Labrapport som Jupyter Notebook på fysikklab våren 2023

Synnøve Arnesen, Dina-Madelen Sandlie Hegna, Andreas Teodor Nysæter, Simen Kristoffer Bedringås, David Persson
Institutt for Teknisk kybernetikk, NTNU Institutt for Elektroniske systemer, NTNU

Introduksjon

Dette labforsøket går ut på å studere en kule som ruller (uten å gli) på en berg-og-dal-bane. Forsøket har blitt simulert numerisk i python hvor programmet har regnet ut og plottet relevante størrelser som blant annet baneform, farten til kulen v(x) og kulens kinetiske energi K. Forsøket har også blitt gjennomført fysisk. Her har rulleforsøkene blitt filmet og kulens bevegelse er blitt analysert i sporingsprogrammet "Tracker" [1]. Ved hjelp av grunnleggende mekanikk og loven om energibevaring har det vært mulig å beregne blant annet kulens totale kinetiske energi K. Den grunnleggende mekanikken som er benyttet i dette forsøket er basert på Newtons tre lover som ble presentert i 1686 i "Principia Mathematica Philosophiae Naturalis." [2]. Til slutt har resultatene fra simuleringen og det fysiske forsøket blitt sammenlignet og diskutert.

Teori

Når en kule ruller ned en bane med kjent form gitt av $y(x)$ og det antas at mekanisk energi er bevart, vil det være mulig å regne ut størrelser som hastighet, friksjonskraft mellom kula og overflaten, og rulletiden (med hjelp av numerikk). For å regne ut disse størrelsene må det også antas at den første- og andrederiverte av y, altså y' og y'' er kjent.

Kulas treghetsmoment mhp. rotasjonsaksen(som går gjennom massesenteret CM) er gitt ved

$$I_0 = cmr^2.$$

Her er m kulas masse, r kulas radius og c en konstant som kan settes til $\frac{2}{5}$ under antagelsen at kula er ei kompakt kule med uniform massefordeling. Videre er det nyttig å anta at banens krumningsradius er mye større enn kulas radius, siden dette medfører at kulas massesenter følger samme form som banen, bare forskjøvet i y-retning.

I kulas startpunkt $x = 0$ er høyden $y(0) = y_0$. Det kan antas at kula starter i ro, og dermed blir den mekaniske energien

$$E = U_0 = mgy_0.$$

Her er nullpunktet for U definert som $y = 0$. Kulas rotasjonsenergi er gitt ved $E_{rot} = 1/2I\omega^2$, og med innsetting for c og $\omega = v/r$ blir dette til

$$K = \frac{1+c}{2}mv^2,$$

der v er hastigheten. Bevaring av energi gir

$$U_0 = K + mgy,$$

$$\iff v = \sqrt{\frac{2g(y_0 - y)}{1+c}}$$

i et punkt på banen der høyden er y og hastigheten er v. Siden y er en funksjon av posisjonen x, kan også farten betraktes som en funksjon $v(x)$.

Krumningen (med fortegn), som også er den inverse av krumningsradiusen til grafen $y(x)$ er gitt ved

$$\kappa = \frac{y''}{(1 + (y')^2)^{\frac{3}{2}}},$$

som umiddelbart gir sentripetalakselerasjonen

$$a_\perp = v^2/R = v^2\kappa = \frac{2g(y_0 - y)}{1+c}\frac{y''}{(1 + (y')^2)^{\frac{3}{2}}}$$

Fortegnet til κ avhenger av banens krumning slik at κ er positiv når banen krummer oppover, og negativ når banen krummer nedover (κ har samme fortegn som y''). Dette vil også bli fortegnet til $a_\perp$, som er konsistent med å ha positiv retning oppover.

med innsatt $c = \frac{2}{5}$ blir hastigheten

$$\sqrt{\frac{10g(y_0 - y)}{7}}.$$

Newtons lov, sett fra et koordinatsystem som står normalt på banen og med normalkraften som positiv retning i y-aksen gir oss

$$N - mg\cos\beta = ma_\perp.$$

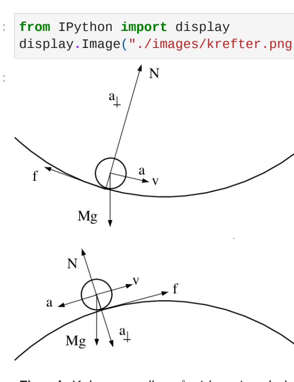
Her er β banens helningsvinkel. Dette kan omformes til

$$N = m(g\cos\beta + a_\perp).$$

Dette viser at når banen peker oppover, er $a_\perp > 0$, altså peker $a_\perp$ i samme retning som N, som gjør at N blir større enn $mg\cos\beta$. Når banen peker nedover, skjer det motsatte, altså er $a_\perp < 0$ og peker i motsatt retning av N, og N blir mindre enn $mg\cos\beta$.

