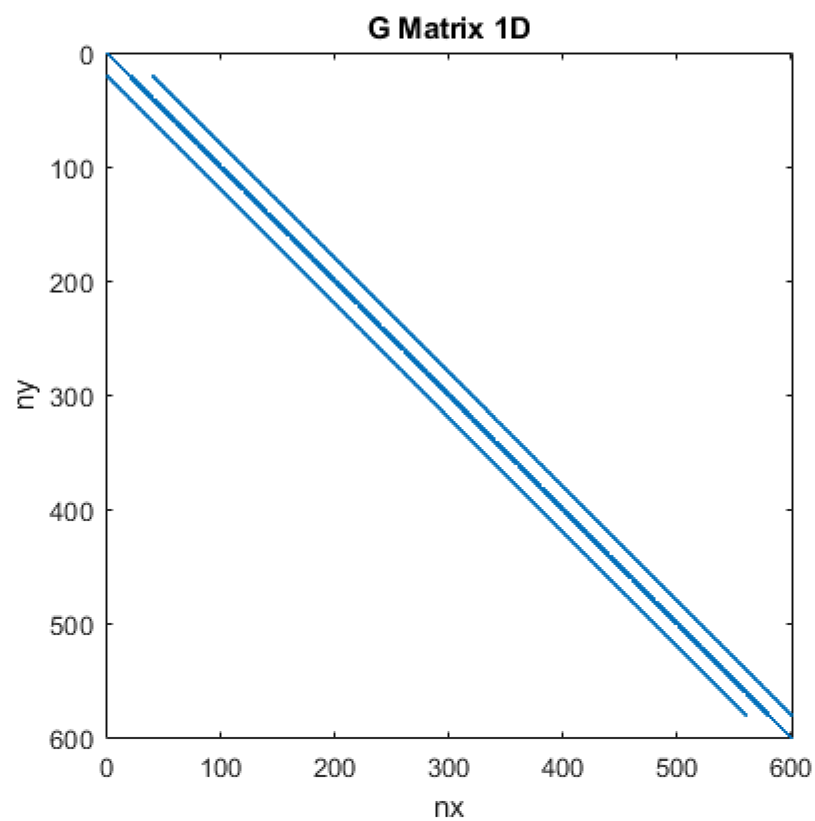
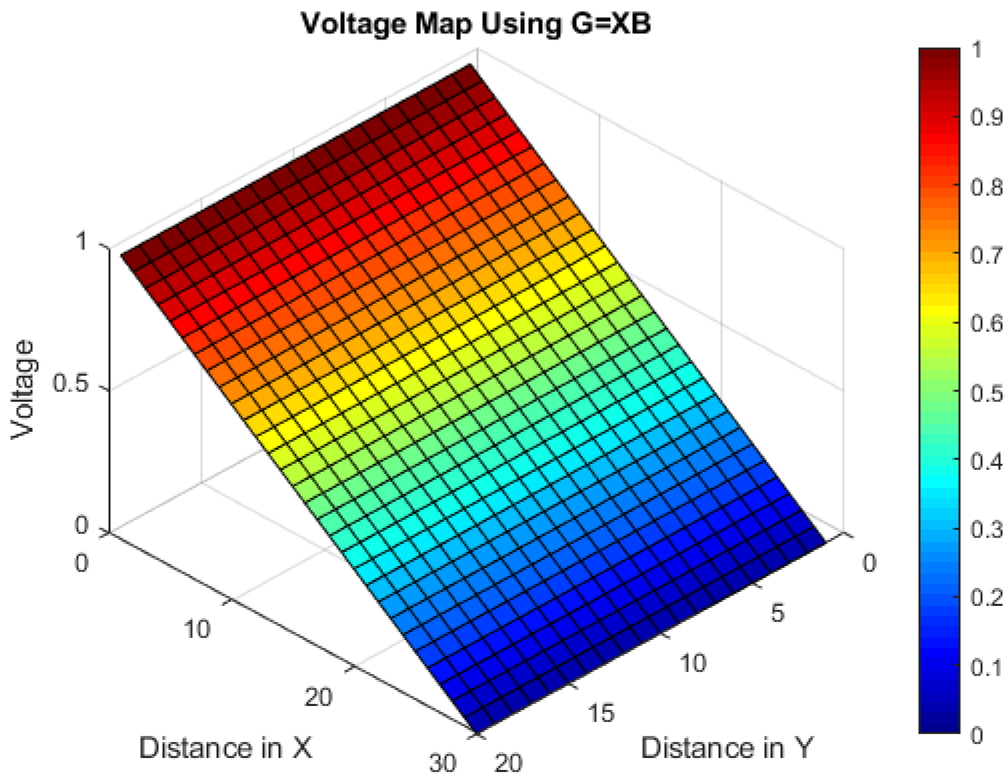
```
clear
clc


nx = 30; %number of rows
ny = 20; %number of columns
G = zeros(nx*ny, nx*ny);
G2 = zeros(nx*ny, nx*ny);
iterations = 250;           %Used in the analyitical solution

colormap(jet)

B = zeros(nx*ny,1);
B2 = zeros(nx*ny,1);

m_0 = 9.11e-31; %mass of electrons

%Used to calculate sthe N value for the G matrix
fn = @(i, j) j + (i-1)*ny;



%1D SOLUTION STARTS HERE
for w = 1:nx % x
    for s = 1:ny % y
%         G(1, 1) = 1;
%         G(nx, 1) = 1;
        n = fn(w, s);
        nxm = fn(w-1, s);
        nxp = fn(w+1, s);
        nym = fn(w, s-1);
        nyp = fn(w, s+1);
        if w == 1                   %Check to see if on left boundary
            G(n,:) = 0;
            G(n, n) = 1;
            B(n) = 1;
        elseif w == nx              %Check to see if on right boundary
            G(n,:) = 0;
            G(n, n) = 1;
            B(n) = 0;
        elseif s == 1               %Check bottom boundary
            G(n,:) = 0;
            G(n, n) = -3;
            G(n, nxp) = 1;
            G(n, nxm) = 1;
            G(n, nyp) = 1;
        elseif s == ny              %Check top boundary
            G(n,:) = 0;
            G(n, n) = -3;
            G(n, nxp) = 1;
            G(n, nxm) = 1;
            G(n, nym) = 1;
```

