MEK1100, vår 2021

Ioanna Maria Lazarou

Obligatorisk oppgave 2 av 2

April 30, 2021

Det fulle hastighetsfeltet er: $v = u\hat{i} + v\hat{j} + w\hat{k}$

(a) Vi laster ned fila og les den inn Python. Vi sjekker at hver matrise har 194 punkter i x-retning og 201 punkter i y-retning, tilsammen 38994 punkter i xy-planet. Vi sjekker også at de to vektorene har 194 punkter i x-retning.

Til slutt sjekker vi at griddet i xy-planet er regulært med intervall 0.5 mm i begge retninger og at y-koordinatene spenner ut hele diameteren til røret.

Koden:

```
import numpy as np
import matplotlib.pyplot as plt

### a)
import scipy.io as sio
data = sio.loadmat('data.mat')
x = data.get('x')
y = data.get('y')
u = data.get('u')
v = data.get('v')
xit = data.get('xit')
yit = data.get('yit')
print(f"{x}\n")
print(y)

dat = [x,y,u,v,xit,yit]

#Sjekker informasjon om matrisene og vektorene
for j in range(6):
    value = []
    indx = ['x','y','u','v','xit','yit']
    for i in dat:
        value += [np.shape(i)]
        punkt = [(x * y) for x, y in value]
    if j < 4:
        print(f"Matrisen {indx[j]} har (x,y) = {value[j]}")
    elif j >= 4:
        print(f"Vektoren {indx[j]} har (x,y) = {value[j]}")
print(f"Tilsammen [x,y,u,v,xit,yit] har {punkt} punkter i xy-planet")

#test funksjoner
#Sjekk at griddet i xy-planet er regul rt med intervall 0.5 mm i begge retninger
def test_griddet(x):
    tol = 1E-14
```

```
    for i in x:
        for j in range(194-1):
            t = i[j+1]-i[j]
            success = abs(t - 0.5) < tol
            assert success

# Sjekk at y-koordinatene spenner ut hele diameteren til r ret
def test_y_kord(ykord):
    tol = 1E-14
    ykord = []
    for i in y:
        ykord += [i[1]]
    for j in range(194):
        t = ykord[j+1]-ykord[j]
        success = abs(t - 0.5) < tol
        assert success

#kaller test funksjoner
test_griddet(x)
test_y_kord(y)

"""
[[ 0.    0.5   1.   ... 95.5  96.   96.5]
 [ 0.    0.5   1.   ... 95.5  96.   96.5]
 [ 0.    0.5   1.   ... 95.5  96.   96.5]
 ...
 [ 0.    0.5   1.   ... 95.5  96.   96.5]
 [ 0.    0.5   1.   ... 95.5  96.   96.5]
 [ 0.    0.5   1.   ... 95.5  96.   96.5]]

[[-50.   -50.   -50.   ...  -50.   -50.   -50. ]
 [-49.5  -49.5  -49.5  ...  -49.5  -49.5  -49.5]
 [-49.   -49.   -49.   ...  -49.   -49.   -49. ]
 ...
 [ 49.    49.    49.   ...   49.    49.    49. ]
 [ 49.5   49.5   49.5  ...   49.5   49.5   49.5]
 [ 50.    50.    50.   ...   50.    50.    50. ]]

Matrisen x har (x,y) = (201, 194)
Matrisen y har (x,y) = (201, 194)
Matrisen u har (x,y) = (201, 194)
Matrisen v har (x,y) = (201, 194)
Vektoren xit har (x,y) = (1, 194)
Vektoren yit har (x,y) = (1, 194)
Tilsammen [x,y,u,v,xit,yit] har [38994, 38994, 38994, 38994, 194, 194] punkter
i xy-planet
"""
```

Alt ser ut til å være riktig. Vi ser at matrisene og vektorene har riktig antall punkter og at intervallene er 0.5 mm og at y går fra -50 mm til 50 mm.

