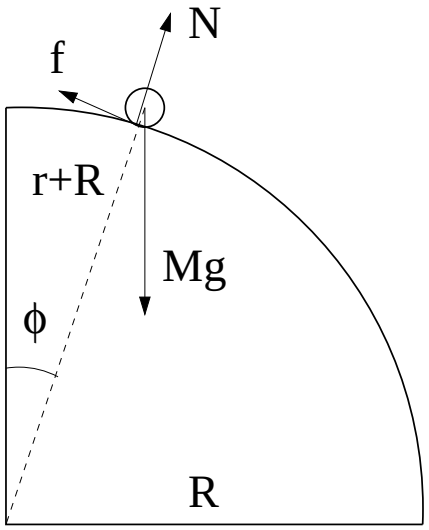
FY1001/TFY4145 Mekanisk fysikk. Institutt for fysikk, NTNU. Høsten 2014.
Veiledning: 7. - 10. oktober. Innleveringsfrist: Mandag 13. oktober kl 14.

Øving 6

Obligatorisk øving som i stor grad er basert på bruk av MATLAB

Oppgaven, i meget korte trekk

- Bruk MATLAB til å løse bevegelsesligningene numerisk for ett eller flere legemer som ruller nedover på utsiden av en kvartsirkel, fra toppen ($\phi = 0^+$) og fram til der legemet mister kontakten med underlaget ($\phi = \phi_k$).
- Les *eksperimentelle* data (t, x, y) fra tekstfiler inn i MATLAB og sammenlign de numeriske beregningene med eksperimentene.
- Presenter og diskuter resultatene i besvarelsen.



Kvalitativ beskrivelse av problemstillingen

I denne øvingen skal vi studere legemer som ruller og glir (slurer) på en vertikaltstilt sirkel. De ulike legemene befinner seg på utsiden av sirkelen og er ikke festet til underlaget, så det er tilstrekkelig å betrakte en kvartsirkel, som i figuren til venstre. Hvis det ikke er friksjon mellom legeme og underlag, blir problemet enkelt å løse: Kreftene som virker på legemet er tyngdekraften $G = Mg$ og normalkraften N fra underlaget, og legemets tyngdepunkt får en akselerasjon A bestemt av Newtons 2. lov (N2), parallelt med sirkelbanen pga tyngdens parallellkomponent, og normalt sirkelbanen pga N og tyngdens normalkomponent. Vi kjenner sammenhengen mellom legemets hastighet V og akselerasjonens normalkomponent (sentripetalakselerasjonen). Dessuten kan normalkraften N ikke godt bli mindre enn null. Denne visdommen gjør oss i stand til å avgjøre hvor legemet mister kontakten med underlaget og endrer bevegelsesmønster, fra sirkelbevegelse til typen "skrått kast". (Jfr oppgave 2b i øving 3: Energibevarelse kombinert med N2 normalt på sirkelbanen.)

Friksjon kompliserer bildet. Vi begrenser oss til legemer som har rotasjonssymmetri mhp en akse gjennom tyngdepunktet, normalt på bevegelsesretningen, samt en veldefinert "rulleradius" r som angir avstanden fra tyngdepunktet til kontaktpunktet mellom legeme og underlag (så lenge disse to *har* kontakt, selvsagt). Legemets treghetsmoment mhp nevnte akse kan da alltid skrives på formen

$$I_0 = c \cdot Mr^2,$$

der $c = 1$ for en tynn ring eller hul sylinder, 1/2 for en kompakt sylinder, 2/5 for ei kompakt kule og 2/3 for et tynt kuleskall. Som diskutert i forelesningene er det friksjonskraftens dreiemoment $\tau = fr$ som sørger for at legemet begynner å rulle og at vinkelhastigheten øker:

$$fr = I_0\dot{\omega}.$$

I starten vil f være stor nok til at vi får ren rulling og kan bruke rullebetingelsene

