

MODELLING OF INTEGRATED DEVICES
ELEC 4700

Finite Difference Method

Author: Tony Lam
Student number: 101077253
Date submitted: March 1, 2021

1 Finite difference Method

1.1 Part 1 Ramp

A ramp 1D simulation is created with the boundary condition of 1V at $X=0$ and 0V at $X=L$.

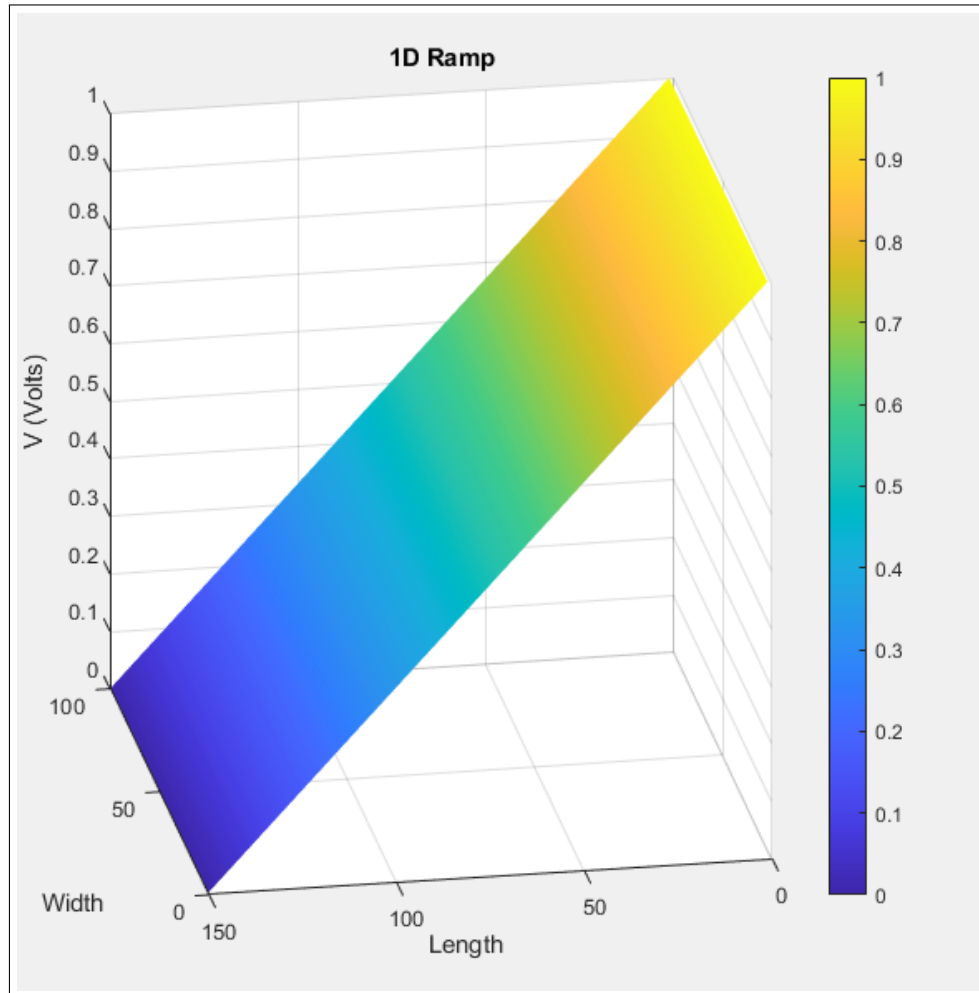


Figure 1: 1-D Ramp Simulation

1.2 Part 1 Saddle

A saddle 2D simulation is created with the boundary X condition of 1V at $X=0$ and 0V at $X=L$. The Y boundary condition is 0V at $y=0$ and $Y=W$.

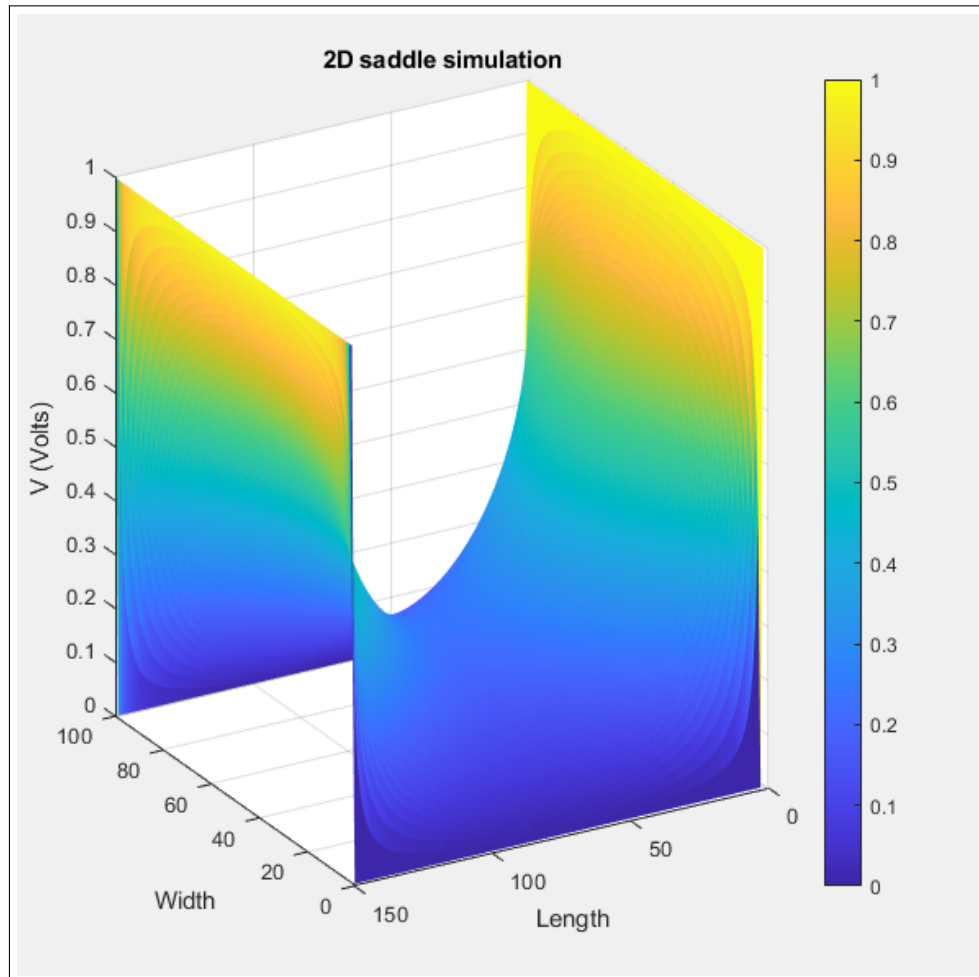


Figure 2: 2-D Saddle Simulation

1.3 Analytical Solution of Saddle

Using the Equation(1) below:

$$V(x,y) = \frac{4V_0}{\pi} \sum_{n=1,3,5..}^{\infty} \frac{1}{n} \frac{\cosh\left(\frac{n\pi x}{a}\right)}{\cosh\left(\frac{n\pi b}{a}\right)} \sin\left(\frac{n\pi y}{a}\right) \quad (1)$$

