EE2703 : Applied Programming Lab
Assignment 5

Dharahas
EE20B041

March 6, 2022

Assignment

We wish to solve for the currents in the resistor.For that,
The potential at each point is the average of the neighbouring potentials.

$$\phi_{i,j} = \frac{\phi_{i+1,j} + \phi_{i-1,j} + \phi_{i,j+1} + \phi_{i,j-1}}{4}$$

Code

Defining parameters

```
Nx = 25
Ny = 25
radius = 8
Niter = 1500
```

Initialising potential array

```
phi = np.zeros((Nx,Ny))
x = np.arange(int(-Nx/2), int(Ny/2)+1)
y = x

X, Y = np.meshgrid(x, y)
ii = np.where(X*X + Y*Y <= 8*8)
phi[ii]=1

plt.scatter(ii[0], ii[1], color="r", s=5)
plt.xlabel("x")
plt.ylabel("y")
plt.title("Potential plot")
plt.show()
```

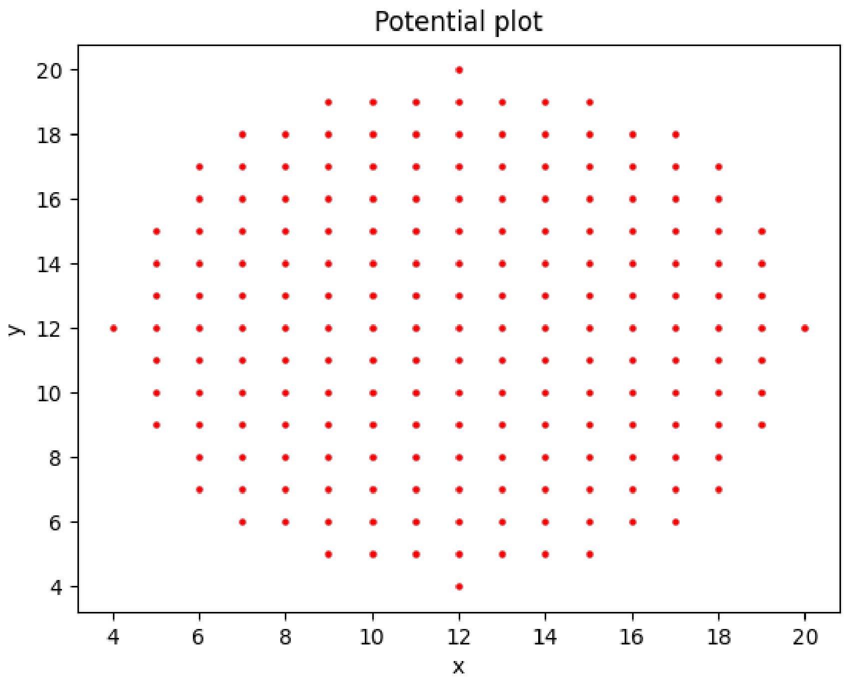

Figure 1: Initial potential on the plate

Performing iterations

```
error = [0 for x in range(Niter)]
for k in range(0,Niter):
    oldphi = phi.copy()
    phi[1:-1,1:-1] = 0.25*(phi[1:-1,0:-2] + phi[1:-1, 2:] +      phi[0:-2, 1:
    error[k] = (abs(phi - oldphi)).max()
```

Applying boundary conditions :

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

Error analysis

Plotting the errors on loglog and semilog scale :

```
x = [n for n in range(Niter)]

plt.loglog(x, error, label="loglog scale")
plt.xlabel("number of iterations")
plt.ylabel("log(error)")
plt.title("Error vs No. of iterations on loglog scale")
plt.show()

plt.semilogy(x, error, label="semilog scale")
plt.xlabel("number of iterations")
plt.ylabel("error")
plt.title("Error vs No. of iterations on semilog scale")
plt.show()
```