$$V = r\omega \quad ; \quad A = r\dot{\omega} = r\alpha.$$

Men etter hvert som helningen og vinkelen ϕ øker, blir normalkraften N - og dermed den maksimale friksjonskraften $f_{\max} = \mu_s N$ - mindre og mindre, og til slutt for liten til at ren rulling kan opprettholdes. Legemet går ved en viss vinkel ϕ_r over til å "slure", med vinkelhastighet $\omega < V/r$. Det betyr at relativ hastighet mellom legeme og underlag ikke lenger er lik null i kontaktpunktet (slik tilfellet er ved ren rulling), men snarere $v_{\text{rel}} = V - \omega r > 0$. Det må igjen bety at underlaget utfører et (negativt) friksjonsarbeid på legemet, mer presist $dW_f = -f\,ds = -f\,v_{\text{rel}}\,dt$ i løpet av et lite tidsintervall dt. Men sluring eller ei, legemets hastighet øker på vei nedover og blir før eller senere så stor at normalkraften N fra underlaget reduseres til minimumsverdien null. Det roterende legemet mister kontakten med underlaget, og bevegelsen til massesenteret kan deretter beskrives som et skrått kast.

En enda grundigere diskusjon av problemet vil bringe luftmotstand og skruballproblematikk på banen, og for legemer med liten masse (f.eks en bordtennisball) kan vi ikke forvente at slike effekter vil være neglisjerbare. I denne øvingen ser vi bort fra luftmotstand og avbøyning pga skru (Magnus-effekten).

Aktuelle ligninger og uttrykk.

N2 langs banen:

$$Mg\sin\phi - f = MA.$$

N2 normalt banen:

$$Mg\cos\phi - N = M\frac{V^2}{r+R}.$$

N2 for rotasjon om legemets tyngdepunkt (CM):

$$fr = I_0\alpha.$$

Legemets treghetsmoment mhp symmetriaksen gjennom CM:

$$I_0 = cMr^2.$$

Akselerasjon langs banen:

$$A = \frac{dV}{dt}.$$

Hastighet langs banen:

$$V = (r+R)\frac{d\phi}{dt} = (r+R)\Omega.$$

Maksimal friksjonskraft fra underlaget på legemet:

$$f_{\max} = \mu_s N.$$

Så langt generelle ligninger og uttrykk som gjelder så lenge det er kontakt med underlaget, slik at legemets CM hele tiden befinner seg i avstand $r + R$ fra origo (sirkelens sentrum).

I starten, så lenge $0 < \phi \leq \phi_r$, har vi *ren rulling*. Da gjelder sammenhengene (kjent fra forelesningseksemplet om rulling på skråplan):

$$\begin{aligned} V &= \omega r \\ A &= \alpha r \\ f &= I_0\alpha/r = cMA \\ A &= g\,\frac{\sin\phi}{c+1} \end{aligned}$$

Vinkelen ϕ_r for overgangen fra ren rulling til sluring bestemmes av ligningen

$$f = f_{\max},$$

som resulterer i

$$(c+3)\cos\phi_r - 2 = \frac{c}{\mu_s}\sin\phi_r.$$

Denne ligningen kan løses analytisk: Sett $x = \cos\phi_r$, $\sqrt{1-x^2} = \sin\phi_r$, og kvadrer på begge sider av ligningen. Dette gir en andregradsligning for x, der den ene av de to løsningene (den største x-verdien) tilsvarer den aktuelle ϕ_r i 1. kvadrant ($0 < \phi_r < \pi/2$). Men strengt tatt har vi ikke behov for denne løsningen: I MATLAB-programmet kan vi ganske enkelt sammenligne verdiene for f og $f_{\max}$ underveis, der vi i utgangspunktet antar ren rulling og $f = cMA$. Dersom $f \leq f_{\max}$, *har* vi ren rulling, som antatt. Og dersom $f > f_{\max}$, har vi *ikke lenger* ren rulling, hvoretter vi endrer friksjonskraften (til $f = \mu_k N$, som gjelder for sluring) og andre berørte størrelser (som A, ω og α). Dette gir en numerisk fastleggelse av ϕ_r, som eventuelt kan settes inn i ligningen ovenfor til slutt, for å sjekke at det er samsvar mellom "numerikken" og den analytiske løsningen.

Fra og med $\phi = \phi_r$ har vi altså sluring. Programmet kjører så videre inntil legemet mister kontakten med underlaget, ved vinkelen ϕ_k bestemt ved at $N(\phi_k) = 0$.

Numerisk løsning av bevegelseslignigene.

