Table of Contents

ELEC 4700 Assignment 2 Agberien Stephen

Part 1: Electrostatic Potential in Rectangular Region

Simple 2D Laplace case

Part 1a)

```
W = 20;
L = 30;

v0 = 1;

dx = 1;
dy = 1;
nx = L / dx;
ny = W / dy;

G = sparse(nx*ny, nx*ny);
F = zeros(nx*ny, 1);

% for i = 1:nx
%     for j = 1:ny
%
%         n = i + (j - 1) * nx;
%         nym = i + (j - 2) * nx;
%         nyp = i + j * nx;
%         nxm = i - 1 + (j - 1) * nx;
%         nxp = i + 1 + (j - 1) * nx;
%
%         if i == 1
%             G(n, n) = 1;
%             F(n) = v0;
%         elseif i == nx
%             G(n, n) = 1;
%             F(n) = 0;
%         elseif j == 1
%             G(n, n) = 1;
%         elseif j == ny
%             G(n, n) = 1;
%         else
%             G(n, n) = -4;
```

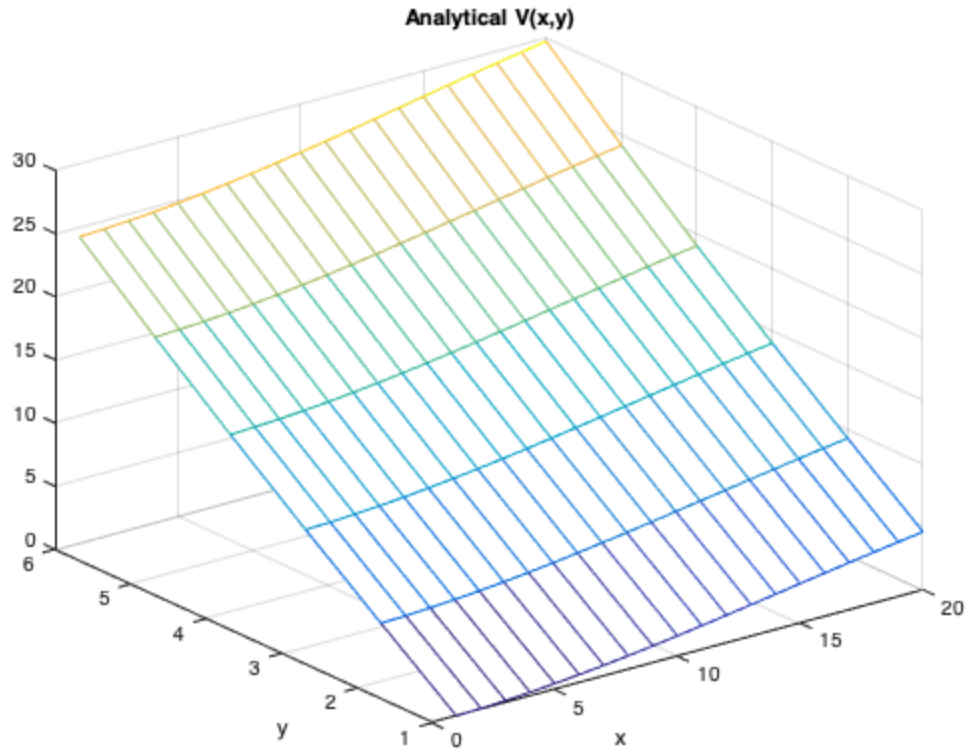
```
%             G(n, nxm) = 1;
%             G(n, nxp) = 1;
%             G(n, nym) = 1;
%             G(n, nyp) = 1;
%             F(n) = 0;
%         end
%     end
% end
```

