

Oblig 2

MEK1100

Lasse Steinnes

23. April 2018

Introduksjon

Formålet med oppgaven var å analysere og beskrive et hastighetsfelt i luftmasser og væske, målt langs lengdesnittet til en sylinder (radius: 5 cm). Datasettet ble samlet ved hjelp av metoden Particle Imaging Velocimetry (PIV), i Hydrodynamisk Laboratorium ved matematisk institutt. Hastighetsfeltet for røret kan beskrives som

$$\vec{v} = u\hat{i} + v\hat{j} + w\hat{k}.$$

Målingene ble kun foretatt for x- og y-komponenten, der \hat{i} går horisontalt langs røret og \hat{j} går vertikal. Det ble ikke foretatt noen målinger for \hat{k} , i bredden på sylinderet.

a) Nedlastning og oversikt over dataene

All beskrivelse av metode som følger hører til koden i vedlegg 1.

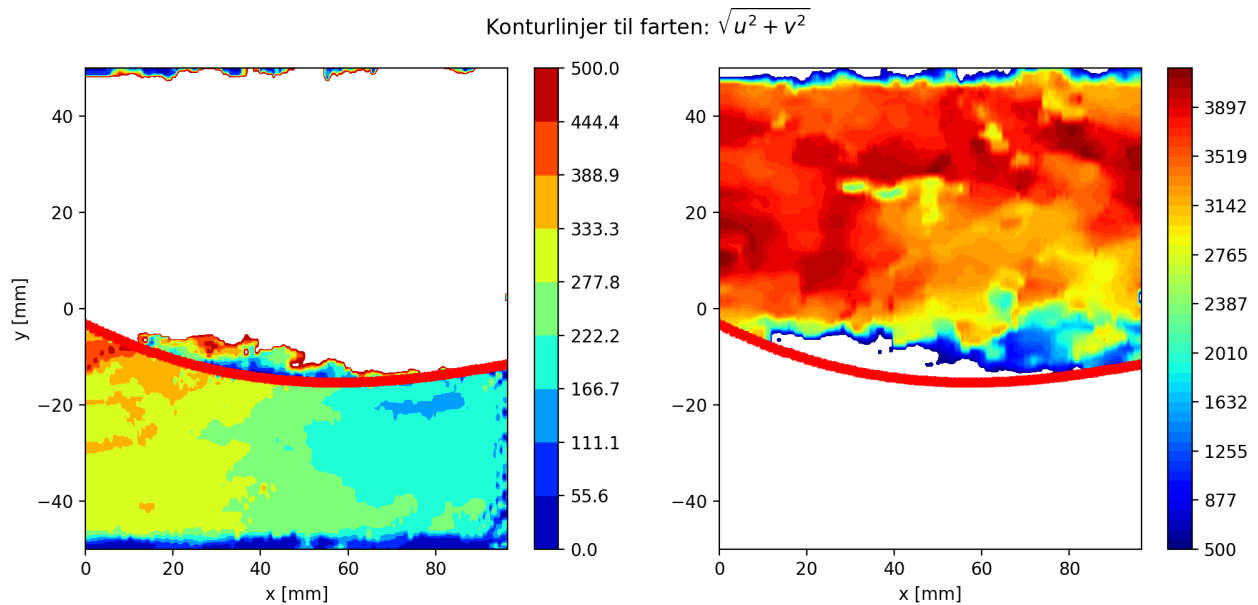
Datatabellene ble lastet ned ved hjelp av biblioteket scientific python's modul "input and output" ved hjelp av kommandoen loadmat for å laste dataene inn i programmeringsmiljøet, og data.getfor å tildele objekter spesifikke datatyper. Ved hjelp av numpy's shape ble det sjekket at hver matrise inneholdt 194 kolonner tilsvarende x-koordinater, og 201 rader tilsvarende y-koordinater. Den samme metoden ble benyttet for å sjekke at det var 194 punktmålinger for skillelinjen. Fra datasettet er det tydelig at matrisene X og Y tilsvarer et rutenett (grid) for å beskrive x-y-planet.

For å kontrollere at punktene hadde punktmålinger med intervall 0.5 mm, ble det kjørt en stillegående test som tar differansen mellom alle punkter til den spesifikke matrisen. Y-koordinatene spant ut hele diameteren til røret ($10\text{cm} = 100\text{mm}$), noe som ble beregnet

ved å ta summen av absoluttverdien til første og siste punktmåling i Y-matrisen. Dette følger fra et symmetrisk tverrsnitt.

b) Farten til gass -og væskefasen

Konturlinjer gir en oversikt over området der et skalarfelt har den samme verdien. Konturfelt ble fremstilt for farten $\mathbf{v} = \sqrt{u^2 + v^2}$ ved hjelp av pyplotmodulens kommando "contourf", som er tilgjengelig i matplotlib-biblioteket (1, vedlegg 2).