The analytical iteration result is shown in the Figure (3) below

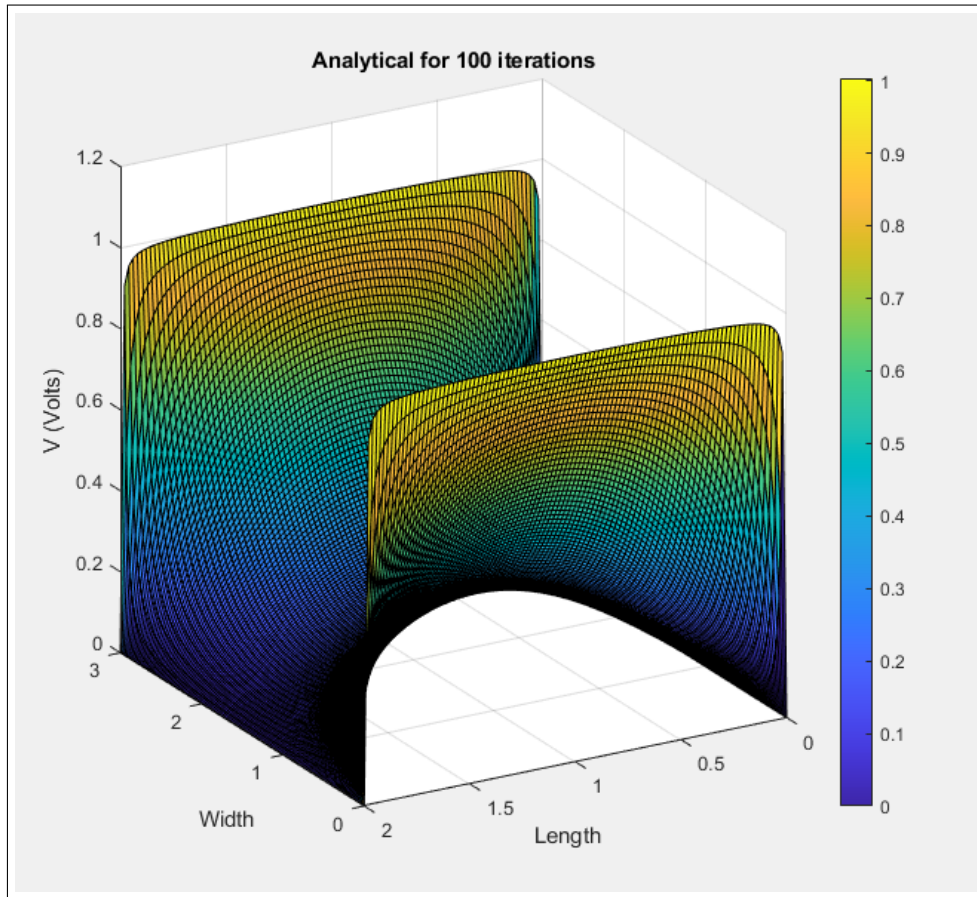


Figure 3: 2-D Saddle Simulation

Meshing can be a trade off between accurate and time. A denser meshing can provide better results but will take longer to produce and a less dense mesh can be inaccurate but much faster for more complex simulations.

Both method is fairly accurate, however the analytical solution took longer to produce and the 1V in the X=0 and X=L is slightly rounded at Y=0 and Y=W.

2 Bottle-neck and current flow

A bottle neck is added to the region where σ is 1 outside the box and $\sigma = 10^{-2}$ inside the box.