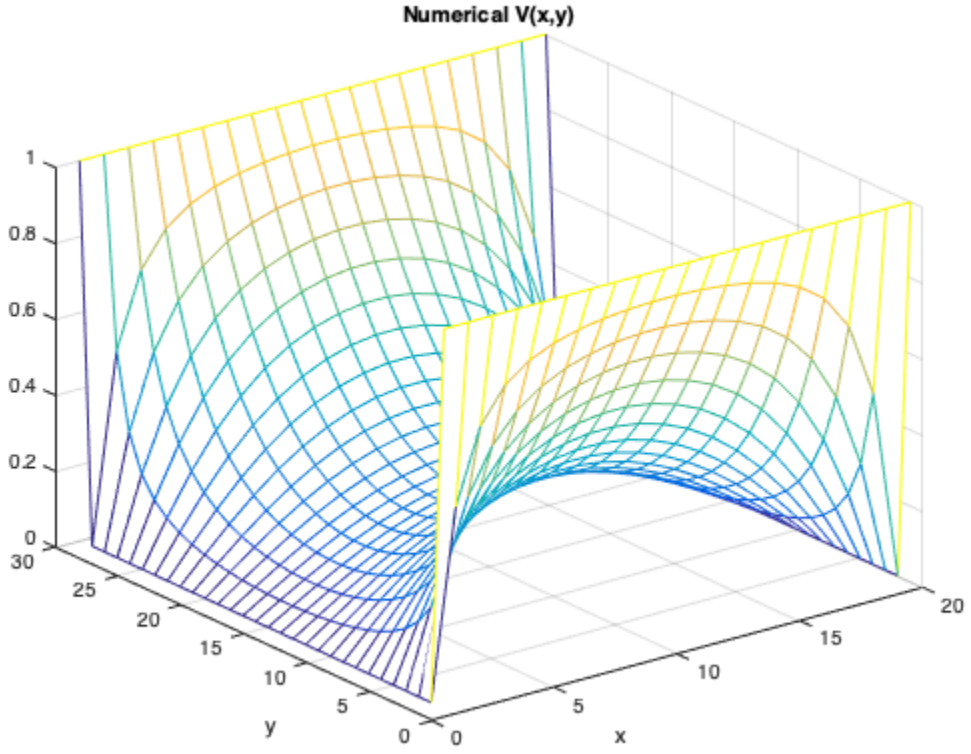
Part 1 b)

same for loop as part A

```
for i = 1:nx
    for j = 1:ny

        n = i + (j - 1) * nx;
        nym = i + (j - 2) * nx;
        nyp = i + j * nx;
        nxm = i - 1 + (j - 1) * nx;
        nxp = i + 1 + (j - 1) * nx;

        if i == 1
            G(n, n) = 1;
            F(n) = v0;
        elseif i == nx
            G(n, n) = 1;
            F(n) = v0;
        elseif j == 1
            G(n, n) = 1;
        elseif j == ny
            G(n, n) = 1;
        else
            G(n, n) = -4;
            G(n, nxm) = 1;
            G(n, nxp) = 1;
            G(n, nym) = 1;
            G(n, nyp) = 1;
            F(n) = 0;
        end
    end
end


a = 30;
b = 20;
sum = 0;

for x = 1:nx
    for y = 1:ny
        for n = 1:2:1000
            sum = sum + 1/n * cosh(n*pi*x/a) * sin(n*pi*y/a) /
 cosh(n*pi*b/a);
```

```
        end
        V2(x, y) = 4/pi * sum;
    end
end

figure(1)
mesh(V2)
title('Analytical V(x,y)')
xlabel('x')
ylabel('y')

V2 = zeros(nx, ny);
V = G\F;

for i = 1:nx
    for j = 1:ny
        n = i + (j - 1)*nx;
        V2(i, j) = V(n);
    end
end

[X, Y] = meshgrid(1:1:ny, 1:1:nx);
figure(2)
mesh(X, Y, V2)
title('Numerical V(x,y)')
xlabel('x')
ylabel('y')
```


Analytical V(x,y)
30
25
20
15
10
5
0
6
5
4
3
2
1
y
0
5
10
15
20
x


Numerical V(x,y)
1
0.8
0.6
0.4
0.2
0
30
25
20
15
10
5
0
y
0
5
10
15
20
x

MESHING AND COMMENTS ON NUMERICAL VS ANALYTICAL

- The analytical solution significantly differs from the numerical solution and seems to be very innaccurate.
- This error can possibly be accounted for in the number of iterations of the sum that were performed.
- The analytical solution requires the sum to be iterated an infinite amount of times.
- Under these specific conditions, the numerical solution is more accurate and shows less error.

