3 - LIKNINGER

La oss begynne med et sitat av en stor matematiker.

> *"It is impossible to exaggerate the extent to which modern applied mathematics has been shaped and fueled by the general availability of fast computers with large memories. Their impact on mathematics, both applied and pure, is comparable to the role of the telescopes in astronomy and microscopes in biology."*
>
> — *Peter Lax, Siam Rev. Vol. 31 No. 4*

Fysiske modeller formuleres stort sett som likninger som involverer fysiske størrelser. I de enkleste tilfellene er modellen en algebraisk likning, slik som $s = s_0 + vt$ eller $F = ma$ eller $\frac{1}{2}mv^2 = mgh$, og løsningen et tall. På skolen trente du veldig mye på å løse algebraiske likninger som

$$3x + 5 = 2$$

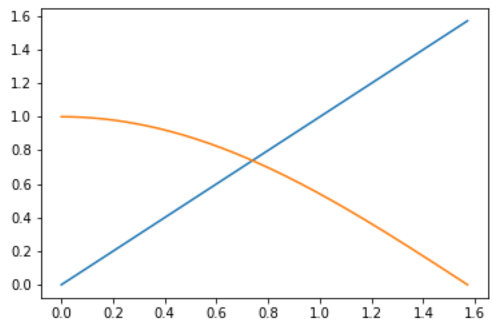
eller

$$x^2 + 2x + 1 = 0.$$

Denne tolv år lange løsningstreningsleiren gir et skjevt bilde av virkeligheten. De aller fleste likninger som modellerer noe interessant kan ikke løses for hånd på noen praktisk måte. La oss fleske til med en ganske vanskelig oppgave.

$\boxed{1}$ Prøv å løse likningen $x = \cos x$.

Her er et plot av funksjonene $f(x) = x$ og $g(x) = \cos x$. Hvis du fant løsningen, vil jeg gjerne vite hvordan du gjorde det. Det er lett å se at de to funksjonene krysser hverandre, så løsningen til likningen eksisterer, men den er ikke så lett å finne uten en kalkulator eller en datamaskin.
Løsningen kalles Dottietallet og er oppkalt etter en professor i fransk. Professor Dottie var gift med en matematiker, og oppdaget at hvis hun skrev inn et tilfeldig tall på kalkulatoren hans og trykket gjentatte ganger på cosinusknappen, endte hun alltid opp på omtrent 0.739085.



$\boxed{2}$ Gjenta professor Dotties eksperiment i Python.

Professor Dottie gjenoppdaget noe som kalles **fikspunktiterasjonen**. [1] Hun oppdaget at rekursjonen

$$x_{n+1} = \cos x_n$$

lager en følge som sakte men sikkert konvergerer mot den korrekte løsningen $L \approx 0.739085$.

Fikspunktiterasjonen er et eksempel på en av mange numerisk likningsløsere. En numerisk likningsløser er noe som produserer en følge $\{x_n\}$ som sakte men sikkert jobber seg inn mot løsningen til en likning. Det er nyttig å kjenne til numeriske likningsløsere, slik at man ikke blir stående fast bare fordi man støter på en likning som er umulig å løse med penn og papir.

$\boxed{3}$ Gjenta professor Dotties eksperiment i Python, men nå teller du hvor mange iterasjoner det tar før det stabiliserer seg på $L \approx 0.739085133215161$. Er du fersk i python kan du gjøre slik:

```
import numpy as np
x=1
for i in range(20):
    x=np.cos(x)
    print(x)
```

[1] https://www.maa.org/sites/default/files/Kaplan2007-131105.pdf

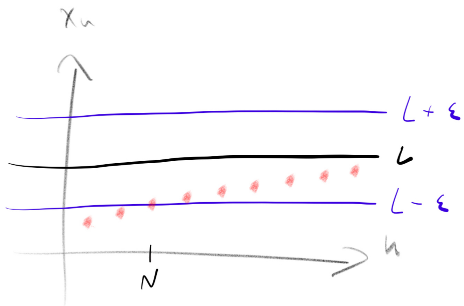
Nå er vi kommet til en klassisk situasjon. Det er lett å repetere professor Dotties eksperiment og se at desimalene stabiliserer seg. Men om vi skal skjønne ordentlig hva som skjer, er vi nødt til å innføre en del tekniske begreper. Her er det første:

Konvergent følge

Vi sier at en følge $\{x_n\}$ er konvergent dersom det finnes en L slik at det for enhver $\epsilon > 0$ finnes N slik at $n > N$ impliserer

$$|x_n - L| < \epsilon.$$

Dette er en definisjon som får mange studenter til å fryse på ryggen. Men det er dessverre den eneste måten å gjøre det på slik at alt blir presist. Tenk på L som det følgen konvergerer til (for eksempel løsningen til $x = \cos x$), og på ϵ som vilkårlig valgt bitte lite tall. Konvergens betyr at alle ledd etter x_N ligger nærmere L enn ϵ, og at du kan finne en slik x_N uansett hvor liten ϵ blir.



Når vi nå har definert konvergens, kan vi også definere fikspunktiterasjonen:

Fikspunktiterasjonen

Fikspunktiterasjonen

$$x_{n+1} = g(x_n)$$

produserer en følge som kan konvergere mot en løsning av likningen

$$x = g(x).$$

Om fikspunktiterasjonen konvergerer til noe fornuftig, kommer an på g og på x_0. La oss ta en oppgave som illustrerer dette.

$\boxed{4}$ Likningen

$$x \ln x = 1$$

kan skrives om til formen

$$x = g(x)$$

på minst to forskjellige måter. Finn de to åpenbare, og prøv fikspunktiterasjonen

$$x_{n+1} = g(x_n)$$

på begge.

Det finnes mange numeriske likningsløsere. En annen klassiker er **Newtons metode**. Denne baserer seg på at likningen som skal løses er skrevet på formen

$$f(x) = 0.$$

Den leter altså etter nullpunkter til funksjoner, og er gitt ved

$$x_{n+1} = x_n - \frac{f(x_n)}{f'(x_n)}.$$

$\boxed{5}$ Løs likningen $x = \cos x$ med Newtons metode og tell antall iterasjoner slik som i oppgave 3. (Du må skrive om likningen litt.) Hva ser du?

Det første elementet x_0 i følgen kalles **initialgjetningen**. Alle numeriske likningsløsere trenger en slik for å komme igang. (I koden i oppgave 3 er $x_0 = 1$.) Noen numeriske likningsløsere er mer sensitive på initialgjetningen enn andre, og forskjellige numeriske likningsløsere oppfører ganske ulikt.

$\boxed{6}$ Løs likningen $x = \cos x$ med både Newtons metode og fikspunktiterasjonen og eksperimenter med forskjellige initialgjetninger. Hva ser du?

Nå tar vi en oppgave fra den virkelige verden. Denne er visst en klassiker på maskiningeniørstudiet.

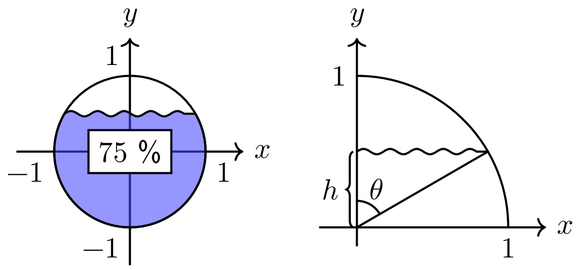
$\boxed{7}$ La h være vannstanden i et kloakkrør med radius 1 m, og la $h = \cos\theta$.

Når vannet fyller 75 % av rørets tverrsnitt, løser θ ligningen

$$2\sin 2\theta + \pi - 4\theta = 0.$$

Finn vannstanden h med fikspunktiterasjonen og Newtons metode.

(Figur: Marius Thaule.)