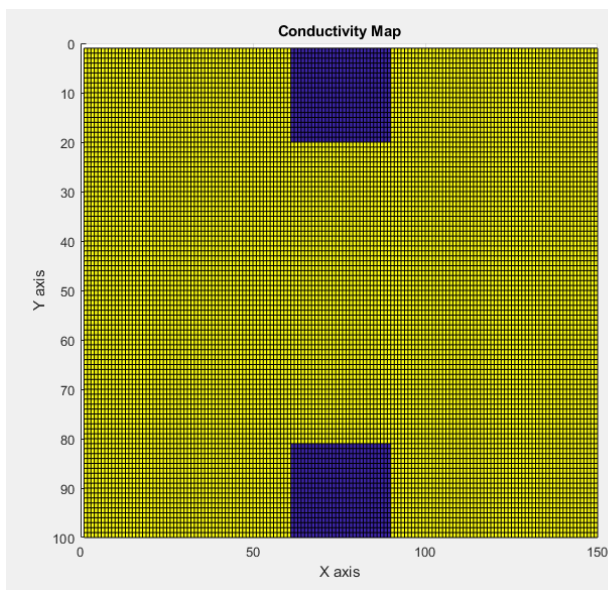


Figure 4: Conduction Map

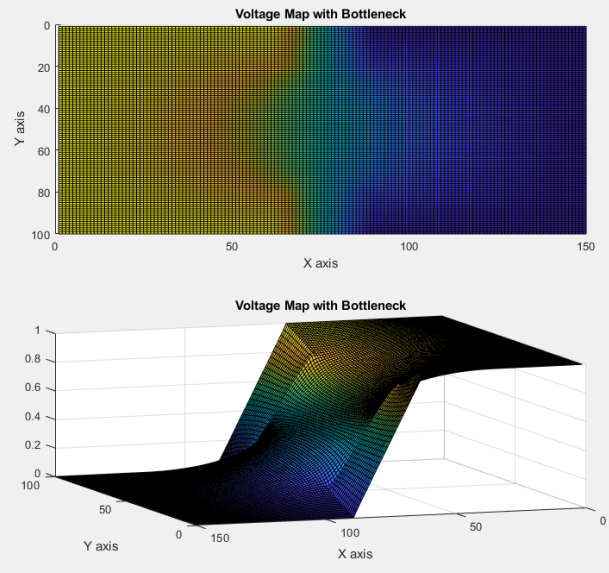


Figure 5: Voltage Map

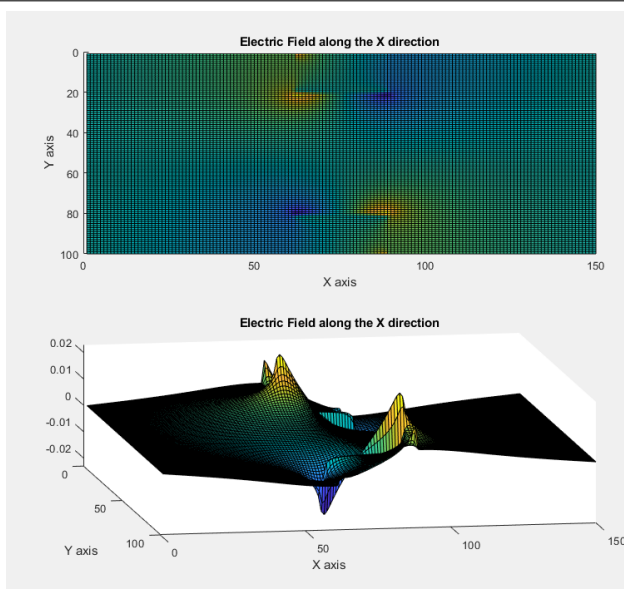


Figure 6: Electric Field along X

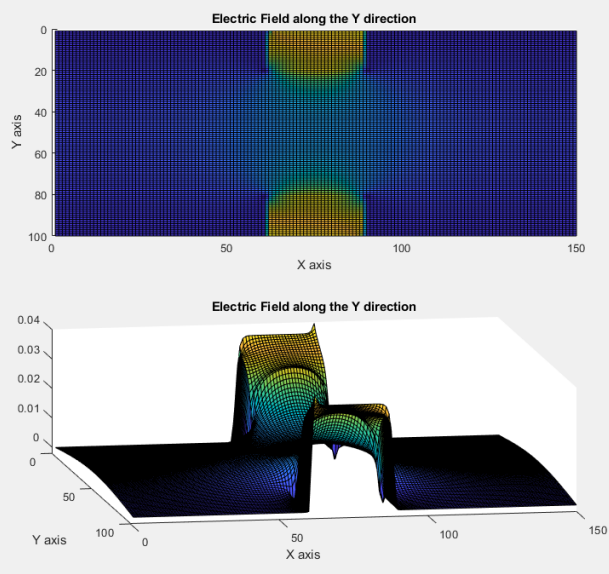


Figure 7: Electric Field along Y

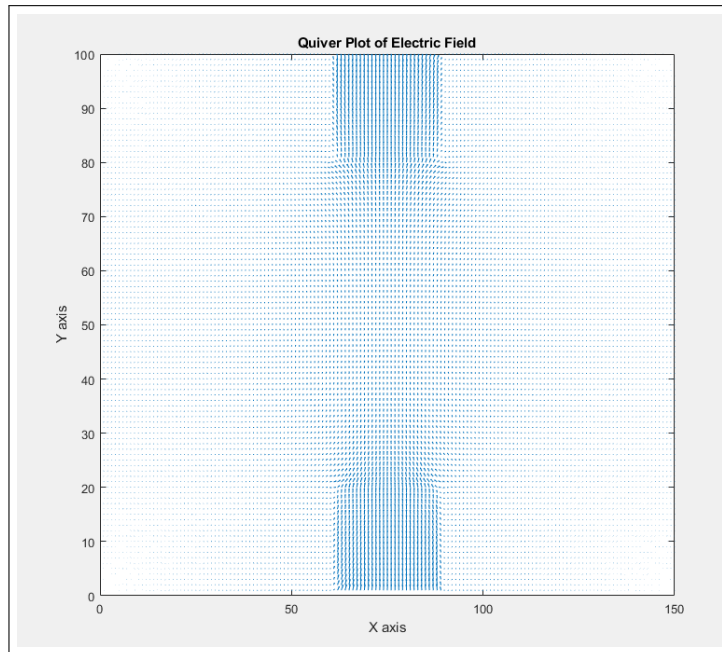


Figure 8: Electric Field

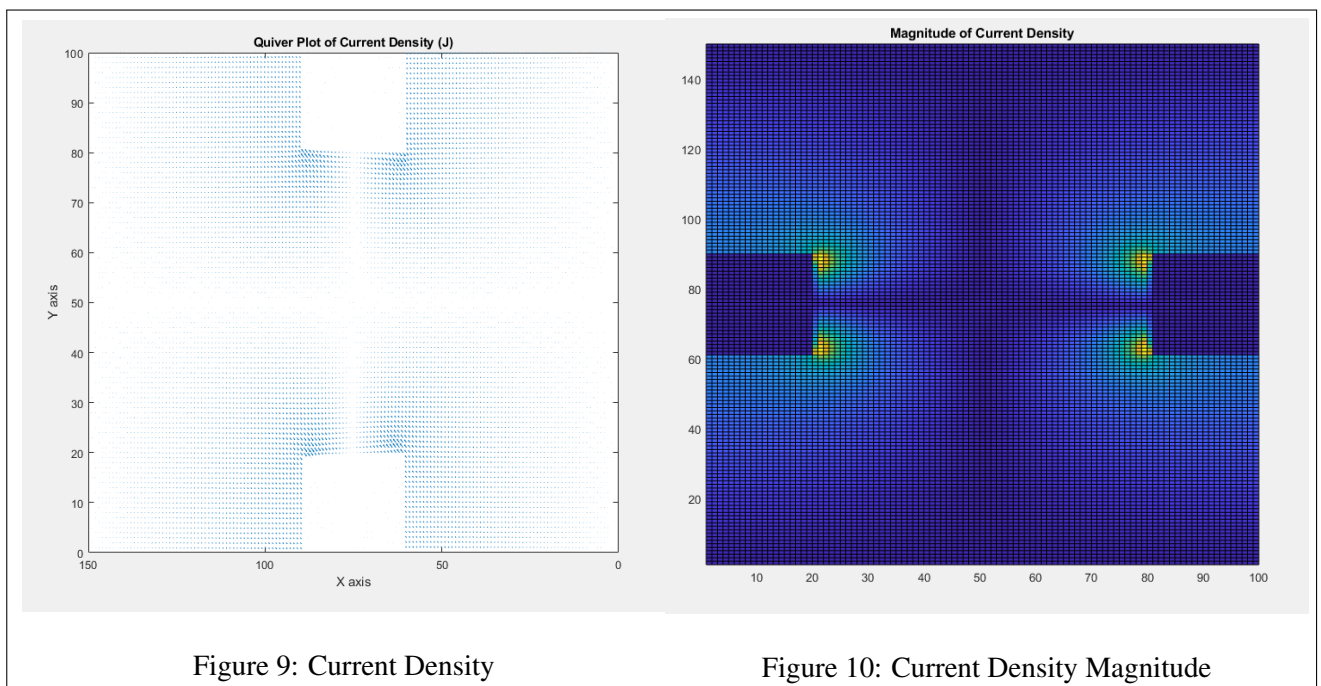


Figure 9: Current Density

Figure 10: Current Density Magnitude

2.1 Narrowing of the Bottle-neck

The bottle neck is narrowed.