Part 2: Current Flow in Rectangular Region

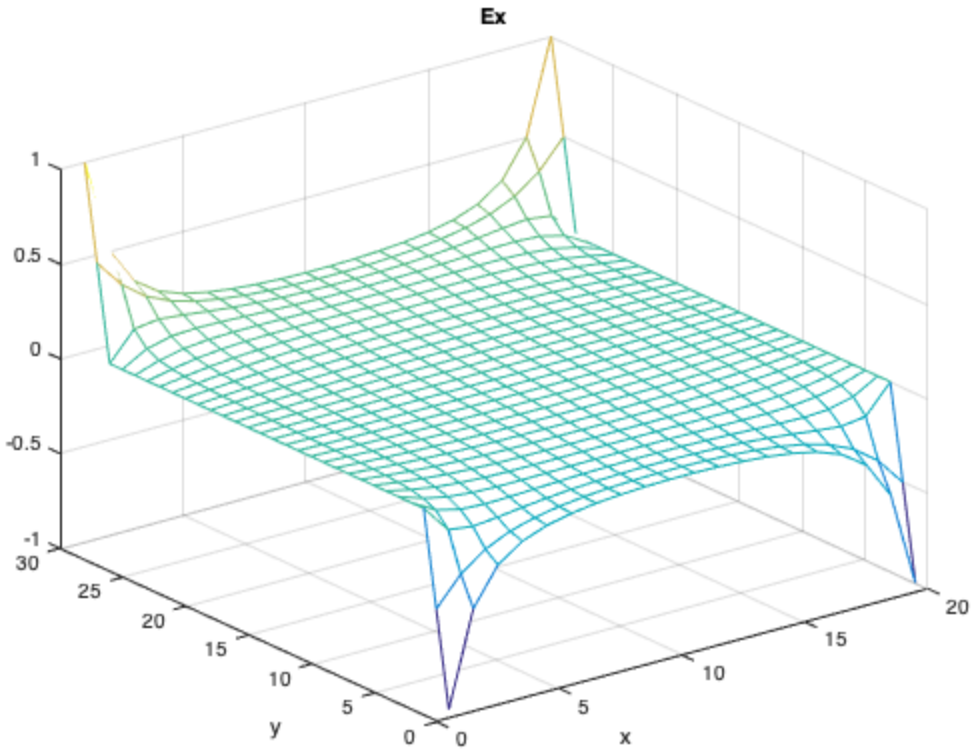
Solving current flow in a rectangular region using Finite Difference

