Assignment 5

Kathir Pagalavan EE20B056

March 6, 2022

1 Aim

To Solve Laplace equation to find out the voltage distribution in a conducting surface given its boundary conditions and to find the current density and temperature distribution pattern.

2 Introduction

A cylindrical wire of size 0.35 cm by 0.35 cm is soldered to the middle of a copper plate and its voltage is held at 1 Volt. One side of the plate is grounded, while the remaining are floating. The plate is 1 cm by 1 cm in size.

3 Assignment 3

3.1 Defining Parameters

We have chosen a 25x25 grid with the circular wire of radius 0.35cm located at the centre $V = 1V$ by default. All these parameters can be modified by passing arguments in the program

```
#size of plate is 1*1 fixed
Nx=25; # size along x
Ny=25; # size along y
radius=0.35;# radius of central lead
Niter=1500; # number of iterations to perform

if(len(sys.argv)==3):
        Nx = int(sys.argv[1])
        Ny = int(sys.argv[2])
        Niter = int(sys.argv[3])
elif(len(sys.argv)==1):
    print()
else:
    print("Invalid argument input!")
    exit()
```

3.2 Initializing Potential

We start by creating an zero 2-D array of size Nx x Ny. then the circular wire region potential is initialised to $1V$.

```
x = linspace(-0.5,0.5,Nx)
y = linspace(0.5,-0.5,Ny)
X,Y = meshgrid(x,y)
phi = zeros((Nx,Ny))
ii = where(X*X + Y*Y <= radius*radius)
phi[ii] = 1.0
```

3.3 Performing Iterations

3.3.1 Updating Potential

From the Laplace's equation we infer the result that the potential at any point is the average of the potentials at all the points around it.

$$\phi_{i,j} = 0.25 * (\phi_{i-1,j} + \phi_{i+1,j} + \phi_{i,j+1} + \phi_{i,j-1}) \tag{1}$$

```
    phi[1:-1,1:-1] = 0.25*(phi[1:-1,0:-2] + phi[1:-1,2:] + phi[0:-2,1:-1] + phi[2:,1:-1])
```

3.3.2 Applying Boundary Conditions

The bottom boundary is grounded. The other 3 boundaries have potential gradient only tangential to the surface.

```
    phi[1:-1,0] = phi[1:-1,1]
    phi[1:-1,-1] = phi[1:-1,-2]
    phi[0,1:-1] = phi[1,1:-1]

    phi[ii] = 1.0
```

3.3.3 Calculating error

Error is taken as the maximum of the difference in the estimated potentials at each of the point.

```
errors=zeros(Niter)
for k in range(Niter):
errors[k]=(abs(phi-oldphi)).max()
```

3.3.4 Plotting the errors

The error data, fitted taking all the values to an exponential, fitted taking the values after 500 iterations to an exponential, and all of the data is plotted on a semi-log graph

```
c_approx_500 = lstsq(c_[ones(Niter-500),arange(500,Niter)],log(errors[500:]),rcond=None)
a_500,b_500 = exp(c_approx_500[0][0]),c_approx_500[0][1]

c_approx = lstsq(c_[ones(Niter),arange(Niter)],log(errors),rcond=None)
a, b = exp(c_approx[0][0]), c_approx[0][1]
```