In [ ]:
```python
from IPython import display
display.Image("./images/krefter.png", width=300)
```

Out[ ]:



Figur 1. Kule som ruller på et krumt underlag. Her er v kulas fart, a er baneakselerasjon, $a_\perp$ er sentripetalakselerasjon, Mg er tyngdekraft, f er friksjonskraft og N er normalkraft.

Den siste størrelsen som kan være nyttig å utlede nå er den statiske friksjonskraften fra banen på kula. Ved valg av helningsvinkelen β slik at $\beta < 0$ når banen heller nedover, og $\beta > 0$ når banen heller oppover får stigningstallet

$$y' = \frac{dy}{dx} = \tan\beta$$

riktig fortegn. De tangentielle kreftene er friksjonskraften F og tyngdekraftens horisontale komponent $-mg\sin\beta$. Hvis banen heller nedover blir $\beta < 0$, $F < 0$ og $a = \frac{dv}{dt} > 0$. N2 gir da

$$-mg\sin\beta + F = ma.$$

Hvis banen derimot heller nedover blir $\beta > 0$, $F > 0$, og $a = \frac{dv}{dt} < 0$. N2 gir da

$$mg\sin\beta - F = -ma,$$

altså er ligningen lik uavhengig av helningen til banen. Baneakselerasjonen kan regnes ut som den deriverte av banefarten $v(y) = \sqrt{\frac{2g(y_0-y)}{1+c}}$, altså

$$a = -\frac{g\sin\beta}{1+c}.$$

Her brukes $\frac{dy}{dt} = v_y = -v\sin\beta$. Med $c = \frac{2}{5}$ blir $a = -\frac{5g\sin\beta}{7}$. Innsetting gir

$$F = ma + mg\sin beta = \frac{c}{1+c}mg\sin\beta,$$

eller $F = \frac{2mg\sin\beta}{7}$ for kulen. En forutsetning som har blitt tatt er at kula ruller rent og ikke slurer på underlaget. Det betyr at den statiske friksjonskraften F ikke kan overstige den maksimale verdien $|F| = \mu_s|N|$.

Tidsutvikling

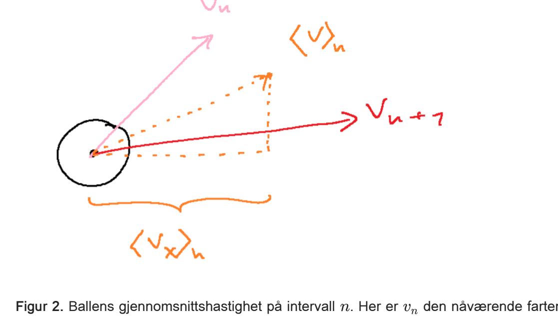
Ingen av ligningene hittils har inneholdt tiden på noe vis, som følger av at utgangspunktet har vært bevaring av mekanisk energi. For å finne kulas tidsutvikling er det ofte nødvendig å gå helt tilbake til Newtons andre lov og bruke numerikk, men i tilfellet hvor kula ruller på an bane med bestemt form $y(x)$ kan det gjøres enklere ved å bruke at hastigheten v i hvert punkt y ver gitt vedo $v = \sqrt{\frac{10g(y_0-y)}{7}}$. Tiden dt som brukes på en forflytning dx er da gitt direkte ved $dt = \frac{dx}{v_x}$