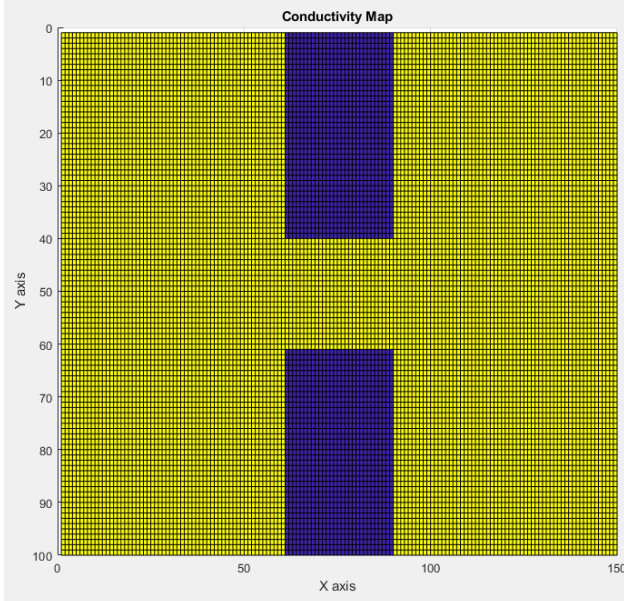


Figure 11: Conduction Map

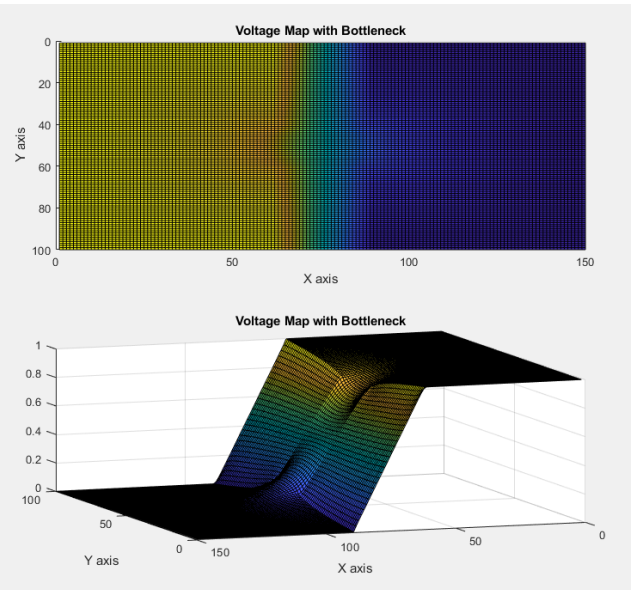


Figure 12: Voltage Map

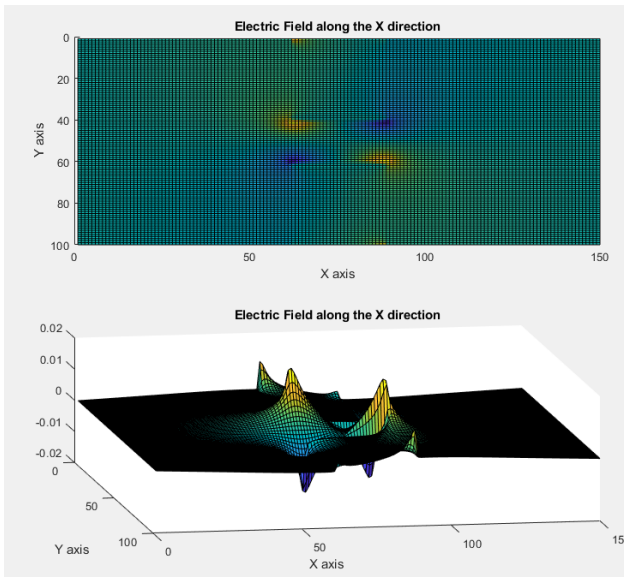


Figure 13: Electric Field along X

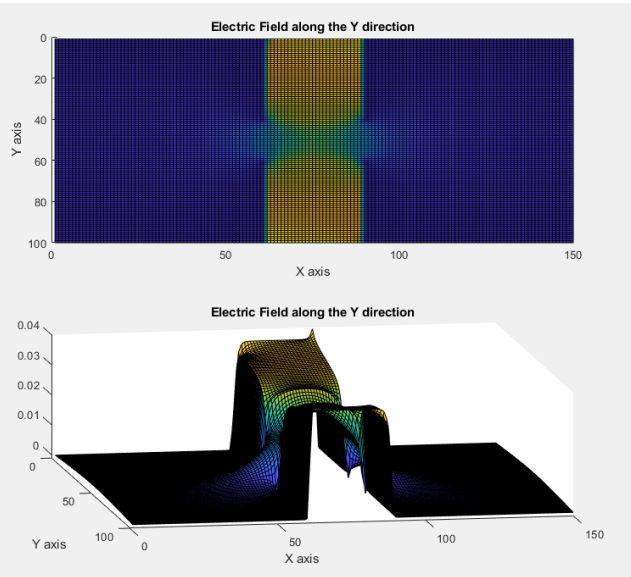


Figure 14: Electric Field along Y

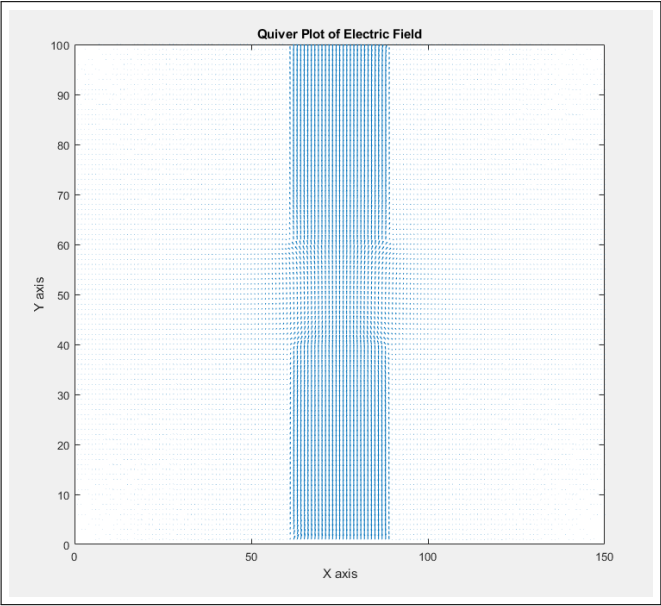


Figure 15: Electric Field

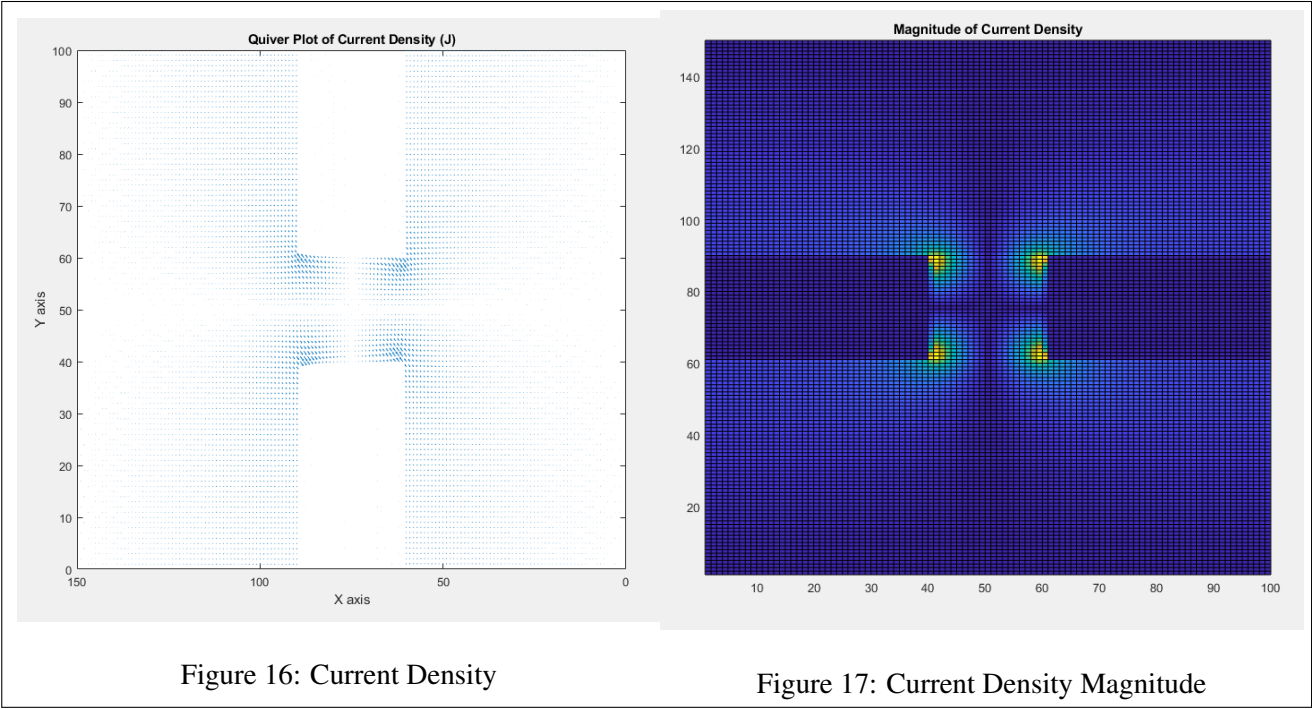


Figure 16: Current Density

Figure 17: Current Density Magnitude

2.2 Different Delta value

The σ value is kept at 1 outside the box but is change to $\sigma = 10^{+2}$ inside the box.

