FLORIDA INSTITUTE OF TECHNOLOGY
MECHANICAL AND AEROSPACE ENGINEERING DEPARTMENT

MAE 5150-E1: Computational Fluid Dynamics

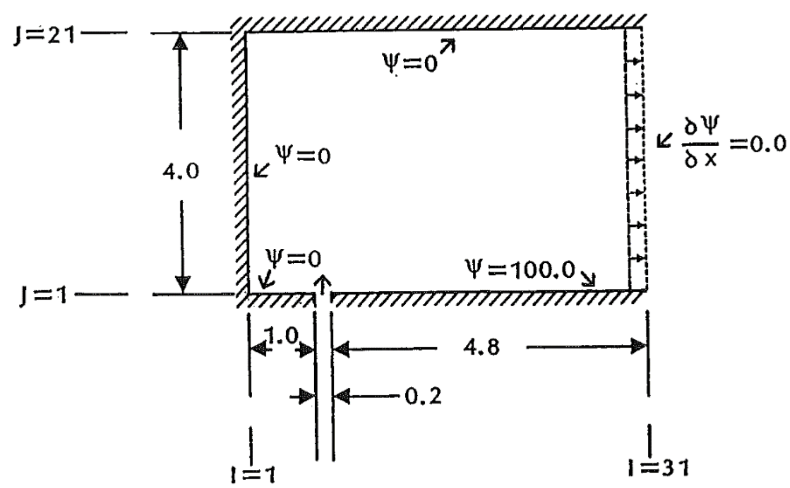
Fall 2017

Coding Project 2

Name: Max Le

Due October 17, 2017

A two-dimensional inviscid, incompressible fluid is flowing steadily through a chamber between the inlet and the outlet, as shown in the figure. It is required to determine the streamline pattern within the chamber.



For a two-dimensional, incompressible flow, the continuity equation is expressed as

$$\frac{\partial u}{\partial x} + \frac{\partial v}{\partial y} = 0$$

A stream function Ψ may be defined such that

$$u = \frac{\partial \Psi}{\partial y} \quad \text{and} \quad v = -\frac{\partial \Psi}{\partial x}$$

Recall that a streamline is a line of constant stream function. Furthermore, vorticity is defined as

$$\Omega = \nabla \times V$$

for which

$$\Omega_z = \frac{\partial v}{\partial x} - \frac{\partial u}{\partial y}$$

For an irrotational flow, the vorticity is zero. Therefore,

$$\frac{\partial v}{\partial x} - \frac{\partial u}{\partial y} = 0$$

Substituting the definitions of the stream function into the above equations yields

$$\frac{\partial}{\partial x}\left(-\frac{\partial \Psi}{\partial x} \right) - \frac{\partial}{\partial y}\left(\frac{\partial \Psi}{\partial y} \right) = 0$$

or

$$\frac{\partial^2 \Psi}{\partial x^2} + \frac{\partial^2 \Psi}{\partial y^2} = 0$$

The goal in this problem is to obtain the solution of this elliptic partial differential equation using the various numerical techniques discussed earlier. The solution will provide the streamline pattern within the chamber.

Since the chamber walls are streamlines, i.e. lines of constant Ψ, we will assign values for these streamlines, as shown in the figure. Solve this problem with codes using the following techniques:

a) Point Gauss-Seidel	c) Point SOR
b) Line Gauss-Seidel	d) Line SOR

For all methods, the step sizes are specified as

$$\Delta x = 0.2, \qquad \Delta y = 0.2 \qquad ERRORMAX = 0.01$$

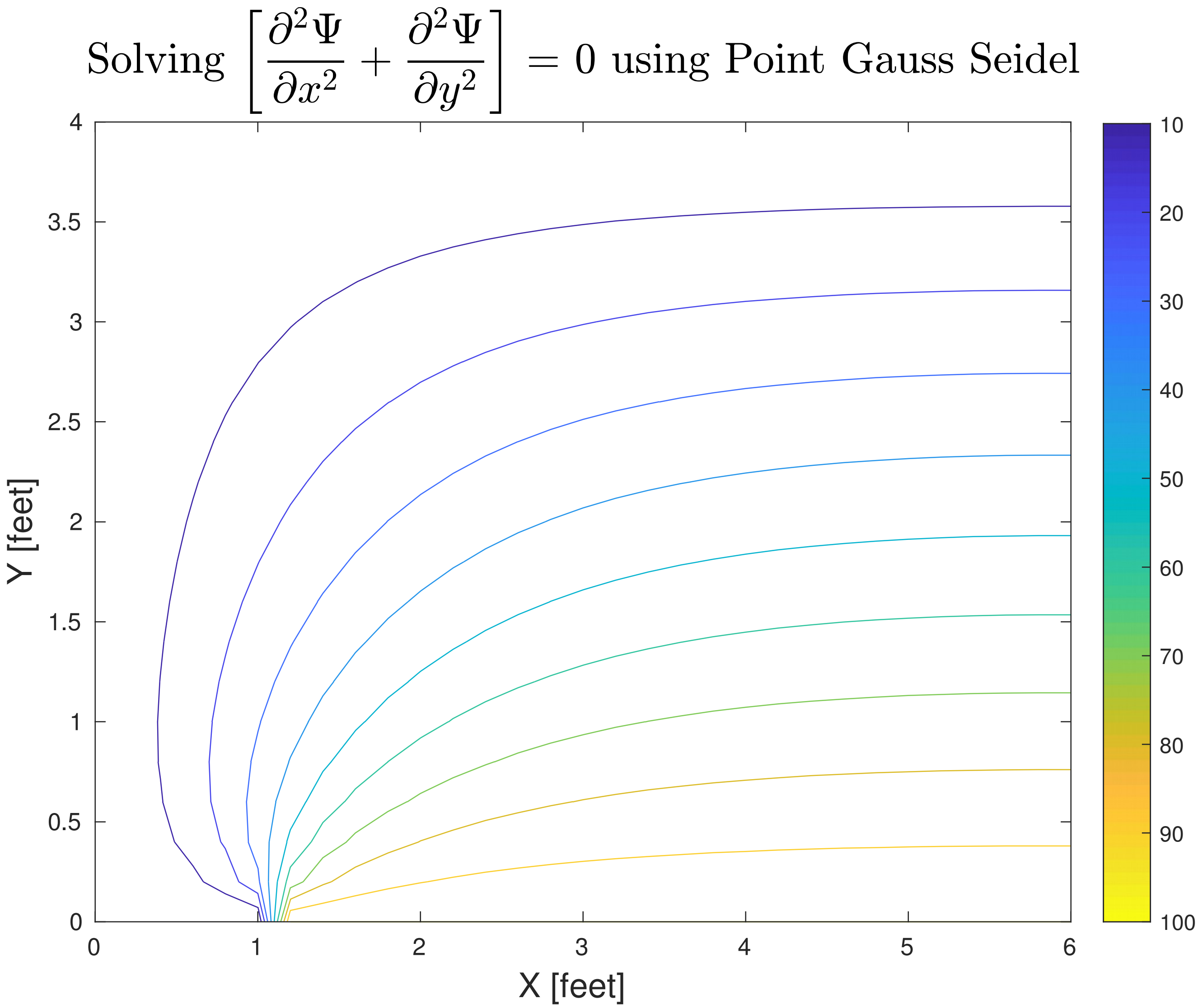
with convergence criterion *ERROR* < *ERRORMAX*, where

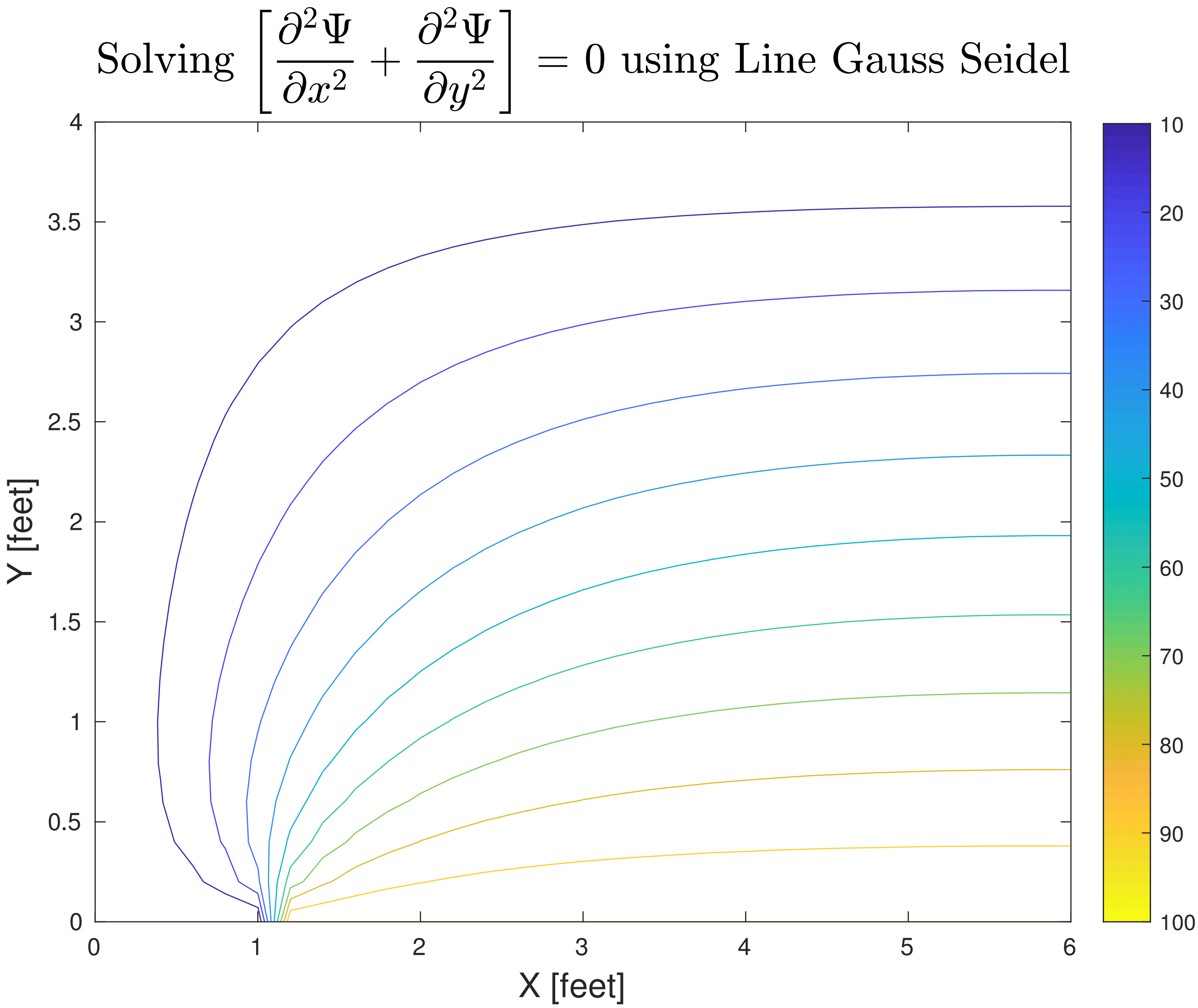
$$ERROR = \sum_{\substack{i=2 \\ j=2}}^{\substack{j=JM-1 \\ i=IM-1}} \left| \Psi_{i,j}^{k+1} - \Psi_{i,j}^{k} \right|$$

