Bevegelsesmotstand

Yang Min Wang and Dag Kristian Dysthe*
Fysisk institutt, UiO
(Sist endret March 13, 2018)

Målet med denne oppgaven er å lære litt om fluiddynamikk og bevegelsesmotstand, om dimensjonsanalyse og om bruk av kamera og billedbehandling som måleinstrument.

I. INNLEDNING

Bevegelsesmotstand eller "drag" (engelsk) begrenser f.eks. hvor fort vi kan sykle og hvor fort ting faller mot bakken. Å utlede analytiske formler for luftmotstanden krever noen sider med fluiddynamikk og resultatene gjelder bare for idealiserte former og grensebetingelser. Vi skal i stedet bruke dimensjonsanalyse som er et kraftig verktøy for å finne nødvendige sammenhenger mellom fysiske størrelser i et komplekst problem uten å måtte skrive ned alle de fysiske lovene som gjelder. Eksperimentet vi skal gjøre er å undersøke de forskjellige typene bevegelsesmotstand som dimensjonsanalysen forutsier. Eksperimentet går ut på å følge bevegelsen til kuler som faller ned gjennom en væske. Målingen av bevegelsen til kulen kunne vi gjort med kameraet på mobiltelefonen, det er et fabelaktig måleinstrument dere kan bruke når som helst.

II. DIMENSJONSANALYSE FOR BEVEGELSESMOTSTAND

For å angi fysiske [1] størrelser kvantitativt benyttes *måleenheter;* så som meter, sekund, kilo og ampere. La Q være en fysikalsk størrelse (f.eks. en fart). Vi har da eksempelvis:

$$\begin{aligned} Q &= 3.6 \text{ kilometer/time} \\ &= 60 \text{ meter/minutt} \\ &= 1 \text{ meter/sekund} \\ &= 100 \text{ centimeter/sekund} \end{aligned} \tag{1}$$

Måleenhetene er bestemt ut fra praktiske vurderinger, tradisjoner og internasjonale konvensjoner. Hvilken enhet som benyttes kan til en viss grad være situasjonsbestemt og derfor vilkårlig, som eksemplet ovenfor viser. For bedre å kunne skille mellom det praktiske (spesifikke måleenheter) og det prinsipielle (relasjonene mellom enhetene), er det vesentlig å kunne dele de fysiske størrelsene inn i klasser uavhengig av spesifikke måleenheter. Vi sier at to størrelser, Q_1 og Q_2, har lik *fysisk dimensjon* dersom de kan angis ved samme måleenheter, eller: dersom forholdet mellom dem kan uttrykkes som et reelt tall.

Det finnes mange dimensjonsklasser. Det viser seg imidlertid at det er mulig å velge et lite antall *grunnleggende dimensjonsklasser* som basis for alle de andre dimensjonene: M, L og T [2].

Her er en del eksempler på fysiske størrelser, deres dimensjoner uttrykt i de grunnleggende dimensjonsklassene og SI-måleenhetene.

Størrelser	dimensjoner	SI-enheter
Lengde	L	m
Tid	T	s
Masse	M	kg
Areal	L^2	m^2
Volum	L^3	m^3
Hastighet	LT^{-1}	m/s
Aksellerasjon	LT^{-2}	$\mathrm{m/s}^2$
Kraft	MLT^{-2}	newton
Tetthet	ML^{-3}	$\mathrm{kg/m}^3$
Arbeid	ML^2T^{-2}	joule
Effekt	ML^2T^{-3}	watt
Trykk, Stress	$ML^{-1}T^{-2}$	pascal
Viskositet (kinematisk)	L^2T^{-1}	m^2/s
Viskositet (dynamisk)	$ML^{-1}T^{-1}$	pascal s

Vi skal nå gjøre en dimensjonsanalyse for å finne ut noe om kraften F_d på en kropp i bevegelse i en væske [3]. Vi er interessert i hvordan F_d (luftmotstanden når du sykler, væskemotstanden til en hval, et menneske eller en krill) avhenger av størrelsen til kroppen som beveger seg, hastigheten og egenskapene til væsken. Vi må da begynne med en liste over de fysiske størrelsene vi tror kan være viktige [4]. Vi bruker notasjonen $[x]$ = "er dimensjonen til x".