Med $k+1$ jevnt fordelte punkter $\{x_0, x_1, \ldots x_{k-1}, x_k\}$, og tilsvarende banehøyder definert er hastigheten v_n og helningsvinkelen β_n i posisjonen (x_n, y_n) kjent. Det er da mulig å regne ut tidssteget Δt_n mellom x_{n-1} og $x_n (n = 1, 2, \ldots k)$. Horisontalkomponenten til hastigheten i posisjon x_n er gitt ved $v_{x,n} = v_n\cos\beta$. Den gjennomsnittle horisontalkomponenten til hastigheten på intervall nr. n er gitt ved

$$\langle v_x\rangle_n = \frac{1}{2}(v_{x,n-1} + v_{x,n}).$$

In [ ]:
```python
display.Image("./images/ballhastighet.png", width=500)
```

Out[ ]:



Figur 2. Ballens gjennomsnittshastighet på intervall n. Her er v_n den nåværende farten, v_{n+1} farten i neste tidssteg, $\langle v\rangle_n$ gjennomsnittsfarten i intervallet n, og $\langle v_x\rangle_n$ gjennomsnittsfarten i x-retning i intervallet n.

Innsettting i definisjonen for Δt_n fra tidligere gir da

$$\Delta t_n = \frac{\Delta x_n}{\langle v_x\rangle_n} = \frac{2\Delta x_n}{v_{x,n-1} + v_{x,n}}.$$

Det horisontale steget Δx_n vil alltid bare være x_n - x_{n - 1}. Dette gir til slutt at kula starter i posisjonen (x_0, y_0) med starthastighet $v_0 = 0$ ved tidspunkt $t_0 = 0$ og passerer posisjonene (x_n, y_n) med hastighet v_n og ved tidspunkt

$$t_n = \sum_{j=1}^{n}\Delta t_j, \qquad n = 1, \ldots, k$$

Dette brukes selvfølgelig til å estimere tidsforløpet numerisk.

Metode

Nødvendig utstyr for eksperiment:

- Vater
- Kule
- Berg-og-dal-bane
- Merkepenner
- Kamerastativ

Eksperimentet går ut på at en kule skal rulle ned en kjent bane med 8 skruehøyder. Skruehøydene i dette prosjektet kan man finne i figur 3. Banen er tildelt av NTNU og skruehøydene er randomisert for hver gruppe. Forsøket skal filmes med et telefonkamera som er montert i et stativ med en standard filmkvalitet på 1080p og en bildefrekvens på 30. Det er viktig at både banen og kameraet er i vater. Kulens initielle fart er 0 meter i sekundet og startpunktet er toppunktet på banen. Hvert av forsøkene er en videosnutt av kulen som ruller ned banen, hvor hver videosnutt starter i det kulen blir sluppet, og slutter når kulen er på sluttpunktet i banen.

Prosjektet er gjort over 4 økter hvor hver økt fokuserer på en hoveddel av oppgaven. Den første økten brukes for å organisere gruppen og finne en samarbeidsplattform. I dette prosjektet er det brukt Git. Økt 2 er for numerikk og beregninger av verdier som skal sammenlignes med resultater fra økt 3. I økt 3 gjennomføres eksperimentet. Eksperimentet blir filmet og kjørt gjennom analyseverktøyet Tracker for å hente ut resultater. I økt 4 skal resultater fra økt 2 og 3 analyseres.

Første økt ble brukt til å sette opp en samarbeidsplattform for gruppen, og sørge for at all nødvendig programvare for resten av prosjektet er lastet ned. Alle deltakerne skal sette opp en Github profil, en av deltakerne lager et prosjekt som deles med resten av gruppen. Git-programvaren lastes ned til prosjekt deltagernes datamaskiner. Dette er brukt i samsvar med Visual Studio Code. Visual Studio Code er et koderedigeringsprogram som lar prosjektdeltakerne skrive og kjøre kode i Python og Jupiter Lab. Deretter er det brukt Git terminal kode som "git push", "git init", "git add" og "git pull" for å distribuere kode og verdier. Verktøyene brukt for kode redigering, distribusjon og utregninger i prosjektet er GitHub, Visual Studio Code og Tracker.