(b) Vi ser på strukturen i både gassfasen og væskefasen. Vi lager et konturplott av hastighetskomponentene i xy-planet. Vi lager en variabel $H = sqrt(u^2 + v^2)$ der u og v er matrisene som inneholder komponentene i hastighetsfeltet. Dessuten tegner vi inn posisjonen til skilleflaten i plottet, ved å bruke en serie med en svart farge som skiller seg ut fra konturplottet. Vi kan se konturplottet i 1

Koden:

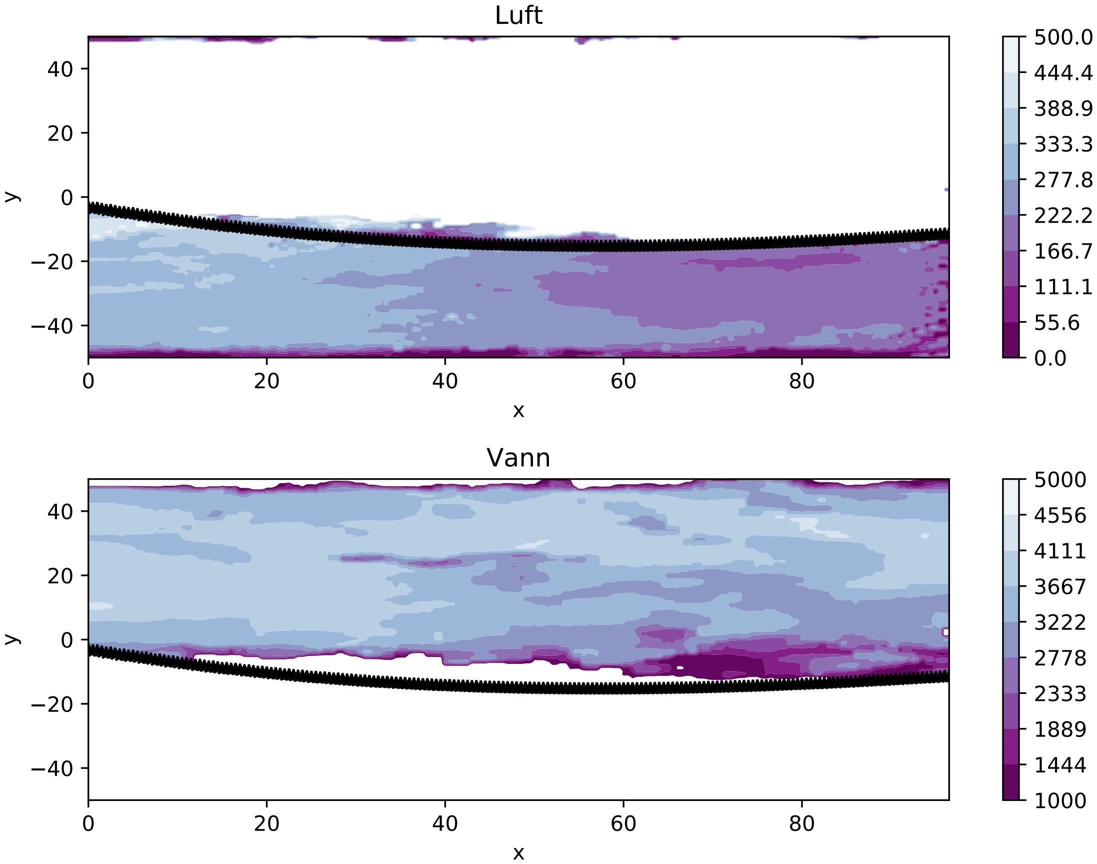

Figure 1: Konturplott som illustreres farten $\sqrt{u^2+v^2}$ for hastighetskomponentene i xy-planet med fargeskala. Den skilleflaten vises med svart farge.

```
### b)

H = np.sqrt(u**2 + v**2) #farten for hastighetskomponentene i xy-planet
plt.figure(figsize=(8, 6))
plt.subplot(2, 1, 1)
plt.title('Luft')
plt.xlabel('x')
plt.ylabel('y')
plt.plot(xit, yit, "k*")
plt.contourf(x, y, H, np.linspace(0, 500, 10), cmap=plt.cm.BuPu_r)
plt.colorbar()

plt.subplot(2, 1, 2)
plt.title('Vann')
plt.xlabel('x')
plt.ylabel('y')
plt.plot(xit, yit, "k*")
plt.contourf(x, y, H, np.linspace(1000, 5000, 10), cmap=plt.cm.BuPu_r)
plt.colorbar()

plt.tight_layout()
plt.savefig("b.pdf")
```

```
plt.show()

"""
(plot)
"""
```

(c) Vi lager et vektor pilplott med hastigheten i xy-planet $u\hat{i} + v\hat{j}$. Vi betrakter tre mindre områder i hastighetsfeltet ved å lage tre rektangler definert ved indeksene til hjørnene (ix, iy). Vi bruke forskjellige farger for hver side i rektanglene: rødt nede (side 1), grønt på høyre side (side 2), blått oppe (side 3) og svart på venstre side (side 4). Vi kan se figuren i 2.

