

---

## Table of Contents

Question 1a .....	1
Question 1B .....	2
Question 2 A .....	5
Question 2 B .....	8

## Question 1a

```
% select dimension of the box (can be changed)
nx=75; %length of box
ny=50; %width of box

% create the G matrix
g=sparse(nx*ny);
b=zeros(1,nx*ny);

for i = 1:nx
    for j=1:ny
        n=j+(i-1)*ny;
        %left side
        if i==1
            g(n,:)=0;
            g(n,n)=1;
            b(n)=1;
        %right side
        elseif i==nx
            g(n,:)=0;
            g(n,n)=1;

        %bottom
        elseif j==1
            g(n,n)=-3;
            g(n,n+1)=1;
            g(n,n-1)=1;
            g(n,n+ny)=1;
        %top
        elseif j==ny
            g(n,n)=-3;
            g(n,n+1)=1;
            g(n,n-1)=1;
            g(n,n-ny)=1;

        else %bulk node
            g(n,n)=-4;
            g(n,n+1)=1;
            g(n,n-1)=1;
            g(n,n+ny)=1;
            g(n,n-ny)=1;
```

---

```

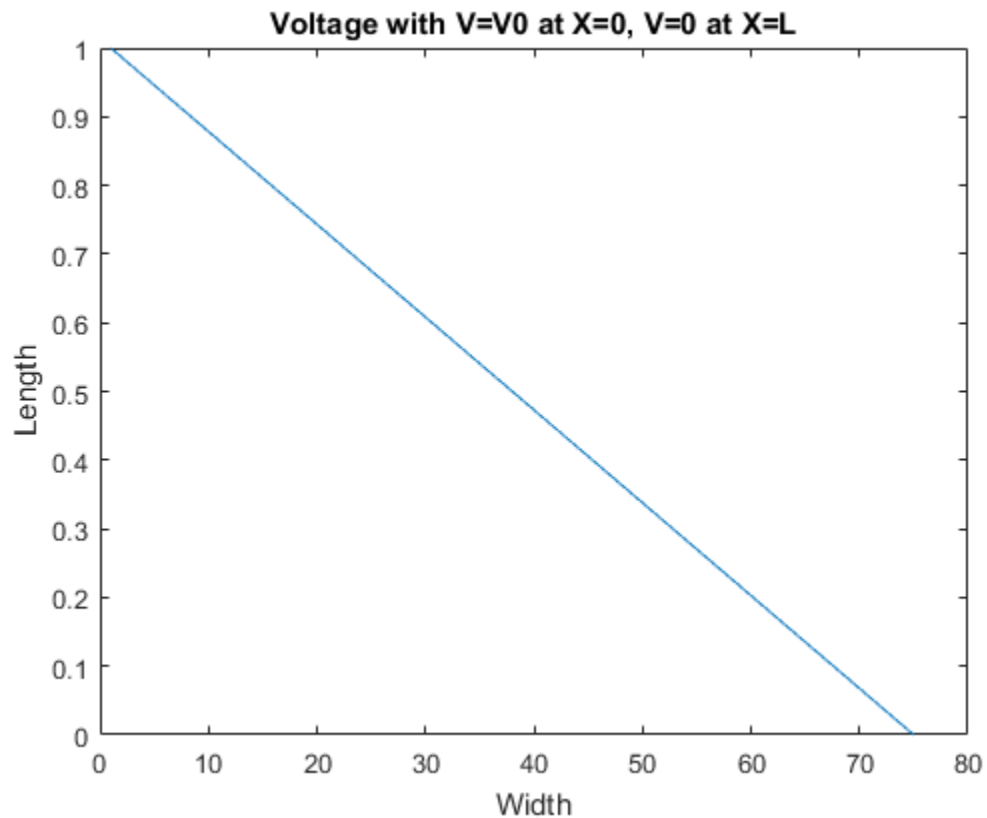
        end
    end
end

E=g\b';

%reconstruct the nx by ny matrix
d=zeros(nx,ny);
for i = 1:nx
    for j=1:ny
        n=j+(i-1)*ny;
        d(i,j)=E(n);
    end
end
%plot the results
plot(d)
xlabel('Width')
ylabel('Length')
title('Voltage with V=V0 at X=0, V=0 at X=L')

% This is the expected result.  Voltage falls off linearly with
distance

```



## Question 1B

```
nx=75; %length
```

---

```

ny=50; %width

g=sparse(nx*ny);
b=zeros(1,nx*ny);

for i = 1:nx
    for j=1:ny
        n=j+(i-1)*ny;
        if i==1
            g(n,:)=0;
            g(n,n)=1;
            b(n)=1;
        elseif i==nx
            g(n,:)=0;
            g(n,n)=1;
            b(n)=1;
        elseif j==1
            g(n,n)=1;

        elseif j==ny
            g(n,n)=1;

        else %bulk node
            g(n,n)=-4;
            g(n,n+1)=1;
            g(n,n-1)=1;
            g(n,n+ny)=1;
            g(n,n-ny)=1;
        end
    end
end

E=g\b';

d=zeros(nx,ny);
for i = 1:nx
    for j=1:ny
        n=j+(i-1)*ny;
        d(i,j)=E(n);

    end
end

figure(1)
surf(d)
xlabel('Width')
ylabel('Length')
title('Calculated Surface plot of V with 1B boundary conditions')
view(-60,37)
%plot analytical solution

y=0:ny;
x=-(nx+1)/2:(nx-1)/2; %theoretical solution occurs from -x to x