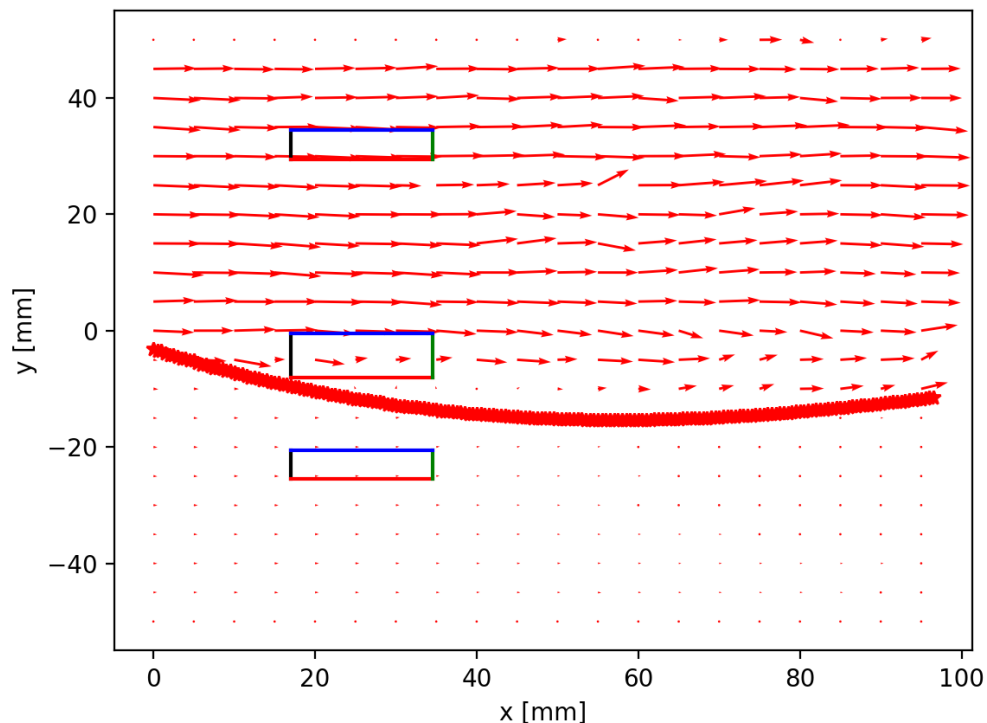


Figur 1: Konturlinjene til farten $\mathbf{v} = \sqrt{u^2 + v^2}$. Til venstre: Konturlinjer for $\mathbf{v} < 500$. Til høyre: Konturlinjer for $\mathbf{v} > 500$.

Hovedsakelig var farten i lufta større enn i væsken (1). Imidlertid var konturlinjene over skilleflaten for $\mathbf{v} < 500$. Dette kan være en konsekvens av at væsken bremset ned gassfasen der disse var i fysisk kontakt med hverandre. Den samme effekten kan observeres der gassfasen møter veggen iylinderet, øverst på y-aksen.

c) Vektorfelt for hastigheten

Et vektorfelt for hastigheten viser retningen og den relative størrelsen til hastigheten i hvert punkt som tas med i den grafiske fremstillingen. Det ble laget et vektorfelt for hastigheten $\vec{v} = u\hat{i} + v\hat{j}$ i xy-planet, ved hjelp av pyplot's "quiver", med uthevede områder av interesse (2, vedlegg 3).



Figur 2: Vektorfelt av hastigheten $\vec{v} = u\hat{i} + v\hat{j}$. De tre rektanglene indikerer henholdsvis to områder i gassfasen, og en i væskefasen.