- Hastigheten til kroppen i forhold til væsken (langt unna): v, $[v] = LT^{-1}$
- Størrelsen til kroppen: r, $[r] = L$
- Tettheten til væsken: ρ, $[\rho] = ML^{-3}$
- Viskositeten til væsken: μ, $[\mu] = ML^{-1}T^{-1}$
- Motstandskraften på kroppen: F_d, $[F_d] = MLT^{-2}$

Vi har brukt vår fysiske intuisjon til å eliminere mange andre mulige fysiske størrelser som massen og fargen til

*Electronic address: d.k.dysthe@fys.uio.no

kroppen eller væsken, temperaturen (vi antar at den bare inngår gjennom ρ og μ), osv. Nå skal vi bruke Buckingham's Π-teorem som sier:

> Antall uavhengige, dimensjonsløse parametre som trengs for å beskrive dette systemet er antall fysiske størrelser som inngår minus antall uavhengige dimensjoner som de størrelsene består av.

Det er 3 uavhengige dimensjoner som inngår: M, L og T. Siden vi har 5 fysiske størrelser vi tror skal være med i beskrivelsen av problemet så sier Π-teoremet at det skal være 5-3=2 uavhengige, dimensjonsløse variabler i beskrivelsen av fenomenet. Vi har nå stor frihet i å velge hvilke 2 variabler vi vil bruke.

Reynolds-tallet R_e er en viktig dimensjonsløs variabel for å skille mellom når strømningen er laminær og når den er turbulent. De to typene strømninger vil også være to grensetilfeller for strømning rundt en kropp, så vi velger Reynoldstallet

$$R_e = \rho v r/\mu \tag{2}$$

som den ene variabelen. Nå ønsker vi å finne den andre dimensjonsløse variabelen. Siden R_e ikke inneholder F_d søker vi nå en dimensjonsløs variabel som inneholder F_d. Vi begynner da med å velge hvordan vi skal fjerne dimensjonen M i F_d. Her har vi to valg, vi kan bruke ρ eller μ.

Vi bruker først ρ: $[F_d/\rho] = T^{-2}L^4$. Så vil vi fjerne dimensjonen T: $[F_d/\rho v^2] = L^2$ og så L:

$$C_R = \frac{F_d}{\rho v^2 r^2}. \tag{3}$$

C_R er et dimensjonsløst tall som vi vil kalle R-motstandskoeffisienten.

Vi bruker nå μ: $[F_d/\mu] = L^2T^{-1}$, fjerner tiden med $[F_d/\mu v] = L$ og så lengden:

$$C_S = \frac{F_d}{\mu v r}. \tag{4}$$

C_S er et dimensjonsløst tall som vi vil kalle S-motstandskoeffisienten.

Hva betyr de to motstandskoeffisientene C_R og C_S? Da vi valgte å bruke ρ så ser vi at F_d er uavhengig av viskositeten. Dette tilsvarer at R_e er veldig stort og at turbulens er viktigere enn laminær strømning. Denne type strømning har Rayleigh beskrevet (derav R-en i C_R). Det motsatte grensetilfellet er når R_e er liten og viskositeten bestemmer kraften på kroppen gjennom den laminære strømningen rundt kroppen. Denne typen strømning kalles Stokes-strømning etter ham som først regnet ut laminær strømning rundt en kule. Vi vet enda ikke noe om hvor stor eller liten Reynolds-tallet R_e må være for at vi skal være i det ene eller det andre grensetilfellet. Men vi vet nå at hvis vi er i det turbulente grensetilfellet så skal $F_d/r^2 \propto v^2$, mens hvis vi er i det laminære grensetilfellet så er $F_d/r \propto v$. Dette kan vi bruke til å analysere eksperimentelle data for hvor fort en kropp faller i tyngdefeltet gjennom en væske.