```

---

---

```

[vx,vy]=meshgrid(x,y);

f=4/pi*cosh(pi*vx/ny)/cosh(pi*(nx+1)/2/ny).*sin(pi*vy/ny);

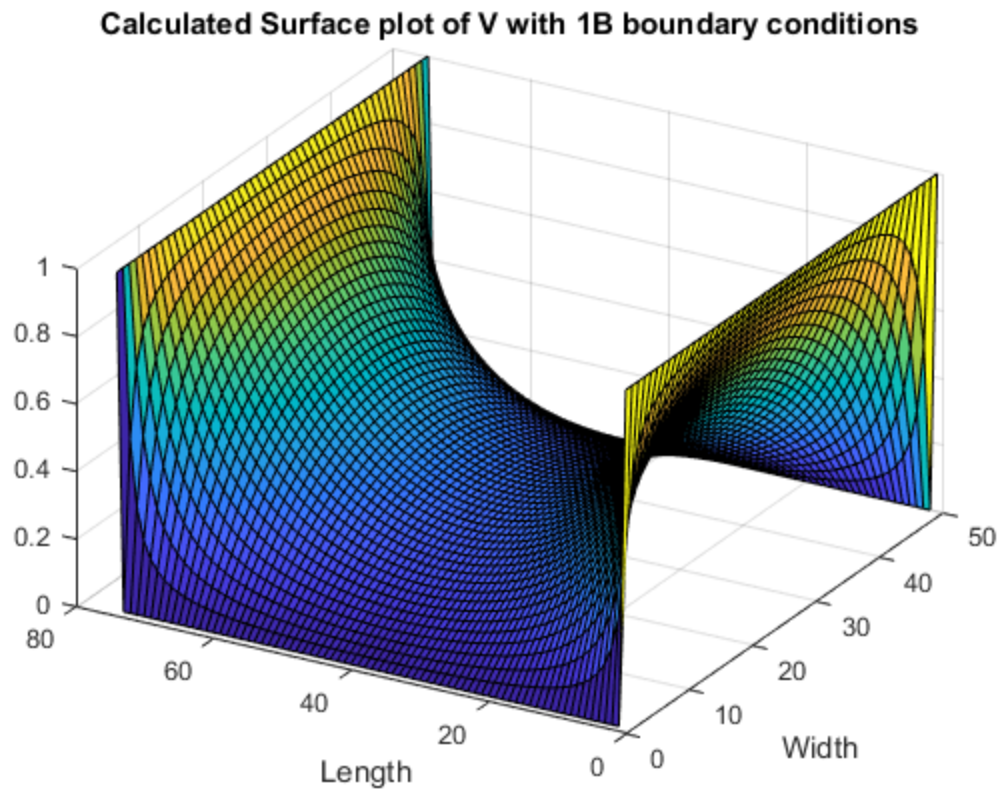
for n=3:2:100
    f=f+(4/pi)*(1/n)*cosh(n*pi*vx/ny)/cosh(n*pi*(nx+1)/2/ny).*sin(n*pi*vy/ny);
end

figure(2)
surf(vy,vx+38,f)

view(-60,37)
xlabel('Width')
ylabel('Length')
title('Analytical Surface plot of V with 1B boundary conditions')

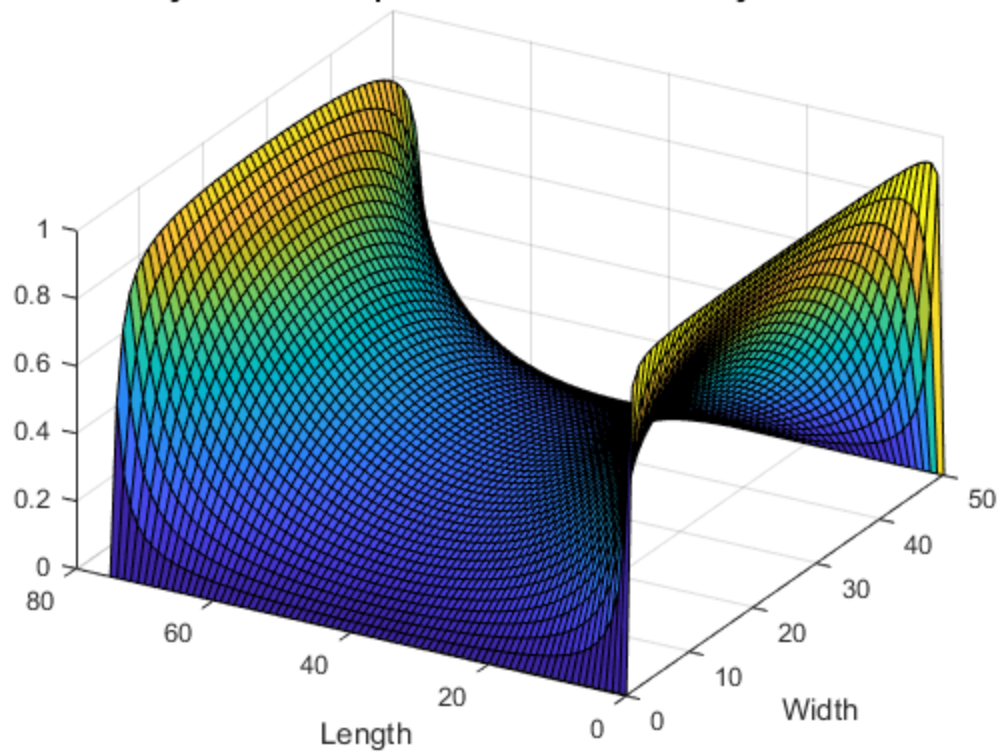
% The calculated solution matches the analytical results. This
% supports
% that the calculated solutions are correct, and represent the system
% we
% are attempting to model
% The meshing on the plots is good, as we can see that there are not
% any
% large discontinuities between elements.