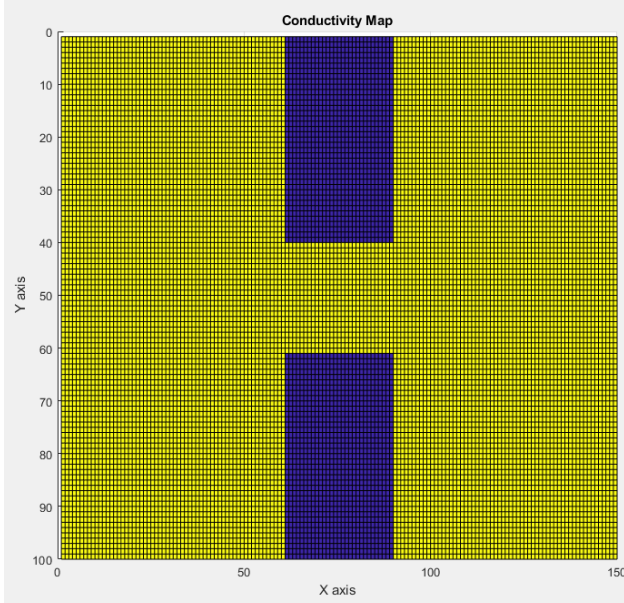


Figure 18: Conduction Map

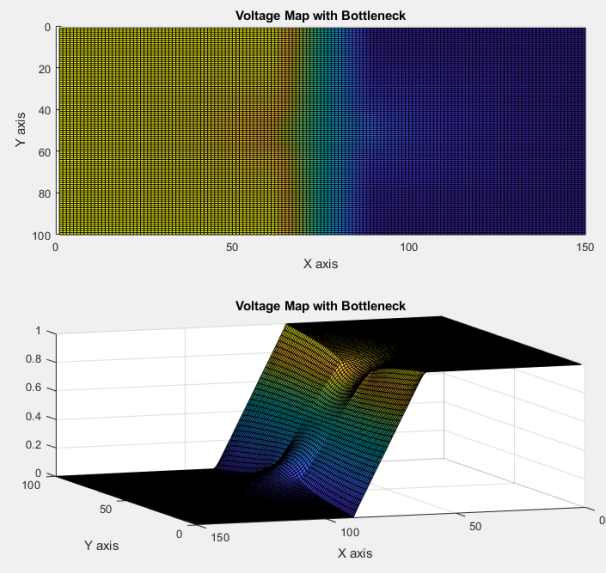


Figure 19: Voltage Map

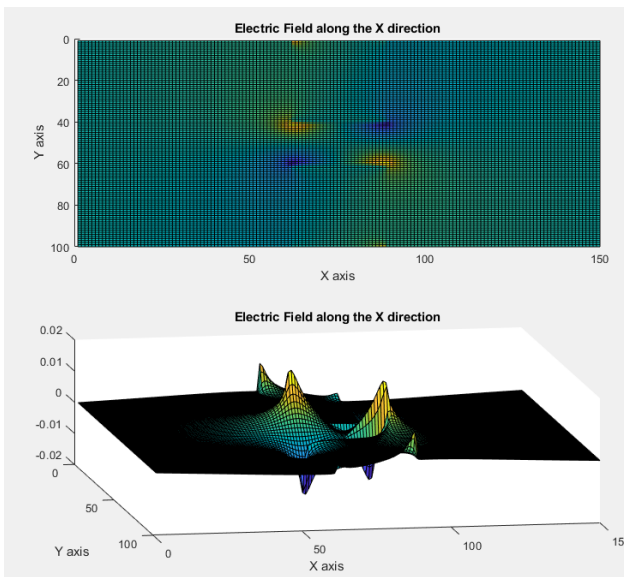


Figure 20: Electric Field along X

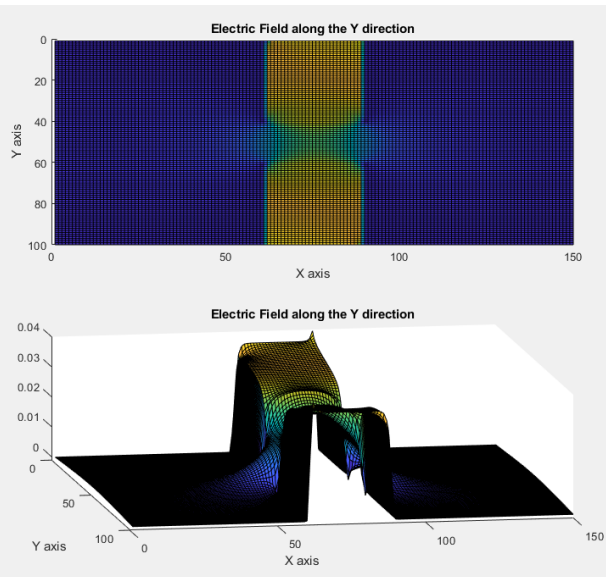


Figure 21: Electric Field along Y

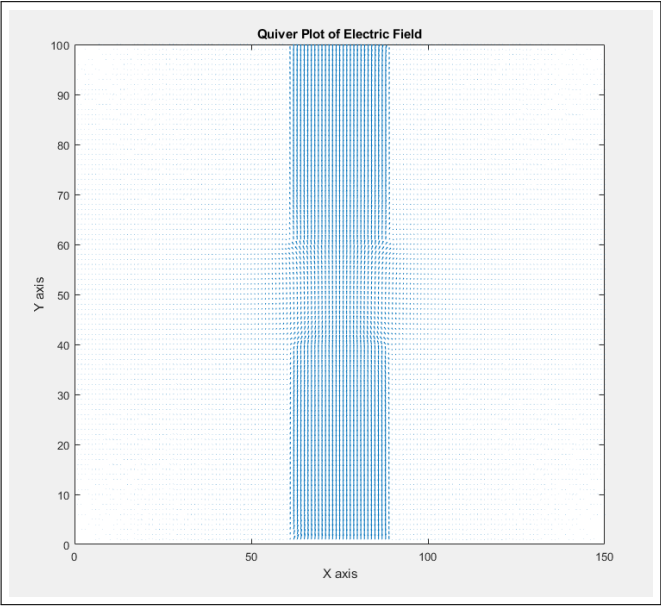


Figure 22: Electric Field

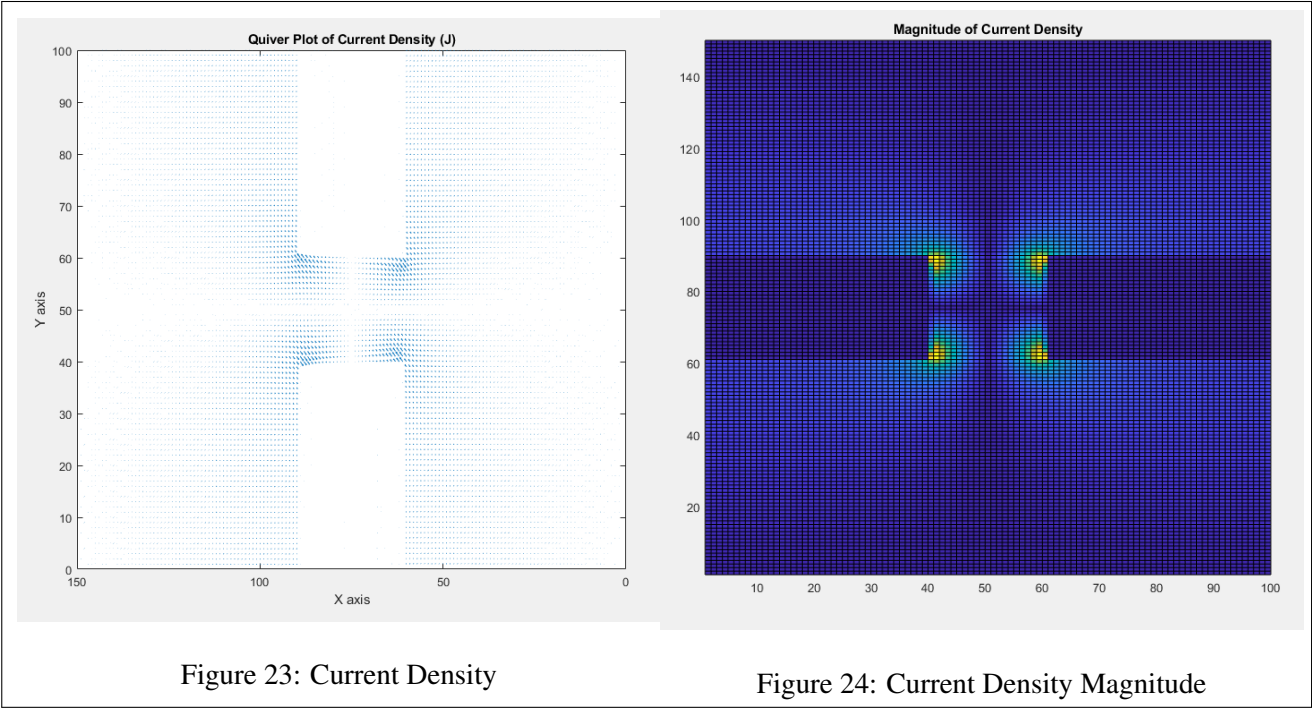


Figure 23: Current Density

Figure 24: Current Density Magnitude

3 Appendix

3.1 Main Script

```

1  % clear all
2  close all
3  clear
4  clearvars -GLOBAL
5  clc
6  format shorte
7
8  set(0, 'DefaultFigureWindowStyle', 'docked')
9
10 global L W V0
11
12 L = 3;
13 W = 2;
14 V0 = 1;
15
16 %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
17 %% Variables to be changed below %%
18 %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
19
20 nx = 150;                % Number of X elements
21 ny = 100;                % Number of Y elements
22 Iteration = 100;
23
24 %=====
25 %===== Part 1A Ramp =====
26 %=====
27 Part1a_Ramp(nx,ny);
28
29 %=====
30 %===== Part 1B Saddle =====
31 %=====
32 Part1b_Saddle(nx,ny,Iteration);
33
34
35 %=====
36 %===== Part 2A =====
37 %=====
38
39 Part2a(nx,ny);
40
41 Part2c_Narrowing(nx,ny);
42
43 Part2d_DiffSigma(nx,ny);

```

3.2 Function for Part 1 Ramp

```

1 function [outputArg1,outputArg2] = Part1a_Ramp(nx,ny)
2
3 solutMat = zeros(nx,ny);
4 solutVect = zeros(nx*ny,1);
5 G = sparse(nx*ny, nx*ny);
6
7 for i = 1:nx
8     for j = 1:ny
9
10         n = j+(i-1)*ny;
11
12         if i == 1                %Left side
13             G(:,n) = 0;
14             G(n,n) = 1;
15             solutVect(n) = 1;
16
17         elseif i == nx           % Right size
18             G(:,n) = 0;
19             G(n,n) = 1;
20
21         elseif j == 1            %Bottom y-axis
22             G(n,n) = -3;
23             G(n,n+ny) = 1;
24             G(n,n-ny) = 1;
25             G(n,n+1) = 1;
26
27         elseif j == ny           %Top y-axis
28             G(n,n) = -3;
29             G(n,n+ny) = 1;
30             G(n,n-ny) = 1;
31             G(n,n-1) = 1;
32
33         else %Handle standard middle cases
34
35             G(n,n) = -4;
36             G(n,n+ny) = 1;
37             G(n,n-ny) = 1;
38             G(n,n+1) = 1;
39             G(n,n-1) = 1;
40         end
41     end
42 end
43
44 solutVect = G\solutVect;
45
46 for i = 1:nx
47     for j = 1:ny
48         n = j+(i-1)*ny;
49         solutMat(i,j) = solutVect(n);
50     end

```



```
51 end
52
53 figure(1)
54 surf(solutMat, 'edgecolor', 'none')
55 colorbar;
56 title('1D Ramp');
57 ylabel('Length');
58 xlabel('Width');
59 zlabel('V (Volts)');
60 view(-100,20)
61
62 end
```

3.3 Function for Part 1 Saddle

```

1 function [outputArg1,outputArg2] = Part1b_Saddle(nx,ny,Iteration)