Det er egentlig litt rart at vi har kommet frem til formlene for Rayleigh (3) og Stokes (4) drag uten å bruke noe fluidmekanikk (bare litt intuisjon om at viskositet og væsketetthet må bety noe samt en ide om Reynoldstallet). Denne måten å gå frem på er svært nyttig for mer komplekse problemer der vi ikke har analytiske ligninger. Neste skrittet da er det vi skal gjøre nå: gjøre eksperimenter og bruke resultatene av dimensjonsanalysen til å hjelpe oss å analysere dataene og se sammenhenger i dem.

III. LABORATORIEØVING

Vi skal bruke et kamera med objektiv tilkoblet PC-en til å måle på kuler med forskjellig radius og masse som faller ned gjennom en gjennomsiktig sylinder fylt med olje (Shell Tellus S2 M 68 Industrial Hydraulic Fluid). Deretter bruker vi billedbehandling i Matlab til å finne bevegelsen av kulene.

A. Oppgaver

a. Karakterisering av avbildingssystemet

- Åpne programmet `uEye Cockpit` i "expert mode" (dvs. start programmet og velg File→Mode→Expert fra menyen).

- Snu kameraet mot "lyskutteren" (Frame Rate Counter) og fokuser på lysdiodene. Sett kameraet på nær maksimal billedrate (Frames Per Second, FPS), f.eks. 70 FPS [5], og kort lukkertiden, f.eks. 1/250 s. Sett gain til minimum og finstill lukkertid og evt. gamma for å få best mulig kontrast. Skru på lyskutteren og signalgeneratoren og still inn den laveste ikke-null frekvensen på signalgeneratoren som gjør at mønstret av opplyst lysdioder er konstant [6]. Siden det er 60 lysdioder som opplyses på en runde er billedraten frekvensen delt med 60.

- Hvorfor stiller inn "den laveste ikke-null frekvensen"?

- Lagre en sekvens til fil, åpne filen med `VideoReader` i `MATLAB` og sjekk at billedraten ("framerate") stemmer overens med frekvensmålingen. **NB!** Det er avgjørende viktig at sekvensen ikke mangler noen "frames". **Billedsekvensen må lagres på PCens lokaldisk `C:`.**

- Anslå hvor nøyaktig du mener informasjonen om billedraten er.

- Fokuser på en kule på den avstanden en fallende kule vil ha under forsøket. Finstill på nytt om nødvendig lukkertid og evt. gamma for å få best mulig kontrast.
- Ta og lagre noen stillbilder av en meterstokk på posisjoner fallende kuler vil ha under forsøket.

b. Bevegelsesmotstand på fallende kuler: De fysiske størrelsene som inngår i fenomenet bevegelsesmotstand er hastigheten, v, som måles ved billedbehandling, viskositeten μ og tettheten ρ til væsken (se databladet for oljen som vi skal bruke), radiene, r, til kulene som måles med skyvelær og massene, m, til kulene som måles med en vekt. Det er tre krefter som virker på kulen: tyngdekraften $G = mg$, oppdriften $B = \rho 4\pi r^3/3$ (Archimedes lov), og bevegelsesmotstanden $F_d(v)$ (friksjonskraften fra væsken). I motsetning til kontaktfriksjonen mellom faste legemer er denne friksjonskraften avhengig av farten v. Når summen av disse kreftene er null vil kulen ha konstant hastighet v_c, som vi kaller terminalhastigheten. Dette inntreffer når