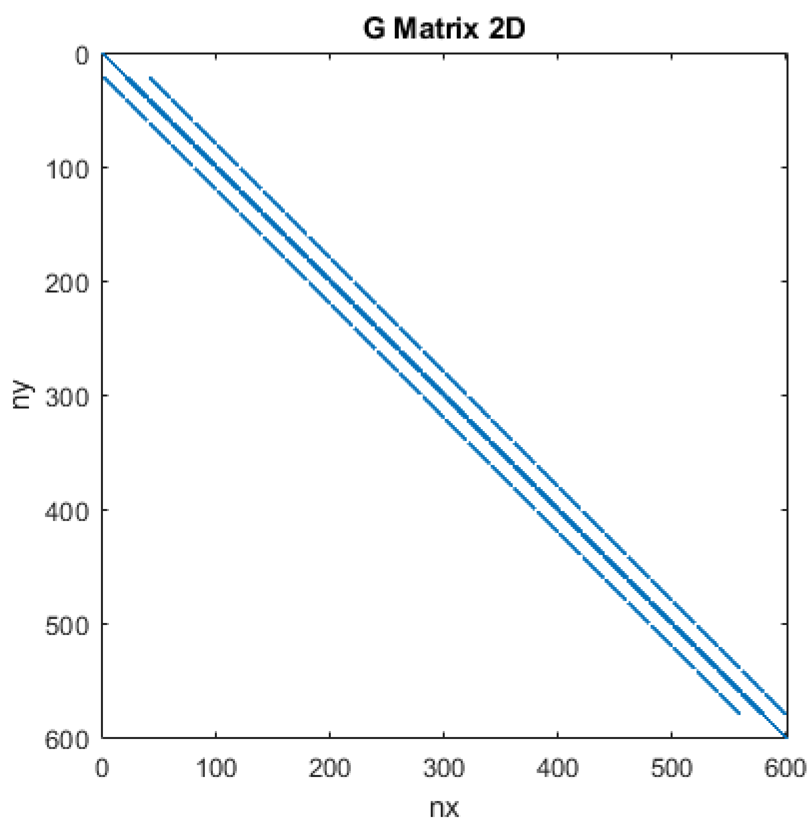
```
        else                        %Somewhere in the middle :)
            G(n,:) = 0;
            G(n, n) = -4;
            G(n, nxp) = 1;
            G(n, nxm) = 1;
            G(n, nyp) = 1;
            G(n, nym) = 1;
        end
    end
end


figure(1)
spy(G); % 2.a.iv
title('G Matrix 1D');
xlabel('nx'); ylabel('ny');
grid on;


X = G\B;



for w = 1:nx % x
    for s = 1:ny % y
        n = s+(w-1)*ny;
        plot1(w, s) = X(n);
    end
end

figure(2)
surf(plot1);
colormap(jet)
title('Voltage Map Using G=XB')
xlabel('Distance in Y')
ylabel('Distance in X')
zlabel('Voltage')
view(135, 45);
colorbar




%2D SOLUTION STARTS HERE
for ww = 1:nx % x
    for ss = 1:ny % y
%         G(1, 1) = 1;
%         G(nx, 1) = 1;
        n = fn(ww, ss);
        nxm = fn(ww-1, ss);
        nxp = fn(ww+1, ss);
        nym = fn(ww, ss-1);
        nyp = fn(ww, ss+1);
```

