

Oblig3 i FYS-MEK/F 1110 våren 2006

Utforsking av et kaotisk system.

Les gjennom hele oppgaven (også neste side) før du starter med praktisk arbeid.

I denne obligen ønsker vi at du utforsker bevegelsen til en pendel som blir utsatt for et harmonisk kraftmoment i tillegg til tyngde og luftmotstand. Det er utarbeidet et skriv om kaotiske system tilgjengelig under “Materiell fra enkelte forelesninger” på kurswebsidene, og det er gitt en del tips om programmeringen i et skriv tilgjengelig fra “Gruppeoppgaver” på kurswebsidene. I utgangspunktet forventer vi at alle punktene nedenfor besvares, men vi krever ikke at både punkt *e* og *f* er besvart for å få godkjent obligen. Punkt *f* er antakelig det vanskeligste og det som tar aller mest kjøretid, men det er også det morsomste punktet!

a) Lag en figur som viser geometri og krefter/kraftmoment o.l. som virker på pendelen beskrevet i notatet “Kaotiske system i fysikken”. Referer til grunnleggende fysiske lover, og vis her ved at bevegelsesligningen til en pendelen, kan skrives som:

$$\tau(\theta, t) = -mgL\sin\theta - qL^2\omega + \tau_{y0}\sin(\omega_y t) = mL^2\alpha \quad (1)$$

hvor m er massen til pendelen, L pendelens lengde, g tyngdens akselerasjon, θ vinkelutslaget, ω vinkelhastigheten, α vinkelakselerasjonen, τ_{y0} en amplitude for det ytre påtrykte kraftmomentet, ω_y det påtrykte kraftmomentets vinkelhastighet, og t er tiden. Det antas at luftmotstanden er proporsjonal med farten og at luftmotstandskoeffisienten er q .

b) Lag et dataprogram (fortrinnsvis i Matlab) som kan beregne pendelens bevegelse ut fra ligning 1 samt definisjonene av vinkelhastighet og vinkelakselerasjon. Programmet skal lagre posisjon (dvs vinkel), vinkelhastighet og tid i tre ulike arrays og plote vinkelutslag som funksjon av tid. Programmet skal også generere faseplottet for bevegelsen. Vis at den enkleste variant av Eulers metode *ikke* gir tilfredsstillende resultat for “Sett A” (se neste side) av parametre og initialbetingelser. Juster programmet slik at du bruker Euler-Cromer’s metode i stedet (evt 4. ordens Runge-Kutta) og vis at denne metoden fungerer tilsynelatende ok. Bruk denne metoden i resten av obligen.

c) Beregn for hånd hvilken initiell vinkelhastighet som svarer til at man bare så vidt kommer opp til toppunktet når den initielle vinkelen er null (henger rett ned). Bestem periodetiden til bevegelsen for “Sett A” av parametre, men der initial-vinkelhastigheten nå varieres fra nær null til ca 50 % høyere verdi enn det som skal til for at pendelen svinger forbi toppunktet. Periodetidene kan leses ut av Matlab-grafene, gjerne ved å bruke zoom-funksjonen for å få en noenlunde nøyaktig avlesning. Lag et plot som viser hvordan periodetiden endrer seg med den initielle vinkelhastigheten. Dette plottet skal inn i rapporten du lager for obligen.

I rapporten må du også ta med et plot som viser vinkelutslag vs tid og et faseplot, som begge tilsvarer det tilfellet at man kommer *nesten* helt til topps før pendelen vender og faller ned igjen. Alle plottene skal inngå i rapporten du lager for obligen, sammen med beskrivelser og kommentarer til resultatene.

d) Kjør nå programmet for to sett parametre der det ytre kraftmomentet og friksjon er koblet inn (“Sett B” og “Sett C” sist i denne oppgaveteksten). For hvert av settene skal et plot som

viser vinkel som funksjon av tid tas med i rapporten din. Beskriv med ord resultatene du får (forsøk å tenke fysikk!).

e) Bestem (omtrentlig) Lyapunov-eksponenten for pendelen under betingelsene gitt i “Sett B” og “Sett C”. Du må modifisere programmet litt for å få gjort dette. (Det kan lønne seg å lagre programmet du har hatt hittil under nytt navn først, og så gjøre endringene etterpå slik at du ikke ødelegger det programmet du allerede har fått til. Kommenter gjerne ut de programlinjene som du ikke har bruk for i denne varianten av programmet.) Ta med to plot som viser hvordan du har bestemt Lyapunov-eksponenten for de to tilfellene. Fortell med ord hva de to ulike Lyapunov-eksponentene sier om systemet.

f) Lag en tredje variant av dataprogrammet ditt ved å ta utgangspunkt i en av de to forrige variantene og juster det slik at du får plottet et Poincaré-snitt gjennom faseplottet. Det vil da være umulig å ta vare på alle vinkler og hastigheter og tider under bevegelsen. Ta i stedet *bare* vare på punktene $\theta(t)$ og $\omega(t)$ der $\omega_p t$ passerer et heltalls multippel av 2π , og plott disse i et punktplot (ikke linjeplot) slik det er gjort i notatet om kaotiske system. Forkast gjerne de ca 40 første punktene for å unngå at innsvingningsforløpet når bevegelsen starter skal prege plottet. Bruk gjerne “Sett D” som utgangspunkt for beregningene her, men du kan med fordel undersøke hvordan Poincaré-snittet endrer seg dersom du endrer på ω_p (eller τ_{p0}) slik at du får *minst to* klart forskjellige Poincaré-snitt-plott. Ta med et passe utvalg av slike plot i rapporten din.

MERK: Skriv en fyldig rapport over arbeidet ditt med obligen, der utledningene inngår, der resultatene beskrives med ord i tillegg til plot, der resultatene kommenteres, og der det legges ved minst en versjon av programmet. Rapporten skal leveres enten fullstendig i papirformat eller fullstendig elektronisk i form av én fil i .pdf, .ps eller .doc-format.

Obligen leveres til Siw Bruer på ekspedisjonskontoret eller sendes som vedlegg til mail til <s.m.m.bruer@fys.uio.no>. **Skriv fullt navn og gruppenummer såvel i rapporten som i “subject linjen” i tilfelle du leverer elektronisk!** Obligen kan også sendes pr post til Siw Bruer, Fysisk institutt, Boks 1048 Blindern, 0316 Oslo. Leveringsfrist **fredag 28.april kl 1500**

Parametre for bruk i obligen (størrelser er angitt i SI-enheter, uten at selve enheten er oppgitt):

Sett A:	Sett B:	Sett C:	Sett D:
$m = 1.0$	$m = 1.0$	$m = 1.0$	$m = 1.0$
$L = 1.0$	$L = 1.0$	$L = 1.0$	$L = 1.0$
$g = 9.8$	$g = 9.8$	$g = 9.8$	$g = 9.8$
$q = 0.0$	$q = 1.0$	$q = 1.0$	$q = 0.7$
$\tau_{y0} = 0.0$	$\tau_{y0} = 1.0$	$\tau_{y0} = 30.0$	$\tau_{y0} = 20.0$
$\omega_y = 2.0$	$\omega_y = 2.0$	$\omega_y = 3.3$	$\omega_y = 2.0$
total_tid = 40	total_tid = 40	total_tid = 100	total_tid = 1000++
delta_t = 1/200	delta_t = 1/200	delta_t = 1/200	delta_t = 1/200
theta_i = 0.0	theta_i = 0.0	theta_i = 0.0	theta_i = 0.0
omega_i = 2.0	omega_i = 2.0	omega_i = 2.0	omega_i = 2.0