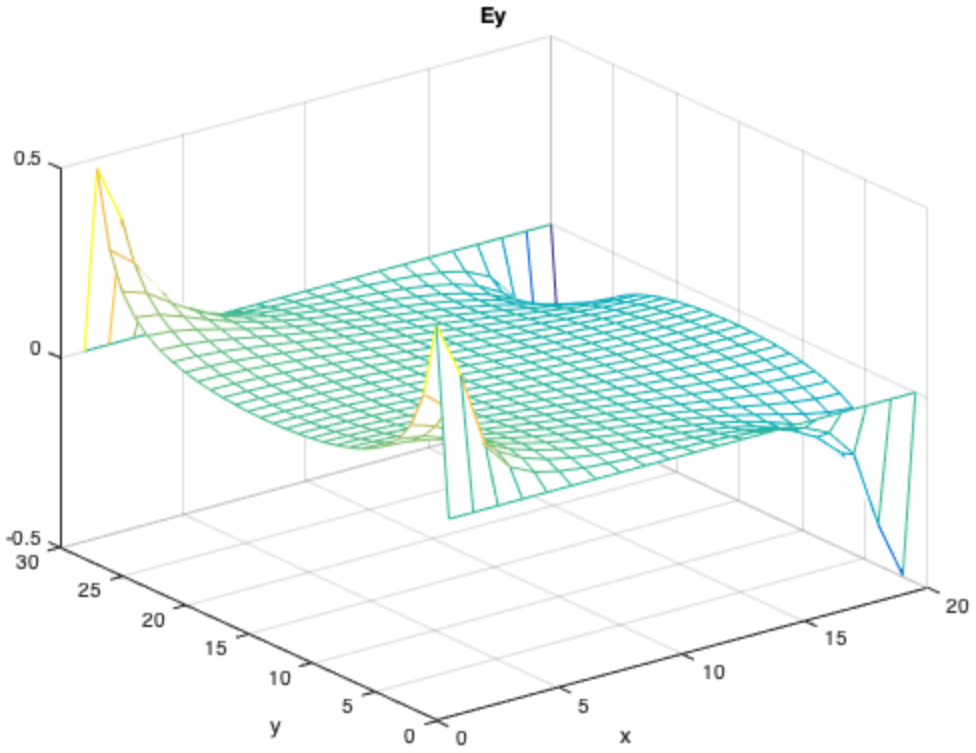
```
W = 20;
L = 30;

v0 = 1;
dx = 1;
dy = 1;
nx = L / dx;
ny = W / dy;

G = sparse(nx*ny, nx*ny);
F = zeros(nx*ny, 1);

for i = 1:nx
    for j = 1:ny

        n = i + (j - 1) * nx;
        nym = i + (j - 2) * nx;
        nyp = i + j * nx;
        nxm = i - 1 + (j - 1) * nx;
        nxp = i + 1 + (j - 1) * nx;

        if i == 1
            G(n, n) = 1;
            F(n) = v0;
        elseif i == nx
            G(n, n) = 1;
            F(n) = v0;
        elseif j == 1
            G(n, n) = 1;
        elseif j == ny
            G(n, n) = 1;
        else
            G(n, n) = -4;
            G(n, nxm) = 1;
            G(n, nxp) = 1;
            G(n, nym) = 1;
            G(n, nyp) = 1;
            F(n) = 0;
```

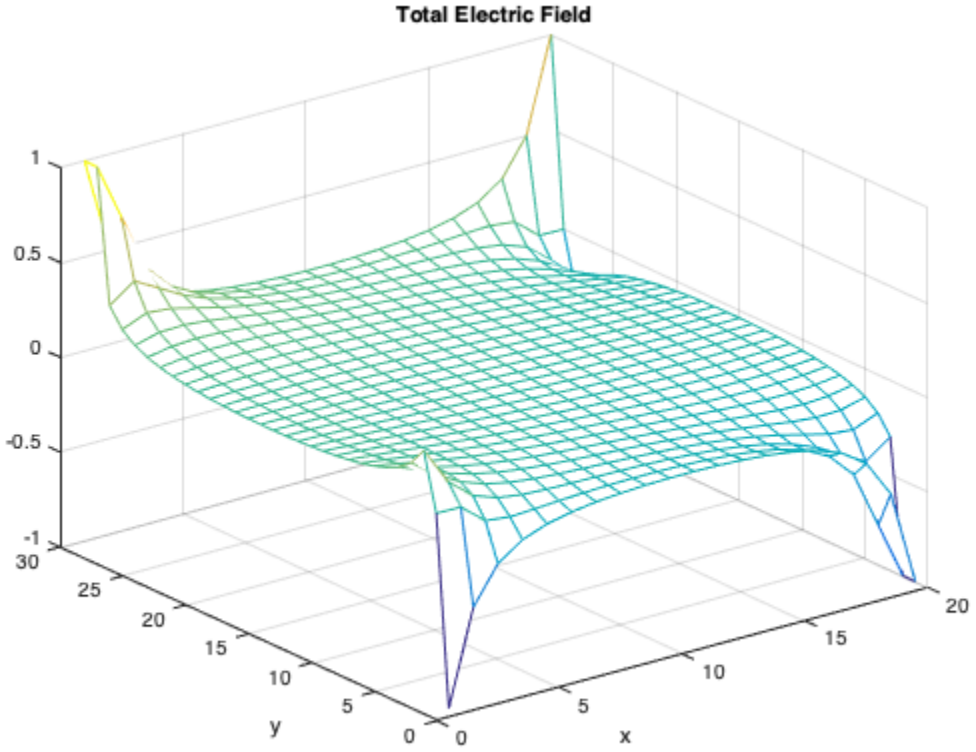
```
        end
    end
end

V2 = zeros(nx, ny);
Ex = zeros(nx, ny);
Ey = zeros(nx, ny);
V = G\F;

for i = 1:nx
    for j = 1:ny
        n = i + (j - 1)*nx;
        V2(i, j) = V(n);
    end
end

for i = 1:nx
    for j = 1:ny
        if i == 1
            Ex(i, j) = V2(i+1, j) - V2(i, j);
        elseif i == nx
            Ex(i, j) = V2(i, j) - V2(i-1, j);
        else
            Ex(i, j) = (V2(i+1, j) - V2(i-1, j))/2.0;
        end

        if j == 1
            Ey(i, j) = V2(i, j+1) - V2(i, j);
        elseif j == ny
            Ey(i, j) = V2(i, j) - V2(i, j-1);
        else
            Ey(i, j) = (V2(i, j+1) - V2(i, j-1))/2.0;
        end
    end
end

figure(3)
mesh(Ex)
title('Ex')
xlabel('x')
ylabel('y')

figure(4)
mesh(Ey)
title('Ey')
xlabel('x')
ylabel('y')

figure(5)
mesh(Ex + Ey)
title('Total Electric Field')
xlabel('x')
ylabel('y')
```