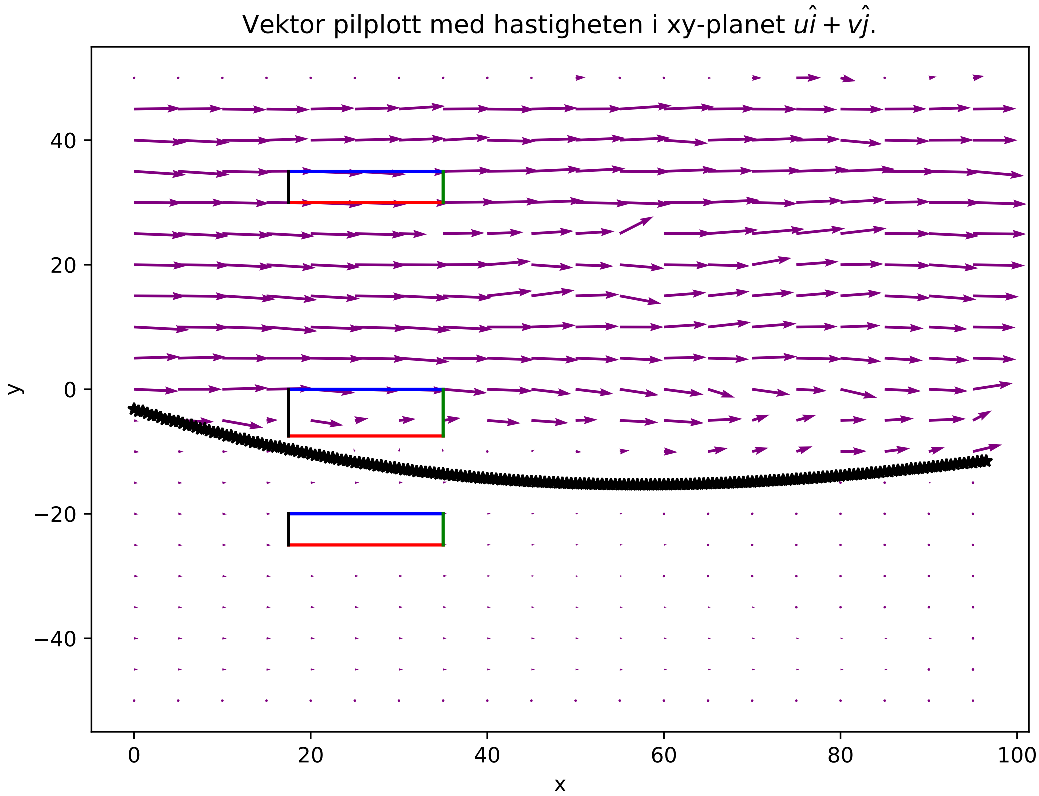

Figure 2: Vektor pilplott med hastigheten i xy-planet ui + vj. Tre rektangler er definert ved indeksene til hjørnene (ix, iy). Den skilleflaten indikeres med svart farge. To av rektanglene ligger i gassfasen(over skilleflaten) og ett av rektanglene ligger i væskefasen(nedover i plotten).