```



---

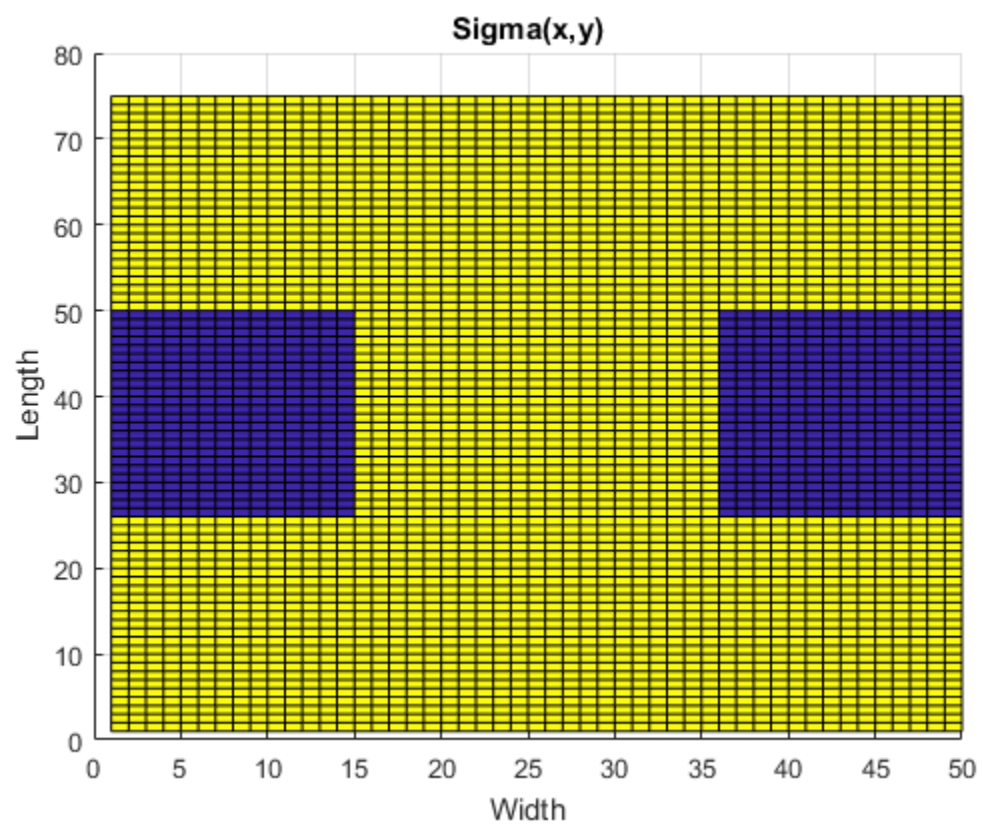
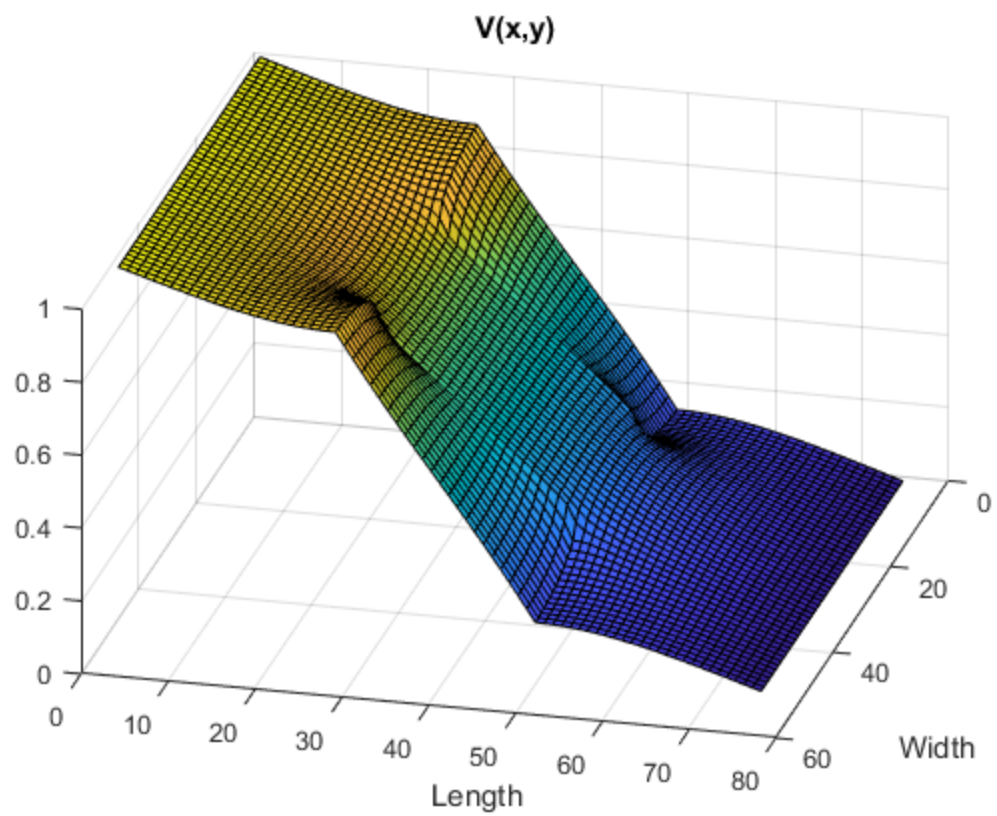
Analytical Surface plot of V with 1B boundary conditions