Use an initial data distribution of $\Psi = 0.0$. Plot the streamline pattern (lines of constant Ψ). Rerun the SOR codes for several values of the relaxation parameter and plot the relaxation parameter versus the number of iterations for these two schemes. In each case, determine the optimal value of the relaxation parameter.

Submit a both a hardcopy of your code and an electronic copy on Canvas.

Note: Except for the purpose of creating plots, MATLAB, Excel, or other commercially available software may not be used for this assignment. You may otherwise program in any language you wish.


Solving $\left[\frac{\partial^2\Psi}{\partial x^2}+\frac{\partial^2\Psi}{\partial y^2}\right]=0$ using Point Gauss Seidel
Y [feet]
X [feet]
0
0.5
1
1.5
2
2.5
3
3.5
4
0
1
2
3
4
5
6
10
20
30
40
50
60
70
80
90
100


Solving $\left[\frac{\partial^2\Psi}{\partial x^2} + \frac{\partial^2\Psi}{\partial y^2}\right] = 0$ using Line Gauss Seidel
Y [feet]
X [feet]
0
0.5
1
1.5
2
2.5
3
3.5
4
0
1
2
3
4
5
6
10
20
30
40
50
60
70
80
90
100

w	Iteration	Time[sec]
0.100	9813	0.289863
0.200	5141	0.140544
0.300	3430	0.096141
0.400	2524	0.070281
0.500	1958	0.053929
0.600	1567	0.042769
0.700	1279	0.034924
0.800	1058	0.028872
0.900	881	0.024065
1.000	737	0.020121
1.100	616	0.016729
1.200	512	0.013936
1.300	423	0.011516
1.400	344	0.009367
1.500	274	0.007498
1.600	210	0.005722
1.700	151	0.004122
1.800	92	0.002550
1.900	99	0.002697
2.000	39686	1.085977