Vektorfeltet for hastigheten og konturlinjene stemmer overens. Hastigheten var størst i gassfasen som ikke var i fysisk kontakt med veggen i cylinderen eller skillelinjen til vannet. Langs kantene av cylinderen ble gassfasen bremsset ned, og på grunn av kurven til skillelinja ble hastigheten ikke bare bremsset ned, men resulterte i virvling langs vannoverflaten. Merk at disse målingene kun gjelder xy-planet. Hastigheten til væsken var svært sakte i forhold til gassfasen, noe som kan forklares fra viskositetsegenskapene til

væsken.

d) Divergensen til \vec{v}

Divergensen er definert til å være

$$\nabla \cdot \vec{v} = \frac{\delta u}{\delta x} + \frac{\delta v}{\delta y} + \frac{\delta w}{\delta z}.$$

Fra definisjonen av den deriverte, kan divergensen beregnes numerisk ved å finne divergensen i hvert punkt. Dette kan gjøres for hver komponent i divergensen ved å ta gjennomsnittet av to intervaller

$$\begin{aligned} \left(\frac{\delta u}{\delta x}\right)_{i,j} &= \frac{1}{2} \left(\frac{u_{i+1,j} - u_{i,j}}{\Delta x} + \frac{u_{i,j} - u_{i-1,j}}{\Delta x} \right) \\ &= \frac{u_{i+1,j} - u_{i-1,j}}{2\Delta x}. \end{aligned}$$

Dette ble utført for alle komponentene, sett bort fra z-komponenten. Dermed tilsvarer ikke divergensen beregnet fra datamålingene, den faktiske divergensen. Konturlinjer for divergensen ble fremstilt med kommandoen "contourf"(3, vedlegg 4).

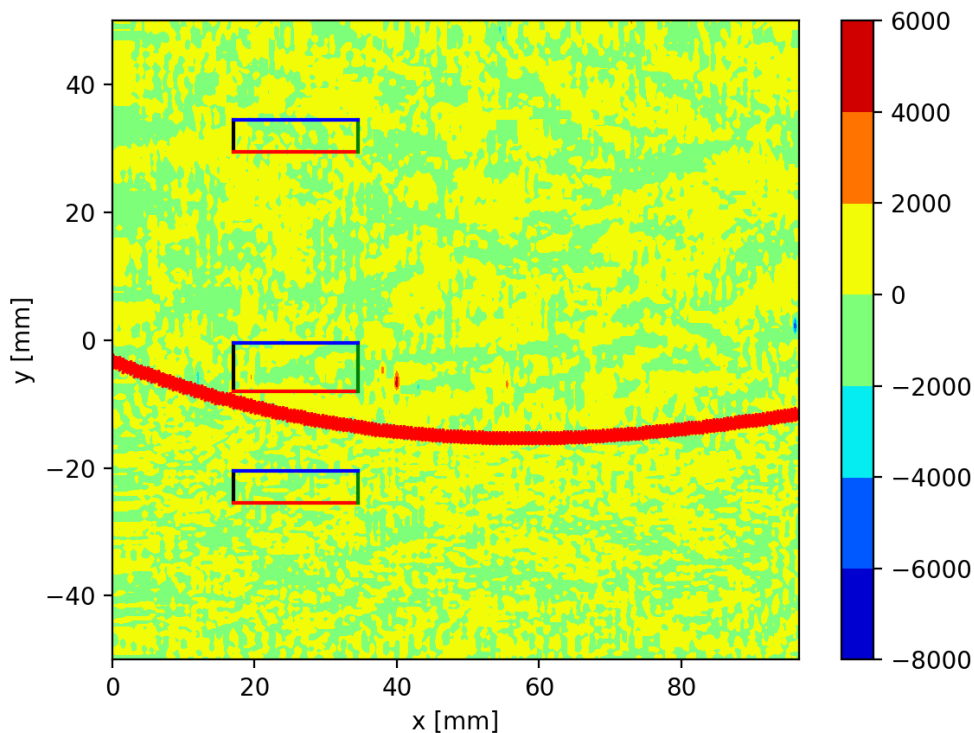
Divergensen var størst i gassfasen, med utbredte områder med konturfarger tilsvarende høyere divergens. Dette samsvarer med høyere hastighet i gassfasen, da divergensen er definert som den integrerte fluksen per volumelement. Den integrerte fluksen vil øke med økt hastighet i retningen ut av overflaten som utspenner volumet.