2 global L W V0
3
4 save = zeros(nx,ny);
5 solution = zeros(nx*ny,1);
6 G = sparse(nx*ny, nx*ny);
7
8 for x = 1:nx
9     for y = 1:ny
10         n = y + (x-1)*ny;
11
12         if x == 1
13             G(n, : ) = 0;
14             G(n, n) = 1;
15             solution(n) = 1;
16         elseif x == nx
17             G(n, : ) = 0;
18             G(n, n) = 1;
19             solution(n) = 1;
20         elseif y == 1
21             G(n, : ) = 0;
22             G(n, n) = 1;
23         elseif y == ny
24             G(n, : ) = 0;
25             G(n, n) = 1;
26         else
27             G(n, n) = -4;
28             G(n, n+1) = 1;
29             G(n, n-1) = 1;
30             G(n, n+ny) = 1;
31             G(n, n-ny) = 1;
32         end
33     end
34 end
35
36 solution = G\solution;
37
38 for i = 1:nx
39     for j = 1:ny
40         n = j+(i-1)*ny;
41         save(i,j) = solution(n);
42     end
43 end
44 figure(2)
45 surf(save, 'edgecolor', 'none')
46 colorbar;
47 title('2D saddle simulation');
48 ylabel('Length');
49 xlabel('Width');
50 zlabel('V (Volts)');

```

```
51 view(-120,20)
52
53 % Analytical solution
54 A_Solve = zeros(ny, nx);
55 x2 = repmat(linspace(-L/2,L/2,nx),ny,1);
56 y2 = repmat(linspace(0,W,ny),nx,1)';
57
58 solution = reshape(solution,[],ny)';
59 for i=1:Iteration
60     n = 2*i - 1;
61     A_Solve = A_Solve + 1./n.*cosh(n.*pi.*x2./W) ...
62         ./cosh(n.*pi.*(L./2)./W).*sin(n.*pi.*y2./W);
63 end
64
65 A_Solve = A_Solve.*4.*V0./pi;
66
67 figure(3);
68 surf(linspace(0,L,nx),linspace(0,W,ny),A_Solve);
69 colorbar;
70 title(sprintf('Analytical for %d iterations', Iteration));
71 ylabel('Length');
72 xlabel('Width');
73 zlabel('V (Volts)');
74 view(-120,20)
75
76 end
```

3.4 Part 2 Current flow through a bottle neck

```

1 function [outputArg1,outputArg2] = Part2c_Narrowing(nx,ny)
2
3 global L W V0
4
5 save = zeros(nx,ny);
6 solution = zeros(nx*ny,1);
7 G = sparse(nx*ny, nx*ny);
8
9 %Question 2
10
11 G = sparse((nx *ny), (nx*ny));
12 V = zeros(1,(nx *ny));
13 Vo = 1;
14
15 conductivity1 = 1; %outside
16 conductivity2 = 1e-2; %inside
17
18 box1 = [(nx* 2/5), (nx* 3/5), ny , (ny *4/5)]; %lower box
19 box2 = [(nx* 2/5), (nx* 3/5),0, (ny *1/5)]; %top box
20
21 bottleNeckVec = zeros(1,10);
22 currentVec = zeros(1,10);
23
24 % conductivity map
25 map = ones(nx,ny);
26
27 for i = 1:nx
28     for j = 1:ny
29         if (i < box1(2) && i >box1(1) && ((j < box2(4)) || (j > box1(4))))
30             map(i,j) = conductivity2;
31         end
32     end
33 end
34 % plot the conduction map
35 figure(4)
36 surf(map);
37 title('Conductivity Map');
38 xlabel('Y axis');
39 ylabel('X axis');
40 view(90,90)
41
42 %Matrix G
43 for i = 1:nx
44
45     for j = 1: ny
46         n = j+ (i-1) *ny;
47
48         if (i == 1)
49             G(n,:) = 0;
50

