End Semester Exam

M Srinivas Reddy

August 1, 2020

1 Abstract

The aim of this assignment is to solve for potential matrix inside a metallic tank partially filled with a liquid. The top plate is connected to the rest of the tank via RLC circuit with the tank acting as a capacitor in series with an AC source.

2 Tasks at Hand

2.1 Potential Matrix

(d) Create a python function that solves Laplace's Equation and obtains ϕ_{mn} on the grid.:

The inputs given to the program are:

- M, the number of nodes along x, including the boundary nodes.
- N, the number of nodes along y, including the boundary nodes.
- k, the height given as the index k corresponding to h.
- δ, The desired accuracy.
- N0 the maximum number of iterations to complete.

The top plate is at 1 Volt. To calculate the potential matrix we first create a MxN matrix and use parallelization techniques and enforce boundary conditions for the top plate(which is at 1V) and other plates(connected to ground).

Algorithm for calculating potential matrix:

```
def tankvol(k):
 prev=tank.copy()
 tank[1:-(k+1),1:-1]=0.25*(tank[1:-(k+1),0:-2]+
 tank[1:-(k+1),2:]+
```

```
tank[0:-(k+2),1:-1]+tank[2:-k,1:-1])
tank[-(k+1),1:-1]=(1/3)*(2*tank[-k,1:-1]+
tank[-(k+2),1:-1])
tank[-k:-1,1:-1]=0.25*(tank[-k:-1,0:-2]+
tank[-k:-1,2:]+
tank[-(k+1):-2,1:-1]+tank[-(k-1):,1:-1])
tank[1:-1,0]=tank[1:-1,1]
tank[1:-1,-1]=tank[1:-1,-2]
tank[-1,1:-1]=tank[-2,1:-1]
tank[1:,0],tank[1:,-1],tank[-1,:],tank[0,:]=0,0,0,Vs
return [tank,prev]
```

(c) How will you parallelize the computation? The m = k row has to be handled differently. Explain why your algorithm is efficient.

- As we can observe in the above code, we made use of different vectors including adjacent columns or rows and found summation of it rather than using a for loop to calculate values of each node separately.
- To separate the calculations at the boundary of the liquid surface we calculated the potential for rows of the vector up to the liquid surface, implemented the conditions at the liquid boundary and then calculated the potential for the rows inside the liquid.

2.2 Electric Field and Charges

(e) Run the code for h/Ly = 0.1,0.2,...,0.9 and obtain Qtop vs h and Qfluid vs h where Qtop is the charge on the top plate held at 1 volt, and Qfluid is the charge on the portion of the wall touching the dielectric fluid. Plot the same. Is it linear? Should it be?:

To obtain Qtop and Qfluid we first need to obtain electric fields inside the tank. To find this, similar to calculation of potential, we can use vectors to find the half of difference between adjacent potentials.

Calculating Electric Fields

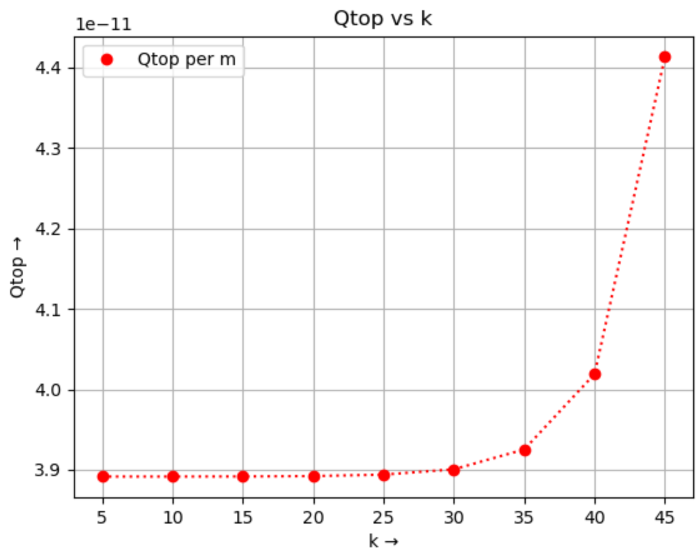
```
Ex=array([zeros(M)]*(N))
Ey=array([zeros(M)]*(N))
Exarr=array([[zeros(M)]*(N)]*9)
Eyarr=array([[zeros(M)]*(N)]*9)
......
Ex[:,0:-1]=-1*(tank[:,1:]-tank[0:,0:-1])*(M/Lx)
Ey[0:-1,:]=-1*(tank[1:,:]-tank[0:-1,:])*(N/Ly)
```

We have Ex and Ey arrays to store electric fields for a particular value of $\frac{h}{Ly}$ and Exarr and Eyarr to store multiple instances of it.
Qtop can be found out by summation of normal electrical field(Ey) at individual nodes multiplied by distance between nodes and ϵ_0. Similar calculation is done for Qfluid but is multiplied by an extra factor of ϵ_r.

