Project 2: Solution to the Laplace equation

Name – Swadesh Suman

McGill ID – 261097252

Objective statement – To solve the Laplace equation on a unit square with Dirichlet boundary conditions u(x, 0) = u(0, y) = u(1, y) = 0, and u(x, 1) = 1. Discretize the second derivatives of u with respect to x and y with a second-order finite-difference spatial discretization. Write the following numerical codes to solve the linear system using the Jacobi, Gauss-Seidel, and Successive Over-Relaxation (SOR) on three grid sizes, 100 × 100, 200 × 200, and 400 × 400.

Problem 1 & 2 -Attached in the pdf in the last pages.

Problem 3 – Demonstrate the solution of the Laplace Equation for the 400 × 400

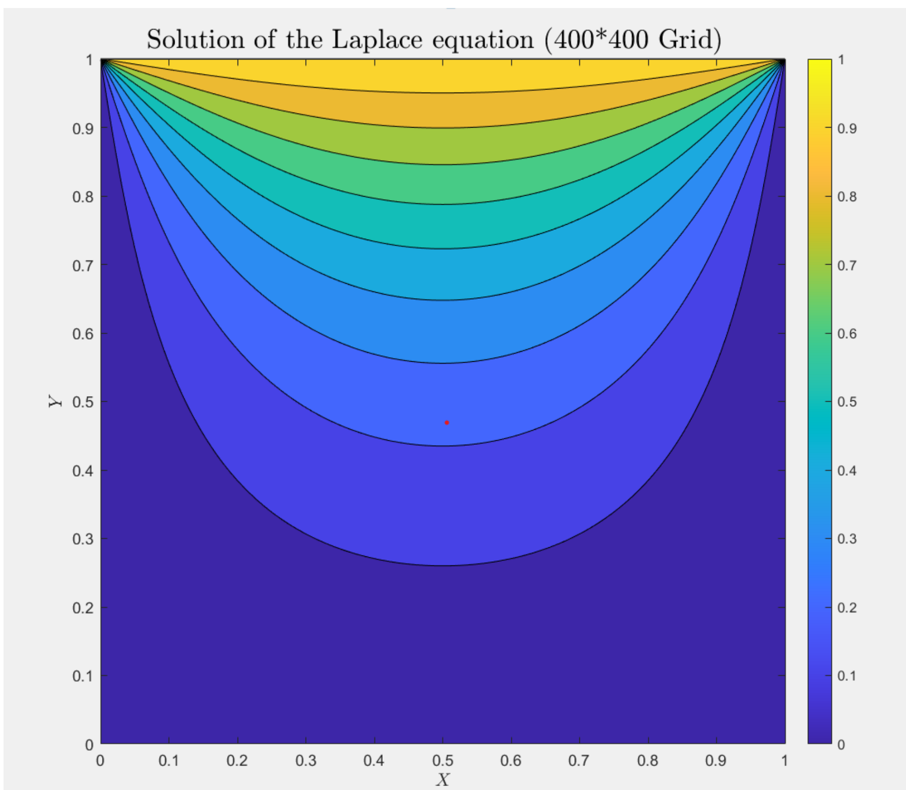

Fig:1 – Contour showing the variation of u

Fig. 1 shows the contour plot of variable u for 400*400 grid points. As per the boundary condition, u is higher on the top surface of the grid. The other three boundaries are set at u = 0.

Problem 4 – Convergence of the log(residual) versus the number of iterations for all three methods on the same plot. Provide a plot for each grid size. Discuss the difference between the schemes. Compute the condition number of the matrix using the Forsythe-Moler method and discuss the results.