```



```

51         G(n,n) = 1;
52
53         V(1,n) = 1;
54
55         elseif (i == nx)
56             G(n,:) = 0;
57
58             G(n,n) = 1;
59
60
61         elseif ((i > 1) && (i < nx) && (j==1))
62
63             G(n, n+ny) = map(i+1,j);
64
65             G(n,n) = -(map(i,j+1)+map(i-1,j)+map(i+1,j));
66
67             G(n,n-ny) = map(i-1,j);
68
69             G(n,n-1) = map(i,j+1);
70
71         elseif ((j == ny) && (i < nx) && (i >1))
72             G(n, n+ny) = map(i+1,j);
73
74             G(n,n) = -(map(i-1,j)+map(i+1,j)+map(i,j-1));
75
76             G(n,n-ny) = map(i-1,j);
77
78             G(n,n+1) = map(i,j-1);
79         else
80             G(n, n+ny) = map(i+1,j);
81
82             G(n,n) = -(map(i-1,j)+map(i+1,j)+map(i,j+1)+map(i,j-1));
83
84             G(n,n-ny) = map(i-1,j);
85
86             G(n,n-1) = map(i,j+1);
87
88             G(n,n+1) = map(i,j-1);
89         end
90     end
91 end
92
93 solution1 = G\V';
94
95 Volt = zeros(nx,ny);
96
97 for i =1:nx
98     for j = 1:ny
99         n = j+(i-1) * ny;
100         Volt(i,j) = solution1(n);
101     end

```

```
102 end
103
104 % Plot of Voltage Map
105 figure(5);
106 subplot(2,1,1);
107 surf(Volt);
108 title('Voltage Map with Bottleneck');
109 xlabel('Y axis');
110 ylabel('X axis');
111 view(90,90)
112
113 subplot(2,1,2);
114 surf(Volt);
115 title('Voltage Map with Bottleneck');
116 xlabel('Y axis');
117 ylabel('X axis');
118 view(-110,20)
119
120 % Electric Field plot
121 [Efx,Efy] = gradient(Volt);
122 Efx = -Efx;
123 Efy = -Efy;
124
125 figure(6)
126 quiver(Efx',Efy');
127 title('Quiver Plot of Electric Field')
128 xlabel('X axis');
129 ylabel('Y axis');
130 axis([0 nx 0 ny]);
131
132 % Electric Field Y
133 figure(7)
134 subplot(2,1,1);
135 title('Electric Field along the Y direction')
136 surface(Efy)
137 xlabel('Y axis');
138 ylabel('X axis');
139 axis([0 ny 0 nx]);
140 view(90,90)
141
142 subplot(2,1,2);
143 title('Electric Field along the Y direction')
144 surface(Efy)
145 xlabel('Y axis');
146 ylabel('X axis');
147 axis([0 ny 0 nx]);
148 view(-280,30)
149
150 figure(8)
151 subplot(2,1,1);
152 title('Electric Field along the X direction')
```

```
153 surface(Efx)
154 xlabel('Y axis');
155 ylabel('X axis');
156 axis([0 ny 0 nx]);
157 view(90,90)
158
159 subplot(2,1,2);
160 title('Electric Field along the X direction')
161 surface(Efx)
162 xlabel('Y axis');
163 ylabel('X axis');
164 axis([0 ny 0 nx]);
165 view(-280,30)
166
167
168 %plotting current density
169 Jx = Efx .* map;
170 Jy = Efx .*map;
171
172 figure(9)
173 quiver(Jx,Jy);
174 title('Quiver Plot of Current Density (J)')
175 xlabel('Y axis');
176 ylabel('X axis');
177 axis([0 ny 0 nx]);
178 view(270,90)
179
180 % Magnitude of current density
181 magnitude = ((Jx .^2) + (Jy .^2)) .^0.5;
182
183 figure(10)
184 title('Magnitude of Current Density')
185 surface(magnitude)
186 axis tight
187
188 currentflow = sum(Jx(nx/2,:));
189
190 end
```

3.5 Part 2 Narrowing bottle neck

```

1 function [outputArg1,outputArg2] = Part2c_Narrowing(nx,ny)
2
3 global L W V0
4
5 save = zeros(nx,ny);
6 solution = zeros(nx*ny,1);
7 G = sparse(nx*ny, nx*ny);
8
9 %Question 2
10
11 G = sparse((nx *ny), (nx*ny));
12 V = zeros(1,(nx *ny));
13 Vo = 1;
14
15 conductivity1 = 1; %outside
16 conductivity2 = 1e-2; %inside
17
18 box1 = [(nx* 2/5), (nx* 3/5), ny , (ny *3/5)]; %lower box
19 box2 = [(nx* 2/5), (nx* 3/5),0, (ny *2/5)]; %top box
20
21 bottleNeckVec = zeros(1,10);
22 currentVec = zeros(1,10);
23
24 % conductivity map
25 map = ones(nx,ny);
26
27 for i = 1:nx
28     for j = 1:ny
29         if (i < box1(2) && i >box1(1) && ((j < box2(4)) || (j > box1(4))))
30             map(i,j) = conductivity2;
31         end
32     end
33 end
34 % plot the conduction map
35 figure(11)
36 surf(map);
37 title('Conductivity Map');
38 xlabel('Y axis');
39 ylabel('X axis');
40 view(90,90)
41
42 %Matrix G
43 for i = 1:nx
44
45     for j = 1: ny
46         n = j+ (i-1) *ny;
47
48         if (i == 1)
49             G(n,:) = 0;
50

```



```

51         G(n,n) = 1;
52
53         V(1,n) = 1;
54
55         elseif (i == nx)
56             G(n,:) = 0;
57
58             G(n,n) = 1;
59
60
61         elseif ((i > 1) && (i < nx) && (j==1))
62
63             G(n, n+ny) = map(i+1,j);
64
65             G(n,n) = -(map(i,j+1)+map(i-1,j)+map(i+1,j));
66
67             G(n,n-ny) = map(i-1,j);
68
69             G(n,n-1) = map(i,j+1);
70
71         elseif ((j == ny) && (i < nx) && (i >1))
72             G(n, n+ny) = map(i+1,j);
73
74             G(n,n) = -(map(i-1,j)+map(i+1,j)+map(i,j-1));
75
76             G(n,n-ny) = map(i-1,j);
77
78             G(n,n+1) = map(i,j-1);
79         else
80             G(n, n+ny) = map(i+1,j);
81
82             G(n,n) = -(map(i-1,j)+map(i+1,j)+map(i,j+1)+map(i,j-1));
83
84             G(n,n-ny) = map(i-1,j);
85
86             G(n,n-1) = map(i,j+1);
87
88             G(n,n+1) = map(i,j-1);
89         end
90     end
91 end
92
93 solution1 = G\V';
94
95 Volt = zeros(nx,ny);
96
97 for i =1:nx
98     for j = 1:ny
99         n = j+(i-1) * ny;
100         Volt(i,j) = solution1(n);
101     end