$$F_g = G - B = F_d(v_c). \tag{5}$$

- Det er ikke praktisk å plukke opp kuler fra oljen så dere får bare ett forsøk per kule. Vær sikker på at alle gruppene er klare til å ta video når kulen slippes.
- Kulene faller med svært forskjellig hastighet fra noen millimeter per sekund til nærmere en meter per sekund.
- Finn massesenteret til kulen i hvert bilde og lag et plot av posisjonen som funksjon av tiden. (Tips: Finn ut hvilke bildenummer kulen er i bildet [start:slutt], lag en `for`-løkke som leser inn bildene og som finner massesenteret for hvert bilde.)
- Hastigheten kan bestemmes på to måter: Ved å ta differansen av posisjonen eller ved å tilpasse en rett linje til den delen av posisjon - tid-kurven som er rett. Plott den momentane hastigheten mellom nabobilder som funksjon av tiden. (Tips: Se Appendiks A.)
- Finn den maksimale hastigheten v_c for tilfellene nevnt ovenfor. (Tips: Her er det lurt å ikke ha med for mye støy.)
- Plott
  - v_c mot F_g/r
  - v_c^2 mot F_g/r^2
  - C_S mot R_e
  - C_R mot R_e

  Bestem hvilken type strømning det er snakk om her.

Appendix A: Matlab-funksjoner til å løse oppgavene

```
diff(P); %Returnerer en vektor som består av differenser av nabo-elementer i vektoren P.

obj = VideoReader(‘‘filnavn.type"); %Returnerer et multimedia objekt og tar inn en videofil av type;
                               .wmv, .mpeg, .avi etc... som vi kan hente informasjon ut fra.

obj = imread('ball1.type'); %Returnerer en MxNx3 matrise, tar inn en bildefil av type format;
                            .png, .jpg etc...

images = read(obj, [#fra #til]); % Leser inn multimedia-objektet ‘‘obj" og lager en matrises NxMx3xB
                               fra det spesifiserte området av bilder som har nr. 1 som laveste index.
                                B = antall bilder.

A = images([y-akse],[x-akse],fargekanal, bildenummer) %Lager en matrise der vi spesifiserer området
vi ønsker å se på, og en bestemt fargekanalen Rød, Grønn eller Blå. Dersom vi ønsker å se på hele
området og alle fargekanalene samtidig skriver vi, A = images(:,:,:, bildenummer).
Hvis man ønsker å starte fra en bestemt index og fortsette til siste tall i matrisenslutten
skriver man; #index_start:end.

image(images(A)); %Dersom man ønsker å se på matrisen.

imagesc(‘‘matrise"); %Skalerer matrisen/bildet.
```

```
colormap(‘‘matrise"); %Lager en fargestolpe over intensitetsnivå for en bestemt kanal.

im2bw(‘‘matrise", grense); %Omgjør matriseelemtene til 0’er og 1’enere utifra kriteriet 0< grense < 1.


------Slik regner man ut massesentret til et 2D-objekt i matlab------
B = bwlabel(‘‘input-matrise", 4); %input-matrise må bestå av 0’ere og 1’ere.
c = regionprops(B,’centroid’);  %Regner ut massesenteret til matrisen B og lagrer informasjonen i
                                en struktur c som er en 1xN matrise.

c.Centroid(1,1); %Returnerer en skalar fra x-posisjonen
                 og c.Centroid(1,2) returnerer y-posisjonen.
```

II. FINNE MASSESENTERET TIL EN 2D-FIGUR

Definisjon av massesenteret:

$$c_m = \frac{\sum_i m_i r_i}{m_i}$$

Vi kan tenke oss at hver celle i matrisen A inneholder en vekt m_i og $r_i = x_i \cdot \vec{i} + y_i \cdot \vec{j}$. Algortimen for å finne massesenteret i x og y-retning:

$$x_m = \frac{\sum_{i,j} A(i,j) \cdot j}{\sum_{i,j} A(i,j)}$$

$$y_m = \frac{\sum_{i,j} A(i,j) \cdot i}{\sum_{i,j} A(i,j)}$$

Men før vi begynner å summere kan vi benytte av oss kontrasten til gjøre regnestykket enklere. Bakgrunnen er svart og har lave intesiteter i alle fargekanaler (R,G,B) i forhold til kulen. Vi kan dermed omgjøre matrisen til en logisk matrise (0’ere og 1’ere). (Tips: bruk matlab-funksjonen `ceil()`; som runder av matrisen til nærmeste heltall.