a) log(residual) versus the number of iterations for 100*100 mesh grid

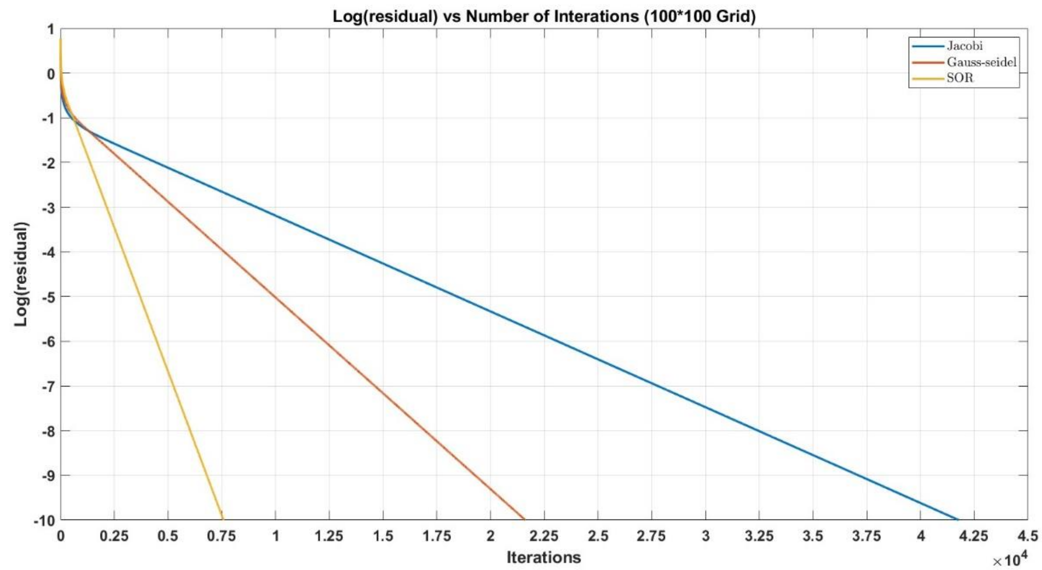

*Fig:2 – Plot of Log(residual) vs Number of Iterations for 100*100 Grid*

b) log(residual) versus the number of iterations for 200*200 mesh grid

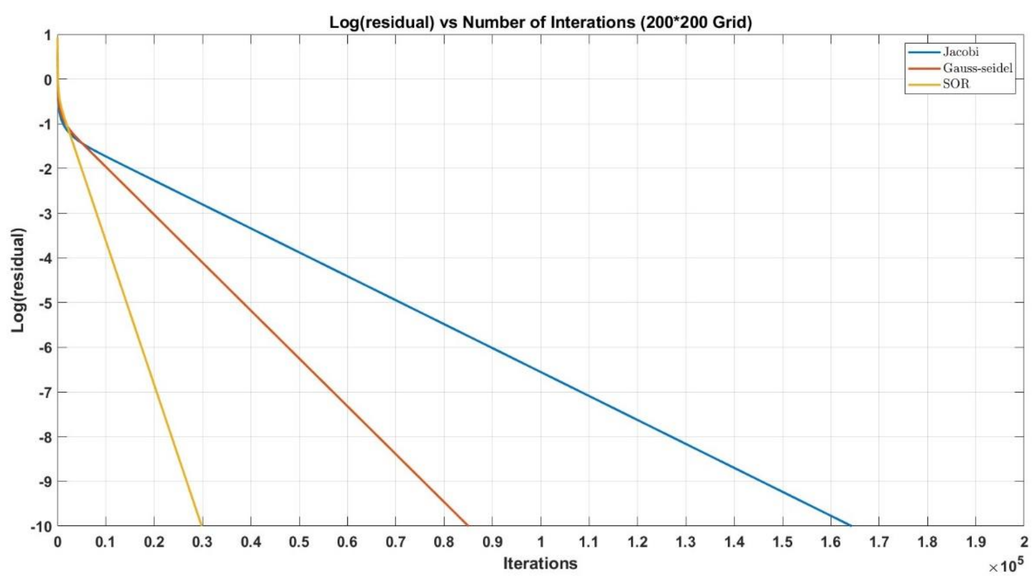

*Fig:3 – Plot of Log(residual) vs Number of Iterations for 200*200 Grid*

c) Log(residual) vs number of iterations for 400 *400 grid

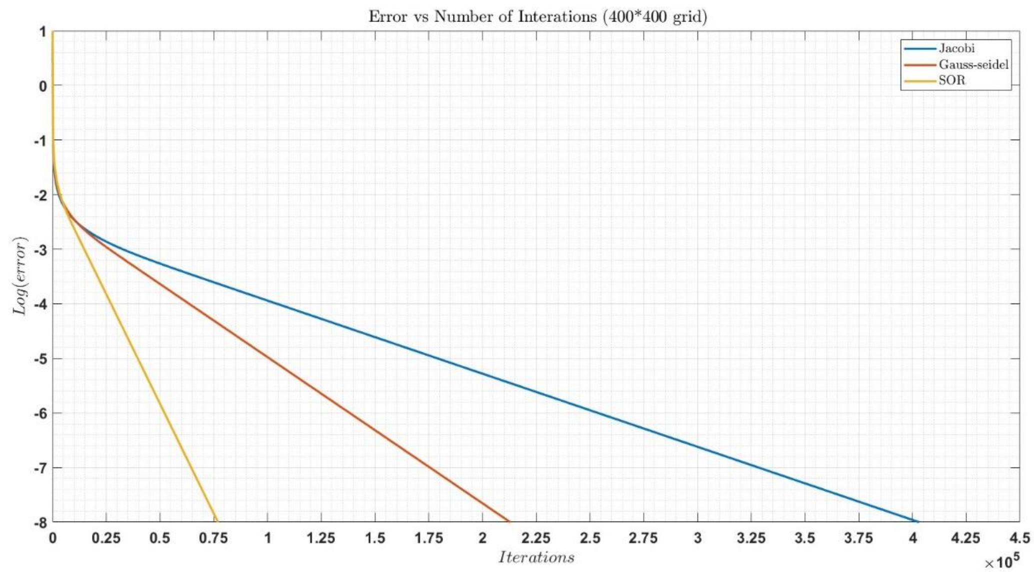

*Fig:4 – Plot of Log(residual) vs Number of Iterations for 400*400 Grid*

Fig 2, 3 & 4 shows the plot of log(residual) vs the total number of iterations taken by Jacobi, Gauss-Seidel, and Successive Over-relaxation (SOR) to reach the desired tolerance value of E-10, E-10, & E-8 respectively.

Discussion –

- The desired tolerance for 400*400 grid is chosen as E-8 as the computational time is relatively much higher compared to other two grid sizes. The total computational time is discussed in details in the next problem 5.
- The SOR method is solved at h = 1.5. (h = over-relaxation factor).
- The total number of iterations taken is highest in Jacobi and lowest in the SOR method. Whereas comparing the grid sizes, it can be seen from the plot that the total number of iterations is highest in 400*400 grid size.
- The number of iterations taken for convergence by SOR is approximately half when compared to that taken by the Gauss-Seidel method. Whereas, the number of iterations taken for convergence by Gauss-Seidel is approximately half that taken by the Jacobi method.
- The condition number calculated for A matrix (400 * 400) using the Forsythe-Moler method is 3.28 e+51. The reason for higher condition number is probably the division by ESP = 5*10^(-t), where t is the number of mantissas. In 64 bit computer, the value is t is 52.
- However, the condition number calculated by the formula given below is 441.97.

condition(A) = ||A|| · ||A−1||

Problem 5 – Convergence of the log(residual) versus the CPU time for all three methods on the same plot. Discuss the difference between the schemes. Comment on the number of vectors and arrays that were necessary for each scheme and compare the algorithms in terms of memory usage