Siden gassen og væsken er inkomprible faser, forventes det at tettheten mellom partiklene i målingene ikke skal endres. Dette innebærer at den totale divergensen er lik 0. Som følge av at divergensen ikke var 0 i målingen for divergensen i xy-planet, må det også ha vært divergens i z-retning for å motvirke divergensen i de andre retningene. Når gassfasene skyver væsken fremover, vil det med andre ord også skyve en del væske til hver av sidene av cylinderen.

e) Virvlingen til \vec{v}

Virvlingen er definert som $\nabla \times \vec{v}$. Vertikalt på xy-planet ligger z-planet, og denne komponenten av virvlingen blir

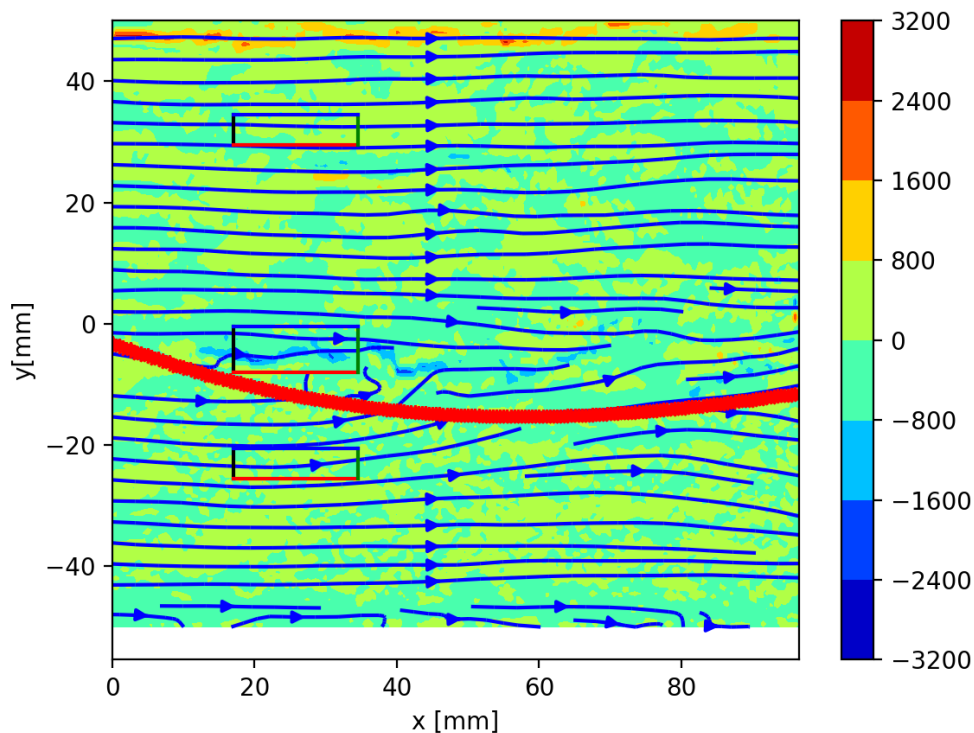
$$\text{curl}(v)\hat{k} = \left(\frac{\delta v}{\delta x} - \frac{\delta u}{\delta y}\right)\hat{k}.$$



Figur 3: Konturlinjer av divergensen i hastighetsfeltet $\vec{v} = u\hat{i} + v\hat{j}$. De tre rektanglene indikerer henholdsvis to områder i gassfasen, og en i væskefasen.

Virvlingen ble beregnet numerisk med en metode tilsvarende tilnærmingen til den deriverte benyttet i seksjon (d). Det ble fremstilt konturlinjer for virvlingen sammen med strømlinjene for hastighetsfeltet (4, vedlegg 5). Strømlinjene har punktene hastighetsfeltet som tangent, og beskriver hvordan en partikkel ville beveget seg langs gass -eller væskefasen.

I det midtre laget av gassfasen (øverste rektangel) var det lite virvling, noe som resulterte i tilnærmet horisontale strømlinjer (4). Imidlertid var det tydelig divergens over og under skillelinjen mellom de to fasene, noe som samsvarer med vektorfeltet for hastigheten (3). I væskefasen bøyer strømlinjene i y-retning, noe som tilsvarer positiv virvling. I gassfasen, rett over skillelinja, var virvlingen svært negativ, fordi gassen bremses ned over bølgetoppen. Langs bunn av sylindren ble væsken bremsset av veggen, og tilsvarende effekt kan observeres i strømlinjene. Den øverste veggen av



Figur 4: Konturlinjer av virvlingen i hastighetsfeltet $\vec{v} = u\hat{i} + v\hat{j}$. De blå linjene illustrer strømlinjer med hastighetsfeltet som tangent. De tre rektanglene indikerer henholdsvis to områder i gassfasen, og en i væskefasen.

sylinderen i y-retning har svært positiv virvling. Denne bremsningen kan forklares ved hjelp av friksjon mellom to faser/kjemiske forbindelser, og man kan forvente at dette skjer langs de andre kontaktoverflatene i cylinderen. Dermed burde det også være virvling x- og y-retningene. Dersom antakelsen om at virvling og friksjon holder vil virvlingen være størst der farten er størst, som følge av at friksjonen øker med hastighet.

