FYS-1120 Oblig 1

Oskar Idland

a)

Ettersom en partikkel i origo vil oppleve like krefter fra alle retninger vil netto feltet være lik null. Uten noen annen informasjon vet vi at potensialet er konstant.

b)

Vi ser på én av linjene først og bruker superposisjon for å finne potensialet av alle linjene. Vi bruker linjen som ligger fremst på x-aksen med endepunkter i $(a, -a, 0)$ og $(a, a, 0)$.

$$V(\mathbf{z}) = \int_C \frac{\rho_l \mathrm{d}l}{4\pi\epsilon_0 R}$$

$$V(\mathbf{z}) = \int_C \frac{Q}{32a\pi\epsilon_0 R}$$

$$\mathbf{R} = \mathbf{r} - \mathbf{r}' = (0, 0, z) - (a, y, 0) = (-a, -y, z)$$

$$V(\mathbf{z}) = \int_{-a}^{a} \frac{Q}{32a\pi\epsilon_0 \sqrt{a^2 + y^2 + z^2}} \mathrm{d}y$$

$$C = \sqrt{a^2 + z^2}$$

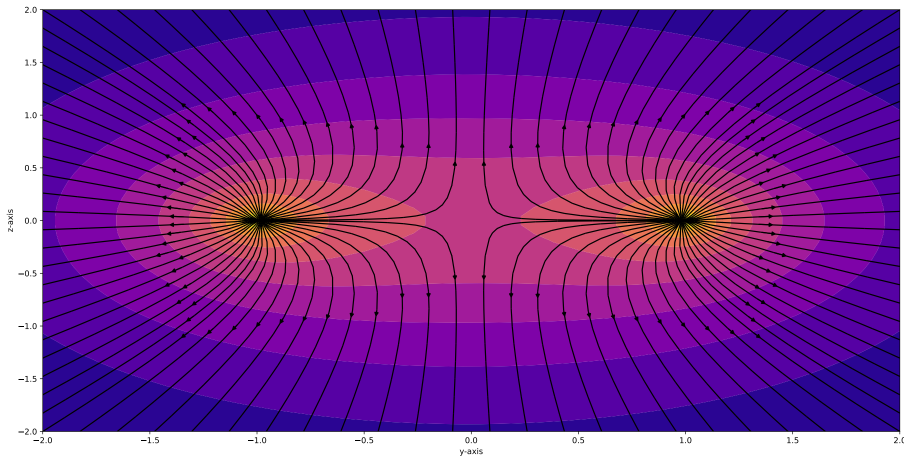
$$V(\mathbf{z}) = \int_{-a}^{a} \frac{Q}{32a\pi\epsilon_0 \sqrt{C^2 + y^2}} \mathrm{d}y$$

$$V(\mathbf{z}) = \frac{Q}{32a\pi\epsilon_0} \left[\operatorname{arcsinh} \frac{y}{C} \right]_{-a}^{a} = \frac{Q}{16a\pi\epsilon_0} \operatorname{arcsinh} \left(\frac{a}{\sqrt{a^2 + z^2}} \right)$$

Nå har vi funnet potensialet fra én av linjene. Ganger vi med fire får vi potesnialet fra alle.

$$V(\mathbf{z}) = \frac{Q}{4a\pi\epsilon_0} \operatorname{arcsinh} \left(\frac{a}{\sqrt{a^2 + z^2}} \right) \tag{1}$$

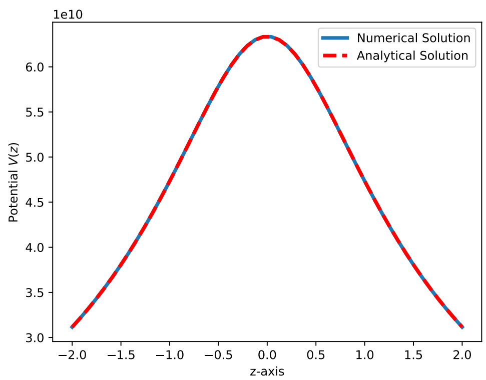
c)



Figur 1: Kontur plot av elektrisk potensialet og strømlinje plot av elektrisk felt

d)

Planet som ble plottet i forrige oppgave inneholder linjen til z-aksen. Vi kan sammenligne den analytiske løsningen med verdiene fra den midterste søylen i matrisen V.