Table 1- Different values of relaxation parameter (w) for Point SOR

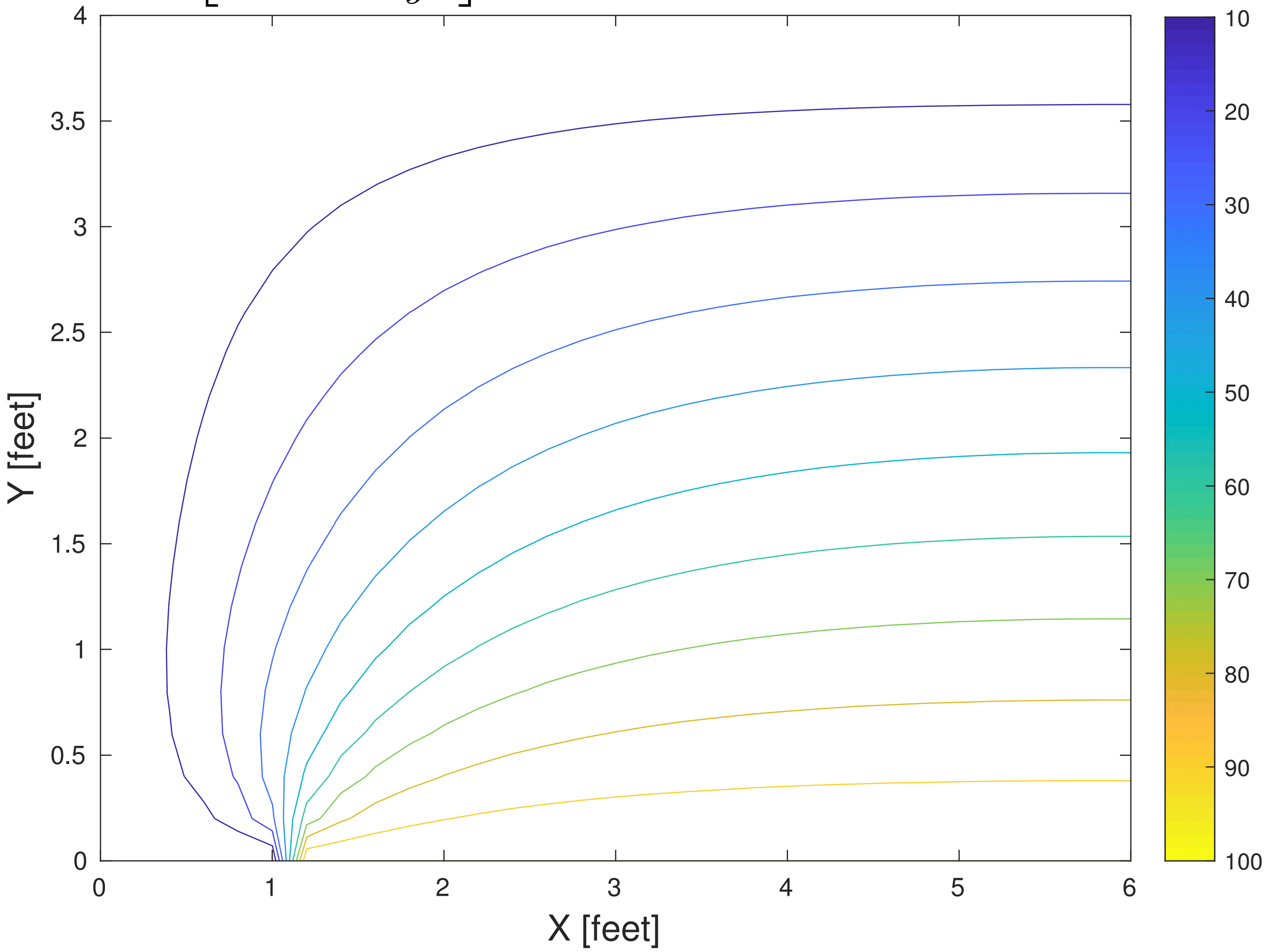
It can be seen that at around w = 1.8 and w = 1.9, the solution converges faster (less iterations and less computational time). After testing with different values between 1.8 and 1.9, it is decided that **w = 1.801** is the optimum relaxation parameter. In the code, setting w = 1.801 takes 91 iterations and 0.002460 seconds.