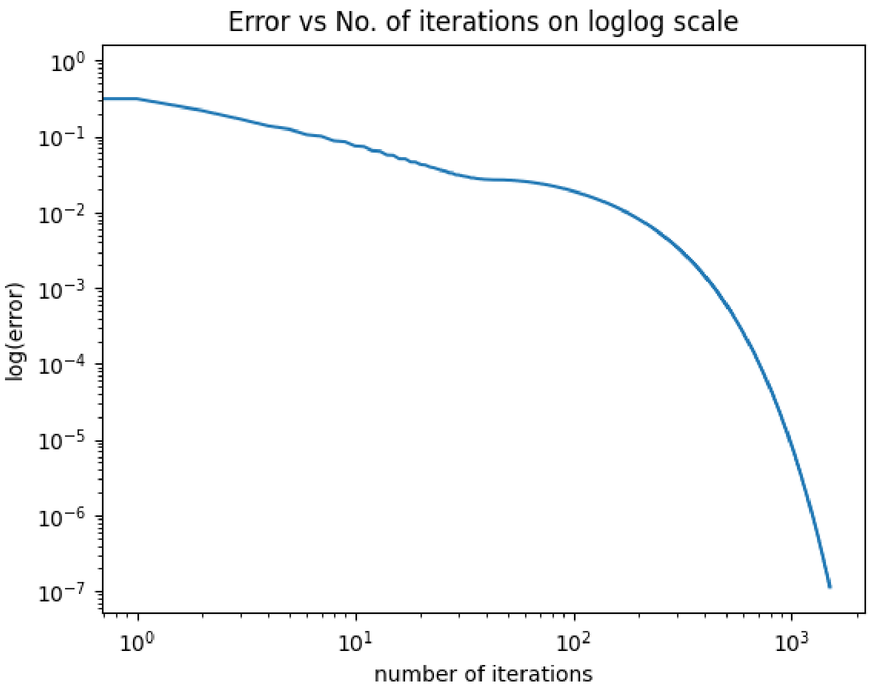

Figure 2: Error on loglog scale

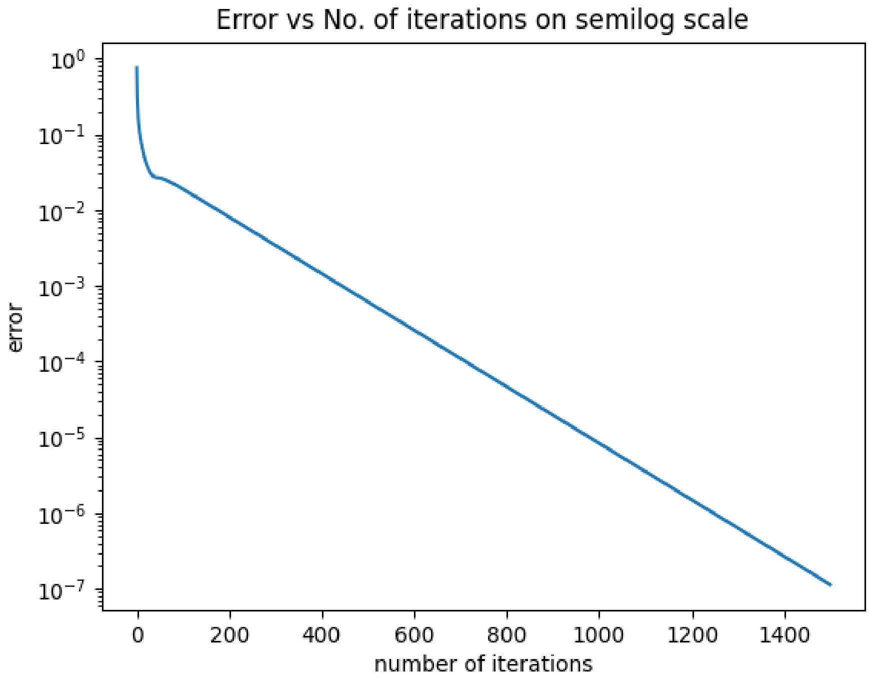

Figure 3: Error on semilog scale

Error was constant initially then decreases exponentially. It is an exponential function of no. of iterations.
Therefore we fit the error in two ways. One for all values of iterations and the other for values after 500 iterations.

The function is :

$$log(y) = log(A) + Bx$$

Fitting for all iterations:

```
b1 = np.log(error)
X1 = np.arange(1, Niter +1)
o1 = np.ones(Niter)
x1 = np.vstack((o1,X1)).T
A1 = np.linalg.lstsq(x1, b, rcond=-1)[0]
```

Fitting for after 500 iterattions :

```
b2 = np.log(error[500:])
```

```
X2 = np.arange(500, Niter)
o2 = np.ones(Niter-500)
x2 = np.vstack((o2, X2)).T
A2 = np.linalg.lstsq(x2,b2, rcond=-1)[0]
```

Plotting the errors:

```
plt.figure(2)
plt.plot(x, np.log(error), label="error")
plt.plot(X1, x1.dot(A1), label="fit1")
plt.plot(X2, x2.dot(A2), label = "fit2", color='g')
plt.legend()
plt.title("Error plot")
plt.xlabel("No. of iterations")
plt.ylabel("Error")
```