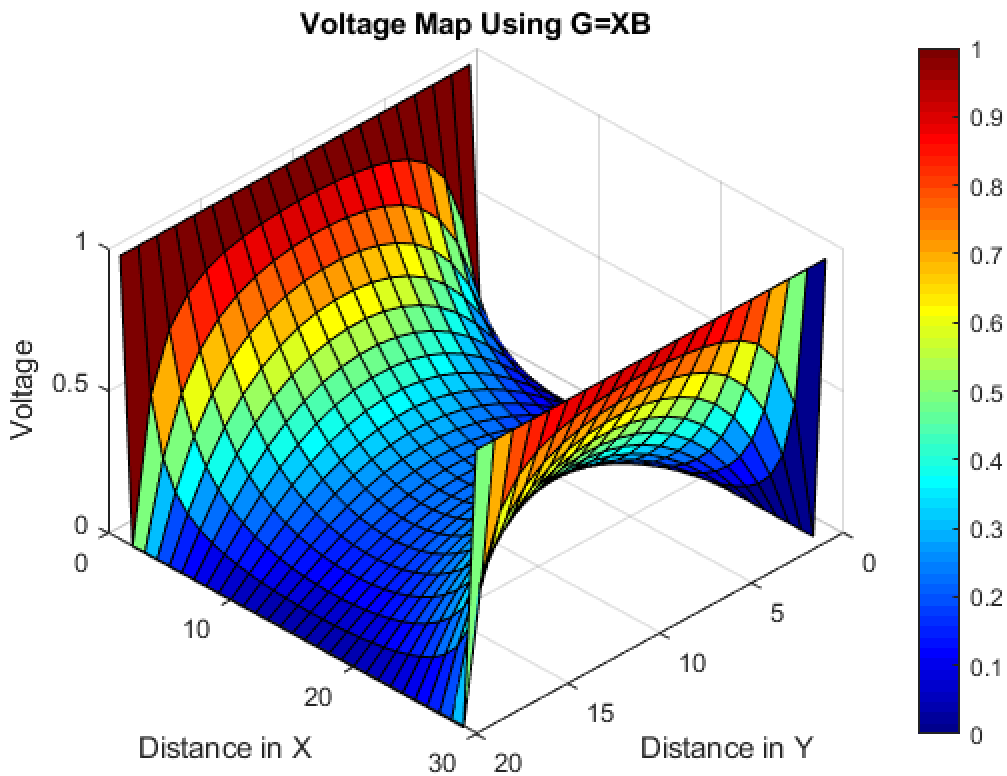
```
        if ww == 1                      %Check left boundary
            G2(n,:) = 0;
            G2(n, n) = 1;
            B2(n) = 1;
        elseif ww == nx                 %Check right boundary
            G2(n,:) = 0;
            G2(n, n) = 1;
            B2(n) = 1;
        elseif ss == 1                  %Check the bottom boundary
            G2(n,:) = 0;
            G2(n, n) = 1;
            B2(n) = 0;
        elseif ss == ny                 %Check top boundary
            G2(n,:) = 0;
            G2(n, n) = 1;
            B2(n) = 0;

        else                            %Somewhere in the middle again!
            G2(n,:) = 0;
            G2(n, n) = -4;
            G2(n, nxp) = 1;
            G2(n, nxm) = 1;
            G2(n, nyp) = 1;
            G2(n, nym) = 1;
        end
    end
end


figure(3)
spy(G2); % 2.a.iv
title('G Matrix 2D');
xlabel('nx'); ylabel('ny');
grid on;


X2 = G2\B2;



for ww = 1:nx % x
    for ss = 1:ny % y
        n = ss+(ww-1)*ny;
        plot2(ww, ss) = X2(n);
    end
end

figure(4)
surf(plot2);
colormap(jet)
title('Voltage Map Using G=XB')
xlabel('Distance in Y')
ylabel('Distance in X')
zlabel('Voltage')
```