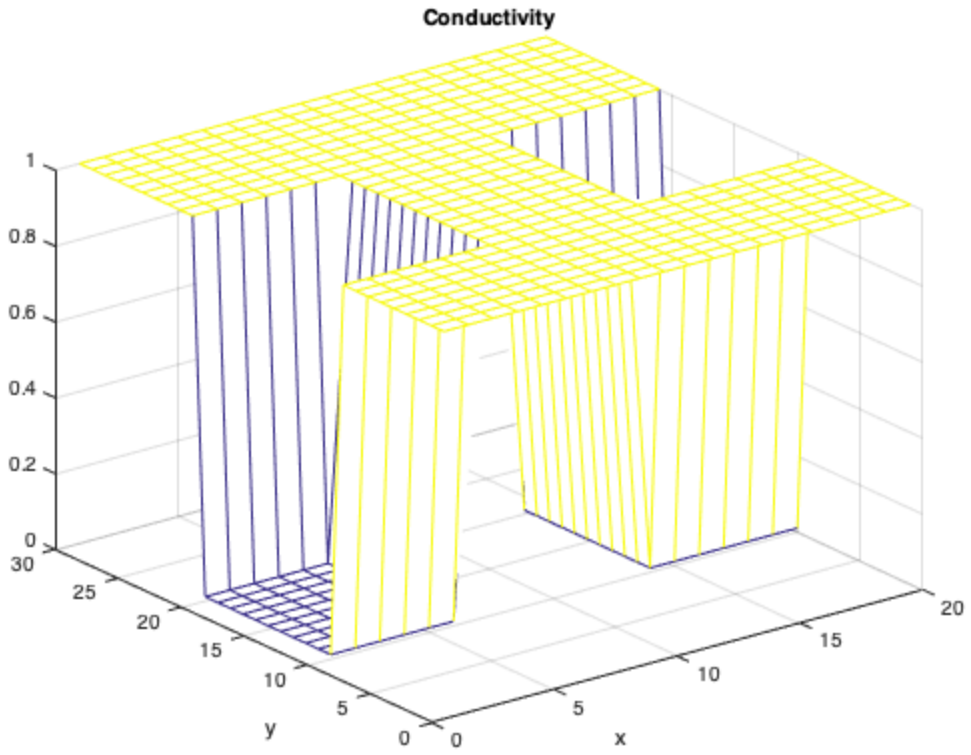
```
sigma = ones(nx, ny);
sigmaLarge = ones(nx, ny);
sigmaSmall = ones(nx, ny);
sigmaLargeBN = ones(nx, ny);
sigmaSmallBN = ones (nx, ny);

for i = 1:nx
    for j = 1:ny
        if j <= (ny/3.0) || j >= (ny*2/3.0)
            if i >= (nx/3.0) && i <= (nx*2/3.0)
                sigma(i,j) = 10^(-2);
                sigmaLarge(i, j) = 100;
                sigmaSmall(i, j) = 10^(-10);
            end
        end
        if j <= (ny/4.0) || j >= (ny*3/4.0)
            if i >= (nx/3.0) && i <= (nx*2/3.0)
                sigmaSmallBN(i,j) = 10^(-2);
            end
        end
        if j <= (ny/2.25) || j >= (ny - ny/2.25)
            if i >= (nx/3.0) && i <= (nx*2/3.0)
                sigmaLargeBN(i,j) = 10^(-2);
            end
        end
    end
end

figure(6)
mesh(sigma)
title('Conductivity')
xlabel('x')
ylabel('y')

E = Ex + Ey;
J = sigma .* E;
figure(7)
mesh(J)
title('Current Density')
xlabel('x')
ylabel('y')

rho = 1./sigma;

I = V2 ./ rho;
figure(8)
mesh(I)
title('Current')
xlabel('x')
ylabel('y')

%Part 2 c)
rhoLargeBN = 1./sigmaLargeBN;
rhoSmallBN = 1./sigmaSmallBN;
```