a) log(residual) versus total CPU time for 100*100 mesh grid

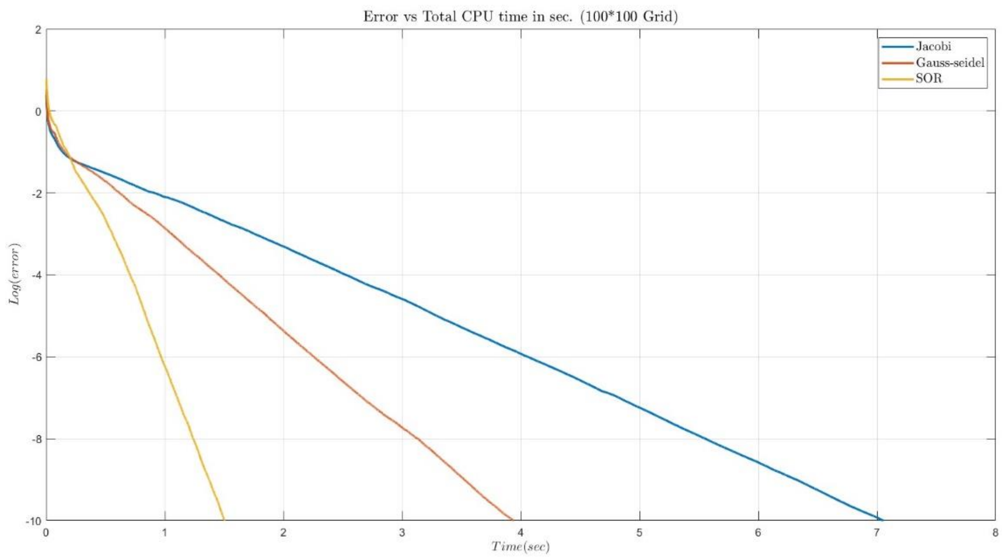

*Fig:5 – Plot of Log(residual) vs total CPU time(sec.) for 100*100 Grid*

b) log(residual) versus total CPU time for 200*200 mesh grid-

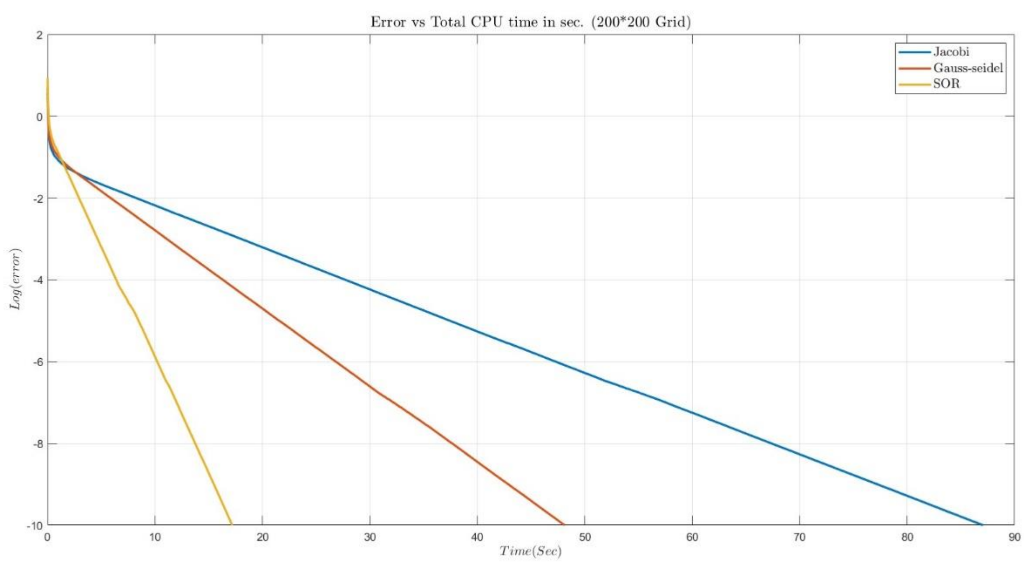

*Fig:6 – Plot of Log(residual) vs total CPU time(sec.) for 200*200 Grid*

c) Log(residual) vs total CPU time for 400 *400 grid -

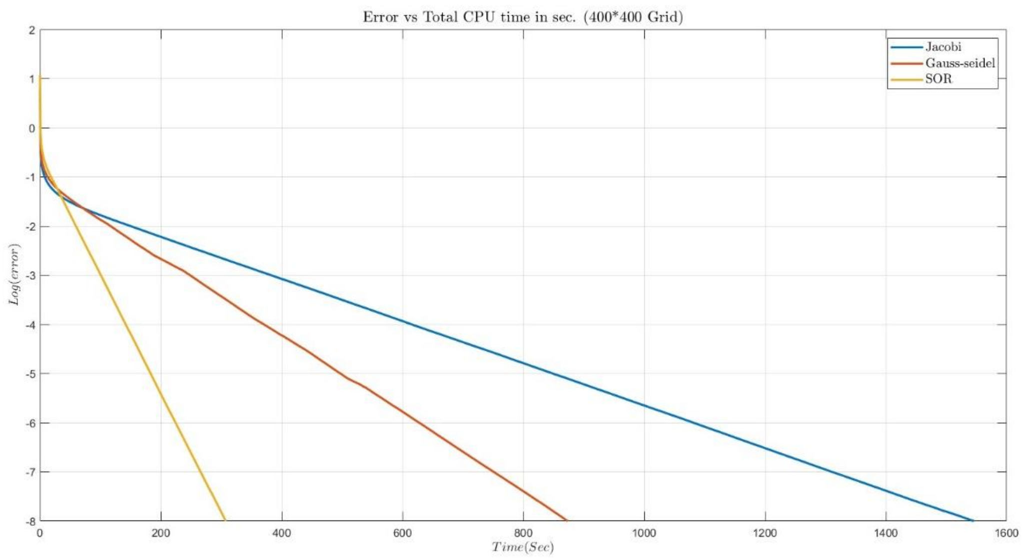

*Fig:7 – Plot of Log(residual) vs total CPU time(sec.) for 400*400 Grid*

Fig 5, 6 & 7 shows the plot of log(residual) vs the total CPU time taken by Jacobi, Gauss-Seidel, and Successive Over-relaxation (SOR) to reach the desired tolerance value of E-10, E-10, & E-8 respectively.

Discussion-

- As it can be seen in the plots, the total CPU time taken by the SOR method (h = 1.5) is the least in all the grids.
- For the 100*100 grid, the total CPU time taken by Jacobi is close to 7 sec, but for 400 * 400 this number goes as high as 1500 sec. Hence for the 400*400 grid, the designed tolerance is taken as E-8.
- For 400 * 400 computational grid, the total CPU time taken by SOR method (h = 1.5) is approximately 175 sec which half of what taken by Gauss-Seidel and more than 5 times less than what taken by the Jacobi method.