Økt 2 er for å regne ut nødvendige teoretiske verdier. For at prosjektet skal være individuelt er baneformen randomisert for hver gruppe. For å finne baneformen legges randomiserte verdier for skruehøydene inn i funksjonen CubicSpline, dette er en funksjon som ligger i interpolate-biblioteket i SciPy. Etter baneformen er satt regnes de teoretiske verdiene til fart v(x), sluttfart, helningsvinkel, normalkraft og friksjonskraft. Verdiene ble regnet ut etter instruksjoner beskrevet i teoridelen. De teoretiske verdiene ble regnet ut med bruk av mattebiblioteket Numpy og grafer plottes ved hjelp av graf- og mattebiblioteket Matplotlib. Tidsutvikling ble brukt for å finne den totale rulletiden til ballen. For å finne tidsutviklingen ble Euler-eksplisitt brukt. I Python ble denne metoden implementert ved bruk av en for-løkke som skrev en Numpy array. Eventuelle tidsutviklinger, verdier og grafer noteres ned for å senere bli sammenlignet med eksperimentets verdier.

Den tredje økten har hovedfokuset på selve eksperimentet. Skruehøydene stilles inn på utdelte baner og det filmes en testvideo for å sørge for at ballen følges i "sporingsprogrammet" Tracker. Når Tracker er konfigurert filmes forsøket ti ganger. Rulletid, sluttfart, total kinetisk energi i sluttpunkt og tapet av den kinetiske energien skal regnes ut for hvert forsøk, her er det sentralt å bruke Tracker for å hente nøyaktige verdier. Tracker fungerer slik at den kan se på videoen og lokalisere punktverdier hvor kulen er. Etter disse verdiene er funnet skal det regnes ut en middelverdi og standardfeil for alt. Middelverdien og standardfeilen for forsøkene sammenlignes med de teoretiske verdiene funnet i økt 2. Det ble ikke gjort noen overordnede modifikasjoner til utførelsen av forsøket, eller innhentingen av resultater som var beskrevet i oppgaveteksten.

For å finne rulletiden, den kinetiske energien i sluttpunktet og tapet av den kinetiske energien ble de målte verdiene fra de 10 forsøkene satt inn i hver sin Numpy array. Middelverdiene ble funnet ved å bruke numpy.average, som finner en gjennomsnittsverdi av hva som er i arrayen. For å finne standardavvik ble funksjonen numpy.std() brukt, denne funksjonen finner standardavviket til det i funksjonen, verdien ble deretter delt på kvadratroten av antall forsøk som ble gjort. De gjeldende sifrene i resultatene var fem. Under eksperimentet ble det ikke gjort overordnede modifikasjoner til utregning.

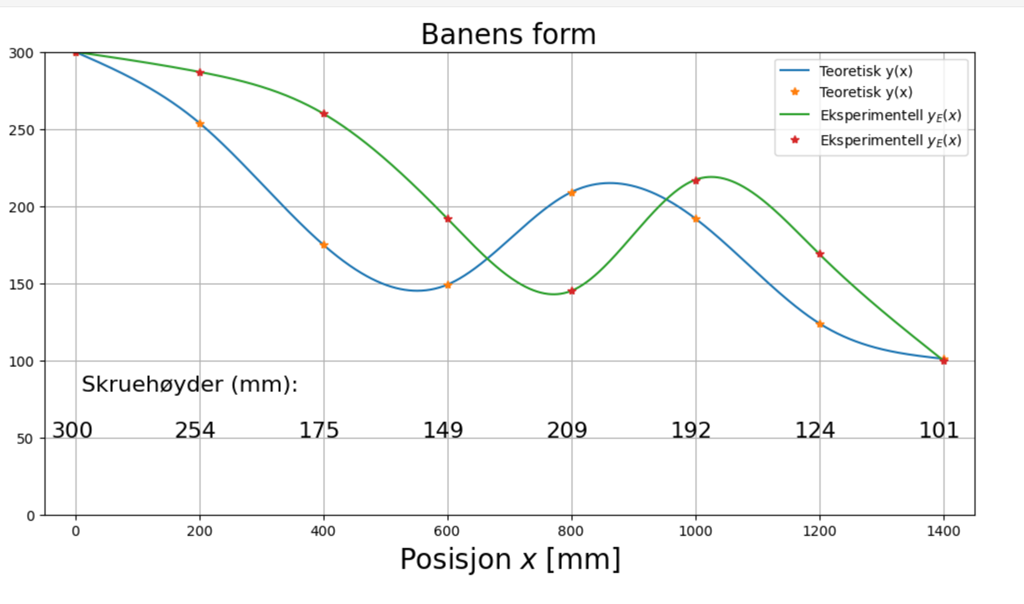
I økt 4 ble resultater fra eksperimentet sammenlignet med teoretiske verdier. Plottingen av grafene skjedde ved hjelp av grafverktøyet til Matplotlib og Numpy arrays. For å sammenligne disse verdiene ble de teoretiske verdiene for baneform, hastighet, slutthastighet, friksjonskraft, normalkraft og friksjonskoeffisient plottet mot middelverdiene til de 10 utførte forsøkene.