f) Sirkulasjonen til hastighetsfeltet

Stokes sats kan utledes fra definisjonen av virvling, slik at

$$\oint \oint_A \vec{n} \cdot (\nabla \times \vec{v}) dA = \oint_S \vec{v} dr.$$

Sirkulasjonen rundt et rektangel i xy-planet kan dermed enten beregnes som et kurveintegral, eller et flateintegral, gitt at kurven er lukket. Beregningene ble utført numerisk ved hjelp av for-løkker, med tilnærming for integrasjon som sum (vedlegg 6). For kurveintegralet betyr dette at

$$\oint_S \vec{v} dx \approx \sum_i^k v_{j,i} \cdot dx,$$

for en fast j-indeks (y-verdi) for den første kurven i x-retning, og tilsvarende for de tre andre kurvene som utgjør rektangelet. Sirkulasjonen beregnet som flateintegral kan tilnærmes numerisk med dobbeltsummen

$$\oint \oint_A \vec{n} \cdot (\nabla \times v) dA \approx \sum_i^k \sum_j^n curl_{j,i} dx dy.$$

Sirkulasjonen ble tilnærmet lik uavhengig av metode (Tabell 1). Utsnittet av Rektangel 1 er tegnet inn øverst i gassfasen, der det var lite virvling. Strømmen i dette området var omtrent horisontalt. Her vil man derfor forvente lite bidrag i sirkulasjonen fra de vertikale sidene (grønn, svart), og mye fra de horisontale. Fordi bidraget i sirkulasjonen avhenger av bevegelsesretningen langs kurven, vil en positiv omdreining gi negativ sirkulasjon på oversiden (blå) og positiv på undersiden (rød). For rektangel 1 stemmer dette overens med datamålingene.

Tabell 1: Sirkulasjonen som kurveintegral og flateintegral [mm^2/s].

Rektangel/side	Rød	Grønn	Blå	Svart	Totalsum kurveintegral	Flateintegral
Rektangel 1	70100.52	266.2736	-68332.85	661.5727	2695.514	2621.559
Rektangel 2	198.4756	300.2166	-61243.46	-231.8276	-60976.60	-61482.54
Rektangel 3	3798.576	163.6094	-4009.835	78.30288	30.65300	-12.21433

Rett over skillelinja var virvlingen sterkt negativ (4), noe som betyr at virvlingen går mot høyre (negativ omdreining om \hat{k}). Dette burde resultere i en negativ sirkulasjon langs den blå og svarte siden i rektangelet, slik datamålingene viser. Imidlertid var ikke sirkulasjonen negativ for den røde og grønne siden i utsnittet, men for den røde undersiden var det betydelig lavere enn for tilsvarende side i rektangel 1. Dette kan skyldes at virvlingene var svært lokal. Dette fanger flateintegralet opp, fordi mye av virvlingene befinner seg midt i flaten av rektangel 2. I vannet, for utsnittet 3, er effekten motsatt under bølgetoppen, med svakt positiv virvling (positiv omdreining). Den svakt positive virvlingen gjør at sirkulasjonen for undersiden og oversiden av rektangelet motvirker hverandre, hvilket resulterer i omtrent null virvling.

De numeriske beregningene presentert ovenfor viser at to forskjellige metoder gir omtrent likt resultat for sirkulasjonen (tabell 1). Dersom nøyaktigheten i datasettet øker

og presisjonen i metoden som fanger opp bevegelsene til partiklene i hastighetsfeltet forbedres, er det mulig at de to metodene kunne gitt svar som lå nærmere hverandre. Eventuelt er det måten den numeriske integrasjonen er utført på som gir utslag, og det er mulig at en høyere nøyaktig kunne blitt oppnådd med bruk av trapesmetoden. Konturfargene for virvling antyder at strømlinjene ikke fanger opp alle de lokale virvlingsfenomene. Med flere datapunkter vil dette også kunne bedres, fordi strømlinjene er tangent til hastighetsfeltet bestående av punktmålinger.

g) Den integrerte fluksen til \vec{v}

For en lukket flate gjelder Gauss sats, som kan utledes fra definisjonen av divergens,

$$\int_{\tau} \nabla \cdot \vec{v} = \int_A \vec{v} \cdot \vec{n} dA.$$

Som følge av at hastighetsfeltet er inkompressibelt, burde den integrerte fluksen være lik 0, fordi $\nabla \cdot \vec{v} = 0$. En numerisk løsning for den integrerte fluksen ble beregnet ved hjelp av høyre side av ligningen over (vedlegg 7, tabell 2).