- For the comparison of the vector and arrays necessary for each scheme, 200* 200 computational grid is used. The table below shows the sizes of each array used in all the three methods for 200*200 grid.

Method	Array Name	Size	Storage Memory
Jacobi	Array to store – Number of Iterations	143689*1	1.096 MB

	Array to store – Error in each iterations	143689*1	1.096 MB
	Array to store – CPU time for each loop	143689*1	1.096 MB
Gauss-Seidel	Array to store – Number of Iterations	74704*1	0.567 MB
	Array to store – Error in each iteration	74704*1	0.567 MB
	Array to store – CPU time for each loop	74704*1	0.567 MB
Successive Over-relaxation method (h = 1.5)	Array to store – Number of Iterations	26408*1	0.201 MB
	Array to store – Error in each iterations	26408*1	0.201 MB
	Array to store – CPU time for each loop	26408*1	0.201 MB

- As can be seen from the above table, the SOR method has the lowest memory usage when compared to that used by the other two methods. At this point of using a 200*200 grid, these values might not be significant, but the actual physical fluid problem may contain millions of nodes and hence in those situations these values will play a significant role in deciding the method to choose for solving the linear systems.

<u>Problem 6 – Demonstrate the order of accuracy of the scheme on a plot where the y-axis is the log of the error and the x-axis is Δx, where both Δx = Δy.</u>