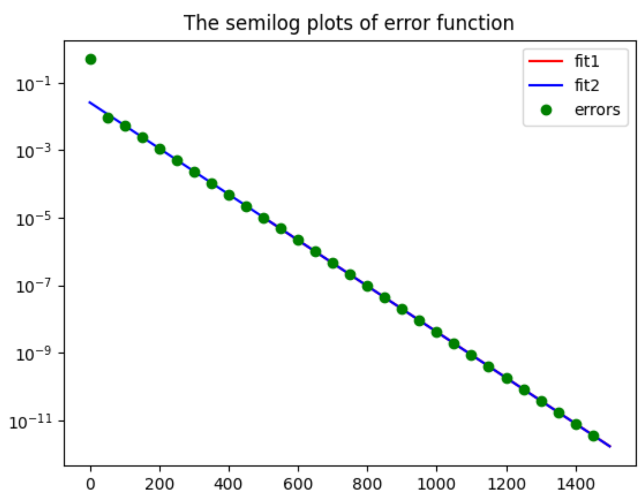

Figure 1: Semi-log plot of error

3.4 Plotting ϕ

```
fig1 = figure(4)
ax = p3.Axes3D(fig1)
title('The 3-D surface plot of the potential')
ax.set_xlabel('x')
ax.set_ylabel('y')
ax.set_zlabel('Potential $(\phi)$')
surf = ax.plot_surface(X, Y, phi, rstride=1, cstride=1, cmap=cm.jet,linewidth=0, antialiased=T
show()

contour(x,y,phi)
plot(x[ii[0]],y[ii[1]],'ro')
xlabel('x')
ylabel('y')
title('The Contour plot of final potential')
```

```
grid()
show()
```

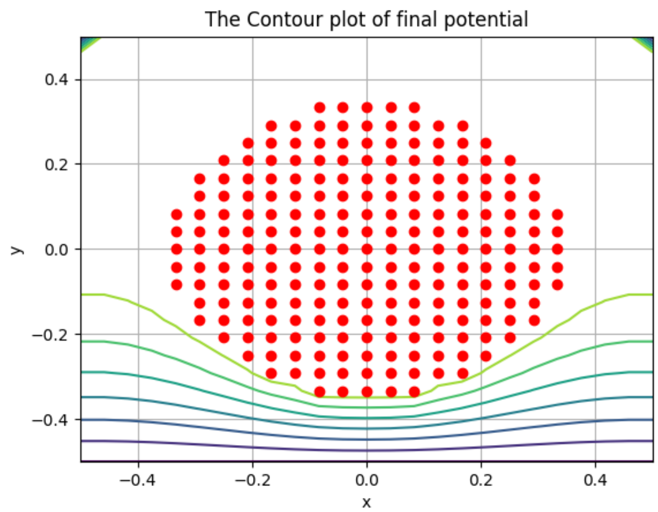

Figure 2: 2d Plot of Potential

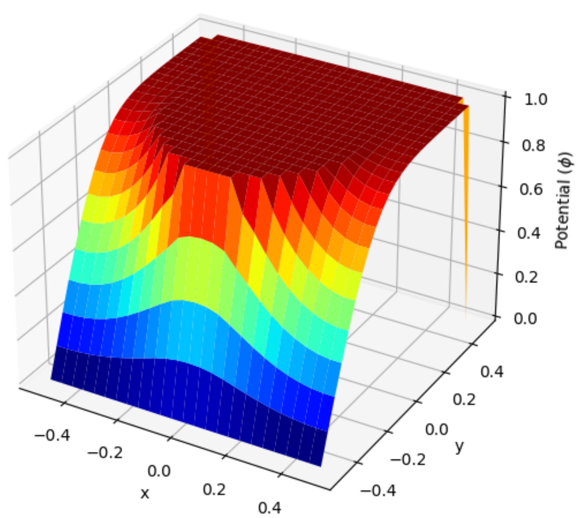

Figure 3: 3d Plot of Potential

3.5 Finding and Plotting J

$$J_{x,ij} = 0.5 * (\phi_{i,j-1} - \phi_{i,j+1}) \tag{2}$$

$$J_{y,ij} = 0.5 * (\phi_{i-1,j} - \phi_{i+1,j}) \tag{3}$$

```
Jx = np.zeros((Nx,Ny))
Jy = np.zeros((Nx,Ny))
Jy[1:-1,1:-1] = 0.5*(phi[1:-1,2:] - phi[1:-1,0:-2])
Jx[1:-1,1:-1] = 0.5*(phi[2:,1:-1] - phi[0:-2,1:-1])
quiver(y,x,Jy[::-1,:],Jx[::-1,:])
contour(x,y,phi)
plot(x[ii[0]],y[ii[1]],'ro')
show()
```