Tabell 2: Integrert fluks til hastighetsfeltet ut langs kantene av utsnittene i xy-planet
[mm³/s]

Rektangel/side	Rød	Grønn	Blå	Svart	Sum integrert fluks
Rektangel 1	1556.868	21664.56	-2059.677	-21056.90	104.8526
Rektangel 2	-5187.564	14782.53	-4074.052	-11997.85	-6476.939
Rektangel 3	-195.5701	1536.821	284.9436	-1750.763	-124.5686

For det første utsnittet ble den totale fluksen positiv (Tabell 2). Det betyr at det strømmer netto gass ut fra dette flateutsnittet. Siden strømlinjen i området var vinklet skrått nedover, førte dette til negativ fluks inn i utsnittet fra venstre sidekant og ovenfra. Det strømmet gass ut fra nedsiden og høyre sidekant. I kontrast ble fluksen for rektangel 3 negativ, fordi strømlinjen var vinklet oppover mot skilleflaten, og retningen til strømlinjen gjenspeiles i strømning ut på høyre sidekant og på oversiden.

Fluksen fra det andre utsnittet ble negativ som følge av virvlingen i negativ omdreiningsretning. Virvlingen medførte en strømning inn fra alle kanter, med unntak av høyre sidekant. Således underbygger både resultatet fra fluksen og sirkulasjonen tilstedeværelsen av en negativ virvling, selv om denne ikke kommer tydelig frem i strømlinjene.

Siden fluksen ifølge Gauss sats burde vært null, må det bety at en fluks ut av flatene i z-retningen utjevnet den totale fluksen i xy-planet. I strømningsfeltet som ble studert må det eksistere en fluks/strømning ut mot kantene i sylindren.

1 Vedlegg

1) Nedlasting og oversikt over dataene

```
1  ### innlevering 2 ###
2
3  # Oppgave a)
4  # Matrisene X,Y
5  # u,v
6  # xit, yit
7
8  import scipy.io as sio
9  import numpy as np
10 import matplotlib.pyplot as plt
11
12 # Dette virker med versjon 7 MAT-filer
13 data = sio.loadmat("data.mat")
14 x = data.get("x")
15 y = data.get("y")
16 u = data.get("u")
17 v = data.get("v")
18 xit = data.get("xit")
19 yit = data.get("yit")
20
21 xlen = np.shape(x); ylen = np.shape(y); ulen = np.shape(u);
22 vlen = np.shape(v); xitlen = len(xit); yitlen = len(yit)
23 datapunkter = (xlen, ylen, ulen, vlen)
24 print('_____')
25 print('Lengden til array')
26 print('_____')
27 for i in datapunkter:
28     print('#datapunkter: {:} '.format(i))
29 print('_____')
30
31 # skal sjekke antall datapunkter for vektorene
32 print('# datapunkter for vektorene x:{:}, datapunkter y:{:} '.format(np.shape(xit), np.←
    shape(yit)))
33 # en akse (rad), 194 punkter.
34
35
36 print(x) # y-ene holdt konstant (y:rader, x:kolonner)
37 print(y) # x-ene holdt konstant # Sammen utgj r disse et grid
38
39 def Test_matrise(x):
40     n = 201
41     m = 194
42     for j in range(1, n):
43         for i in range(1, m):
44             if x[j, i] - x[j-1, i-1] == 0.5: # eventuelt if not
45                 continue
46             else:
47                 print("Ujevn skala")
48                 break
49
50 Test_matrise(x)
```

```

51 Test_matrise(y)      # Ok Kj  re "stille"
52
53 # Sjekk at lengde til y-koord er 100 mm : summere element i matrise y
54 # kan sj  kke ved    summere de to siste leddene
55 diameter = abs(y[0,0]) + abs(y[-1,-1])
56 print('Diameteren er {:.2f}. mm'.format(diameter))
57
58 """ Kj  reeksempel
59 Lasse$ python mek1100oblig2.py
60
61 Lengden til array
62
63 #datapunkter: (201, 194)
64 #datapunkter: (201, 194)
65 #datapunkter: (201, 194)
66 #datapunkter: (201, 194)
67
68 # datapunkter for vektorene x:(1, 194), datapunkter y:(1, 194)
69 [[ 0.    0.5   1.   ..., 95.5 96.   96.5]
70  ...,
71   [ 0.    0.5   1.   ..., 95.5 96.   96.5]]
72
73 [[-50.  -50.  -50.   ..., -50.  -50.  -50. ]
74  ...,
75   [ 50.   50.   50.   ...,  50.   50.   50. ]]
76
77 Diameteren er 100.00. mm
78 """