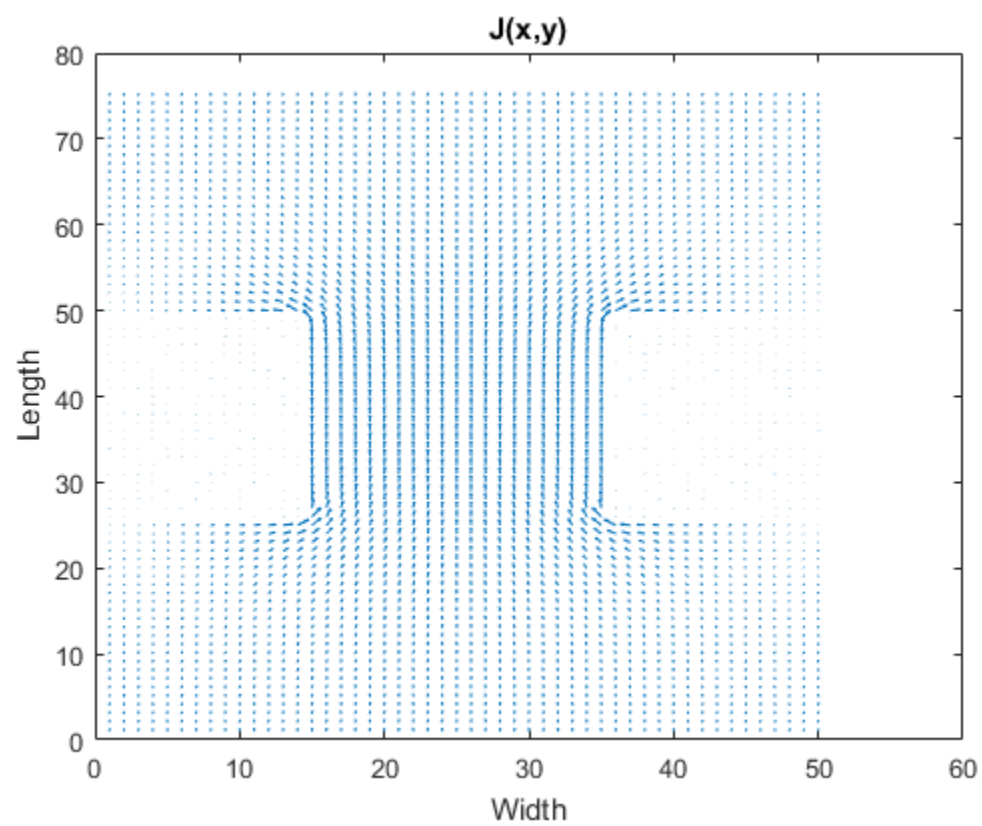
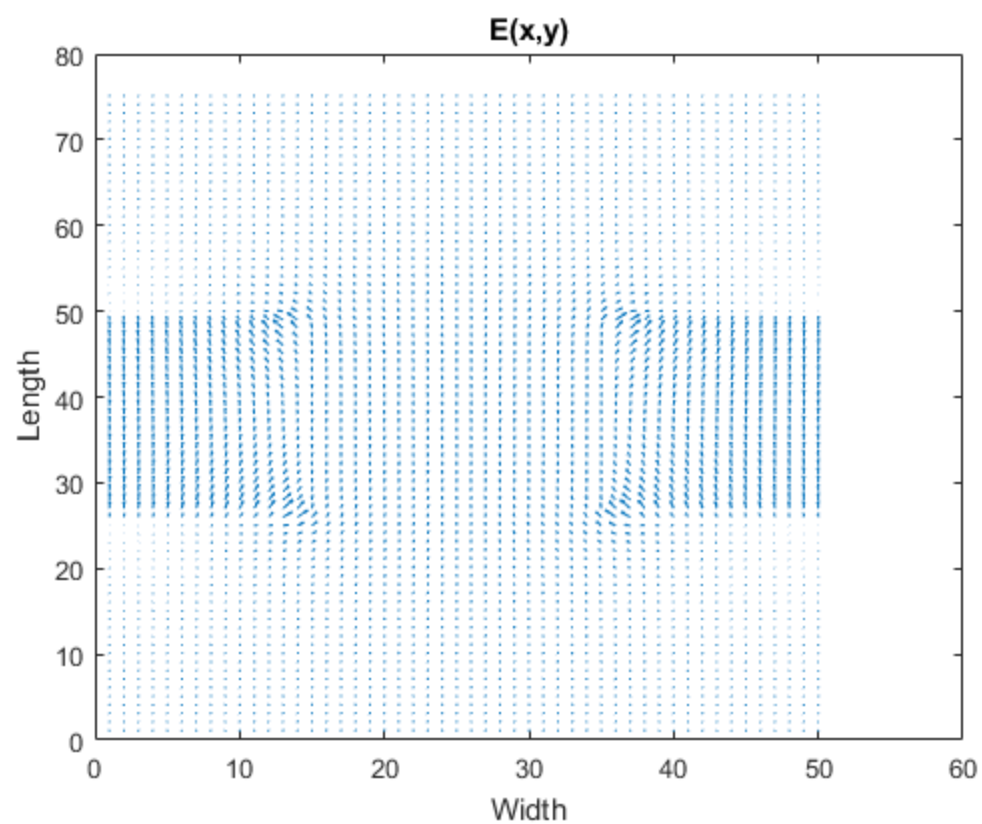