```

```
102 end
103
104 % Plot of Voltage Map
105 figure(12)
106 subplot(2,1,1);
107 surf(Volt);
108 title('Voltage Map with Bottleneck');
109 xlabel('Y axis');
110 ylabel('X axis');
111 view(90,90)
112
113 subplot(2,1,2);
114 surf(Volt);
115 title('Voltage Map with Bottleneck');
116 xlabel('Y axis');
117 ylabel('X axis');
118 view(-110,20)
119
120 % Electric Field plot
121 [Efx,Efy] = gradient(Volt);
122 Efx = -Efx;
123 Efy = -Efy;
124
125 figure(13)
126 quiver(Efx',Efy');
127 title('Quiver Plot of Electric Field')
128 xlabel('X axis');
129 ylabel('Y axis');
130 axis([0 nx 0 ny]);
131
132 % Electric Field Y
133 figure(14)
134 subplot(2,1,1);
135 title('Electric Field along the Y direction')
136 surface(Efy)
137 xlabel('Y axis');
138 ylabel('X axis');
139 axis([0 ny 0 nx]);
140 view(90,90)
141
142 subplot(2,1,2);
143 title('Electric Field along the Y direction')
144 surface(Efy)
145 xlabel('Y axis');
146 ylabel('X axis');
147 axis([0 ny 0 nx]);
148 view(-280,30)
149
150 figure(15)
151 subplot(2,1,1);
152 title('Electric Field along the X direction')
```

```
153 surface(Efx)
154 xlabel('Y axis');
155 ylabel('X axis');
156 axis([0 ny 0 nx]);
157 view(90,90)
158
159 subplot(2,1,2);
160 title('Electric Field along the X direction')
161 surface(Efx)
162 xlabel('Y axis');
163 ylabel('X axis');
164 axis([0 ny 0 nx]);
165 view(-280,30)
166
167
168 %plotting current density
169 Jx = Efx .* map;
170 Jy = Efx .*map;
171
172 figure(16)
173 quiver(Jx,Jy);
174 title('Quiver Plot of Current Density (J)')
175 xlabel('Y axis');
176 ylabel('X axis');
177 axis([0 ny 0 nx]);
178 view(270,90)
179
180 % Magnitude of current density
181 magnitude = ((Jx .^2) + (Jy .^2)) .^0.5;
182
183 figure(17)
184 title('Magnitude of Current Density')
185 surface(magnitude)
186 axis tight
187
188 currentflow = sum(Jx(nx/2,:));
189
190 end
```

3.6 Part 2 Different Delta value

```

1 function [outputArg1,outputArg2] = Part2d_DiffSigma(nx,ny)
2
3 global L W V0
4
5 save = zeros(nx,ny);
6 solution = zeros(nx*ny,1);
7 G = sparse(nx*ny, nx*ny);
8
9 %Question 2
10
11 G = sparse((nx *ny), (nx*ny));
12 V = zeros(1,(nx *ny));
13 Vo = 1;
14
15 conductivity1 = 1; %outside
16 conductivity2 = 1e2; %inside
17
18 box1 = [(nx* 2/5), (nx* 3/5), ny , (ny *3/5)]; %lower box
19 box2 = [(nx* 2/5), (nx* 3/5),0, (ny *2/5)]; %top box
20
21 bottleNeckVec = zeros(1,10);
22 currentVec = zeros(1,10);
23
24 % conductivity map
25 map = ones(nx,ny);
26
27 for i = 1:nx
28     for j = 1:ny
29         if (i < box1(2) && i >box1(1) && ((j < box2(4)) || (j > box1(4))))
30             map(i,j) = conductivity2;
31         end
32     end
33 end
34 % plot the conduction map
35 figure(18)
36 surf(map);
37 title('Conductivity Map');
38 xlabel('Y axis');
39 ylabel('X axis');
40 view(90,90)
41
42 %Matrix G
43 for i = 1:nx
44
45     for j = 1: ny
46         n = j+ (i-1) *ny;
47
48         if (i == 1)
49             G(n,:) = 0;
50

```



```

51         G(n,n) = 1;
52
53         V(1,n) = 1;
54
55         elseif (i == nx)
56             G(n,:) = 0;
57
58             G(n,n) = 1;
59
60
61         elseif ((i > 1) && (i < nx) && (j==1))
62
63             G(n, n+ny) = map(i+1,j);
64
65             G(n,n) = -(map(i,j+1)+map(i-1,j)+map(i+1,j));
66
67             G(n,n-ny) = map(i-1,j);
68
69             G(n,n-1) = map(i,j+1);
70
71         elseif ((j == ny) && (i < nx) && (i >1))
72             G(n, n+ny) = map(i+1,j);
73
74             G(n,n) = -(map(i-1,j)+map(i+1,j)+map(i,j-1));
75
76             G(n,n-ny) = map(i-1,j);
77
78             G(n,n+1) = map(i,j-1);
79         else
80             G(n, n+ny) = map(i+1,j);
81
82             G(n,n) = -(map(i-1,j)+map(i+1,j)+map(i,j+1)+map(i,j-1));
83
84             G(n,n-ny) = map(i-1,j);
85
86             G(n,n-1) = map(i,j+1);
87
88             G(n,n+1) = map(i,j-1);
89         end
90     end
91 end
92
93 solution1 = G\V';
94
95 Volt = zeros(nx,ny);
96
97 for i =1:nx
98     for j = 1:ny
99         n = j+(i-1) * ny;
100         Volt(i,j) = solution1(n);
101     end

```

```
102 end
103
104 % Plot of Voltage Map
105 figure(19)
106 subplot(2,1,1);
107 surf(Volt);
108 title('Voltage Map with Bottleneck');
109 xlabel('Y axis');
110 ylabel('X axis');
111 view(90,90)
112
113 subplot(2,1,2);
114 surf(Volt);
115 title('Voltage Map with Bottleneck');
116 xlabel('Y axis');
117 ylabel('X axis');
118 view(-110,20)
119
120 % Electric Field plot
121 [Efx,Efy] = gradient(Volt);
122 Efx = -Efx;
123 Efy = -Efy;
124
125 figure(20)
126 quiver(Efx',Efy');
127 title('Quiver Plot of Electric Field')
128 xlabel('X axis');
129 ylabel('Y axis');
130 axis([0 nx 0 ny]);
131
132 % Electric Field Y
133 figure(21)
134 subplot(2,1,1);
135 title('Electric Field along the Y direction')
136 surface(Efy)
137 xlabel('Y axis');
138 ylabel('X axis');
139 axis([0 ny 0 nx]);
140 view(90,90)
141
142 subplot(2,1,2);
143 title('Electric Field along the Y direction')
144 surface(Efy)
145 xlabel('Y axis');
146 ylabel('X axis');
147 axis([0 ny 0 nx]);
148 view(-280,30)
149
150 figure(22)
151 subplot(2,1,1);
152 title('Electric Field along the X direction')
```

```
153 surface(Efx)
154 xlabel('Y axis');
155 ylabel('X axis');
156 axis([0 ny 0 nx]);
157 view(90,90)
158
159 subplot(2,1,2);
160 title('Electric Field along the X direction')
161 surface(Efx)
162 xlabel('Y axis');
163 ylabel('X axis');
164 axis([0 ny 0 nx]);
165 view(-280,30)
166
167
168 %plotting current density
169 Jx = Efx .* map;
170 Jy = Efx .*map;
171
172 figure(23)
173 quiver(Jx,Jy);
174 title('Quiver Plot of Current Density (J)')
175 xlabel('Y axis');
176 ylabel('X axis');
177 axis([0 ny 0 nx]);
178 view(270,90)
179
180 % Magnitude of current density
181 magnitude = ((Jx .^2) + (Jy .^2)) .^0.5;
182
183 figure(24)
184 title('Magnitude of Current Density')
185 surface(magnitude)
186 axis tight
187
188 currentflow = sum(Jx(nx/2,:));
189
190 end
```