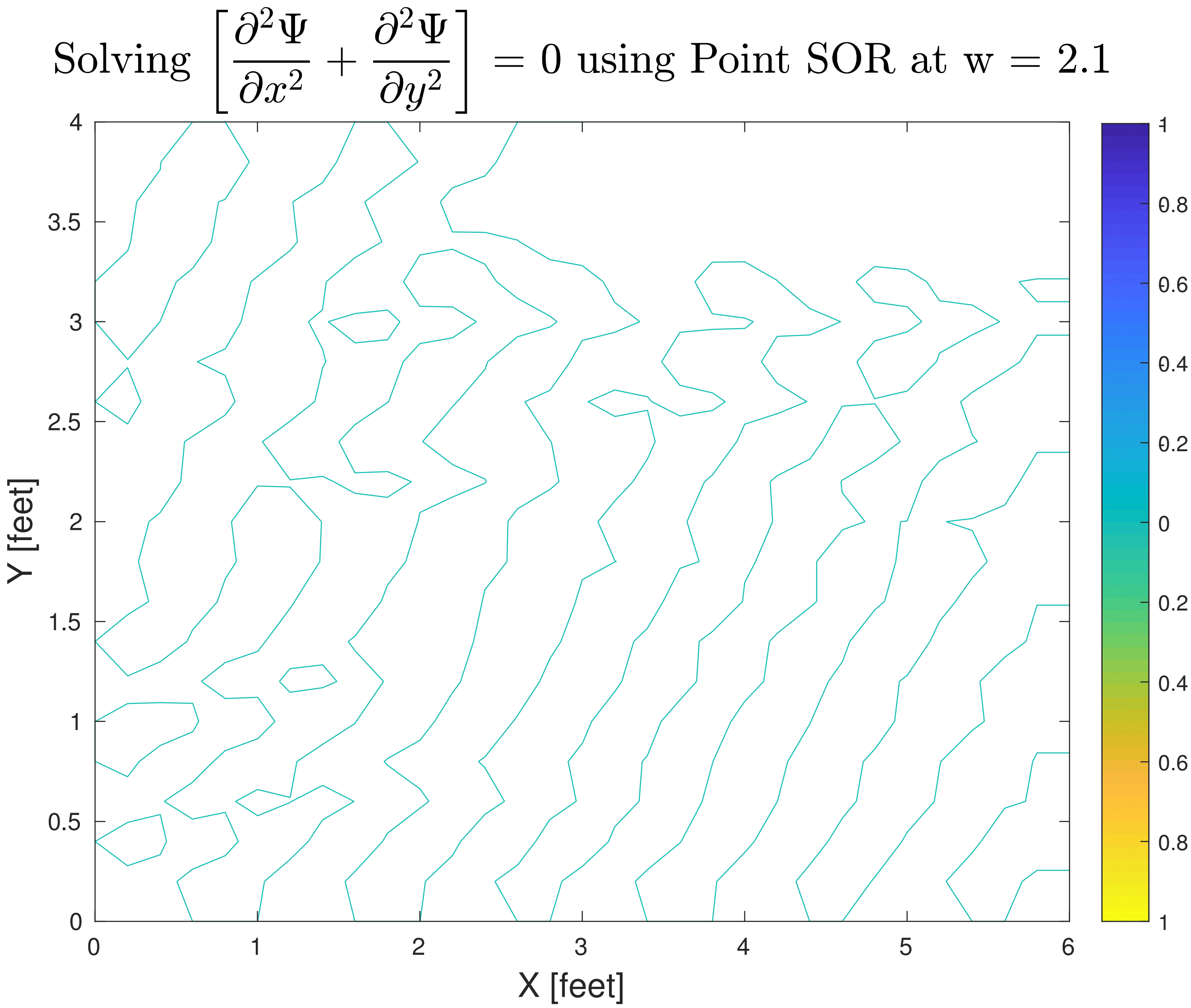
w	Iteration	Time[sec]
0.100	9668	0.397302
0.200	4931	0.188407
0.300	3191	0.119139
0.400	2267	0.089969
0.500	1685	0.063040
0.600	1283	0.048021
0.700	984	0.036855
0.800	753	0.028300
0.900	567	0.021368
1.000	413	0.015536
1.100	283	0.010716
1.200	171	0.006466
1.300	83	0.003198
1.400	2564	0.095974
1.500	566	0.021152
1.600	334	0.012560
1.700	232	0.009338
1.800	170	0.007004
1.900	121	0.005359
2.000	61	0.002851