`for`-løkker er treige i Matlab, som er bygd for å håndere multiplikasjon av hele matrise raskt. Derfor gjør vi enda et triks for å unngå å summere. Vi har en f.eks en matrise A = [0 0 0 1;0 0 1 1;0 0 1 1;0 0 0 1]. Vi trenger å lage en index-matrise I_y dvs. 1’ere på første rad, 2’ere på andre osv. Det gjøres på følgende måte i Matlab ($y_m = 2.5$, $x_m = 3.67$):

```
i = 1:4;
e = ones(1,4)  % 1x4 matrise med 1’ere
b = i’;        % transponerer i og lager en søylematrise
c = b*e        % c er matrise med 1’ere på første rad, 2’ere på andre osv.

I_y = c.*A     % multipliserer elemter i samme posisjon og danner en ny matrise.

m = sum(sum(A));   % summerer alle elementene i A, analogt med å finne total masse.

y_m = sum(I_y*e’)/m % summerer en 4x1 matrise med de forskjellige A(i,j)*i for hver rad og
               deler på total masse. Dette gir massesenteret i x-retning.
```

$$\begin{pmatrix} 1 \\ 2 \\ 3 \\ 4 \end{pmatrix} \cdot \begin{pmatrix} 1111 \end{pmatrix} \rightarrow \begin{pmatrix} 1 & 1 & 1 & 1 \\ 2 & 2 & 2 & 2 \\ 3 & 3 & 3 & 3 \\ 4 & 4 & 4 & 4 \end{pmatrix} .* \begin{pmatrix} 0 & 0 & 0 & 1 \\ 0 & 0 & 1 & 1 \\ 0 & 0 & 1 & 1 \\ 0 & 0 & 0 & 1 \end{pmatrix} \rightarrow \begin{pmatrix} 0 & 0 & 0 & 1 \\ 0 & 0 & 2 & 2 \\ 0 & 0 & 3 & 3 \\ 0 & 0 & 0 & 4 \end{pmatrix} * \begin{pmatrix} 1 \\ 1 \\ 1 \\ 1 \end{pmatrix} = \sum_{i,j} A(i,j) \cdot i$$

$$y_m = \frac{1+1+2+2+3+3+4}{1+1+1+1+1+1+1} = 2.5$$

Tilsvarende må gjøres for x-retningen.

III. UTSTYRSLISTE

- kamera med objektiv
- kuler
- skyvelær
- tommestokk
- vekt
- Frame Rate Counter

---

[1] Med en fysisk størrelse skal vi mene enhver størrelse som må refereres til en enhet for å kvantifiseres.

[2] I tillegg kommer de fysiske størrelsene temperatur, electrisk strøm og lysintensitet, men de har vi ikke bruk for akkurat nå.

[3] Se f.eks. `https://en.wikipedia.org/wiki/Drag_equation`, og referanser der.

[4] Formen til kroppen er nok viktig, men for å unngå den komplikasjonen sier vi at vi er interessert i motstanden til en kropp av samme form når de andre størrelsene varieres.

[5] 70 FPS gjelder nye kamera kjøpt inn i 2016. Hvis det er problematisk å kjøre med høy verdi av FPS, kjør `IDS Camera Manager` og velg Additional functions→Change CPU idle state setting.

[6] En (lydløs) demonstrasjon av teknikken vises på `https://www.youtube.com/watch?v=mwo6pEHgNxE`.