Koden:

```
### c)

n=10 # vi plotter en pil per n=10 punkter
plt.figure(figsize=(8, 6))
plt.quiver(x[::n, ::n], y[::n, ::n], u[::n, ::n], v[::n, ::n], color='purple')

def rektangler(xi,yi,xj,yj):
    x1 = x[yi][xi]; x2 = x[yj][xj]
    y1 = y[yi][xi]; y2 = y[yj][xj]
```

```
    plt.plot([x1,x2],[y1,y1], color='red')
    plt.plot([x2,x1],[y2,y2], color='blue')
    plt.plot([x1,x1],[y1,y2], color='black')
    plt.plot([x2,x2],[y2,y1], color='green')


rektangler(35,160,70,170)
rektangler(35,85,70,100)
rektangler(35,50,70,60)

plt.plot(xit, yit, "k*") # skilleflaten
plt.xlabel('x')
plt.ylabel('y')
plt.title('Vektor pilplott med hastigheten i xy-planet $u\hat{i} + v\hat{j}$.')
plt.savefig('c.pdf')
plt.show()

"""
(plot)
"""
```

(d) Vi regner divergensen til $(ui + vj)$:

$$\nabla \cdot (ui + vj) = (i\frac{\partial}{\partial x} + j\frac{\partial}{\partial y}) \cdot (ui + vj) = \frac{\partial}{\partial x}u + \frac{\partial}{\partial y}v$$

Vi vet at v er definert ved $v = u\vec{i} + v\vec{j} + w\vec{k}$.

Divergensen til v er:

$$\nabla \cdot v = (i\frac{\partial}{\partial x} + j\frac{\partial}{\partial y} + k\frac{\partial}{\partial z}) \cdot (ui + vj + wk) = \frac{\partial}{\partial x}u + \frac{\partial}{\partial y}v + \frac{\partial}{\partial z}w$$

Vi ser at divergensen til $u\vec{i}+v\vec{j}$ er ikke lik som divergensen til v fordi den mangler w-komponenten.

Vi lager et konturplott(fig 3) av divergensen slik at strukturen kommer tydelig fram i både gass og væskefasen. og vi tegneg inn skilleflaten og rektanglene i samme plott.

Koden:

```
### d)

# divegrens
dudx = np.gradient(u, axis=0)
dvdy = np.gradient(v, axis=1)
div = dudx + dvdy

plt.figure(figsize=(8, 6))
divergence = plt.contourf(x, y, div, cmap=plt.cm.BuPu_r)
plt.colorbar(divergence)

plt.plot(xit, yit, "k*") # skilleflaten
rektangler(35,160,70,170)
rektangler(35,85,70,100)
rektangler(35,50,70,60)
```

