Assignment No 5

G Ch V Sairam , EE19B081

25-03-2021

Introduction

We wish to solve for the currents in a resistor. The currents depend on the shape of the resistor.

A cylindrical wire 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.

Combining the continuity equation and Ohm's law , we get

$$\nabla^2\phi = 0$$

Assignment 5

Defining parameters

The user can give the parameters in the commandline if they want to. If no parameters are given , we use the default parameters , giving us a 25 by 25 grid with a circle of radius 8 centrally located maintained at a constant potential of 1V.We choose 1500 as the default number of iterations.

```
if len(argv)==5:
    Nx=int(argv[1])
    Ny=int(argv[2])
    radius=int(argv[3])
    Niter=int(argv[4])
else:
    Nx=25
    Ny=25
    radius=8
    Niter=1500
    print("Default values specified are being used here. If you want to u
```

Initializing potential

We start by creating an 2-D array of size Nx by Ny , consisting of all zeros. Then a list of coordinates lying within the radius is generated and these points are initialized to 1.

```
phi=np.zeros((Nx,Ny),dtype = float)
x = np.linspace(-0.5,0.5,Ny)
y = np.linspace(-0.5,0.5,Nx)
Y,X = np.meshgrid(y,x)
ii=np.where(X**2+Y**2<=((radius/Nx)**2))                    #Points inside th
```

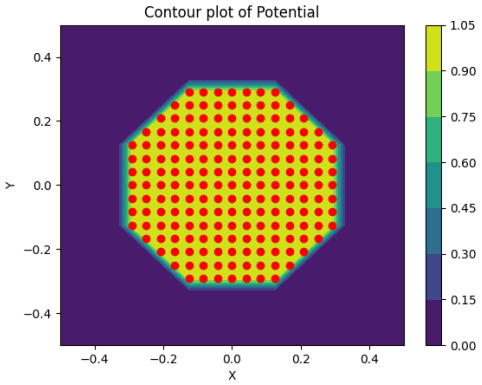

Figure 1: Figure1

Updating potential and finding errors

We use the difference equation to find the value of potential.

$$\phi(i,j) = 0.25 * [\phi(i-1,j) + \phi(i+1,j) + \phi(i,j+1) + \phi(i,j-1)]$$

We also assert the boundaries of the potential.

We will plot the errors on semi-log and log-log plots. We note that the error falls really slowly and this is one of the reasons why this method of solving the Laplace equation is discouraged

```
for k in range(Niter):
        oldphi=phi.copy()
        phi[1:-1,1:-1]=0.25*(phi[1:-1,0:-2]+ phi[1:-1,2:]+ phi[0:-2,1:-1]
        phi[1:-1,0]=phi[1:-1,1]
        phi[1:-1,Nx-1]=phi[1:-1,Nx-2]
        phi[0,1:-1]=phi[1,1:-1]
        phi[ii]=1.0
        errors.append(np.max(np.abs(phi-oldphi)))
```

```
        if errors[k]==0:
                print("The steady state has been reached at",k)
                break
```

Fitting the error and Plotting

We note that the error is decaying exponentially for higher iterations.I have plotted 2 fits. One considering all the iterations(fit1) and another without considering the first 500 iterations. There is very little difference between the two fits.

```
def fit_error(x,y):
        logy=np.log(y)
        mat_x=np.zeros((len(x),2))
        mat_x[:,0]=x
        mat_x[:,1]=1
        B,logA=lstsq(mat_x, np.transpose(logy),rcond=None)[0]
        return (np.exp(logA),B)

def net_error(a,b,N):
        return np.abs(a/b*np.exp(b*(N+0.5)))                    #Find cun

def fit_exp(x,a,b):
        return A*np.exp(B*x)

A,B=fit_error(range(Niter),errors)
A_500,B_500=fit_error(range(Niter)[500:],errors[500:])
```