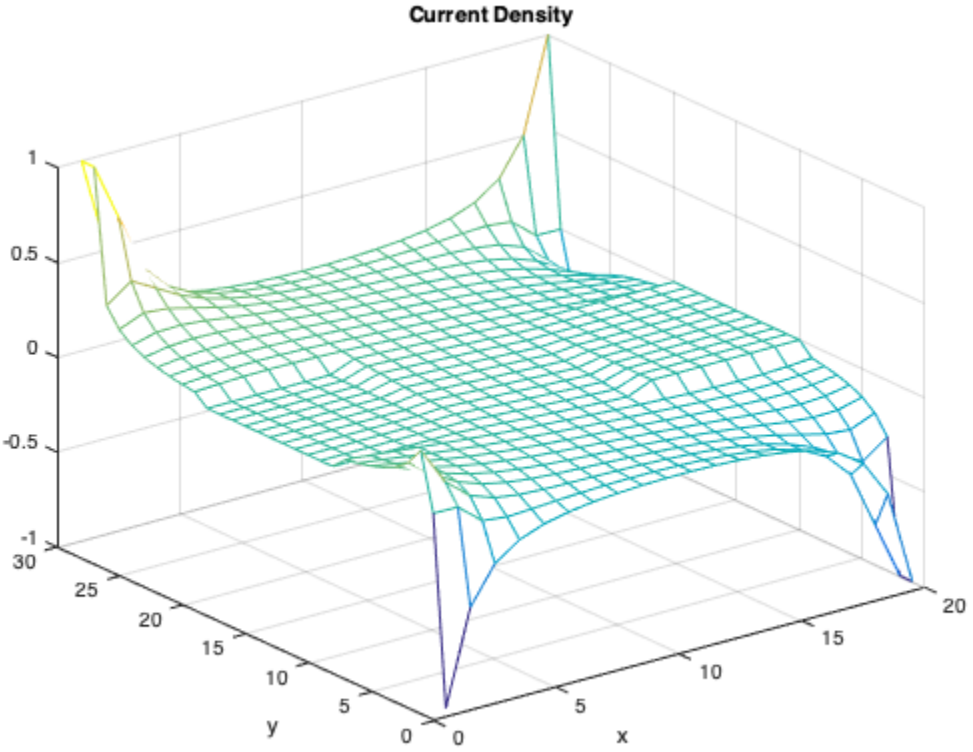
```
figure(9)
subplot(3, 1, 1)
mesh(I)
title('Current vs. Moderate Bottleneck Size')
subplot(3, 1, 2)
mesh(V2 ./ rhoLargeBN)
title('Current vs. Large Bottleneck Size')
subplot(3, 1, 3)
mesh(V2 ./ rhoSmallBN)
title('Current vs. Small Bottleneck Size')

%Part 2 d)
figure(10)
mesh(I)
title('Current vs. Sigma')
```