## Question 2 A

```
current =  
    q2device(1,1e-2,round(nx/3),round(2*nx/3),round(3/10*ny),round(7/10*ny),nx,ny,1);  
current  
%current in (A)
```

```
current =  
  
    8.1958e-06
```





---

## Question 2 B

```
%current vs. mesh size

current=zeros(1,100);
for i = 1:100
    ny=15+(i);
    nx=round(ny*3/2);

    current(i)=q2device(1,1e-2,round(nx/3),round(2*nx/3),round(3/10*ny),round(7/10*ny)
end

x=linspace(1,100);
figure(1)
plot (x,current)
xlabel('mesh size (# of divisions of 1mm)')
ylabel('current (A)')
title('current vs. mesh size')

% Here the current decreases with mesh size. Since the mesh size is
% very
% large at the begining of the graph, it is innaccurate, and gives a
% poor
% estimation of current. As the mesh size increases, the predicted
% value
% approaches the true value. It will reach an asytmote as the
% predicted
% value approaches the real value.

%current vs. increasing width of the Bottle Necks
current=0;
j=0;
for i = 0:15
    j=j+1;
    ny=50;
    nx=75;
    current(j)=q2device(1,1e-2,25,50,round(5+i),round(45-i),nx,ny,0);
end

x=linspace(40,10,length(0:15));
figure(2)
plot (x,current)
xlabel('width of central conductive region')
ylabel('current (A)')
title('current vs. bottleneck width')
view(0,90)

% Here the current increases as the width of the conductive channel
% increases. This is expected because resistance is inversly
% proportional
% to width, so a higher width as a lower resistance, and therefore
% more
% current will flow
```