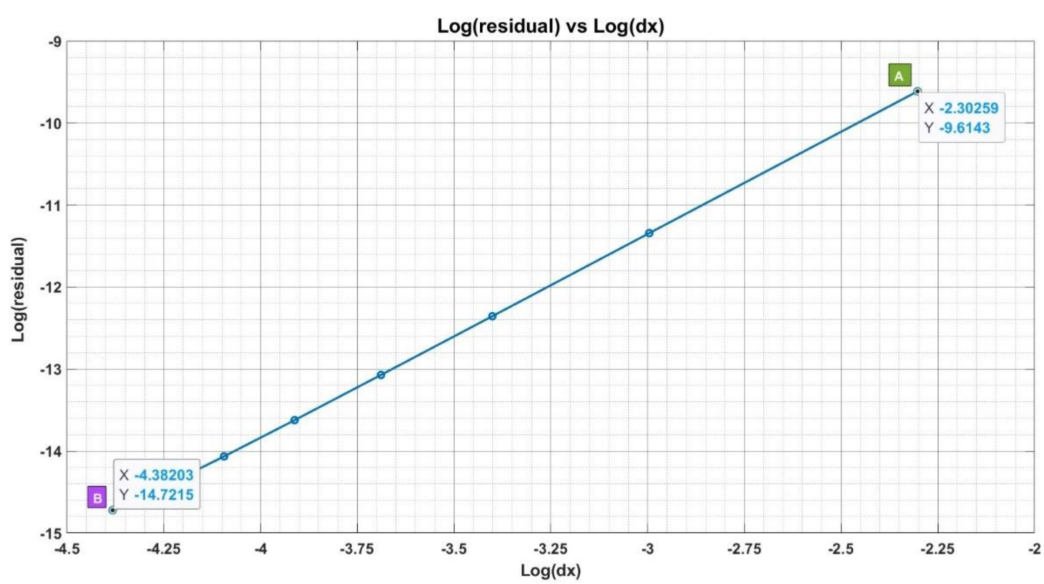

Fig:8 – Plot of Log(residual) vs log(dx) for varying grid sizes

Fig 8 shows the plot of log(residual) vs Log (dx). The error is the difference between the exact solution of the Laplace equation and the u value calculated by the gauss-seidel method. The initial grid size is taken as 10*10, which is then gradually refined till 150*150 grid size each time increasing the grid size in the multiples of 10 (total 15 computational grids).

Discussion –

- Point A in Fig 8, shows the log(dx) as X and log(residual) as Y. From point C each point represents the grid sizes (20*20, 30*30, and so on..).
- Point B represents 80*80 grid size. Where the X value represents the log(dx) and Y value represents the log(residual).
- The error is calculated using the 2^{nd} norm, i.e.,

$$Error = \frac{1}{N}[\sum_{i=1}^{L}(U_{exact} - U)\ \hat{}2 * dx]\hat{}1/2$$

Where, $$U_{exact} = \sum_{n=1}^{inf}\left(\frac{2}{n\pi\sinh(n\pi)}\right)\sinh(n\pi y/L)\sin(n\pi x/L)$$

- The slope of Line BC is 2.4 which is close to 2 as the scheme used for discretizing is second-order accurate.

Problem 7 – Effect of the relaxation parameter on the SOR. Try several different values and discuss your findings. Show plots of the convergence of the log(residual) for various relaxation parameters. Is there an optimum relaxation parameter? Is the optimum the same for all grid sizes?

a) log(residual) versus total number of iterations for different value of h for 100*100 mesh grid

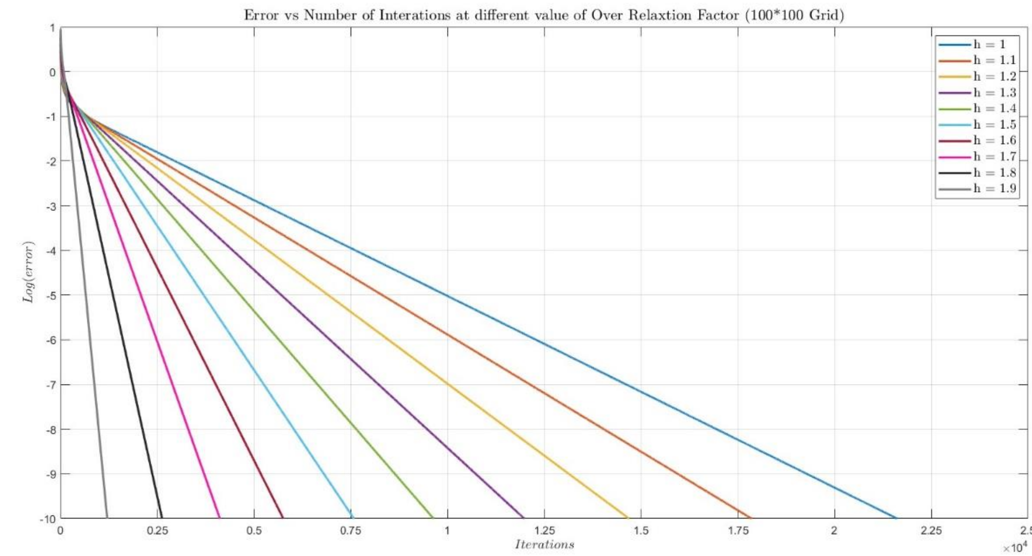

*Fig:9 – Plot of Log(residual) vs total number of iterations for different value of h for 100*100 Grid*

b) log(residual) versus total number of iterations for different value of h for 200*200 mesh grid-