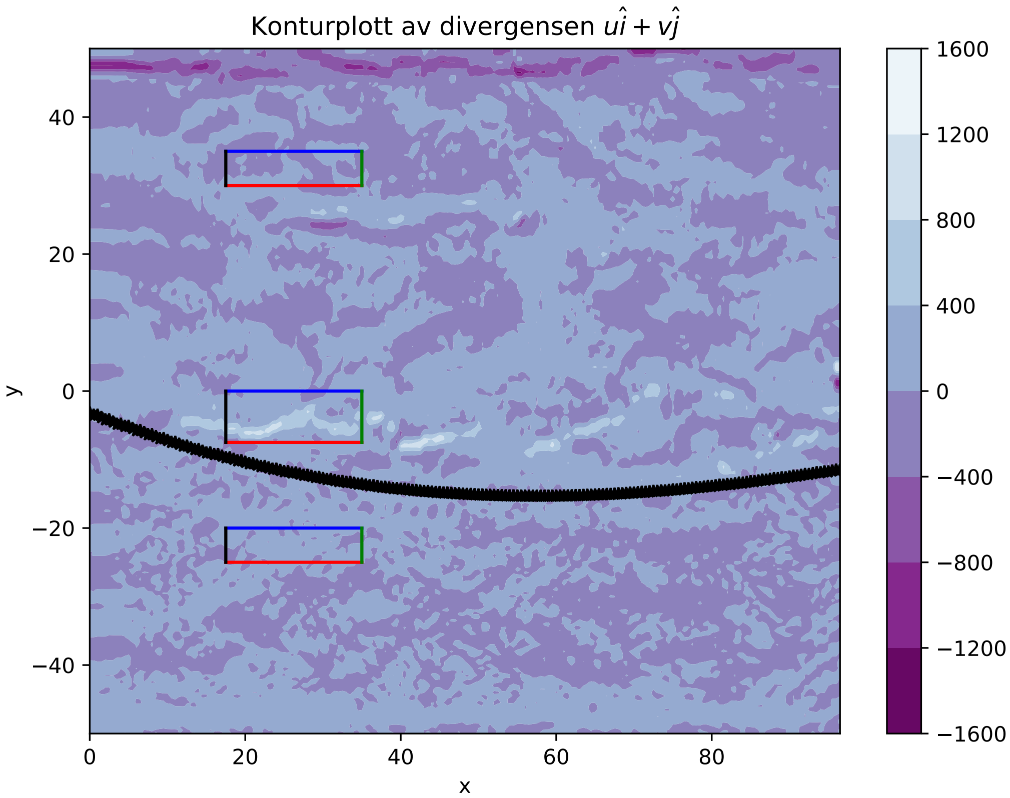

Figure 3: konturplott av divergensen $u\hat{i} + v\hat{j}$ (gass og væskefasen) med samme rektanglene fra oppgave c) og skilleflaten.

```
plt.xlabel('x')
plt.ylabel('y')
plt.title('Konturplott av divergensen $u\hat{i} + v\hat{j}$')
# plt.savefig('d.pdf')
plt.show()
```

Vi antar at både gassen og væska er inkompressible fordi strømingen er betydelig langsommere enn lydhastigheten i luft og i vann.

Konsekvensen er at divergensen til v er 0 fordi siden fluidene er inkompressible betyr dette at de har konstant tetthet ($\nabla \cdot v = 0$)

$$\nabla \cdot v = 0 \Rightarrow \frac{\partial}{\partial x}u + \frac{\partial}{\partial y}v + \frac{\partial}{\partial z}w = 0$$

$$\frac{\partial}{\partial z}w = -(\frac{\partial}{\partial x}u + \frac{\partial}{\partial y}v)$$

$$w = -\int (\frac{\partial}{\partial x}u + \frac{\partial}{\partial y}v)dz$$

Dermed w negerer verdiene for divergensen til $u\vec{i} + v\vec{j}$.

(e) Virvlinga til v:

$$\nabla \times v = \begin{vmatrix} i & j & k \\ \frac{\partial}{\partial x} & \frac{\partial}{\partial y} & \frac{\partial}{\partial z} \\ u & v & 0 \end{vmatrix}$$

$$= -\hat{i}\frac{\partial}{\partial z}v + \hat{j}\frac{\partial}{\partial z}u + \hat{k}(\frac{\partial}{\partial x}v - \frac{\partial}{\partial y}u)$$

k står normalt på både i og j derfor plotter vi virvlingskomponenten $\hat{k}(\frac{\partial}{\partial x}v - \frac{\partial}{\partial y}u)$.

Koden

```
### e)

# virvling
k = np.gradient(v, axis=1) - np.gradient(u, axis=0)

plt.figure(figsize=(8, 6))
plt.contourf(x, y, k, cmap=plt.cm.BuPu_r)

strm = plt.streamplot(x, y, u, v, color=u, linewidth=2, cmap='Purples') # str mlinj
plt.colorbar(strm.lines)

plt.plot(xit, yit, "k*") # skilleflaten
rektangler(35,160,70,170)
rektangler(35,85,70,100)
rektangler(35,50,70,60)

plt.xlabel('x')
plt.ylabel('y')
plt.title('Konturplott av $\mathbf{k}$ virvlingskomponenten')
# plt.savefig('e.pdf')
plt.show()

"""
(plot)
"""
```

Vi kan se konturplott av $\hat{k}$ virvlingskomponenten i figur 4. Vi plotter strømlinjer også for både gass og væskefasen og vi tegn inn skilleflaten i samme plott. Fargen i strømlinjene indikerer gass og væskehastighet. Strømmen for væskefasen er mye tregere enn for gassfasen. Strømningen for væskefasen er veldig turbulent ved veggen mens strømningen for gassfasen er ikke turbulent ved veggen. Strømningen for både for væskefasen og for gassfasen er turbulent ved skilleflaten.

