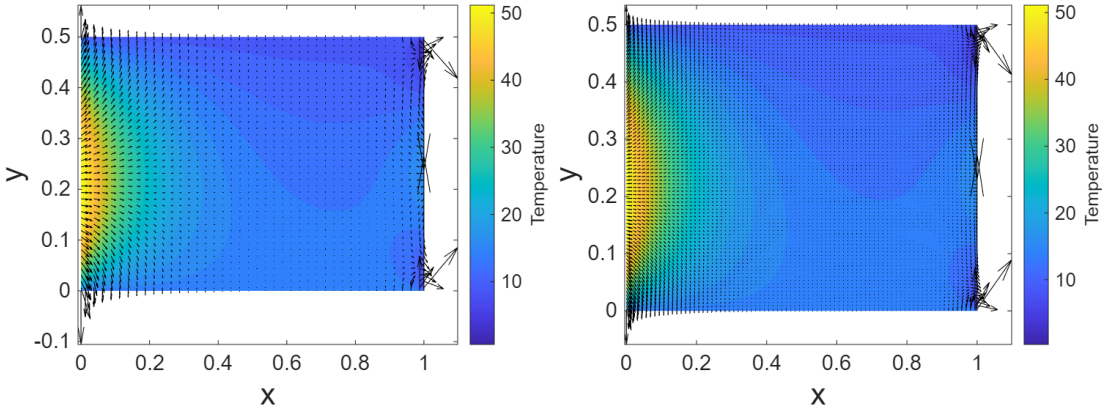
We can observe that the vector plots look similar even though we are increasing the no.of cells to get more accurate solutions. The interior diffusion dominates and the solution is not sensitive to small geometric discretization changes except very close to boundaries where BCs enforce strong gradients.

Also, maximum and minimum temperatures, as well as steep gradients near boundaries, remain nearly unchanged, showing that the finer mesh only smooths the solution without revealing new features. Differences in solution between the grids are very small, indicating that discretization errors are already below a practical level, so the coarse grid can be used safely, saving computation time. Therefore, the solution becomes independent of the mesh you choose in this case.



(a) 10×10 equidistant mesh

(b) 20×20 equidistant mesh



(c) 40×40 equidistant mesh

(d) 80×80 equidistant mesh

Figure 4: Vector plot of heat flux and temperature variation contour plot for all 4 cases

3.2 Effect of Mesh Stretching and Refinement in High-Gradient Regions

We expect large gradients near all four boundaries, especially in the left and right boundary because of the enforced boundary condition causing a huge variation in 'x' and 'y' as observed from the above heat flux plots. So, we refine the mesh (more in the left and right as compared to top and bottom) near the boundaries as follows:

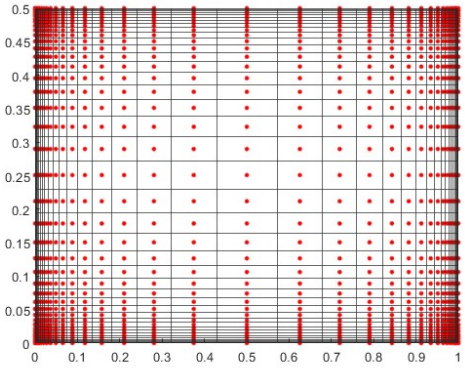
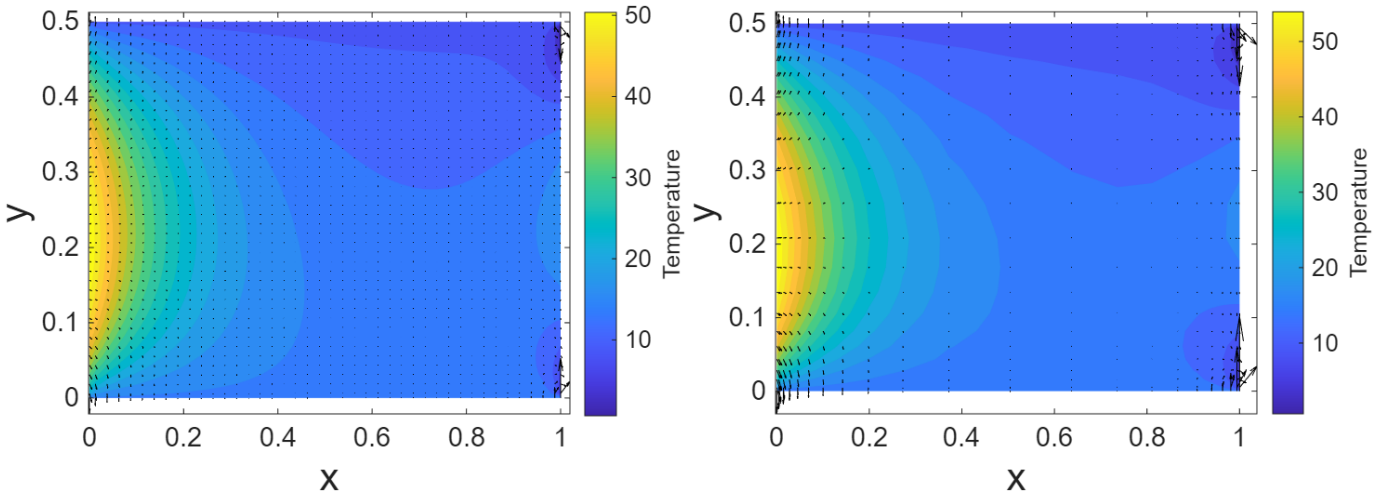