Table 2- Different values of relaxation parameter (w) for Line SOR

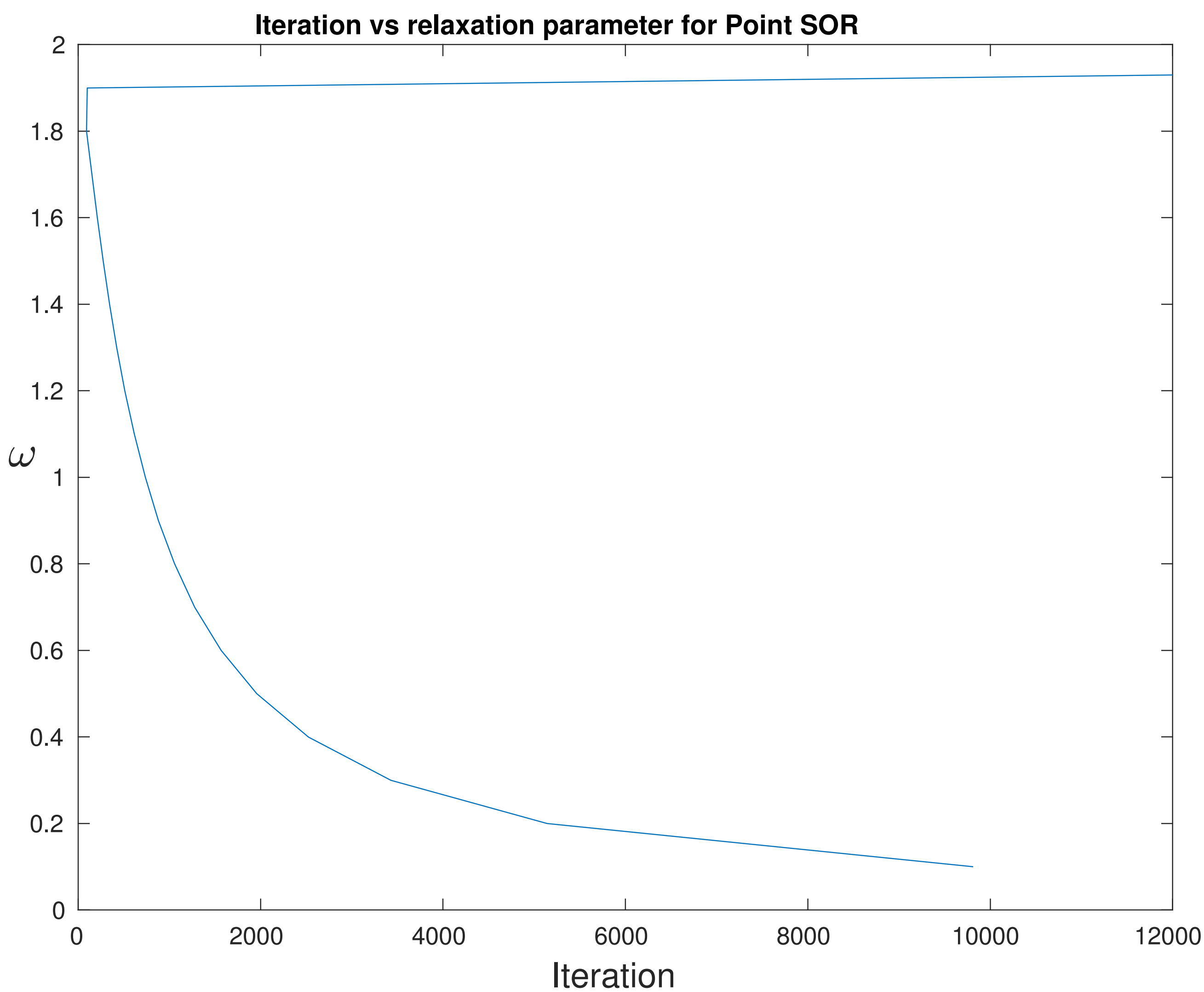
A similar observation can be said for the Line SOR case: at **w = 1.3**, the solution converges the fastest (at 83 iterations and 0.003198 seconds). It can also be seen from later plots of relaxation parameter vs. iteration numbers, that anything higher than the optimum value will not guarantee convergence. As a result, w = 1.3 is chosen as the optimal value for Line SOR.