(f) Nå anvende vi Stokes på rektanglene. Formelen:

$$\int_s \nabla \times v \cdot n ds = \oint_c v \cdot dr$$

Sirkulasjonen til rektanglene kan deles i fire integraler, en per side. Vi oppsummerer disse integralene for hver av rektanglene for å få sirkulasjonen som går rundt dem mot klokka. Et flateintegral betyr dobbelt integral. Vi multiplisere summen med både Δx og Δy. Vi ganger summen med 0.25 fordi begge to er 0.5. I koden under så definerer vi en funksjon for å regne ut kurveintegralet og

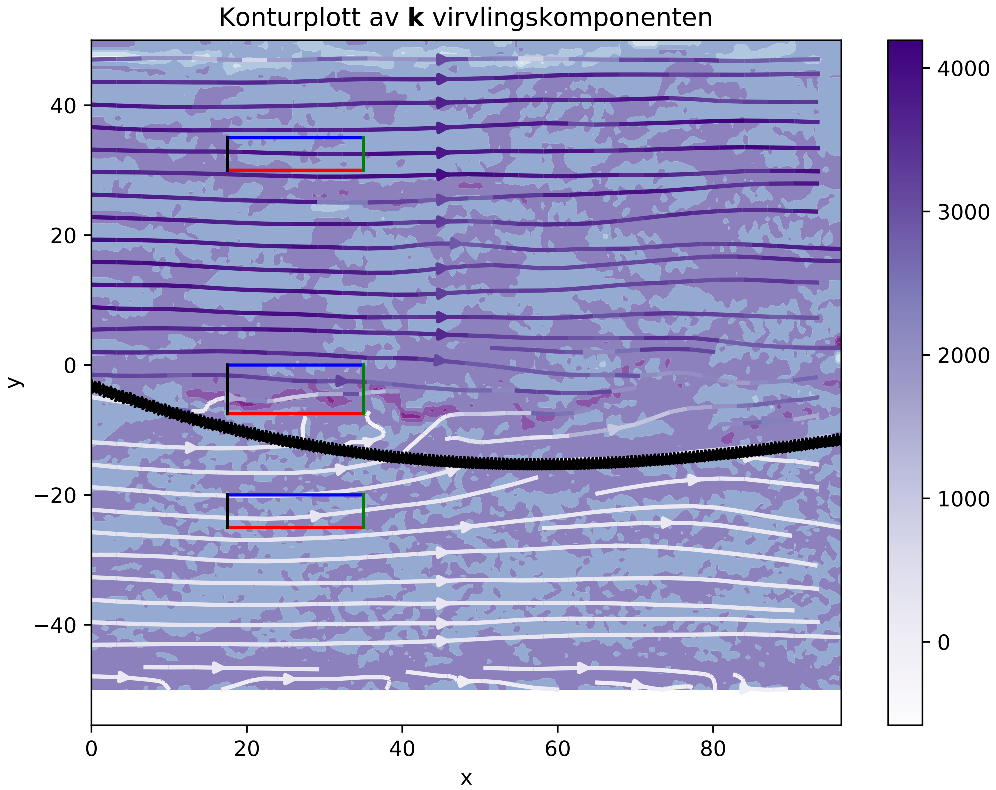

Figure 4: Konturplott (med strømlinjer for både gass og væskefasen med rektanglene og skilleflaten) av $\hat{k}(\frac{\partial}{\partial x}v - \frac{\partial}{\partial y}u)$ virvlingskomponenten.

en for flateintegralet. Disse tar inn koordinatene til nedre venstre, og øvre høyrehjørne, og regner ut sirkulasjonen. For å få en bedre forståelse av sirkulasjonen angir vi verdien til kurveintegralene langs hver side av rektanglene.