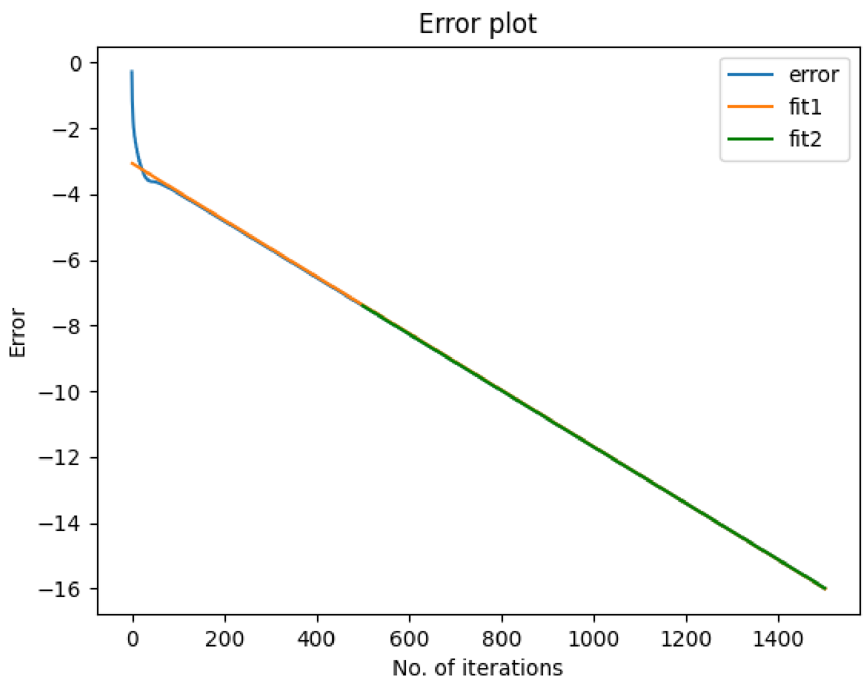

Figure 4: Error plot after fitting

0.0.1 Surface plot

code for surface plot :

```
fig4=plt.figure(4) # open a new figure
ax=p3.Axes3D(fig4) # Axes3D is the means to do a surface plot
plt.title("The 3-D surface plot of the potential")
surf = ax.plot_surface(Y, X, phi.T, rstride=1, cstride=1,
       cmap=plt.cm.jet, linewidth=0, antialiased=False)
fig4.colorbar(surf, shrink=0.5, aspect=5)
plt.show()
```

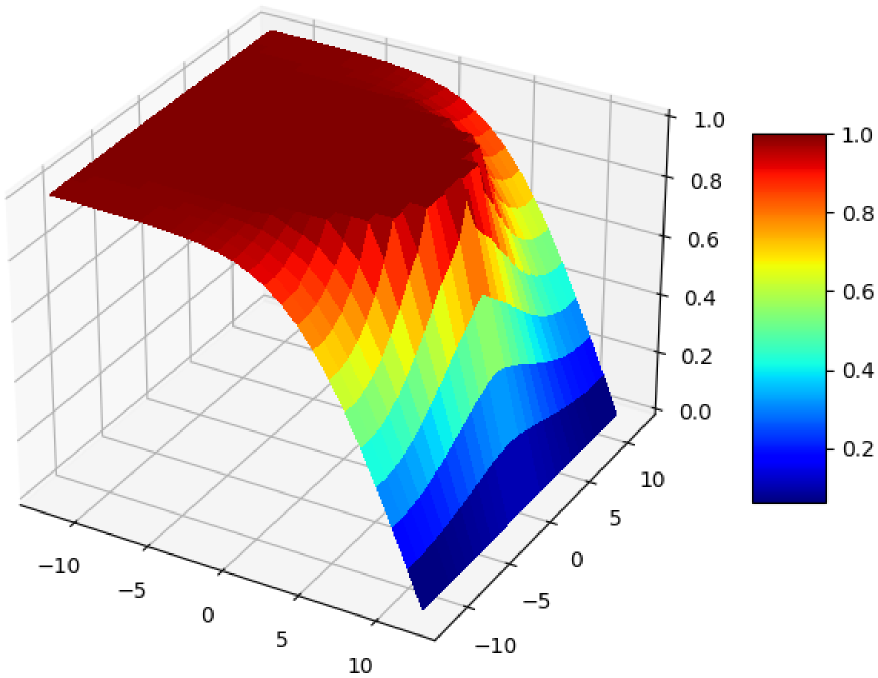

Figure 5: 3D surface plot of potential

Contour plot of potential

```
plt.figure()
plt.plot(ii[0]-Nx/2, ii[1]-Ny/2,"o", color="r")
cp = plt.contourf(Y,X[::-1],phi)
plt.title("Contour plot of potential")
plt.xlabel("x")
plt.ylabel("y")
plt.show()
```