```

2) Grafisk fremstilling av farten til gass -og v  skefasen

```

1  # Oppgave b
2  # skal lage konturlinjer (med vektorpiler)
3  # Skal jobbe med u og v, lage for hver?
4  #plt.contour()
5  #plt.colorbar()
6
7  # plotter vektorplot av hastighetsfelt
8  v_ = np.sqrt(u**2 + v**2)
9  plt.figure(figsize = (12,5))
10
11  lvl1 = np.linspace(0,500,10)
12  ax1 = plt.subplot(121)
13  plt.contourf(x,y,v_, levels = lvl1, cmap = 'jet')
14  plt.plot(xit,yit,'*r')
15  plt.colorbar()
16  plt.ylabel('y [mm]')
17  plt.xlabel('x [mm]')
18
19  lvl2 = np.linspace(500,np.max(v_))
20  ax2 = plt.subplot(122)
21  plt.xlabel('x [mm]')
22  plt.contourf(x,y,v_, levels = lvl2, cmap = 'jet')

```

```

23 plt.colorbar()
24 plt.plot(xit,yit, '*r')
25
26
27 plt.suptitle(r'Konturlinjer til farten:  $\sqrt{u^2 + v^2}$ ') # levels konturer  $\leftrightarrow$ 
    nivÅ : 0-500, 500-maxv
28 plt.show()
29
30 """ KjÅreeksempel
31 Lasse$ python mek1100oblig2.py
32 (plot)
33 """

```

3) Vektorfelt for hastigheten

```

1 # Oppgave c)
2 plt.quiver(x[:,10],y[:,10],u[:,10],v[:,10],units = 'x', color = 'r' $\leftrightarrow$ 
    )
3 plt.plot(xit,yit, '*r')
4 plt.xlabel('x [mm]')
5 plt.ylabel('y [mm]')
6 # tre rektangler:
7 def plot_rektangel(xmin,xmax,ymin,ymax): # (x1,x2)(y1,y2)
8     plt.plot((xmin,xmin),(ymin,ymax),'black')
9     plt.plot((xmin,xmax),(ymin,ymin),'red')
10    plt.plot((xmax,xmax),(ymin,ymax),'green')
11    plt.plot((xmin,xmax),(ymax,ymax),'blue')
12
13 plot_rektangel(x[159,34],x[159,69],y[159,34],y[169,34])
14 plot_rektangel(x[84,34],x[84,69],y[84,34],y[99,34]) #xmin,xmax,ymin,ymax
15 plot_rektangel(x[49,34],x[49,69],y[49,34],y[59,34])
16 plt.show()
17
18 # start:stop:step
19
20 """ KjÅreeksempel
21 Lasse$ python mek1100oblig2.py
22 (plot)
23 """

```

4) Divergensen til \vec{v}

```

1 # d: regne ut divergensen
2 # Vi har at divergens er lik
3 n = 201
4 m = 194
5 div = np.zeros((n,m))
6 for j in range(1,n):
7     for i in range(1,m):

```

```

8         div[j,i] = (u[j,i] - u[j,i-1])/(x[j,i] - x[j-1,i-1])\
9                   + (v[j,i] - v[j-1,i])/(y[j,i] - y[j-1,i-1])
10 #print(div)
11
12 # konturlinjene til divergensen
13
14 plt.contourf(x,y,div, cmap = 'jet')
15 plt.plot(xit,yit, '*r')
16 plt.colorbar()
17 plt.xlabel('x [mm]')
18 plt.ylabel('y [mm]')
19 plot_rektangel(x[159,34],x[159,69],y[159,34],y[169,34])
20 plot_rektangel(x[84,34],x[84,69],y[84,34],y[99,34]) #xmin,xmax,ymin,ymax
21 plot_rektangel(x[49,34],x[49,69],y[49,34],y[59,34])
22 plt.show()
23
24 """ Kj  reeksempel
25 Lasse$ python mek1100oblig2.py
26 (plot)
27 """