Resultater

Den teoretiske baneformen $y(x)$ med tilfeldig genererte skruhøyder avbildet i figur 3, er opphavet til de forventede resultatene av eksperimentet. De samme skruhøydene ble rekonstruert i den fysiske baneformen $y_E(x)$ for å konstantere eksperimentelle resultater.

In [ ]:
```python
display.Image("images/baneform.png")
```

Out[ ]:



Figur 3. Teoretisk baneform sammenlignet med eksperimentall baneform.

Den teoretiske rulletiden på kula var 1.478 s og ble kalkulert med /ref(ligning). Gjennomsnittsrulletiden for kulen ved de 10 forsøkene ble t_E = (1.3067 \pm 0.01227) s. Usikkerheten er her standardfeilen.

Tabell 1 viser slutthastighetene ved de 10 eksperimentelle forsøkene, sammen med den teoretiske kalkulerte slutthastigheten v = 1.66998 s. Den gjennomsnittlige slutthastigheten med standardfeil blir v_E = (1.7075 \pm 0.01663)m/s. Hastigheten $v_E(t)$ for et av de ti forsøkene er plottet med referansehastigheten $v(t)$ i figur 4.

In [ ]:
```python
display.Image("images/tabell_hastighet.png")
```

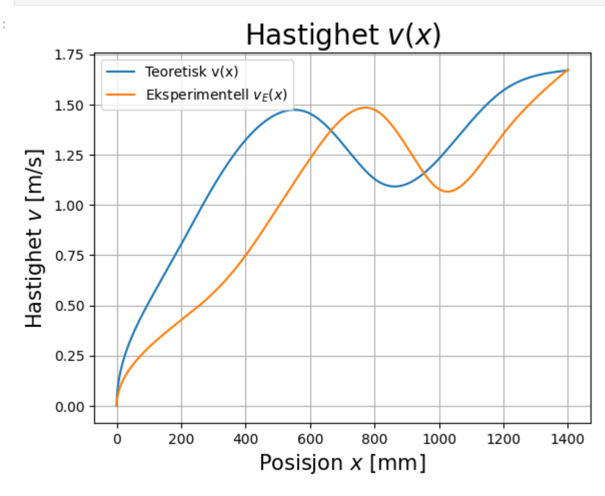
Out[ ]:

Slutthastighet	v [m/s]
1	1.69
2	1.63
3	1.81
4	1.77
5	1.71
6	1.65
7	1.71
8	1.69
9	1.74
10	1.66
Teoretisk	1.67

Tabell 1. Slutthastigheter i 10 forsøk, med referanse

In [ ]:
```python
display.Image("images/hastighet.png")
```

Out[ ]:



Figur 4. Teoretisk hastighet sammenlignet med eksperimentell hastighet.

Middelverdien med standardfeil for den kinetiske energien i de 10 eksperimentelle forsøkene blir kalkulert til $Ekin_E$ = (0.04524 \pm 0.00089) J. Tabell 2 viser de 10 forsøkene med den teoretiske referansen $Ekin$ = 0.0605179 J. Den potensielle energien ved starten av rullebanen er $Epot$ = 0.06051789 J. Med middelverdien for den kinetiske energien og den potensielle energien, kalkuleres middelverdien for tap av mekaniskenergi til $Etap$ = (0.01528 \pm 0.00089) J.