---

```
%current vs. increasing conductivity
j=0;
current=0;
for i = 0.2:.04:0.9
    j=j+1;
    ny=50;
    nx=75;
    current(j)=q2device(i,i/100,50,100,30,70,nx,ny,0);
end
x= 0.2:.04:0.9;
figure(3)
plot (x,current)
xlabel('conductivity of the higher conductivity region')
ylabel('current (A)')
title('current vs.conductivity while \sigma_1=100*\sigma_2')

% Here the current increases as the conductivity increases. This is
% expected because as the conductivity increases, the resistance
% decreases
% and therefore the current increases

%current vs. increasing conductivity
j=0;
current=0;
for i = 1:100
    j=j+1;
    ny=50;
    nx=75;
    current(j)=q2device(1,i/100,50,100,30,70,nx,ny,0);
end
x= 1/100:1/100:1;
figure(4)
plot (x,current)
xlabel('conductivity of the lower conductivity region')
ylabel('current (A)')
title('current vs.conductivity while \sigma_1=1, \sigma_2 varies')

% Here the current increases as the conductivity increases. This is
% expected because as the conductivity increases, the resistance
% decreases
% and therefore the current increases

% Function used in Question 2

function [curr]=q2device(res1,res2,left,right,bottom,top,nx,ny,plot)
nx; %length
ny; %width

g=sparse(nx*ny);
b=zeros(1,nx*ny);
```

---

---

```

sig1=res1;
sig2=res2;

% box = [ left right bottom top] of center high conductive region
box=[left right bottom top];

sigma=zeros(nx,ny);
for i = 1:nx
    for j=1:ny

        if i > box(1) && i < box(2) && j >box(4) %upper box - low cond
            sigma(i,j)=sig2;
        elseif i > box(1) && i < box(2) && j <box(3) %lower box - low
cond
            sigma(i,j)=sig2;
        else %high cond
            sigma(i,j)=sig1;
        end
    end
end

for i = 1:nx
    for j=1:ny
        n=j+(i-1)*ny;
        if i==1 %left
            g(n,:)=0;
            g(n,n)=1;
            b(n)=1;
        elseif i==nx %right
            g(n,:)=0;
            g(n,n)=1;

        elseif j==1 %bottom
            up=(sigma(i,j)+sigma(i,j+1))/2;
            left=(sigma(i,j)+sigma(i-1,j))/2;
            right=(sigma(i,j)+sigma(i+1,j))/2;
            g(n,n)=-(up+left+right);
            g(n,n+1)=up;
            g(n,n-ny)=left;
            g(n,n+ny)=right;

        elseif j==ny %top
            %low conductivity
            down=(sigma(i,j)+sigma(i,j-1))/2;
            left=(sigma(i,j)+sigma(i-1,j))/2;
            right=(sigma(i,j)+sigma(i+1,j))/2;

            g(n,n)=-(up+left+right);
            g(n,n+ny)=right;
            g(n,n-1)=down;
            g(n,n-ny)=left;

```

---

---

```

        else %bulk node

            down=(sigma(i,j)+sigma(i,j-1))/2;
            left=(sigma(i,j)+sigma(i-1,j))/2;
            right=(sigma(i,j)+sigma(i+1,j))/2;
            up=(sigma(i,j)+sigma(i,j+1))/2;

            g(n,n)=-(up+down+right+left);
            g(n,n+1)=up;
            g(n,n-1)=down;
            g(n,n+ny)=right;
            g(n,n-ny)=left;

        end
    end
end

E=g\b';

d=zeros(nx,ny);
for i = 1:nx
    for j=1:ny
        n=j+(i-1)*ny;
        d(i,j)=E(n);

    end
end
if plot
    figure(1)
    surf(d) %V(x,y)
    xlabel('Width')
    ylabel('Length')
    title('V(x,y)')
    view(-256,36)
end

%make sigma(x,y) graph

if plot
    figure(2)
    surf(sigma) %sigma(x,y)
    xlabel('Width')
    ylabel('Length')
    title('Sigma(x,y)')
    view(0,90)
end

[ex,ey]=gradient(d);
ex=-ex;
ey=-ey;
if plot