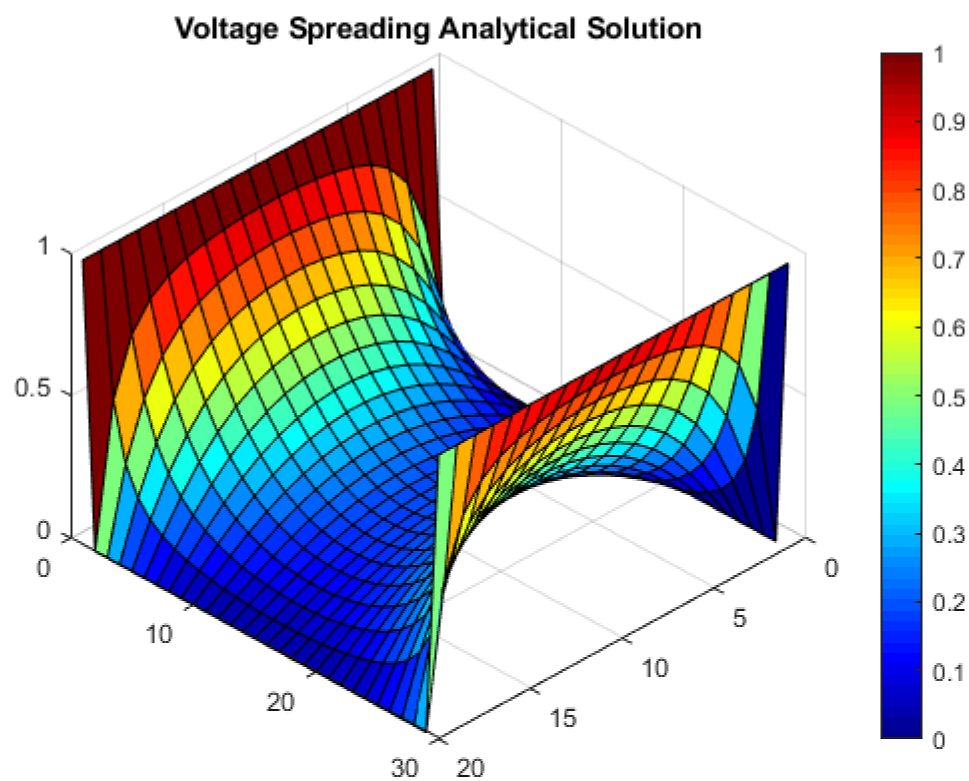
```
view(135, 45);
colorbar



%ANALYTICAL SOLUTION STARTS HERE
%I couldn't figure out how to impolement the equation
%provided in the lab manual, so the Laplace PA example was
%modified to give the result

V2 = zeros(nx, ny); %Voltage array of size nx rows by ny columns
% boundary conditions for V2
for z = 1:ny
    V2(1, z) = 1; %left BC
    V2(nx, z) = 1; %right BC
end
for z = 2:nx-1
    V2(z, 1) = 0; %up BC
    V2(z, ny) = 0; %down BC
end
for k = 1:iterations
    V2(2:nx-1, 2:ny-1) = (V2(1:nx-2, 2:ny-1) + V2(3:nx, 2:ny-1) +
 V2(2:nx-1, 1:ny-2) + V2(2:nx-1, 3:ny))/4;
    % center matrix   =   shifted left    +  shifted right  +
 shifted up      + shifted down
    figure (5)
    surf(V2);
    colormap(jet)
    colorbar
    title('Voltage Spreading Analytical Solution');
    view(135, 45);

end
```


G Matrix 1D
0
100
200
300
400
500
600
ny
0
100
200
300
400
500
600
nx


Voltage Map Using G=XB
Voltage
Distance in X
Distance in Y
0
0.5
1
0
10
20
30
0
5
10
15
20
0
0.1
0.2
0.3
0.4
0.5
0.6
0.7
0.8
0.9
1


G Matrix 2D
ny
nx
0
100
200
300
400
500
600
0
100
200
300
400
500
600


Voltage Map Using G=XB
Voltage
1
0.5
0
0
10
20
30
Distance in X
0
5
10
15
20
Distance in Y
1
0.9
0.8
0.7
0.6
0.5
0.4
0.3
0.2
0.1
0


Voltage Spreading Analytical Solution
1
0.5
0
0
10
20
30
0
5
10
15
20
1
0.9
0.8
0.7
0.6
0.5
0.4
0.3
0.2
0.1
0

Published with MATLAB® R2018a