In [ ]:
```python
display.Image("images/tabell_kinetisk_energi.png")
```

Out[ ]:

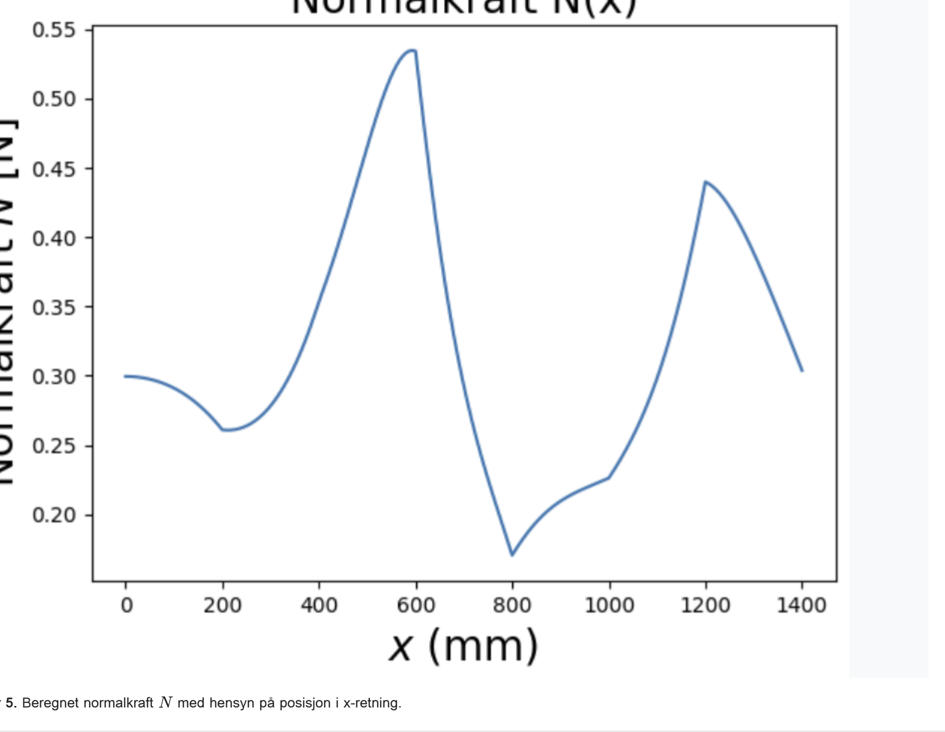
Kinetisk energi	E
1	0.044
2	0.042
3	0.051
4	0.049
5	0.045
6	0.042
7	0.045
8	0.045
9	0.047
10	0.043
Teoretisk	0.061
Potensiell energi start	0.061

Tabell 2. Kinetisk energi i 10 forsøk, med referanse og potensiell energi (start)

Friksjonskraften f gjennom hele banen er vist i figur 5. Normalkraften N vises i figur 6. Forholdet mellom friksjonskraften og normalkraften med hensyn på posisjonen i x-retning gir friksjonskoeffisienten μ, presentert i figur 7.

In [ ]:
```python
display.Image("images/normalkraft.png")
```

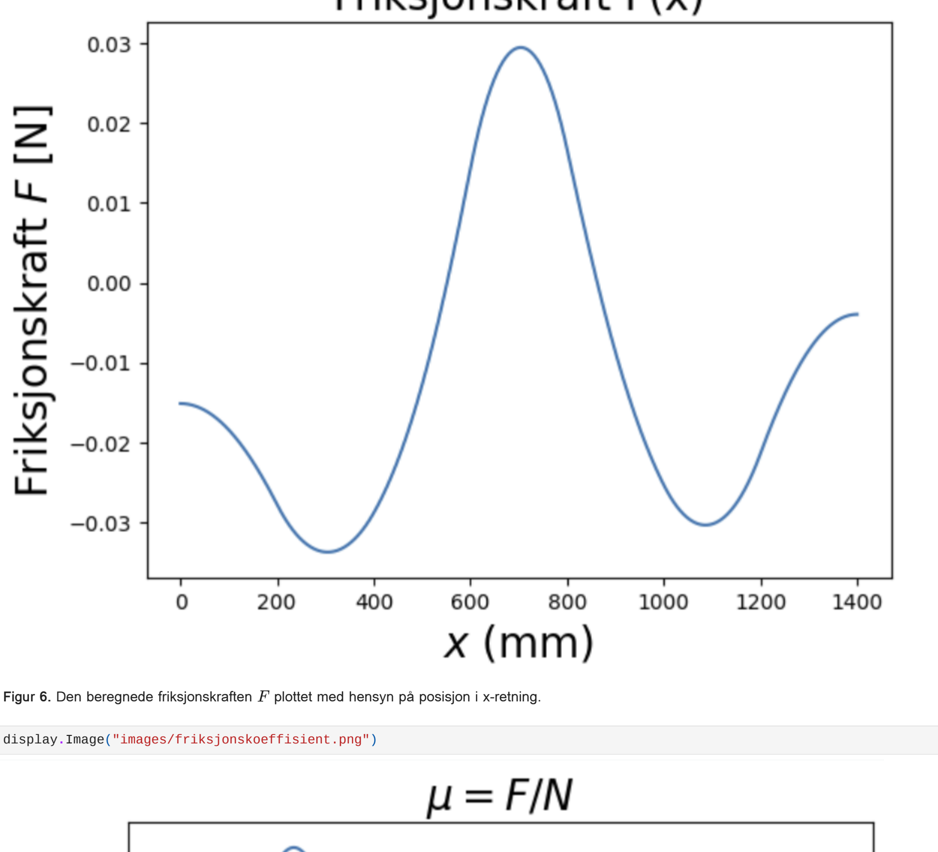
Out[ ]:



Figur 5. Beregnet normalkraft N med hensyn på posisjon i x-retning.

In [ ]:
```python
display.Image("images/friksjonskraft.png")
```

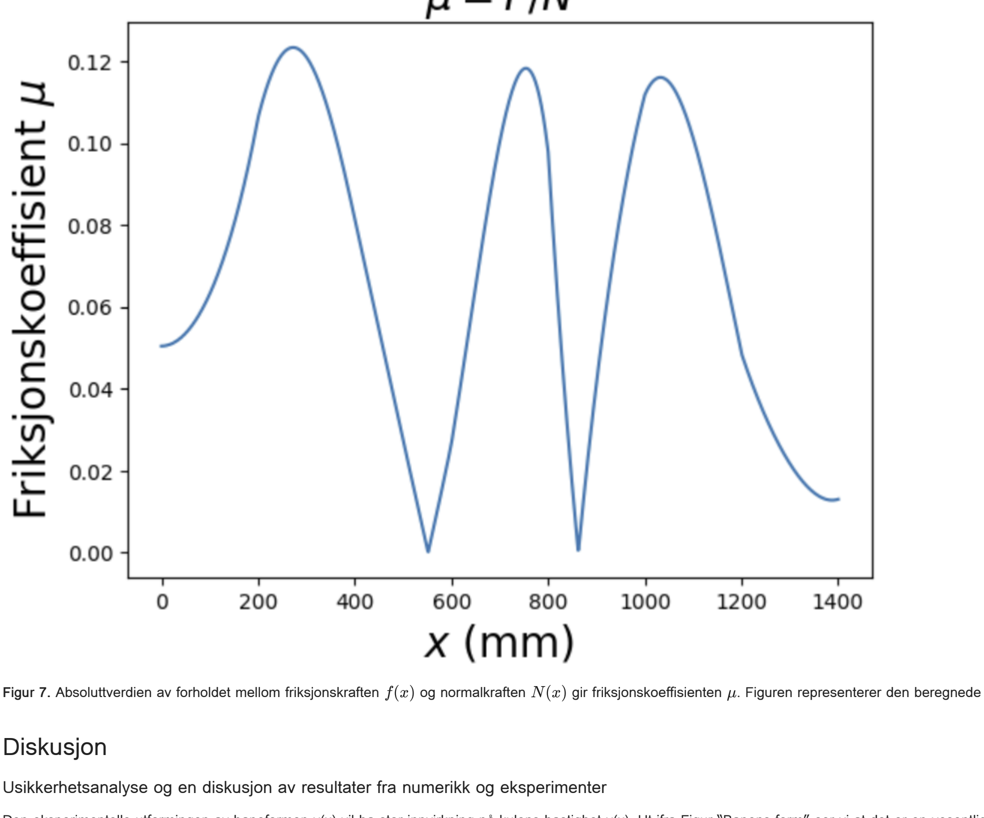
Out[ ]:



Figur 6. Den beregnede friksjonskraften F plottet med hensyn på posisjon i x-retning.

In [ ]:
```python
display.Image("images/friksjonskoeffisient.png")
```

Out[ ]:



Figur 7. Absoluttverdien av forholdet mellom friksjonskraften $f(x)$ og normalkraften $N(x)$ gir friksjonskoeffisienten μ. Figuren representerer den beregnede friksjonskoeffisienten μ med hensyn på posisjonen i x-retning.