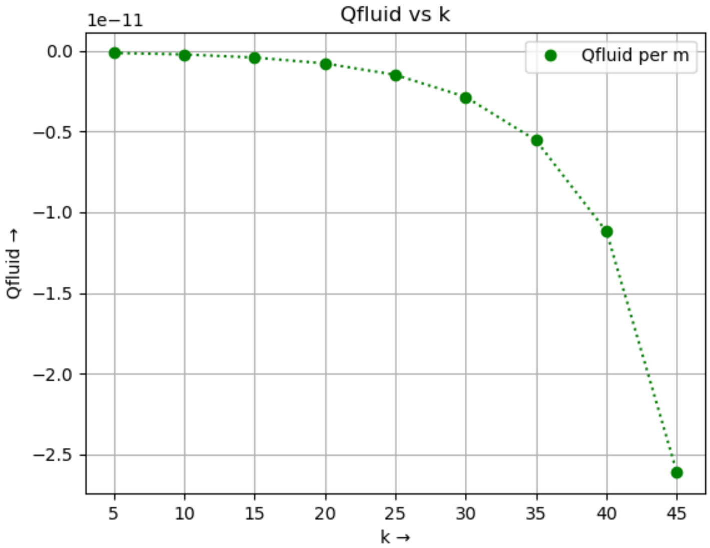
Qtop and Qfluid Calculation

```
for i in range(M):
    Qtop+=(Lx/M)*Ey[1,i]*8.854*10**(-12)
    Qfluid-=(Lx/M)*Ey[-2,i]*8.854*10**(-12)*2
for i in range(N-k-1,N):
    Qfluid+=(Ly/N)*(Ex[i,1]-Ex[i,-2])*8.854*10**(-12)*2
Qtoparr[kind]=Qtop
Qfluidarr[kind]=Qfluid
```

Plot of Qtop vs h/Ly



Plot of Qfluid vs h/Ly



As we can observe the plots are not linear. For understanding the behaviour, let us take a simpler version of this.

- Consider a tank with liquid which is an ideal conductor. If we consider the basic expression C = Aϵ/d we can observe that in this case as liquid level is increasing, the distance d is decreasing and hence capacitance increases and charge increases. Hence the increasing rational curve.
- We can also see that magnitude of Qfluid is also increasing. This is because as h increases the strength of electric field entering the fluid increases and hence the charge increases. This curve is also not linear due to the inverse relation of capacitance with distance.

(f) For h = 0.5, compute Ex and Ey on the at (m+0.5,n+0.5), i.e., at the centre of mesh cells. Show that Dn is continuous at m = k.

We know that at interface of 2 liquids the normal component of displacement vector should be equal.

$$D_{n1} = D_{n2} \tag{1}$$

$$\epsilon_1 E_1 = \epsilon_2 E_2 \tag{2}$$

$$E_1/E_2 = \epsilon_r = 2 \tag{3}$$

Continuity Verification

```
k1=int(karr[1])
E1x=Exarr[1][-k1-2,int(M/2-1)]
```

```
E1y=Eyarr[1][-k1-2,int(M/2-1)]
print('Ex in air: ',E1x)
print('Ey in air: ',E1y)
E2x=Exarr[1][-k1-1,int(M/2-1)]
E2y=Eyarr[1][-k1-1,int(M/2-1)]
print('Ex in fluid: ',E2x)
print('Ey in fluid: ',E2y)
print('ratio of normal components of electric fields ='
,E1y/E2y,'\n')
```

The ratio of electric fields turned out to be 1.9999 (very close to 2) and hence we can conclude that Dn is continuous at the surface of the liquid.

```
Enter Y/N for default values: y
The Number of iterations done for h/Ly= [ 5. 10. 15. 20. 25. 30. 35. 40. 45.]
[1409. 1025. 1067. 1119. 1178. 1232. 1269. 1276. 1176.]

Ex in air:  -0.013698257744842535
Ey in air:  0.35357606820780746
Ex in fluid:  -0.010648288192537146
Ey in fluid:  0.17678803411231084
ratio of normal components of electric fields = 1.9999999999048905
```

(g) Obtain the change in angle of the Electric field at m = k. Does this agree with Snell's Law? Should it?