Figure 5: Refined 40×40 mesh with a stretching factor of 25% and 15 % in the x and y directions respectively

As the mesh is more refined near the boundaries, we expect the solutions to be closer to the true solution. The outcome is given in figure (6):



(a) 40×40 equidistant mesh

(b) 40×40 refined mesh

Figure 6: Comparison of vector plot of heat flux between an equidistant and a refined mesh

As we can see in figure (6), the left and right boundary's contour has become more pronounced because of the refinement. This will be closer to the actual solution as compared to the equidistant case.

Boundary regions where Neumann/Dirichlet conditions and 'k' which varies with y interact gives rise to larger local gradients. Refining the mesh near these boundaries allows the cell-wise temperature gradient (and hence flux) to be resolved more accurately, which is why boundary contours become more pronounced on the refined mesh, the refined mesh captures sharper transitions that the coarse cells average out.

Stretching the mesh near the walls helps capture sharp temperature changes better, but too much stretching makes very long, thin cells, which increases error and slows convergence. We have kept the stretching moderate, so that we get good accuracy near the boundary without losing stability.

3.3 Convergence of the Solver

We plot Residual Error ($\frac{R}{F}$) vs no.of iterations to see how the solver progressively improves it's solution. Here,

$$R = \sum_{all\ cells} |a_P T_P - (a_W T_W + a_E T_E + a_S T_S + a_N T_N + S_u)|$$

$$F = \bar{S} \times L \times H$$

We choose an error tolerance (ϵ) to decide whether the solver has converged or not. Therefore, the equation for convergence criteria is as follows:

$$\boxed{\frac{R}{F} \leq \epsilon}$$

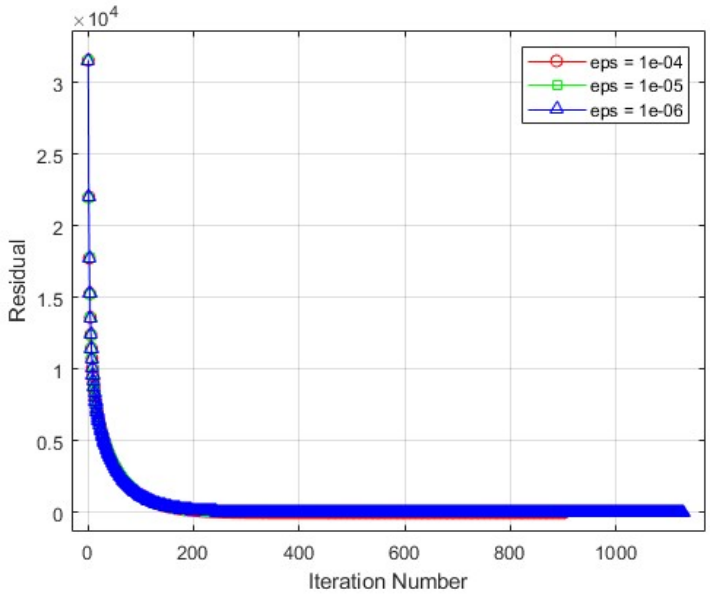


Figure 7: Residual Error vs Iteration Number in a 40×40 equidistant mesh

Error Tolerance (ϵ)	No.of iterations	L2 Norm Error w.r.t $\epsilon = 1e^{-6}$
$1e^{-4}$	902	$3.90\times e^{-6}$
$1e^{-5}$	1015	$3.56\times e^{-7}$
$1e^{-6}$	1128	0