Koden og resultatene:

```
### f)

# sirkulasjon

# kurveintegral
def kurveintegral(x1,y1,x2,y2):
    dt = 0.5
    side1 = sum(u[y1,x1:x2+1]*dt)
    side2 = sum(v[y1:y2+1,x2]*dt)
    side3 = -sum(u[y2,x1:x2+1]*dt)
    side4 = -sum(v[y1:y2+1,x1]*dt)
    sirkulasjon = side1 + side2 + side3 + side4
    return side1, side2, side3, side4, sirkulasjon

# flateintegral
def flateintegral(x1,y1,x2,y2):
```

```
    virvl = np.gradient(v,0.5,axis=1) - np.gradient(u,0.5,axis=0)
    S = np.sum(virvl[y1:y2+1,x1:x2+1])*0.25
    return S

print ('Rektangel 1')
print ('---------------------------------------------------')
a,b,c,d,s = kurveintegral(34,159,69,169)
print("side        kurveintegral")
print(f"1               {a:.3f}")
print(f"2               {b:.3f}")
print(f"3               {c:.3f}")
print(f"4               {d:.3f}")
print (f"kurveintegral: {s:.3f}")
S = flateintegral(34,159,69,169)
print(f"flateintegral: {S:.3f}")
print(f"forskjell (kurve - flateintegral): {np.abs(s - S):.3f}")
print()


print ('Rektangel 2')
print ('---------------------------------------------------')
a,b,c,d,s = kurveintegral(34,85,69,99)
print("side        kurveintegral")
print(f"1               {a:.3f}")
print(f"2               {b:.3f}")
print(f"3               {c:.3f}")
print(f"4               {d:.3f}")
print (f"kurveintegral: {s:.3f}")
S = flateintegral(34,85,69,99)
print(f"flateintegral: {S:.3f}")
print(f"forskjell (kurve - flateintegral): {np.abs(s - S):.3f}")
print()

print ('Rektangel 3')
print ('---------------------------------------------------')
a,b,c,d,s =  kurveintegral(34,49,69,59)
print("side        kurveintegral")
print(f"1               {a:.3f}")
print(f"2               {b:.3f}")
print(f"3               {c:.3f}")
print(f"4               {d:.3f}")
print (f"kurveintegral: {s:.3f}")
S = flateintegral(34,49,69,59)
print(f"flateintegral: {S:.3f}")
print(f"forskjell (kurve - flateintegral): {np.abs(s - S):.3f}")


"""
Rektangel 1
---------------------------------------------------
side        kurveintegral
1               70100.524
2               266.274
3               -68332.856
4               661.573
kurveintegral: 2695.514
```

```
flateintegral: 2621.559
forskjell (kurve - flateintegral): 73.955

Rektangel 2
-----------------------------------------------------------
side        kurveintegral
1               652.329
2               118.499
3               -61243.465
4               -163.303
kurveintegral: -60635.940
flateintegral: -61095.332
forskjell (kurve - flateintegral): 459.392

Rektangel 3
-----------------------------------------------------------
side        kurveintegral
1               5133.348
2               207.910
3               -5410.040
4               78.303
kurveintegral: 9.521
flateintegral: -12.214
forskjell (kurve - flateintegral): 21.735
"""
```

I følge Stokes teorem er kurveintegral lik flatintegral, men men vi kan se at det er forskjell mellom kurven og overflateintegralene. Forskjellen kan for eksempel komme fra den numeriske metoden vår som kun er en tilnærming til gradienten. Vi vet at hastighetsfeltet er en måling som kan inneholde feil og vi kan ha begrenset oppløsning i griddet. Hvis vi ser på verdiene til hver av sidene, i de tre rektanglene som vi regna ut i kurveintegralet, så virker det som om det er høyere hastighet i lufta enn i væsken, og at hastigheten avtar når vi nærmer oss både toppen og bunnen av røret, noe som er forventet siden hastigheten i lufta er mye høyere enn den i vannet. Det stemmer ganske bra med det vi ser når vi ser på vektorplottet til hastighetsfeltet.

Rektangel 1 er det rektangelet som er øverst. Rektangel 3 ligger nederst. Det gir mening at disse tallene er lave siden verdiene kommer til å være ca. like, men med motsatt fortegn, omtrent like mye strømning langs topp og bunn. Rektangel 2 ligger i midten. Kurveintegralet her ble svært høyt negativt. Verdien langs side 1 av rektangelet er veldig mye lavere enn på toppen. Grunnen til dette er at det er masse virvling som er høyt negativt og da gir det mening at sirkulasjonen også er det. Resultatene fra de rektanglene 1 og 3 var forventet ved å se på hastighetsfeltet i området, men resultatene av rektangel 2 var mer overraskende.