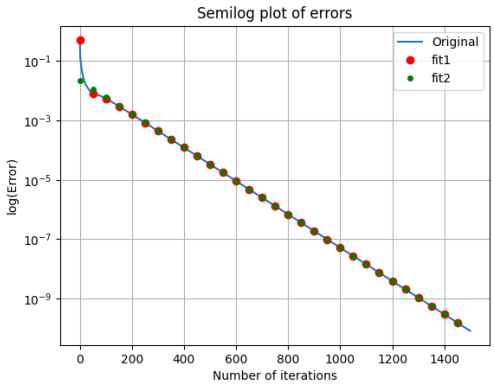

Figure 2: Figure2

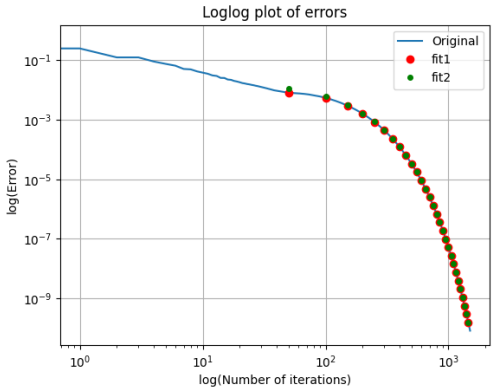

Figure 3: Figure3

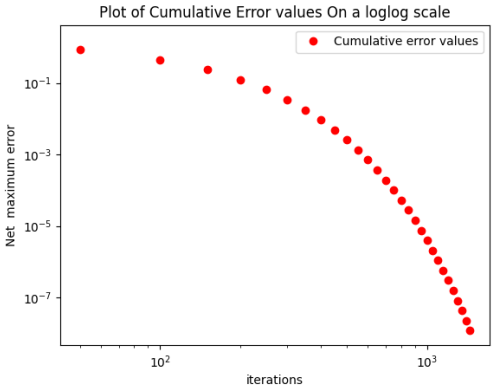

Figure 4: Figure4

Plotting updated potentials

There are 2 ways to plot a 3d figure.We can use the contour plot or we can plot directly the 3d figure using some special modules.

We can plot a 3d figure using the following bit of code:

```
fig1=plt.figure(6)
ax=plt.axes(projection='3d')
plt.title('The 3-D surface plot of the potential')
surf = ax.plot_surface(Y, X[::-1], phi, rstride=1, cstride=1, cmap=plt.cm
savefig("Assgn5_plot6.png")
```

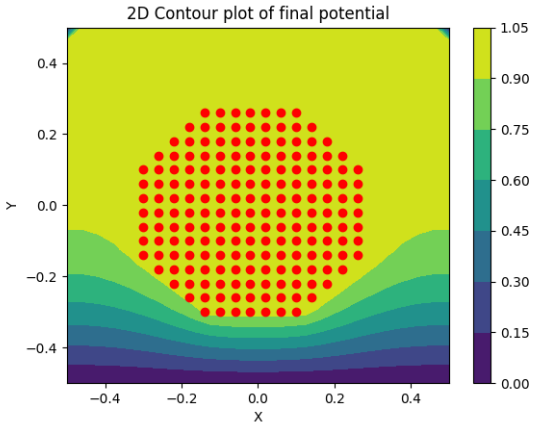

Figure 5: Figure5

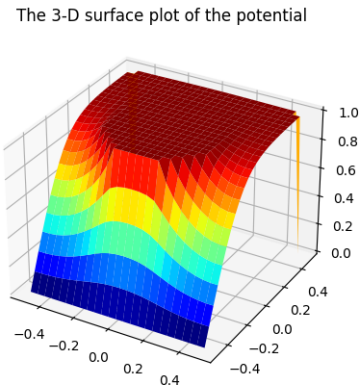

Figure 6: Figure6

Finding current vectors

The current vectors can be calculated as:

$$J(x, ij) = 0.5 * [\phi(i, j-1) - \phi(i, j+1)]$$

$$J(y, ij) = 0.5 * [\phi(i-1, j) - \phi(i+1, j)]$$

We see that, the current fills the entire cross-section and then flows in the direction of the grounded electrode. Hardly any current flows out of the top part of the wire.

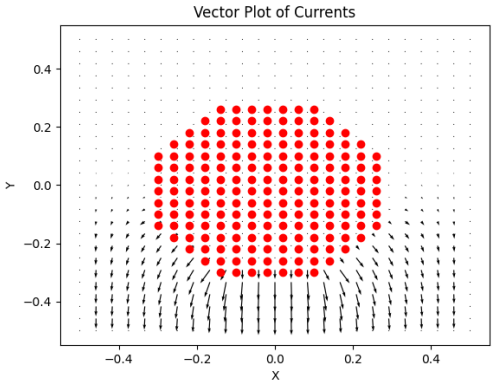

Figure 7: Figure7

Conclusion

We have used discrete differentiation to solve Laplace's Equations.This method of solving Laplace's Equation is known to be one of the worst available. This is because of the very slow coefficient with which the error reduces.