Solving $\left[\frac{\partial^2\Psi}{\partial x^2}+\frac{\partial^2\Psi}{\partial y^2}\right]=0$ using Point SOR at w = 1.801

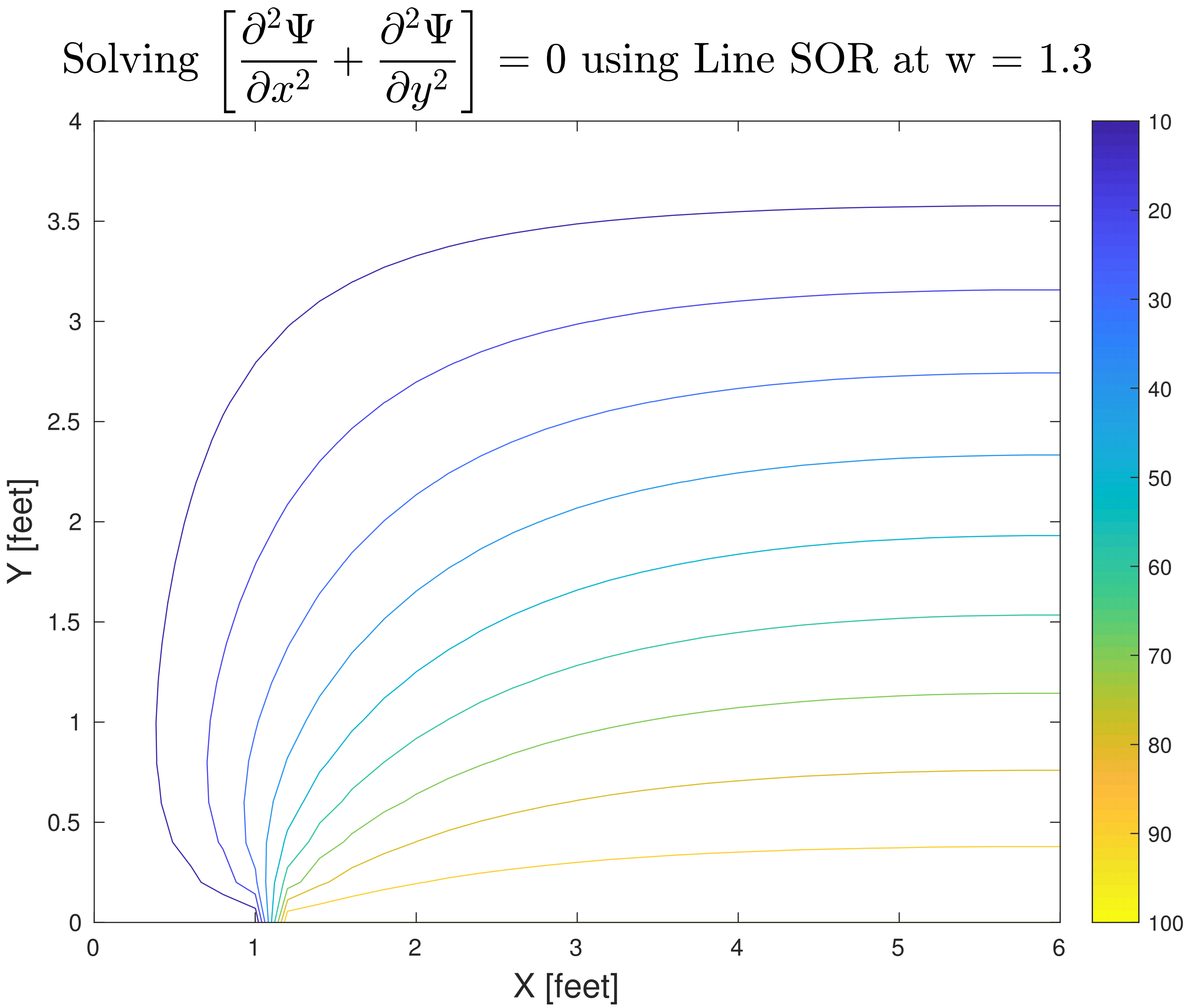




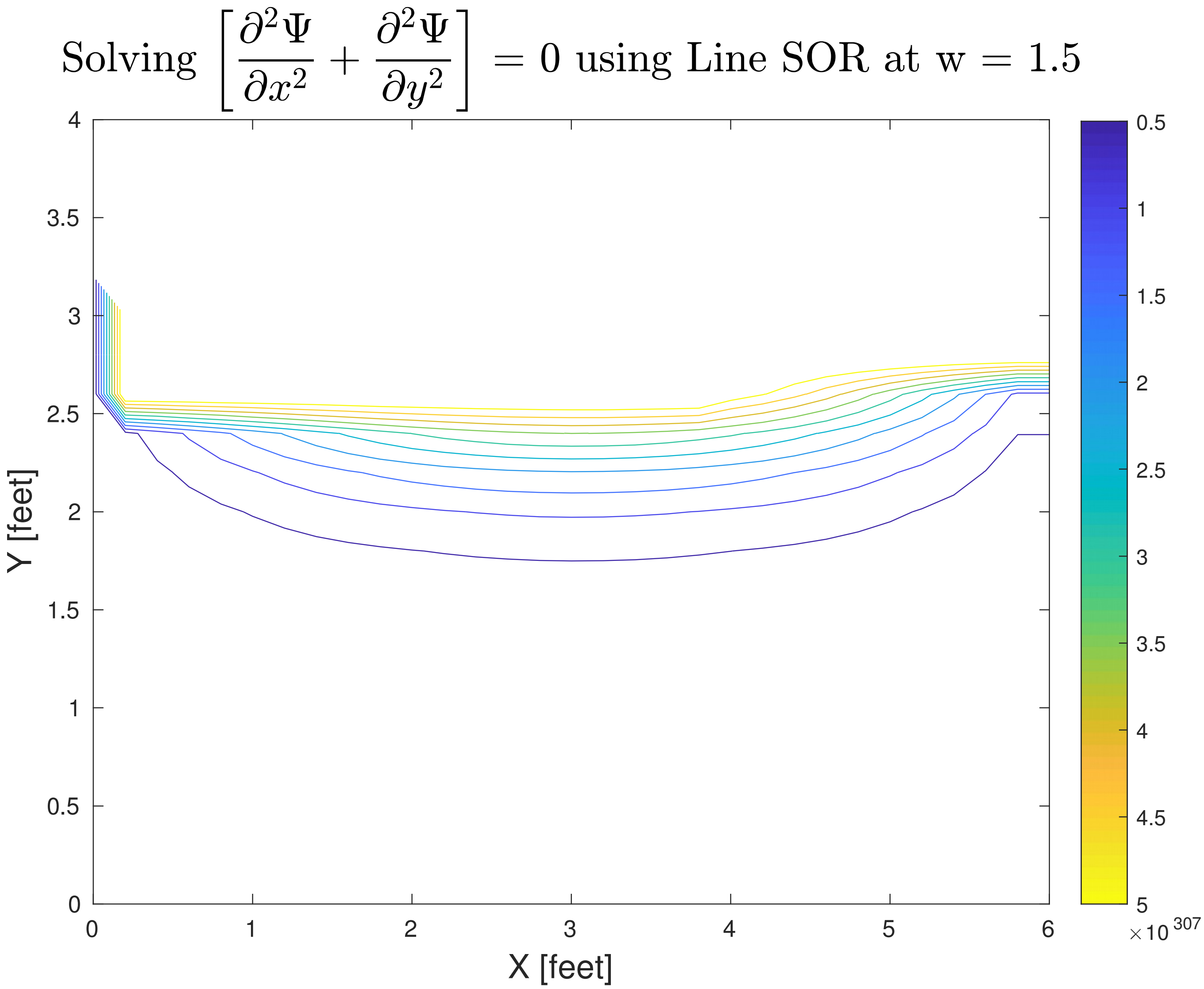
At w>2, the solution blows up and diverges instead.