(g) Siden væsken og gassen er inkompressible anvender vi Gauss sats på rektanglene og forutsette at $\nabla \cdot v = 0$ for det fulle hastighetsfeltet $v = u\hat{i} + v\hat{j} + w\hat{k}$. Fra Gauss sats kan vi da forvente at fluksen gjennom et lukket volum, skal være lik null. Hvis vi da hadde utvitet rektangelet til et prisme, så ville fluksen gjennom top og bunn summert seg sammen slik at den totale fluksen hadde blitt null, dvs. at rektanglene orientert i z-retning hadde fått en integrert fluxs tilsvarende de vi har regnet ut, bare med motsatt fortegn.

Formelen:

$$\int_\sigma v \cdot n d\sigma = \int_\tau \nabla \cdot n d\tau$$

Koden og resultatene:

```
### g)

# integrert flux
def flux_int(u,v,x1,y1,x2,y2):
    dt = 0.5
    I1= -sum(v[y1-1,x1-1:x2])*dt
    I2= sum(u[y1-1:y2,x2-1])*dt
    I3= sum(v[y2-1,x1-1:x2])*dt
    I4= -sum(u[y1-1:y2,x1-1])*dt
    return I1, I2, I3, I4, I1+I2+I3+I4

# gauss sats
def gauss(div,x1,y1,x2,y2):
    return sum(sum(div[y1-1:y2,x1-1:x2]))*0.25

print ('Rektangel 1')
print ('--------------------------------')
I1,I2,I3,I4,flux1 = flux_int(u,v,35,160,70,170)
print("side        kurveintegral")
print(f"1                {I1:.3f}")
print(f"2                {I2:.3f}")
print(f"3                {I3:.3f}")
print(f"4                {I4:.3f}")
print (f"flux: {flux1:.3f}")
gauss1 = gauss(div,35,160,70,170)
print(f"gauss: {gauss1:.3f}")
print()

print ('Rektangel 2')
print ('--------------------------------')
I1,I2,I3,I4,flux2 = flux_int(u,v,35,85,70,100)
print("side        kurveintegral")
print(f"1                {I1:.3f}")
print(f"2                {I2:.3f}")
print(f"3                {I3:.3f}")
print(f"4                {I4:.3f}")
print (f"flux: {flux2:.3f}")
gauss2 = gauss(div,35,85,70,100)
print(f"gauss: {gauss2:.3f}")
print()

print ('Rektangel 3')
print ('--------------------------------')
I1,I2,I3,I4,flux3 = flux_int(u,v,35,50,70,60)
print("side        kurveintegral")
print(f"1                {I1:.3f}")
print(f"2                {I2:.3f}")
print(f"3                {I3:.3f}")
print(f"4                {I4:.3f}")
print (f"flux: {flux3:.3f}")
gauss3 = gauss(div,35,50,70,60)
print(f"gauss: {gauss1:.3f}")

"""
Rektangel 1
```

```
------------------------------------------
side          kurveintegral
1                  1556.868
2                  21664.567
3                  -2059.677
4                  -21056.906
flux: 104.853
gauss: -599.927

Rektangel 2
------------------------------------------
side          kurveintegral
1                  -5187.564
2                  14782.533
3                  -4074.052
4                  -11997.856
flux: -6476.939
gauss: 30792.991

Rektangel 3
------------------------------------------
side          kurveintegral
1                  -195.570
2                  1536.822
3                  284.944
4                  -1750.764
flux: -124.569
gauss: -599.927
"""
```

Den integrerte fluxen ikke blir null derfor vet vi at det er noe bevegelse i z-retning. Fluxen på de fire sidene ble ikke null så det må være en hastighet i z-retningen. Bruker vi da Gauss for å beregne det samme området. Vi har regnet ut alle sidene i de rektanglene. Fluxen til rektangelet 1 er positiv så må det være et negativt element som drar ifra, dermed må det være noe som går inn i planet. Fluksen til rektanglene 2 og 3 er negativ, må det være noe som går ut i planet. Hvis vi ser på hastighetsfeltet ser det ut som at hastigheten holder seg nogen lunde konstant når vi beveger oss bortover x-aksen, og endrer seg mer vi beveger oss langs y-aksen. Vi ser også at hastigheten i x-retningen endrer seg raskest i skilleflaten, der rektangel 2 ligger. Dette ser vi også på fluksen der rektangel 2 er det rektangelet med størst absoluttverdi (6476.939) for den integrerte fluksen.