Diskusjon

Usikkerhetsanalyse og en diskusjon av resultater fra numerikk og eksperimenter

Den eksperimentelle utformingen av baneformen y(x) vil ha stor innvirkning på kulens hastighet v(x). Ut ifra Figur "Banens form" ser vi at det er en vesentlig differanse mellom eksperimentell og teoretisk baneform, dette kan skyldes at forsøkene ble filmet med en dårlig kameravinkel som igjen kan ha ført til at videoanalyseverktøyet "Tracker" ikke klarte å modellere banen riktig. Grafen vist i Figur [banens form] tyder på at eksperimentell baneform er forskjøvet med ca. 200 mm. Dette medfører at påfølgende graf for eksperimentell hastighet vist i Figur [hastighet] også er forskjøvet i forhold til teoretisk hastighet.

Baneformen y(x) vil også påvirke størrelsen N(x). Ved x = 600 mm (y(x) sitt bunnpunkt) vil kula samtidig være ved v(x) sitt eneste toppunkt. Høy hastighet ved et lavt punkt tilsier at kula krever større sentripetal kraft for å følge baneformens krumning. Dette samsvarer med grafen til N(x), hvor x=600 tilsvarer et toppunkt, her vil normalkraften være større enn kulas vekt, fordi den balanserer vekten og den nødvendige sentripetalkraften. Det motsatte tilfellet vil være ved toppunktet til y(x), hvor N(x) tilsvarer -1.3 N, som er mindre enn kulas tyngde. Dette skyldes at kun en del av tyngden virker som normalkraft, mens resten bidrar til den sentripetale banen.

Fra Figur [friksjonskoeffisient] ser vi hvordan friksjonskoeffisienten μ_s varierer med posisjonen x. Her ser vi at den høyeste beregnede friksjonskoeffisienten μ er på ca. 0.12. Den statiske friksjonskoeffisienten μ_s er anslått til å være 0.4. Dermed er det rimelig å anta at den beregnede friksjonskraften f ikke overstiger sin maksimale verdi gitt ved $f = \mu_s|N|$ og at antagelsen om ren rulling er oppfylt.

Som presentert i Tabell [rulletid] er den målte rulletiden til kulen 1.31s, med en standardfeil på +-0.0123s. Den beregnede rulletiden er på 1.48s. Dette avviket kan skyldes at den teoretiske og den eksperimentelle baneformen ikke samsvarer fullstendig. En annen årsak kan være at kulen sin startfart $v_0 > 0$ eller at det er gjort målefeil. Dersom dette avviket skyldes målefeil er trolig den samme feilen gjort i hvert forsøk siden standardfeilen er såpass liten. Med utgangspunkt i beregnet rulletid kan en konkludere med at den målte rulletiden er rimelig.

Som vist i tabell 2 får vi et et tap av kinetisk energi på 0.01528 J. Med tanke på at overflaten til kulebanen er svært jevn (ref. friksjonskoeffisienten i figur 7) kan man anta at tapet av kinetisk energi er rimelig. En tilhørende standardfeil på +-0.00089 J tilsier at det er lite spredning i forsøkene. Eventuelle feilkilder knyttet til lave standardfeil kan være systematiske feil, som at kulens startfart $v_0 > 0$.

Konklusjon

Det har blitt gjennomført et forsøk hvor en kule har blitt sluppet gjennom en kulebane. I forkant av forsøket ble det gjennomført teoretiske beregninger for å simulere oppførselen til kula, disse beregningene skulle sammenlignes med eksperimentelle målinger. Den teoretiske rulletiden var 1.48 s, mens den eksperimentelle rulletiden ble målt til 1.31 s. Den teoretiske sluttfarten var 1.67 m/s, mens den eksperimentelle sluttfarten ble målt til 1.71 m/s, med tilhørende standardfeil 0.016 m/s. Tap av kinetisk energi tilsvarte 0.01528 J med standardfeil +-0.00089 J.

Referanser

[1] Tracker Video Analysis and Modeling Tool, https://physlets.org/tracker/, Douglas Brown, Wolfgang Christian, Robert M Hanson, 2023
[2] Isaac Newton, Philosophiæ naturalis principia mathematica, (1687).