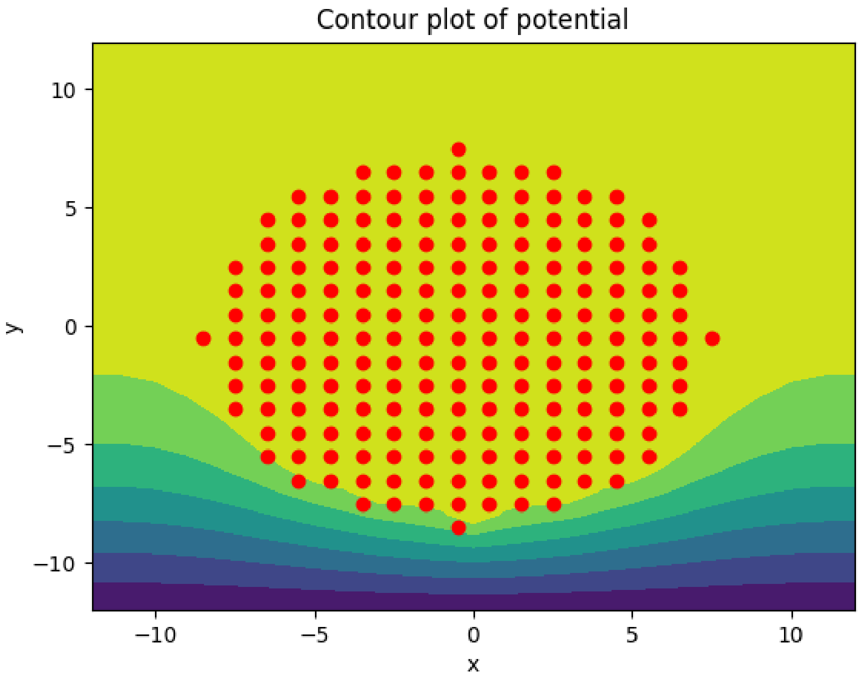

Figure 6: Contour plot of potential

Vector plot of currents

Finding current arrays:

```
Jx = np.zeros((Nx,Ny))
Jy = np.zeros((Nx,Ny))
Jx[1:-1] = 0.5*(phi[2:] - phi[:-2])
Jy[:,1:-1] = 0.5*(phi[:,2:] - phi[:,:-2])
```

Plotting using quiver function:

```
plt.quiver(Y,X ,Jy[::-1,:],Jx[::-1,:])
plt.plot(ii[0]-Nx/2, ii[1]-Ny/2, "o", color="r")
plt.title("Vector plot of currents")
plt.xlabel("x")
plt.ylabel("y")
plt.show()
```

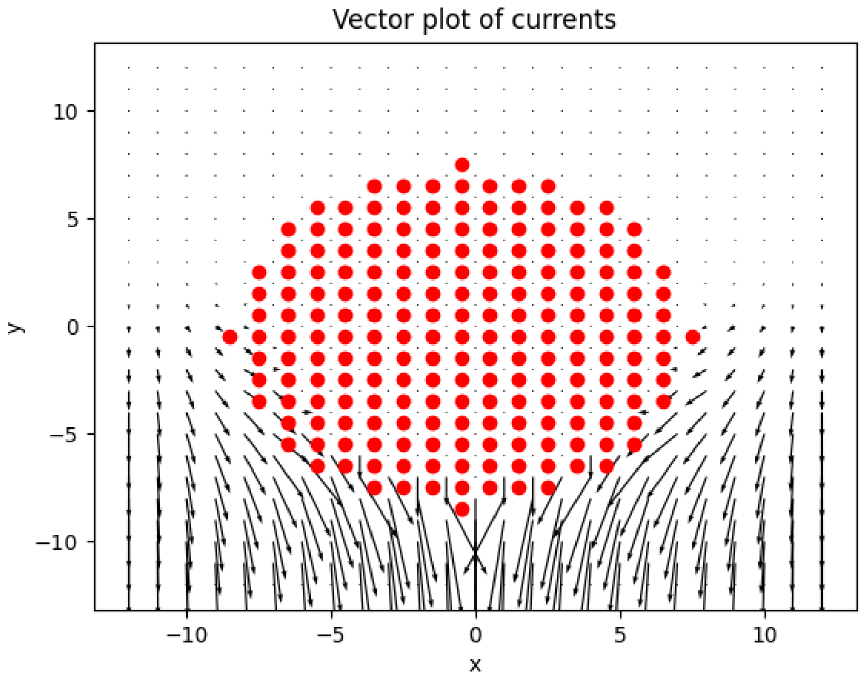

Figure 7: Vector plot of currents

Conclusion

We visualised the current densities and potential in a resistor and how it depends on its geometry. The current density is concentrated below and is perpendicular at the below surface.