Table 2: Error Tolerance vs No.of iterations in a 40×40 equidistant mesh

Lower the error tolerance, we become more and more sure about the final solution's convergence. As we can observe from the table (2), as ϵ decreases, no.of iterations required to converge increases as expected. To check the impact in the solution, we calculate L2 norm error of all the solutions w.r.t the solution from lowest tolerance. These very small differences indicate that even $\epsilon = 1e^{-4}$ produces a solution very close to the solution with $\epsilon = 1e^{-6}$. Therefore,

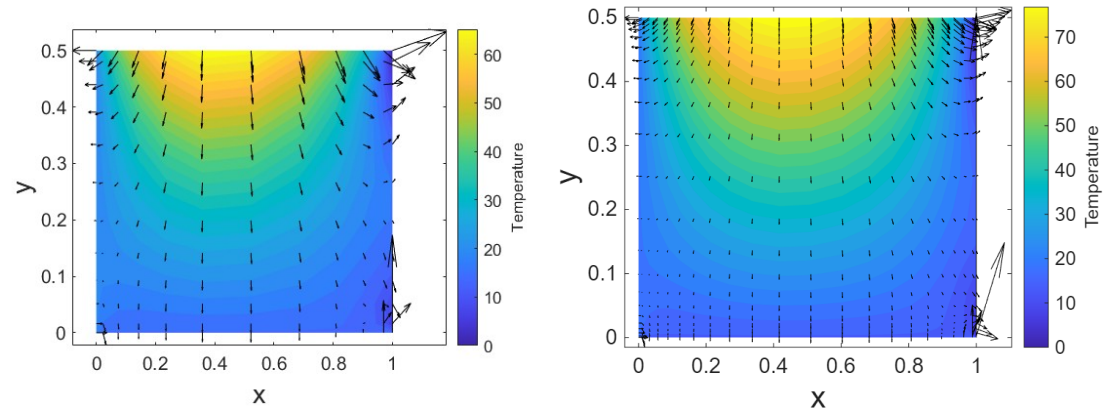
1. Decreasing ϵ improves solution accuracy slightly but increases iterations (computational cost).
2. Beyond $\epsilon = 1e^{-5}$, the improvement is negligible, so a moderate tolerance can be chosen to save computation while maintaining accuracy

3.4 Switching Boundary Conditions

The newly enforced boundary conditions are as follows:

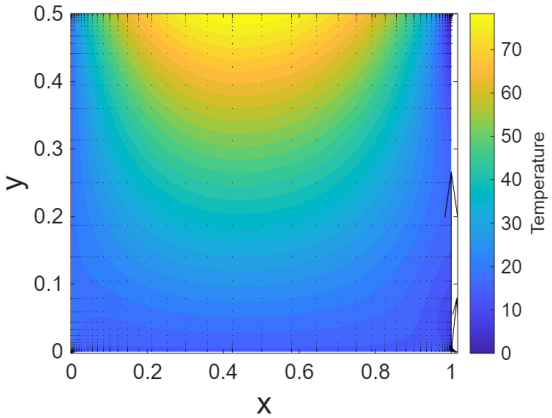
T_1	T_2	T_3	T_4	S	k
15	$5(1-\frac{y}{H})+15\sin(\frac{\pi y}{H})$	$q=-5000$	20	-1.5	$16(1+\frac{y}{H})$

Table 3: New Boundary Conditions (switched T_3 & T_4)