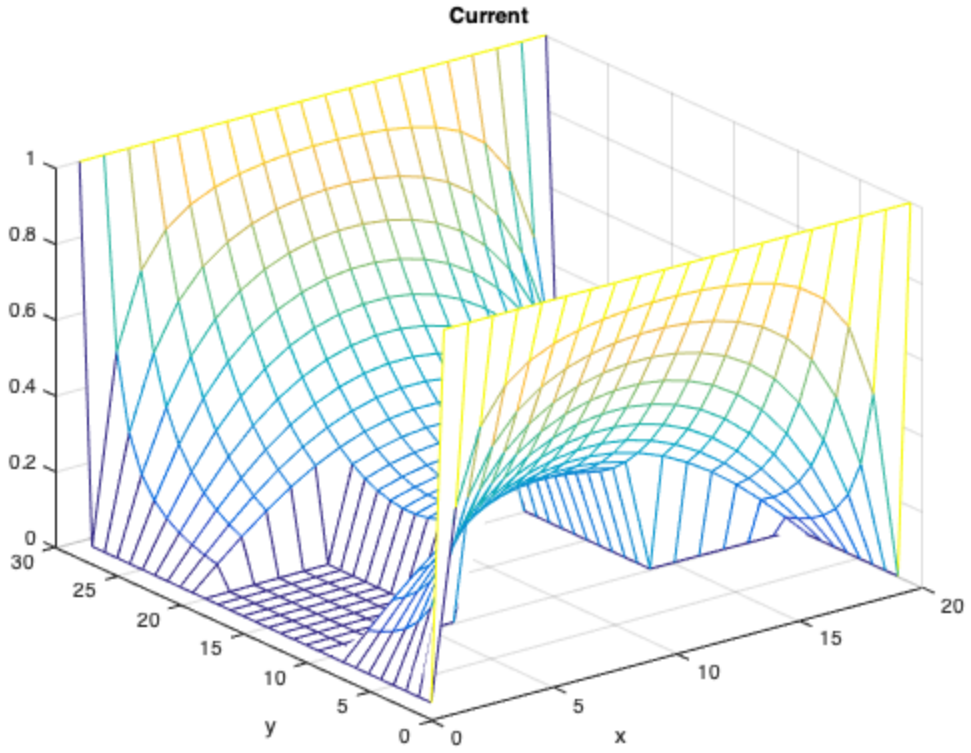


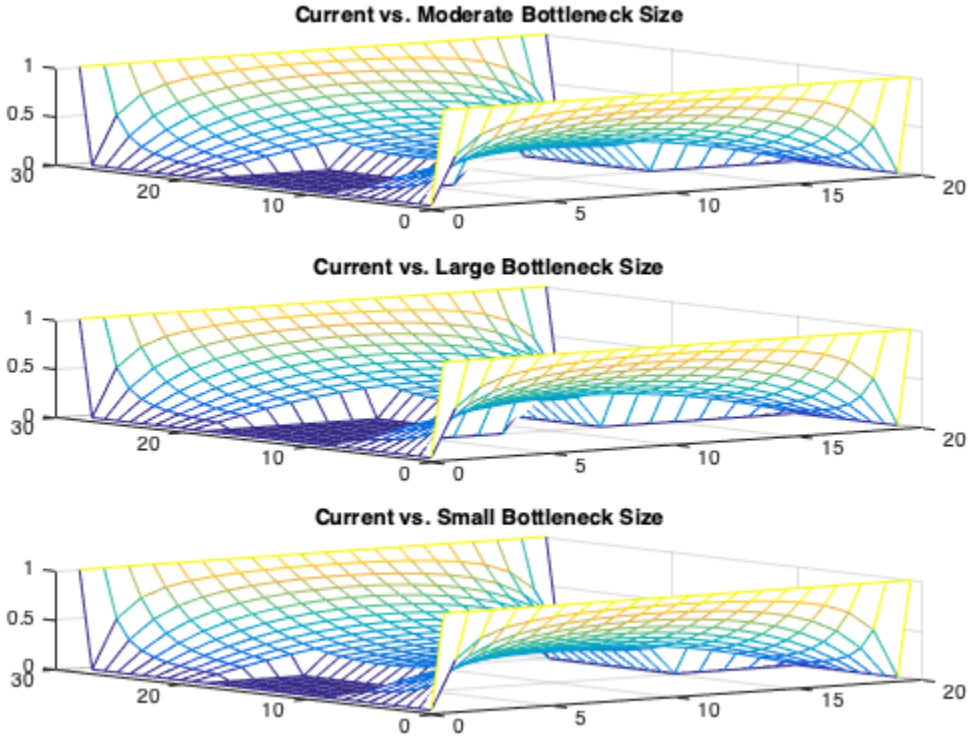

Ey
0.5
0
-0.5
30
25
20
15
10
5
0
y
0
5
10
15
20
x

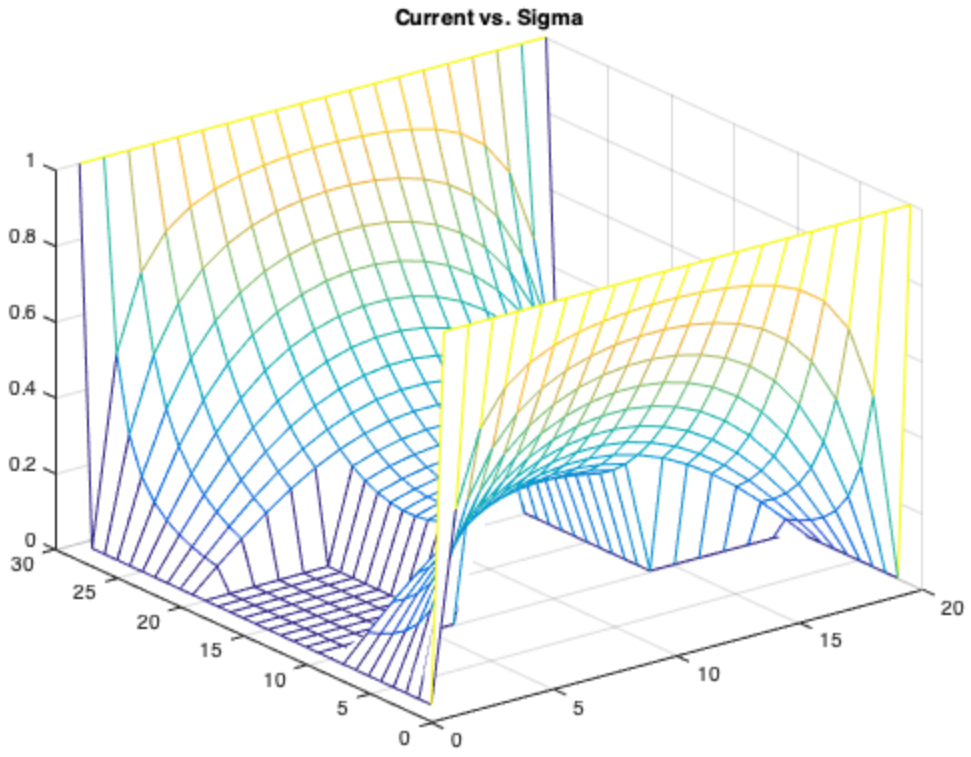

Total Electric Field
1
0.5
0
-0.5
-1
30
25
20
15
10
5
0
y
0
5
10
15
20
x


Conductivity
1
0.8
0.6
0.4
0.2
0
30
25
20
15
10
5
0
y
0
5
10
15
20
x


Current Density
1
0.5
0
-0.5
-1
30
25
20
15
10
5
0
y
0
5
10
15
20
x


Current
1
0.8
0.6
0.4
0.2
0
30
25
20
15
10
5
0
y
0
5
10
15
20
x


Current vs. Moderate Bottleneck Size
Current vs. Large Bottleneck Size
Current vs. Small Bottleneck Size
1
0.5
0
30
20
10
0
0
5
10
15
20



Published with MATLAB® R2015b