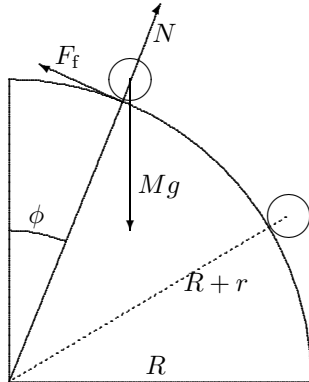


Øving 9

Veiledning: Tirsdag 29 okt. og onsdag 30. okt., se nettsider.

Innlevering: Torsdag 31. okt. kl. 14:00.

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



I denne øvingen skal vi studere legemer som ruller eller 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 kvartssirkel, som i figuren til venstre.

Dersom legemet sklir *friksjonsfritt* på underlaget ($F_f = 0$) blir det ingen rotasjon og problemet er enkelt å løse. Ble gjennomgått i eksempel i forelesning 19.09.13 med bruk av energibevarelse. Kraftene som virker på legemet er tyngdekrafta $G = Mg$ og normalkrafta 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. Normalakselerasjonen er lik sentripetalakselerasjonen.

Dessuten kan normalkrafta 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".

Friksjon kompliserer bildet. La oss diskutere problemet kvalitativt først. For det første gir friksjonskrafta rulling av legemet. Vi begrenser oss til legemer som har rotasjonssymmetri mhp. en akse gjennom tyngdepunktet, og dermed kan rulle om denne akse. Legemets radius er r og lik rulleradius. Legemets treghetsmoment mhp. rotasjonsaksen kan da 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. Det er friksjonskraftas dreiemoment som sørger for at legemet begynner å rulle og at vinkelhastigheten endrer seg,

$$\tau = F_f r = I_0 \dot{\omega}.$$

I starten vil F_f være stor nok til at vi får rein rulling og kan bruke rullebetingelsene.

$$v = r\omega \quad ; \quad a = r\dot{\omega}.$$

Men etter hvert som helningen og vinkelen ϕ øker, blir normalkrafta N - og dermed den maksimale friksjonskrafta $F_{f,\max} = \mu_s N$ - mindre og mindre, og til slutt for liten til at rein rulling kan opprettholdes. Legemet går ved en viss vinkel ϕ_r over til å "slure", og da er vinkelhastigheten $\omega < v/r$. Det betyr at relativ hastighet mellom legeme og underlag ikke lenger er lik null i kontaktpunktet (slik tilfellet er ved rein rulling), men snarere $v_{\text{rel}} = v - \omega r > 0$. Det må igjen bety at underlaget utfører et (negativt) friksjonsarbeid på legemet, i tidsintervallet dt er arbeidet $dW_f = -F_f ds = -F_f v_{\text{rel}} dt$ der $F_f = \mu_k N$.

Med sluring eller ei, legemets hastighet øker på vei nedover og blir før eller senere så stor at normalkrafta N fra underlaget reduseres til minimumsverdien null, og det roterende legemet mister kontakten med underlaget og 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.

Oppgaven i meget korte trekk. Detaljer på siste side.

- Bruk MATLAB til å løse bevegelseslikningene numerisk for ett eller flere legemer som ruller nedover en kvartssirkel, fra toppen ($\phi = 0^+$) og fram til der legemet mister kontakten med underlaget ($\phi = \phi_k$).
- Les *eksperimentelle* data (t, x, y) fra en eller flere filer inn i MATLAB og sammenlign dine numeriske beregninger med eksperimentet/-ene.
- Presenter og diskuter resultatene i besvarelsen.

Alle likninger og uttrykk vi trenger.

N2 langs banen:

$$Mg \sin \phi - F_f = Ma. \quad (1)$$

N2 normalt banen:

$$Mg \cos \phi - N = M \frac{v^2}{r + R}. \quad (2)$$

N2 for rotasjon om legemets tyngdepunkt (CM):

$$F_f r = I_0 \dot{\omega}, \quad \text{der } I_0 = cMr^2. \quad (3)$$

Akselerasjon langs banen:

$$a = \frac{dv}{dt}. \quad (4)$$

Hastighet langs banen:

$$v = (r + R) \frac{d\phi}{dt} = (r + R)\Omega. \quad (5)$$

Så langt generelle likninger 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). Merk at ω er vinkelhastigheten til legemets egenrotasjon mens Ω er vinkelhastigheten for CM's bevegelse på kvartsirkelen.