Figur 2: Numerisk vs Analytisk Løsning

e)

For å finne potensialet fra den øverste firkanten kan vi bruke samme likning som vi fant tidligere 1, men ladningen endrer fortegn og vi må forskyve z-posisjonen. Da får vi følgende.

$$V(z) = -\frac{Q}{4a\pi\epsilon_0} \operatorname{arcsinh}\left(\frac{a}{\sqrt{a^2 + (2a - z)^2}}\right) \tag{2}$$

Legger vi sammen potensialene fra likning 1 & 2 får vi følgende.

$$V_{tot}(z) = \frac{Q}{4a\pi\epsilon_0}\left(\operatorname{arcsinh}\left(\frac{a}{\sqrt{a^2 + z^2}}\right) - \operatorname{arcsinh}\left(\frac{a}{\sqrt{a^2 + (2a - z)^2}}\right)\right) \tag{3}$$

Kapasitansen er gitt ved følgende

$$C = \frac{A\epsilon_0}{d} = \frac{4a^2\epsilon_0}{2a}$$

$$C = 2a\epsilon_0 \tag{4}$$

Kode

```python
import numpy as np
import matplotlib.pyplot as plt
from scipy.constants import epsilon_0
from numba import njit

@njit
def epot4lines(r, L, rho_l, N, a):
    V = 0
    dl = L/N
    for i in range(int(-N/2), int(N/2 + dl)):
        r1 = np.array([a,i*dl,0])
        r2 = np.array([-a,i*dl,0])
        r3 = np.array([i*dl,a,0])
        r4 = np.array([i*dl,-a,0])
        R1 = np.linalg.norm(r - r1)
        R2 = np.linalg.norm(r - r2)
        R3 = np.linalg.norm(r - r3)
        R4 = np.linalg.norm(r - r4)
        dV1 = dl*rho_l / (4*np.pi*epsilon_0*(R1))
        dV2 = dl*rho_l / (4*np.pi*epsilon_0*(R2))
        dV3 = dl*rho_l / (4*np.pi*epsilon_0*(R3))
        dV4 = dl*rho_l / (4*np.pi*epsilon_0*(R4))
        V += dV1 + dV2 + dV3 + dV4
    return V

@njit
def efield4lines(r, L, rho_l, N, a):
    E = np.array([0.0,0.0,0.0])
    dl = L/N
    for i in range(int(-N/2), int(N/2 + dl)):
        r1 = np.array([a,i*dl,0])
        r2 = np.array([-a,i*dl,0])
        r3 = np.array([i*dl,a,0])
        r4 = np.array([i*dl,-a,0])
        R1 = r - r1
        R2 = r - r2
        R3 = r - r3
        R4 = r - r4
        R1_norm = np.linalg.norm(R1)
        R2_norm = np.linalg.norm(R2)
        R3_norm = np.linalg.norm(R3)
        R4_norm = np.linalg.norm(R4)

        k = dl*rho_l / (4*np.pi*epsilon_0)
        dE1 = R1/R1_norm**3
        dE2 = R2/R2_norm**3
        dE3 = R3/R3_norm**3
        dE4 = R4/R4_norm**3
        E += k*(dE1 + dE2 + dE3 + dE4)

    return E

N = 50
L = 2
y = np.linspace(-L, L, N)
z = np.linspace(-L, L, N)
ry, rz = np.meshgrid(y,z)
rho_l = 1
```

```
a = 1
V = np.zeros((N,N))
Ey, Ez = np.zeros((N,N)), np.zeros((N,N))
for i in range(len(V.flat)):
    r = np.array([0, ry.flat[i], rz.flat[i]])
    field = efield4lines(r, L, rho_l, N, a)
    Ey.flat[i], Ez.flat[i] = field[1], field[2]
    V.flat[i] = epot4lines(r, L, rho_l, N, a)



plt.contourf(ry, rz, V, 10, cmap = 'plasma')
plt.streamplot(ry,rz,Ey,Ez, color = 'black', density = .8, broken_streamlines = False)
plt.xlabel('y-axis')
plt.ylabel('z-axis')
plt.show()


# d)

def AnalPot(z,a,Q):
    return Q/(4*np.pi*epsilon_0)* np.arcsinh(a/np.sqrt(a**2 + z**2))

Q = 4*rho_l*L
Vz = V[:,int(N/2)]
z = np.linspace(-L,L,N)
VzAnal = AnalPot(z,a,Q)
plt.plot(z,Vz, label = "Numerical Solution", linewidth = 3)
plt.plot(z,VzAnal, label = "Analytical Solution", linestyle = "--", color = 'red', linewidth = 3)
plt.xlabel('z-axis')
plt.ylabel(r'Potential $V(z)$')
plt.legend()
plt.savefig('Num_vs_Anal.pdf')
plt.show()
```