Bevegelseslignigene er differensialligninger, for eksempel $F = MA = M\,dV/dt$, slik at vi kan skrive

$$dV = \frac{F}{M}\,dt$$

på differensiell form. I numeriske beregninger kan vi selvsagt ikke ta uendelig små tidssteg dt, men vi kan ta et *lite* tidssteg Δt, som gir hastighetsendringen

$$\Delta V = \frac{F}{M}\,\Delta t.$$

Med andre ord, hvis vi kjenner $V(t)$, kan vi beregne $V(t+\Delta t)$ slik:

$$V(t+\Delta t) = V(t) + \frac{F(t)}{M}\,\Delta t = V(t) + A(t)\Delta t.$$

Her skulle egentlig likhetstegnene ha vært erstattet med $\simeq$, men med et tilstrekkelig lite tidssteg Δt blir feilen vi gjør også liten.
I et dataprogram er det hensiktsmessig å lage *tabeller* med verdier for de ulike størrelsene som skal beregnes. Vi lager også en tilsvarende tabell for variablene som de fysiske størrelsene er funksjoner av. I vårt tilfelle er det tiden t som er den aktuelle variabelen, og vi setter

$$t(1) = 0,\ t(2) = \Delta t,\ t(3) = 2\Delta t,\ \ldots\ t(j+1) = j\cdot\Delta t.$$

Alle tidsavhengige fysiske størrelser i det aktuelle problemet puttes inn i tilsvarende tabeller, slik at $V(1) = V(t(1)) = V(t=0)$, $V(j+1) = V(t(j+1)) = V(j\Delta t)$ osv. På denne måten vil element nr $j+1$ i alle tabellene representere de ulike fysiske størrelsene ved samme tidspunkt $t(j+1) = j\Delta t$.
Her er noen størrelser som man vanskelig kan komme utenom i det aktuelle problemet:

$$\begin{aligned} V(j+1) &= V(j) + A(j)\Delta t \\ \Omega(j) &= V(j)/(r+R) \end{aligned}$$

$$
\begin{aligned}
\phi(j+1) &= \phi(j) + \Omega(j)\Delta t \\
A(j) &= g\sin(\phi(j))/(c+1) \quad \text{rulling} \\
f(j) &= cMA(j) \quad \text{rulling} \\
f(j) &= \mu_k N(j) \quad \text{sluring} \\
A(j) &= g\sin(\phi(j)) - f(j) \quad \text{sluring} \\
\alpha(j) &= A(j)/(r+R) \\
N(j) &= Mg\cos(\phi(j)) - MV(j)^2/(r+R) \\
f_{\max}(j) &= \mu_s N(j)
\end{aligned}
$$

For å "komme i gang", må en tilordne startverdier for enkelte størrelser. Det kan for eksempel være $V(1) = 0$ (hvis legemet starter i ro) og $\phi(1) = \phi_0$ (hvis legemets CM starter ved vinkelen ϕ_0). Deretter kan $\Omega(1)$, $A(1)$, $f(1)$, $N(1)$ og $\alpha(1)$ beregnes med formlene ovenfor. Kjennskap til $A(1)$ gir grunnlag for å bestemme $V(2)$, og deretter $\Omega(2)$ og $\phi(2)$, og alle de andre størrelsene for $j = 2$. Og så videre!
For hvert trinn i en slik syklus må vi sjekke (1) om vi fortsatt har kontakt med underlaget (dvs om $N > 0$), og hvis ja, (2) om vi har rulling eller sluring. I MATLAB gjøres slike tester hensiktsmessig med kommandoene `while ... end` eller `if ... else ... end`.

MATLAB-program.

MATLAB-programmet oving6.m kan lastes ned fra hjemmesiden. Programmet bruker tallverdier som er relevante for eksperimentet med bordtennisball. Et filmklipp ligger i fila bordtennisball.mov. Eksperimentelle data for t, x og y ligger i fila bordtennisball.txt. Dataene leses inn i programmet, og eksperimentelle hastighetskomponenter bestemmes numerisk,

$$V_x^{\exp}(j) = \frac{x(j+1) - x(j-1)}{t(j+1) - t(j-1)},$$

og tilsvarende for $V_y^{\exp}$. Mot slutten av programmet plottes V_x og V_y, bestemt numerisk fra teori og numerisk fra eksperiment, i samme figur, slik at de kan sammenlignes. Detaljer om fire ulike eksperimenter kommer i neste avsnitt.