```

---

---

```

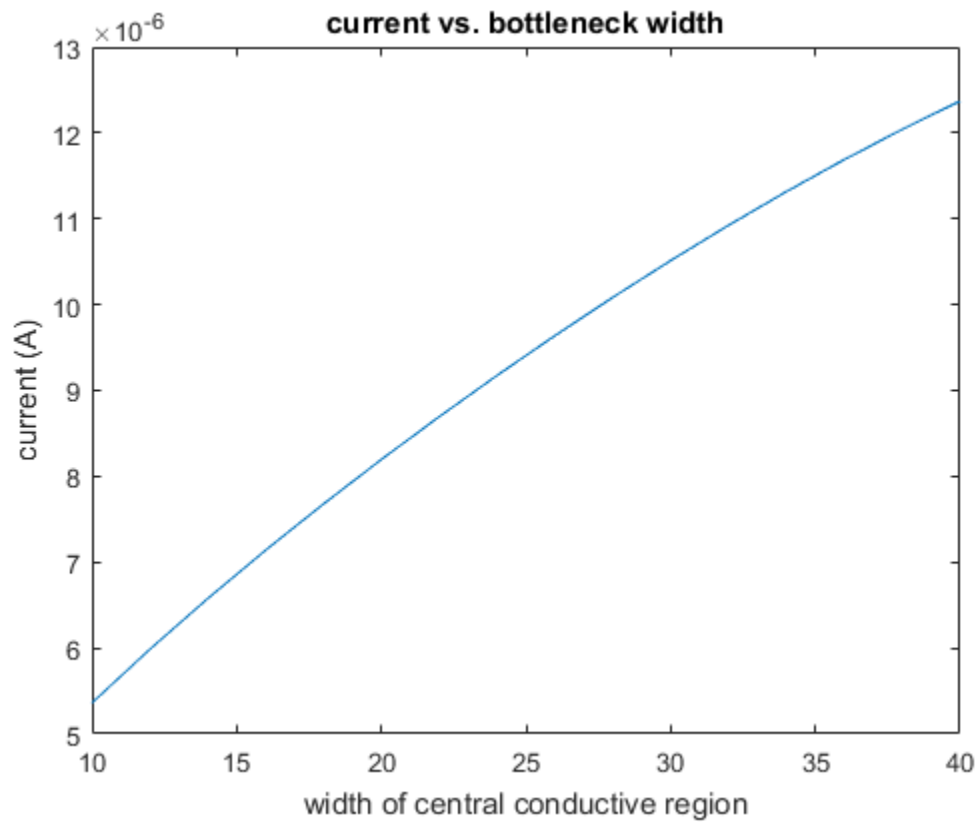
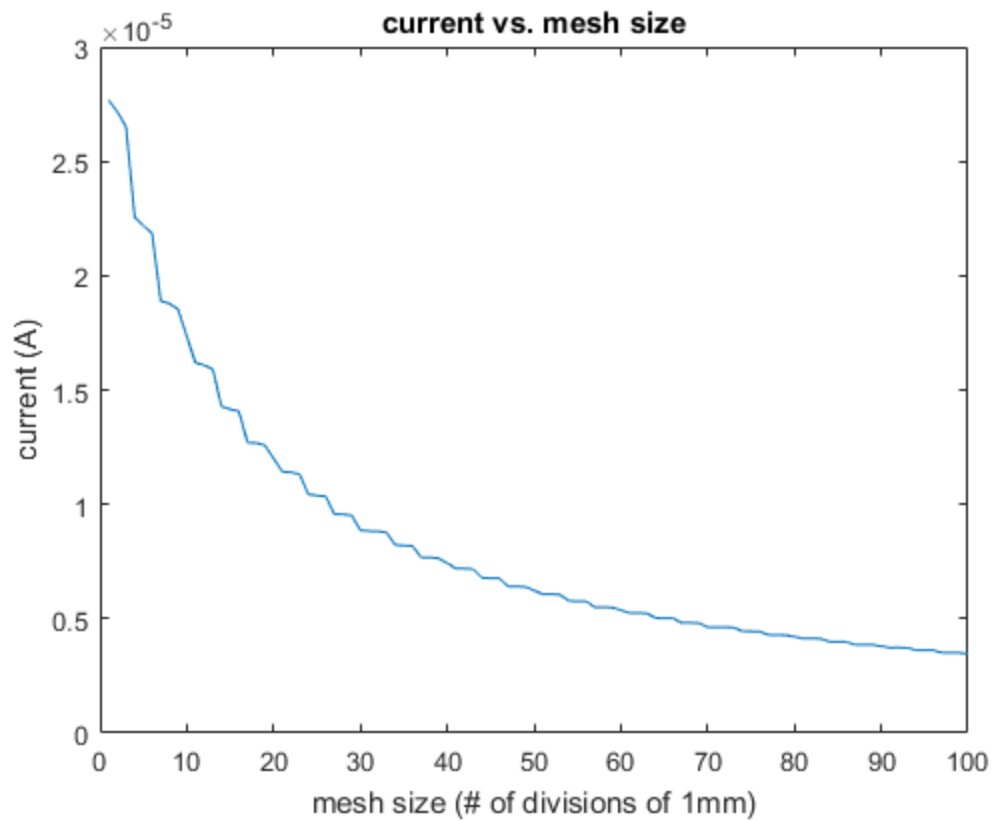
        figure(3) %E(x,y)
        quiver(ex,ey)
        xlabel('Width')
        ylabel('Length')
        title('E(x,y)')
    end

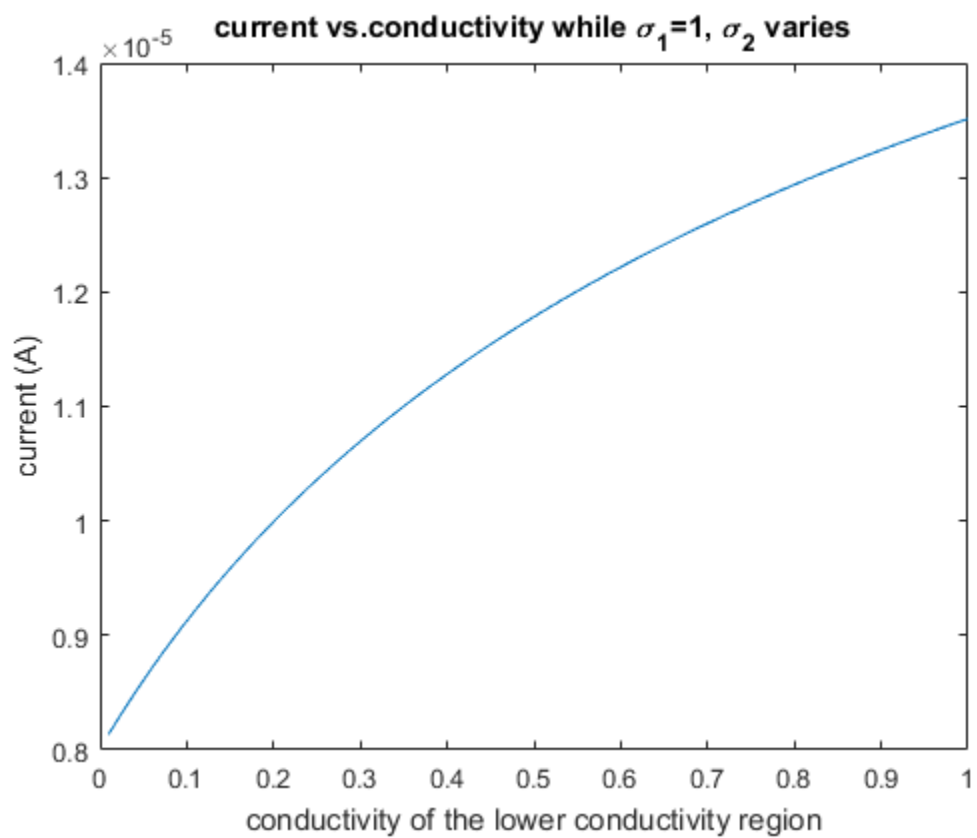
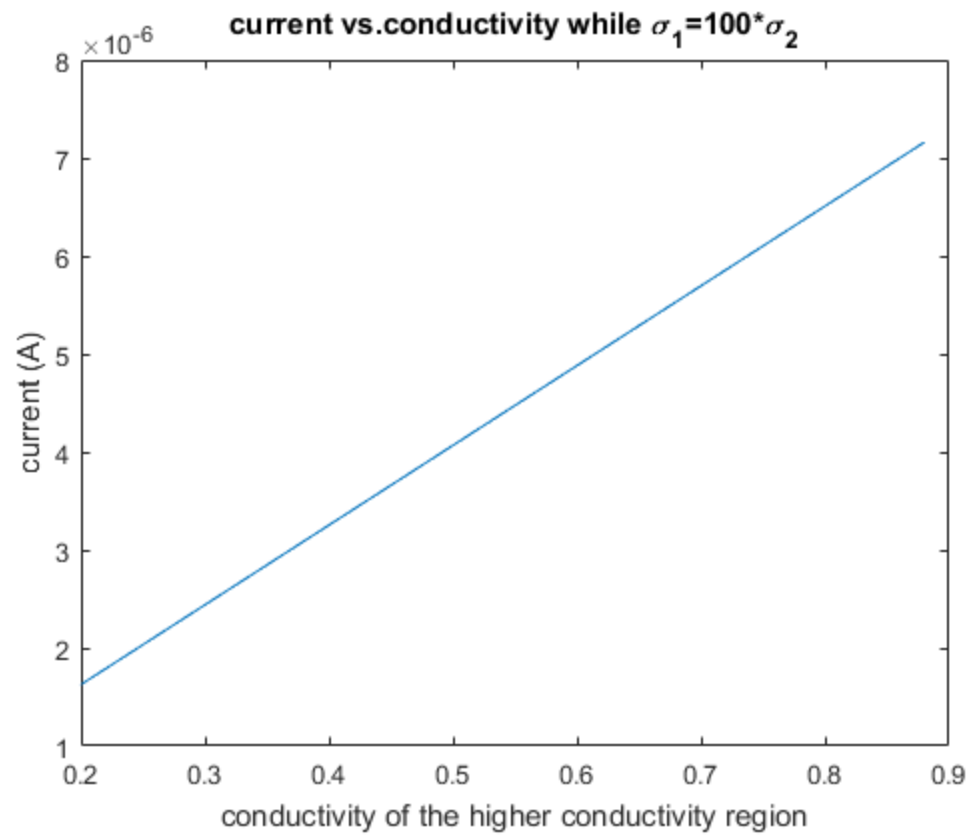
    Jx = ex.*sigma;
    Jy = ey.*sigma;
    if plot
        figure(4)
        quiver(Jx,Jy)
        xlabel('Width')
        ylabel('Length')
        title('J(x,y)')
    end

    %find current between the two contacts (assuming they are on the right
    and
    %left side of the device. also assume the device is 1mmx1mm)

    curr=0;
    for j=1:ny
        %current = current + current_density * area of 1 element
        curr=curr+Jy(1,j)*(1e-3/ny);
    end
end

```





---

*Published with MATLAB® R2017a*