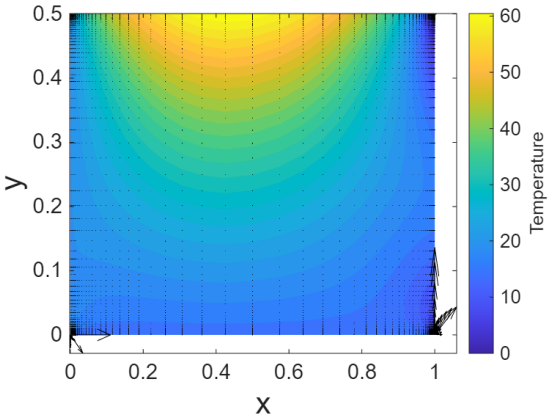


(a) 10×10 refined mesh

(b) 20×20 refined mesh



(c) 40×40 refined mesh



(d) 80×80 refined mesh

Figure 8: Vector plot of heat flux of 4 different refined mesh (15% in 'x' & 25% in 'y')

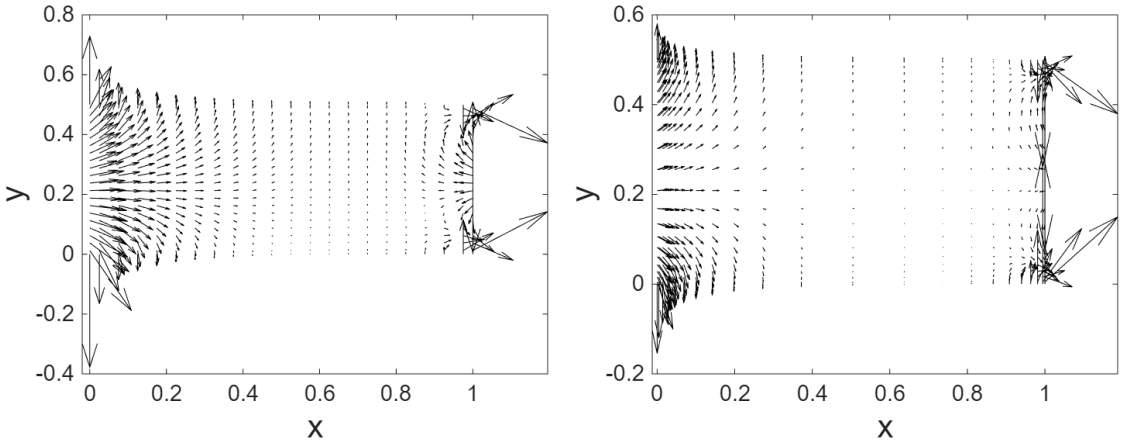
Mesh (Refined)	Max T (New BC)	Loc (New BC)	Max T (New BC)	Loc (Old BC)
10×10	77.7	(1,6)	48.0	(6,1)
20×20	79.3	(1,11)	44.6	(11,1)
40×40	80.9	(1,21)	42.5	(21,1)
80×80	81.2	(1,41)	42.0	(41,1)

Table 4: Comparison of different BCS with Max Temp (K) and Location in (row,col) of the mesh

The heat flux plot reflects the changed boundary conditions, where there is a heat source from the top boundary diffusing downward. The global temperature distribution reorganises so that isotherms align with the new Dirichlet boundaries and flux streamlines connect sources and sinks along new paths. As a result of this, we see a shift in the value of maximum temperature and in the location of it, which turns out to be symmetrically opposite in both cases as observed from table (4). We have also changed the refinement to have a more refined mesh near the upper boundary to accomodate the heat source at the top.

3.5 Heat Flux vector plot

We have plotted the heat flux vectors for all the solutions given above. For the boundary conditions in **Case 4:** The direction and magnitude of heat flow, shows a strong, constant heat influx at the left boundary due to the enforced Neumann condition ($q = -5000$). Heat is then seen to exit through the colder top and right boundaries. The material's thermal conductivity, $k = 16(1 + \frac{y}{H})$, increases with height, causing heat to follow the more conductive, path of least resistance.



(a) 20×20 equidistant mesh (b) 20×20 refined mesh: (25% 'x' & 15% 'y')

Figure 9: Heat Flux vector plots without contours

The refined mesh, with smaller cells concentrated near the boundaries, accurately captures the steep temperature gradients, resulting in sharply defined contours. In contrast, the coarse, equidistant mesh averages out these sharp transitions, leading to a smoother, less physically accurate solution. Therefore, the refined mesh is needed for resolving the true behavior in these high-gradient regions and producing a solution closer to the actual physical solution.