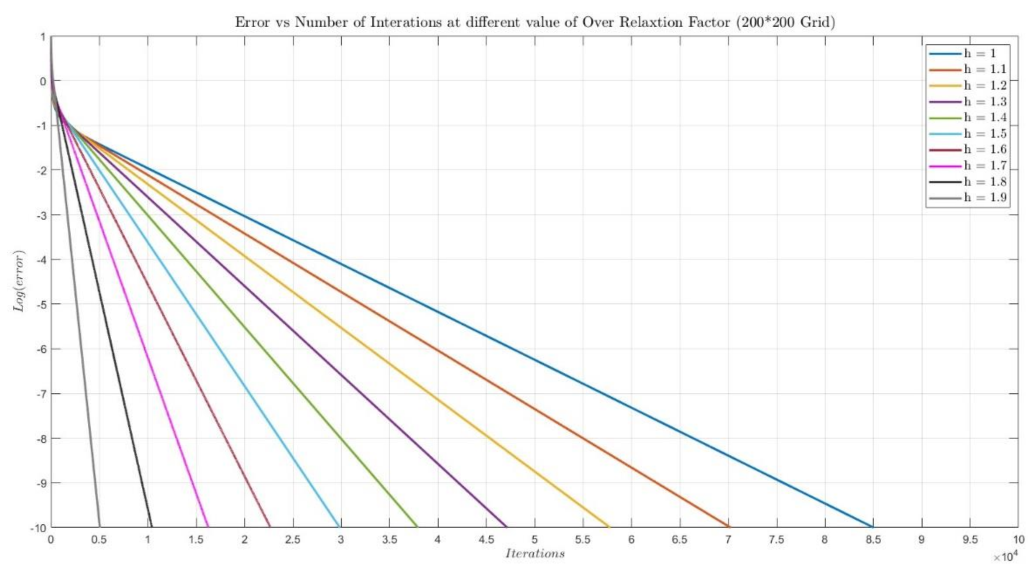

*Fig:10 – Plot of Log(residual) vs total number of iterations for different value of h for 200*200 Grid*

c) Log(residual) vs the total number of iterations for different value of h for 400 *400 grid -

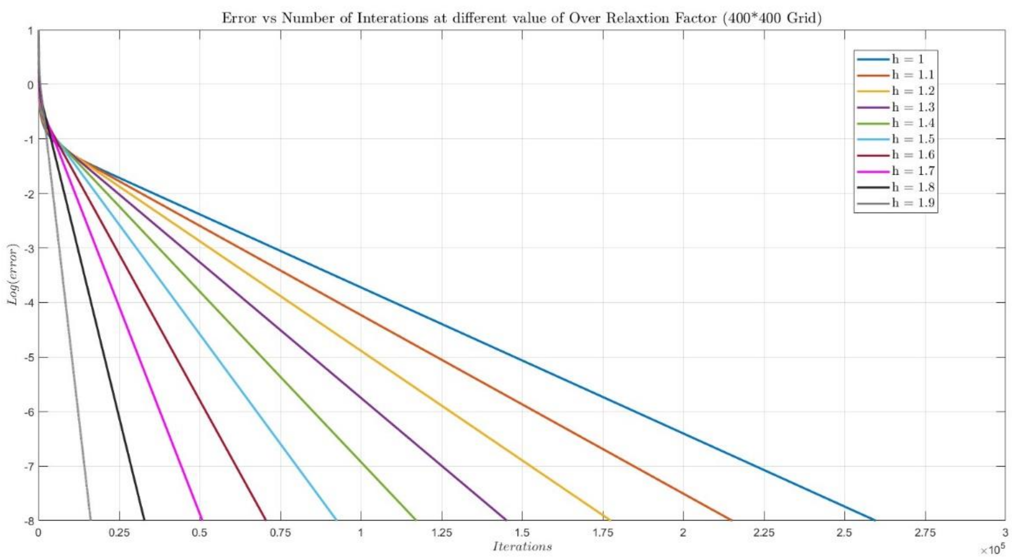

*Fig11 – Plot of Log(residual) vs total number of iterations for different value of h for 400*400 Grid*

Fig 9, 10 & 11 shows the plot of log(residual) vs the total number of iterations for different value of h in the 100*100,200*200 & 400*400 grid to reach the desired tolerance value of E-10, E-10, & E-8 respectively.

Discussion –

- For calculation of the optimum value of h in the SOR method, the value of h is varied from 1 to 1.9.
- With the increase in the value of h, the total number of iterations taken to converge continues to decrease to 1.9. Hence there is no optimum value of h is in this range.
- This is the case with all three grid sizes.
- However, at h = 2, the solution diverges. Hence there might be an optimum value of h between the 1.9 to 2 range.