Eksperimentelt.

Eksperimenter er utført med fire legemer: bordtennisball, metallsylinder, taperull og volleyball.

Bordtennisball:
Radius: 19 mm.
Treghetsmoment: $c = 2/3$.
Materiale: Hard plast.
Startvinkel: $\phi_0 = 0.047$ rad.
Filer: bordtennisball.mov og bordtennisball.txt

Metallsylinder:
Radius: Ytre radius 19 mm, indre radius 17 mm.
Treghetsmoment: $c = 0.90$.
Materiale: Metall (messing).
Startvinkel: $\phi_0 = 0.027$ rad.
Filer: sylinder.mov og sylinder.txt

Taperull:
Radius: Ytre radius 37.5 mm, indre radius 12.5 mm.
Treghetsmoment: $c = 5/9$.
Materiale: Myk plast.
Startvinkel: $\phi_0 = 0.045$ rad.
Filer: taperull.mov og taperull.txt

Volleyball:
Radius: 101 mm.
Treghetsmoment: $c = 2/3$.
Materiale: Myk plast.
Startvinkel: $\phi_0 = 0.055$ rad.
Filer: volleyball.mov og volleyball.txt

Oppgaven, en mer detaljert beskrivelse

- Last ned txt-filene og oving6.m og legg dem i samme katalog (mappe). Start opp MATLAB og kjør oving6.m for å sjekke at det fungerer som det skal. Gå gjennom programmet slik at du forstår det som gjøres.
- Lagre en kopi av oving6.m, for eksempel med filnavn taperull.m (avhengig av hvilket legeme du har tenkt å studere). Foreta nødvendige endringer av parametre: Friksjonskoeffisientene, legemets radius r, størrelsen c som inngår i treghetsmomentet, startvinkelen $\phi(1)$, starthastigheten $V(1)$. Kjør programmet og sammenlign med eksperimentet. Du må antagelig finjustere noen av parametrene for å få godt samsvar mellom teoretiske og eksperimentelle data.
- Identifiser beregnede vinkler for overgangen mellom ren rulling og sluring, ϕ_r, og for der legemet mister kontakten med underlaget, ϕ_k. Sjekk om den numerisk beregnede ϕ_r stemmer med ligningen for ϕ_r i teksten ovenfor (side 3).
- **Innlevering**: Lever pr epost til studass. Du må ha med figuren med fire subplott (som i oving6.m) for minst ett av de andre tre legemene, i pdf-format. Legg inn en linje i MATLAB-programmet slik at figuren inneholder navnet ditt. Det er ikke nødvendig å sende MATLAB-programmet til studass. Skriv, i eposten eller i pdf-figuren, hvilke parameterverdier du har brukt (for μ_s, μ_k og $V(1)$). Kommenter kort resultatene: Samsvar mellom teori og eksperiment; samsvar mellom numerisk beregnet ϕ_r og teoretisk ϕ_r.

Forslag til frivillige tilleggsoppgaver: Bestem ϕ_k basert på de eksperimentelle resultatene. (Tips: Betrakt avstanden fra origo til legemets massesenter.) Undersøk hvordan det ligger an med energibevarelse i den numeriske løsningen av ligningene, og eventuelt i de eksperimentelle dataene. Sammenlign numerisk og eksperimentell hastighet. Plott effekttap (pr masseenhet) pga sluring. Plott relativhastigheten i kontaktpunktet mellom objekt og underlag. Plott friksjonskraften.

Tips om PDF-figurer i MATLAB
MATLAB er nok ikke verdens beste program for å lage vakre figurer, men det holder for mange praktiske formål. Et minimumskrav, dersom du ønsker å ta en MATLAB-generert PDF-figur inn i et LaTeX-dokument som f.eks labrapporten, er at overflødig tomrom rundt selve figuren ikke blir med i det endelige produktet. Det er flere måter å unngå dette på, og du kan kanskje en allerede. Hvis ikke, lagre filene barefigur.m, plottgraf.m og pdfilatex.tex i samme katalog, start opp MATLAB, kjør programmet barefigur.m og kompiler LaTeX-fila pdfilatex.tex. Les innholdet i pdfilatex.pdf, og se hva som skjer dersom du lager figuren baregraf.pdf uten å bruke funksjonen barefigur.m.