I starten, så lenge $0 < \phi \leq \phi_r$, har vi *rein rulling*. Da gjelder

$$v = \omega r \quad (6)$$

$$a = \dot{\omega} r \quad (7)$$

$$F_f = I_0 \dot{\omega} / r = cMa \quad (8)$$

$$a = g \frac{\sin \phi}{c + 1} \quad (9)$$

Vinkelen ϕ_r for overgangen fra rein rulling til sluring bestemmes av at friksjonen er lik maksimal friksjonskraft:

$$F_f = F_{f,\max} = \mu_s N. \quad (10)$$

I MATLAB-programmet kan vi sjekke om legemet fortsatt ruller ved å sammenligne verdiene for F_f og $F_{f,\max}$ underveis, der vi i utgangspunktet antar rein rulling og $F_f = cMa$. Dersom $F_f \leq F_{f,\max}$, har vi *rein rulling*, som antatt. Og dersom $F_f > F_{f,\max}$, har vi *ikke lenger* rein rulling. Dette gir en numerisk fastleggelse av ϕ_r . Analytisk kan ϕ_r bestemmes fra likningene ovenfor. Likningen vi kommer fram til er

$$(c + 3) \cos \phi_r - 2 = \frac{c}{\mu_s} \sin \phi_r. \quad (11)$$

Fra og med $\phi = \phi_r$ har vi altså sluring og friksjonskrafta settes til $F_f = \mu_k N$. Programmet kjører så videre med denne friksjonskrafta og tilhørende likninger. (Obs likningene (6)-(9) gjelder ikke lenger.) I MATLAB-programmet sjekkes for hver iterasjon om $N > 0$. Når iterasjonene finner at $N < 0$ har legemet mistet kontakten med underlaget, og vi har $\phi = \phi_k$. Videre bevegelse er skrått kast, som vi ikke skal analysere numerisk (programmet stoppes).

Numerisk løsning av bevegelseslikningene. (Jfr øving 7, oppgave 4.)

Bevegelseslikningene er differensiallikninger, 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:

$$v(t + \Delta t) = v(t) + \frac{F(t)}{M} \Delta t = v(t) + a(t)\Delta t. \quad (12)$$

Her skulle egentlig likhetstegnene ha vært erstattet med \simeq , men med tilstrekkelig lite tidssteg Δt blir feilen vi gjør også "tilstrekkelig liten".

I et dataprogram (som f.eks. MATLAB) 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

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

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 $t(j + 1) = j\Delta t$. Med disse definisjonene vil de numeriske likningene

du får bruk for være blant disse:

$$\begin{aligned}v(j+1) &= v(j) + a(j)\Delta t \\ \Omega(j) &= v(j)/(r+R) \\ \phi(j+1) &= \phi(j) + \Omega(j)\Delta t \\ a(j) &= g \sin(\phi(j))/(c+1) \quad \text{rulling} \\ F_f/M \rightarrow f(j) &= ca(j) \quad \text{rulling} \\ F_f/M \rightarrow f(j) &= \mu_k n(j) \quad \text{sluring} \\ a(j) &= g \sin(\phi(j)) - f(j) \quad \text{sluring} \\ \dot{\Omega} \rightarrow \alpha(j) &= a(j)/(r+R) \\ N(j)/M \rightarrow n(j) &= g \cos(\phi(j)) - v(j)^2/(r+R) \\ f_{\max}(j) &= \mu_s n(j)\end{aligned}$$

Merk at vi bruker kraft per masseenhet i programmet (f og n). Forsikre deg om at du kjenner igjen hver av likningene over. F.eks. representerer den første likningen likn. (12) og mange av likningene representerer de analytiske likningene (1)-(10).