Snell's law states that the ratio of sine of incident and refracted angle is equal to ratio of refractive indices or in the case of EM waves $\mu * \epsilon * \sigma$

$$\mu_1\epsilon_1\sigma_1 sin(\theta_r) = \mu_2\epsilon_2\sigma_2 sin(\theta_i)$$

Snells Law Verification

```
E1=complex(Exarr[4][-k1-1,13],Eyarr[4][-k1-1,13])
E2=complex(Exarr[4][-k1-2,13],Eyarr[4][-k1-2,13])
snell=cos(angle(E1))/cos(angle(E2))
print('The ratio of incident,refracted sine is:',snell)
```

Since the angles calculated from the complex numbers are angles made with the x-axis, cos(angle) was written to consider the correct angles. The ratio turned out to be around 0.9 and so Snell's law is not valid. This is because electric field on its own does not represent an EM Wave and so ratio will not be equal.

```
The ratio of sine of incident,refracted angle is: 1.2980658176747837
```

2.3 Resonant Frequency

(b) Develop your algorithm to determine h from the observed resonant frequency and present it.

We know that the top plate is connected to the rest of the tank via a series RLC circuit and an AC source, where the tank is acting as a capacitor. In a series RLC circuit, we know that resonant frequency of the circuit is

$$\omega = \frac{1}{\sqrt{LC}}$$

If the value of the inductor is L and the width of the tank is z and , then
Algorithm for calculating h:

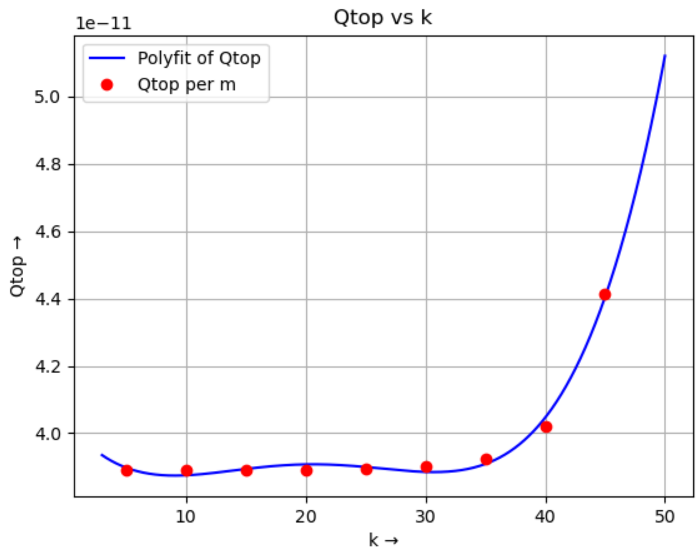
```
#Finding Q
omega = Value1
L = Value2
z = Value3
C = 1/((omega**2)*L)
Qtopres = C/z*

#Using Polyfit
deg=4
coeff=polyfit(karr,Qtoparr,deg)
coeff[deg]-=Qtopres
ksol=root(coeff)

#Condition to find correct root
for i in deg:
    if angle(ksol[i])==0 and ksol[i]<=N:
        hsol=ksol[i]*Ly/N
        print('h=',hsol)
    else:
        print('No Solution')
```

As we calculated Qtop previously for for different values of h/Ly, we can fit the data to a polynomial Q(h). As the resonant frequency is given, we can the calculate Qtopres from it and find the corresponding h value in the polynomial.

Plot of 4th degree polynomial fit to Qtoparr



If we increase the degree of the polynomial, the polynomial will be better fit to the data. Here a polynomial of degree 4 is chosen. Also once we get the solutions, another condition of k being real and positive were implemented to find the exact solution.

3 Other Algorithms Involved

3.1 Inputs to the Program

The program has been designed to accept multiple inputs from the user for obtaining the values of M,N,k,δ,N0. The user has the option of choosing the default values or custom inputs by pressing [Y/N].

```
opt=input('Enter Y/N for default values: ')
if opt=='Y' or opt=='y':
    M=25
    N=50
    k=6
    delta=0.0000001
    N0=1500
elif opt=='N' or opt=='n':
    M=int(input("Number of nodes along x(M): "))
    N=int(input("Number of nodes along y(N): "))
    k=int(input("Index height of liquid: "))
    delta=float(input("Least value of error: "))
    N0=int(input('Max no of iterations: '))
else:
```

```
    print('Invalid Input')
    quit()
```

The inclusion of Δ(Distance between nodes) was not possible as the distance was already fixed given M,N and Lx,Ly.