```

5) Virvlingen til \vec{v}

```

1 # Oppgave e) Virvlingen i rundt z-aksen
2 curl = np.zeros((n,m))
3
4 for j in range(1,n-1):
5     for i in range(1,m-1):
6         curl[j,i] = (v[j,i+1] - v[j,i-1])\
7                   - (u[j+1,i] - u[j-1,i])
8 # velger midtpunkter
9
10 #print(curl)
11 plt.contourf(x,y,curl, cmap = 'jet')
12 plt.plot(xit,yit, '*r')
13 plt.colorbar()
14 plt.xlabel('x [mm]')
15 plt.ylabel('y [mm]')
16 plot_rektangel(x[159,34],x[159,69],y[159,34],y[169,34])
17 plot_rektangel(x[84,34],x[84,69],y[84,34],y[99,34]) #xmin,xmax,ymin,ymax
18 plot_rektangel(x[49,34],x[49,69],y[49,34],y[59,34])
19 plt.streamplot(x,y,u,v, color = "blue") # plt.streamline()
20 plt.show()
21
22 """ Kj  reeksempel
23 Lasse$ python mek1100oblig2.py
24 (plot)
25 """

```

6) Sirkulasjonen til hastighetsfeltet

```

1 # Oppgave f) Stokes og greens sats på rektanglene
2 # 1) Sirkulasjon regnet som kurveintegral
3 print(np.shape(v)) #201,194 y, x
4
5 def sirkulasjon(jmin,jmax,imin,imax):
6     int_list = [0,0,0,0]
7     for i in range(imin,imax+1):
8         int_list[0] += u[jmin,i]*0.5
9     for j in range(jmin,jmax+1):
10        int_list[1] += v[j,imax]*0.5
11    for i in range(imin,imax+1):
12        int_list[2] -= u[jmax,i]*0.5
13    for j in range(jmin,jmax+1):
14        int_list[3] -= v[j,imin]*0.5
15    return int_list[0], int_list[1], int_list[2], int_list[3], sum(int_list)
16
17 shrek1 = sirkulasjon(159,169,34,69)
18 shrek2 = sirkulasjon(84,99,34,69)
19 shrek3 = sirkulasjon(49,59,34,59)
20 print(shrek1)
21 print(shrek2)
22 print(shrek3)
23
24 # 2 ) Sirkulasjonen som flateintegral
25 def sirksurf(jmin,jmax, imin,imax):
26     sum_ = 0
27     curlarea = curl*0.5*0.5
28     sum_curlarea = 0
29
30     for j in range (jmin,jmax+1):
31         for i in range(imin,imax+1):
32             sum_curlarea += curlarea[j,i]
33     return (sum_curlarea)
34
35
36 sirksurf1 = sirksurf(159,169,34,69)
37 sirksurf2 = sirksurf(84,99,34,69)
38 sirksurf3= sirksurf(49,59,34,69)
39 print(sirksurf1,sirksurf2,sirksurf3)
40
41 """ KjÅreeksempel
42 Lasse$ python mek1100oblig2.py
43 (70100.523878614273, 266.27357615858688, -68332.856099786746, 661.57273770969914, ←
44 2695.5140926958193)
45 (198.47559740489203, 300.21661027011692, -61243.464778495952, -231.82759129461652, ←
46 -60976.600162115559)
47 (3798.5759031398288, 163.60937776314384, -4009.8351655432498, 78.302877021285482, ←
48 30.652992381008559)
49
50 2621.55869628 -61482.5409893 -12.2143338642
51 """

```

7) Den integrerte fluksen til \vec{v}

```

1 # g) gauss sats: integrert fluks
2 # direkte:
3 def fluks_area(jmin,jmax,imin,imax):
4     int_list = [0,0,0,0]
5     for i in range(imin,imax+1):
6         int_list[0] -= v[jmin,i]*0.5 #(x[jmin,i+1] - x[jmin,i])
7         int_list[1] += v[jmax,i]*0.5
8     for j in range(jmin,jmax+1):
9         int_list[2] += u[j,imax]*0.5
10        int_list[3] -= u[j,imin]*0.5
11    return int_list[0],int_list[1],int_list[2],int_list[3], sum(int_list)
12
13 print(fluks_area(159,169,34,69))
14 print(fluks_area(84,99,34,69))
15 print(fluks_area(49,59,34,69))
16
17 # Kj  reeksempel
18 """
19 Lasse$ python mek1100oblig2.py
20
21 (1556.8679439413959, -2059.6771847938708, 21664.567474322168, -21056.905628561482,  
22 104.8526049082102)
23 (-5187.5640330678907, -4074.0522144394345, 14782.532896182345, -11997.85583077298,  
24 -6476.9391820979599)
25 (-195.5701479258336, 284.9436464350764, 1536.8217966413547, -1750.7639611955597,  
26 -124.56866604496213)
27 """

```