For å ”komme i gang”, må en tilordne startverdier for enkelte størrelser. Vi kan f.eks. starte med legemet i ro ved en gitt liten vinkel ϕ_0 , da er $v(1) = 0$ og $\phi(1) = \phi_0$. 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` og/eller `if ... else ... end`.

MATLAB-program.

MATLAB-programmet `oving09.m` kan lastes ned fra nettets nettside. 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,

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

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 for de fire eksperimenter, utført med bordtennisball, metallsylinder, teiprull og volleyball.

Kvartsirke: Radius: 785 mm.

Bordtennisball:

Radius: 19 mm.

Treghetsmoment: $c = 2/3$.

Startvinkel: $\phi_0 = 0,047$ rad.

Filer: `bordtennisball.mov` og `bordtennisball.txt`

Metallsylinder:

Radius: Ytre radius 19 mm, indre radius 17 mm.

Treghetsmoment: $c = 0,90$.

Startvinkel: $\phi_0 = 0,027$ rad.

Filer: `sylinder.mov` og `sylinder.txt`

Teiprull:

Radius: Ytre radius 37,5 mm, indre radius 12,5 mm.

Treghetsmoment: $c = 5/9$.

Startvinkel: $\phi_0 = 0,045$ rad.

Filer: `teiprull.mov` og `teiprull.txt`

Volleyball:

Radius: 101 mm.

Treghetsmoment: $c = 2/3$.

Startvinkel: $\phi_0 = 0,055$ rad.

Filer: `volleyball.mov` og `volleyball.txt`

- Last ned txt-filene og oving09.m og legg dem i samme katalog (mappe). Start opp MATLAB og kjør oving09.m for å sjekke at det fungerer som det skal. Åpne oving09.m, gå gjennom programmet og prøv å forstå det som gjøres.
- Velg deg minst ett legemes bevegelse å studere nærmere. Hvis du f.eks. velger sylinderen lagrer du en kopi av oving09.m til filnavn cylinder.m. Foreta nødvendige endringer av parametre. Kjør programmet og sammenlign med eksperimentet. Du må selv justere de to friksjonskoeffisientene etter behov. Underlaget er laget av metall. Sylinderen er også av metall, mens volleyballen og teiprullen er en slags myk plast eller gummi. Prøv deg fram, eller se om du finner ”passende” verdier ved hjelp av litteratur, google etc.
- Identifiser beregnede vinkler for overgangen mellom rein rulling og sluring, ϕ_r , og for der legemet mister kontakten med underlaget, ϕ_k . Sjekk om den numerisk beregnede ϕ_r stemmer med likn. (11) for ϕ_r (sett inn dine funne verdier).
- Vinkelen ϕ_r for overgangen fra rein rulling til sluring bestemmes av likningen (11) ovenfor. Vis hvordan du kommer fram til denne likningen på basis av likn. (2) ved å bruke kriteriet $f = f_{\max}$ samt likningene (1), (8), (9) og energibevaring (fram til sluring starter).
- Frivillige tilleggsoppgaver:
 - Hvordan vil du gå fram for å bestemme ϕ_k basert på de eksperimentelle resultatene?
 - Undersøk hvordan det ligger an med energibevarelse i den numeriske løsningen av likningene, og eventuelt i de eksperimentelle dataene. Sammenlign numerisk og eksperimentell hastighet. Plott effekttap (per masseenheter) pga. sluring.
 - Plott relativhastigheten i kontaktpunktet mellom objekt og underlag.
 - Plott friksjonskrafta.
 - Likn. (11) kan forsåvidt løses ganske enkelt analytisk: Sett $x = \cos \phi_r$, $\sqrt{1 - x^2} = \sin \phi_r$, og kvadrer på begge sider. Dette gir en andregradslikning for x , der den ene av de to løsningene tilsvarer den aktuelle ϕ_r i 1. kvadrant ($0 < \phi_r < \pi/2$). Gjør dette og sjekk om dinne funne verdier for ϕ_r stemmer med dette.
- Innlevering: Lever på papir eller per epost til studass. Du må ha med figurer (som i oving09.m) for minst ett av de andre tre legemene. Skriv hvilke parameterverdier du har brukt. Diskuter/kommenter kort resultatene.
- Lykke til!