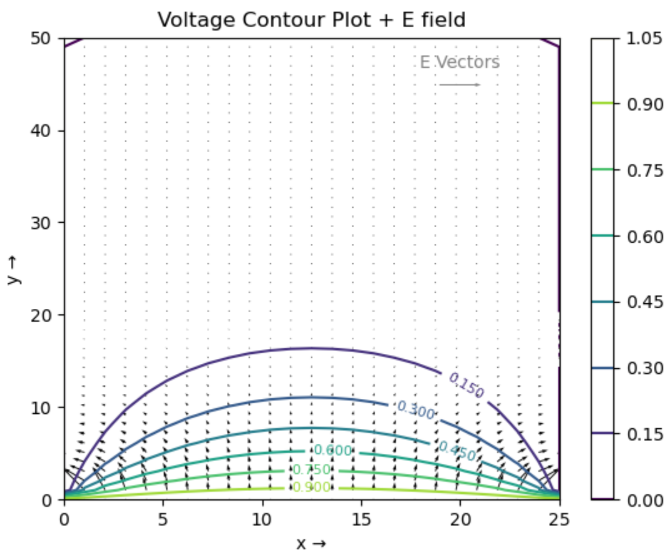
3.2 Quiver and Contour Plots

Plotting of voltage contour and quivers along the direction of electric fields was fairly simple.

```
figure(0)
title('Voltage Contour Plot + E field')
cont=contour(x,y,tankarr[4])
colorbar(cont)
clabel(cont,inline=True,fontsize=8)
xlabel('x \u2192',size=10)
ylabel('y \u2192',size=10)
q1=quiver(x,y,Exarr[2],Eyarr[2],scale=1.5,
scale_units='inches',label='E Vectors')
quiverkey(q1,0.8,0.9,0.5,label='E Vectors',
labelcolor='grey',color='grey')
```

The **quiverkey** was used to create a legend for the electric field.

Plot of Potential and E Field



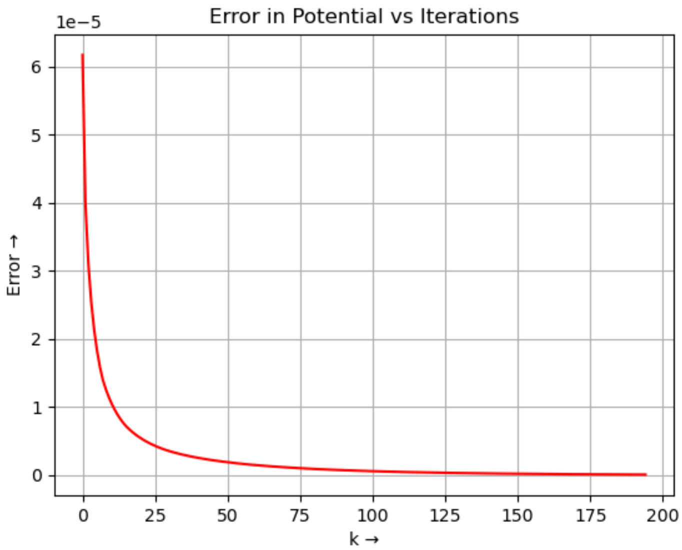
3.3 Error Calculation to a certain Accuracy

To calculate error we first create an array of 'ones' of length N0. We then calculate the error for the first time without applying any condition and then subsequent times by enforcing condition of **error**$\leq \delta$ and break the loop after the condition is satisfied.

```
k=int(k)
kind=int(k*10/N)-1
for i in range(N0):
    if error[kind][i-1]>=delta:
        tankvol(k)
        error[kind][i]=abs(tankvol(k)[0]-
        tankvol(k)[1]).max()
    else:
        maxiarr[kind]=i
        break
#array of no of iterations done for each value of h/Ly
print('The Number of iterations done for h/Ly=',karr)
print(maxiarr)
```

We do this for different values of h/Ly and hence different values of k and store them in a matrix of suitable size. We can also see that we are storing the value of i after the accuracy limit has been reached in an array called maxiarr[]. In this way we can find out number of iterations taken to reach the accuracy limit for different values of k.

Plot of Error vs Iterations for different values of k



We can clearly observe that the error is exponentially decreasing with increase in iterations as expected.

4 Conclusion

On analysing this problem we can conclude that a tank filled with liquid can be used to change resonant frequency by change the height of the liquid. Also use of python made this 3d problem much simpler to analyse and change additional parameters if required. We observed that we could easily determine the level of liquid inside the tank by changing the frequency of the voltage source. This saves us from directly measuring the level of the liquid in the tank using physical methods.