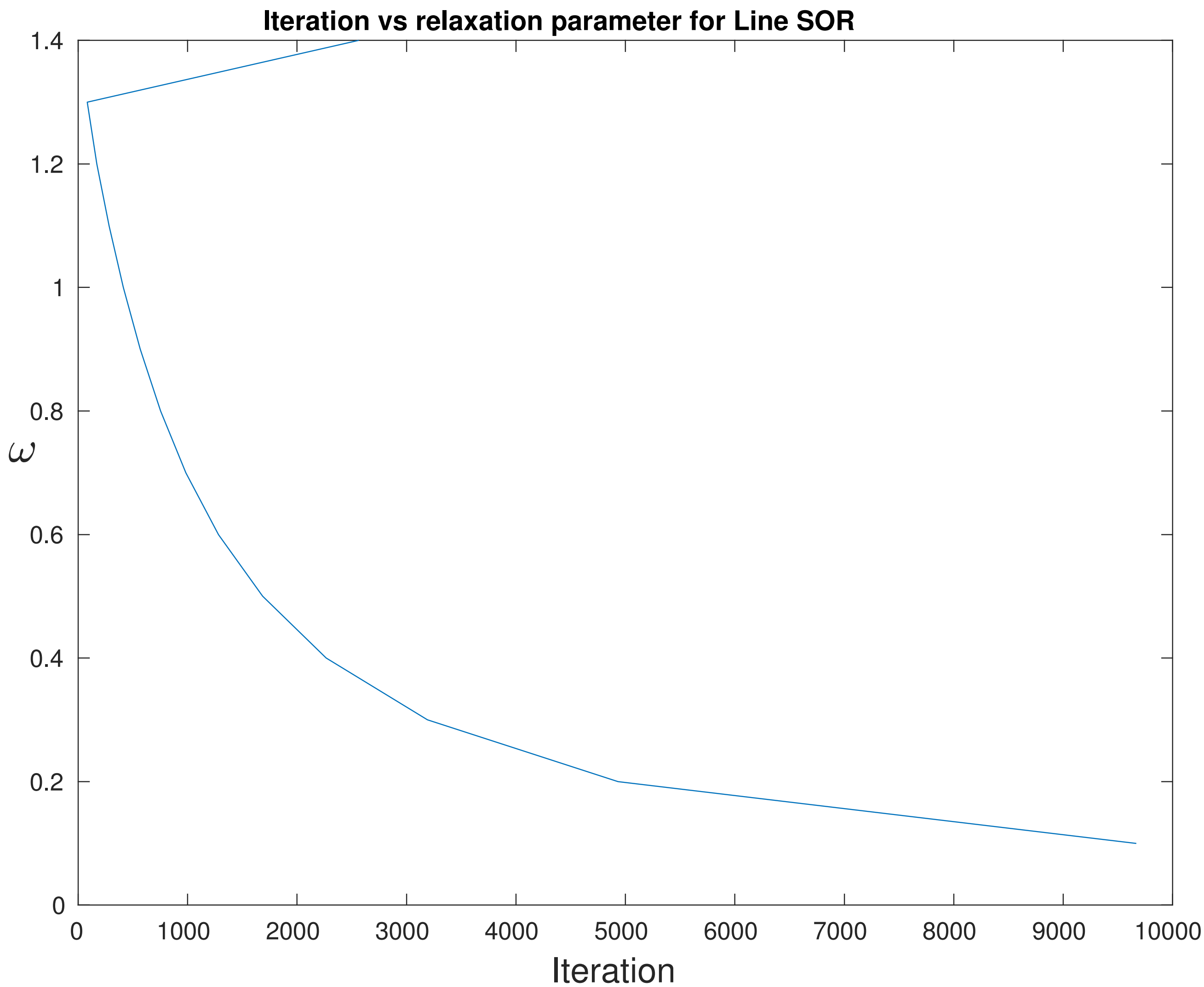
It can be seen that higher W will give lower Iterations. However, very high W will cause the Iterations to rise and from later plots, the solution can also diverge. The optimum relaxation parameter is shown clearly in this plot, where 1.8<w<1.9.



Solving $\left[\frac{\partial^2\Psi}{\partial x^2}+\frac{\partial^2\Psi}{\partial y^2}\right]=0$ using Line SOR at w = 1.3
Y [feet]
X [feet]
0
0.5
1
1.5
2
2.5
3
3.5
4
0
1
2
3
4
5
6
10
20
30
40
50
60
70
80
90
100



For Line SOR, the solution reaches optimum convergence at a lower w than Point SOR. Still, passing this optimal value causes divergence. We can see that the code tries to keep the derivative boundary condition (dPsi/dx = 0) on the right hand side, but due to the divergence of earlier data points, this does not help it to converge.



Line SOR converges faster than Point SOR, and also reaches optimal value of relaxation parameter faster. The same trend can be observed: higher W will give lower Iterations, but very high W will cause divergence. It can also be seen clearly that w = 1.3 is the optimum value for this scheme (less iterations).