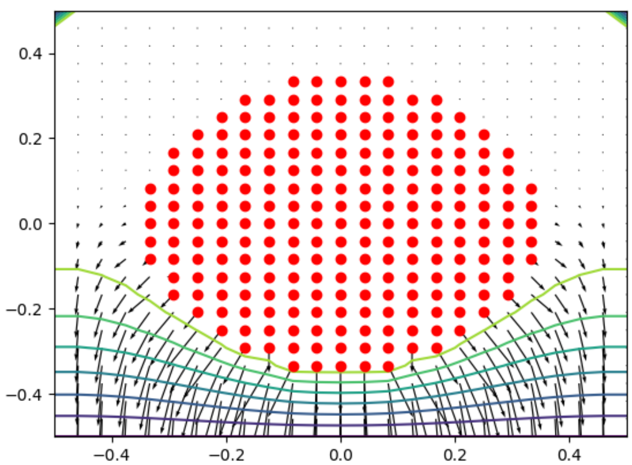

Figure 4: Vector plot of current flow

3.6 Finding temperature at different regions in the plate

We solve the Laplace equation for heat flow:

$$\nabla^2 T = -\frac{q}{\kappa} = -\frac{|J|^2}{\sigma\kappa}$$

We again assume the electrical and thermal conductivites of copper for the plate

```
T = np.zeros((Nx,Ny))
T[:,:] = 300
sigma = 6*(10**7)
```

```
kappa = 386
#in the following calculation J is actually dV and hence it is divided with the distance betwe
for i in range(Niter):
    T[1:-1,1:-1] = 0.25*(T[1:-1,0:-2] + T[1:-1,2:] + T[0:-2,1:-1] + T[2:,1:-1] + ((((Jx*Nx)**2
    T[1:-1,0]=T[1:-1,1]
    T[1:-1,Nx-1]=T[1:-1,Nx-2]
    T[0,1:-1]=T[1,1:-1]
    T[ii] = 300.0
fig1=figure(4)
ax=p3.Axes3D(fig1)
title('The 3-D surface plot of the temperature')
ax.set_xlabel('x')
ax.set_ylabel('y')
ax.set_zlabel('Temperature')
ax.plot_surface(X, Y, T, rstride=1, cstride=1, cmap=cm.jet, linewidth=0, antialiased=True)
show()
```

This is very similar to the previous case of finding potential except that at initial condition, everything is at 300K, and the boundary conditions set the central circle and ground to 300K for all iterations.

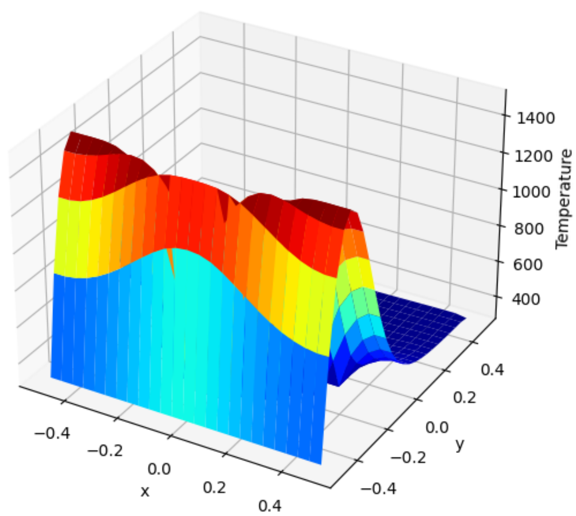

Figure 5: Temperature distribution plot

4 Conclusion

We have used Laplace's equation in its difference equation format to find the voltage distribution in the given plate given its boundary conditions. We also found out the current density distribution and the temperature distribution on the surface.