Nå kommer en litt teoretisk side. En likning på formen

$$x_{n+1} = h(x_n)$$

kalles en som nevnt **rekursjon** eller en **iterasjon**. Siden numeriske likningsløsere produserer en følge som i noen tilfeller konvergerer mot løsningen til likningen vi interesserer oss for, vil vi aldri finne løsningen helt nøyaktig, men vi kan bestemme oss for å slutte når vi synes vi har nok desimaler. Et vanlig triks for å vite når man skal gi seg, er å se på når det ikke lengre er endring i desimalene fra iterasjon til iterasjon. Da har datamaskinen har gjort så godt den kunne med den koden den fikk servert. Her er en presis definisjon av "ingen endring etter desimal sånn og sånn":

Cauchyfølge

Vi sier at en følge $\{x_n\}$ er en cauchyfølge dersom det for enhver $\epsilon > 0$ finnes en N slik at $m, n > N$ impliserer

$$|x_n - x_m| < \epsilon.$$

Definisjonen av cauchyfølge og konvergent følge ser veldig like ut, men det er en subtil forskjell - i definisjonen av cauchyfølge nevnes ikke hva følgen konvergerer til. Tenk igjen på ϵ (gresk bokstav, uttales epsilon) som et bitte lite tall. Definisjonen av cauchyfølge sier at dersom du går langt nok ut i følgen (altså at m og n er store), er det ingen endring i desimalene til venstre for en eller annen desimal gitt av størrelsen på ϵ. I et enkelt numerisk eksperiment slik som Dottie-eksperimentet sjekker vi strengt tatt ikke konvergens, vi sjekker bare om vi tror vi har en cauchyfølge. Det går bra, for i $\mathbb{R}$ er cauchyfølge og konvergent følge det samme. Dette er ikke sant i $\mathbb{Q}$, og derfor har vi $\mathbb{R}$:
`https://en.wikipedia.org/wiki/Construction_of_the_real_numbers`

$\boxed{8}$ Følgende oppgave illustrerer en av måtene å konstruere $\mathbb{R}$ på. Løs likningen

$$x^2 - 2 = 0$$

med Newtons metode, og svar på følgende to spørsmål:

- Er det sant at $x_n \in \mathbb{Q}$ for alle n dersom $x_0 \in \mathbb{Q}$?
- Dersom $\{x_n\}$ konvergerer, konvergeres det mot et rasjonalt tall?

En vanlig datamaskin regner stort sett med seksten desimaler. Seksten desimalers nøyaktighet kalles **maskinpresisjon**. Å skjønne nøyaktig hva datamaskinen gjør, er litt for komplisert for oss. Se her:
`https://www.math.ntnu.no/emner/MA2501/2008v/flyttall.pdf`
dersom du er interessert. Dersom du vil lese et helt kompendium om slikt, kan du lese dette:
`https://www.uio.no/studier/emner/matnat/math/MAT-INF1100/h21/kompendiet/matinf1100.pdf`

$\boxed{9}$ Hva slags ϵ korresponderer maskinpresisjon til?

Forresten kalles gjerne løsningen til en likning en **rot**. Dette er greit å vite når man prater med gamle folk. Man bruker også ordet "rot" om nullpunktet til en funksjon.

10 Likningen

$$x^2 = \cos x$$

har to røtter, og likningen kan skrives om til $x = g(x)$ på minst to åpenbare måter. Prøv fikspunktiterasjonen og Newtons metode på begge. Den ene løsningen er

$$L = 0.8241323123025225.$$

Hvor mange iterasjoner må til før det ikke er endring i desimalene?

11 Likningen

$$x^2 + x + 1 = 0$$

har som alle vet to komplekse røtter. Du kan finne disse numerisk, men du må fortelle Python at du har til hensikt å jobbe med komplekse tall:

```
import numpy as np
x=1+0*1j
for i in range(100):
    x=-np.sqrt(x-1)
    print(x)
```

Hvordan finner man den andre løsningen?

12 Vi vet jo at det kan være vanskelig å finne nullpunktene til et polyom, selv om algebraens fundamentalteorem garanterer at de finnes. Her kan numeriske likningsløsere være til hjelp. Faktoriser polynomet

$$x^3 + x^2 - 3x - 3 = 0.$$

13 Her er en oppgave som slenger ut noen spørsmål vi kan svare på mot slutten av semesteret. Polynomene

$$p(x) = x^4 - 10x^3 + 35x^2 - 50x + 24$$

og

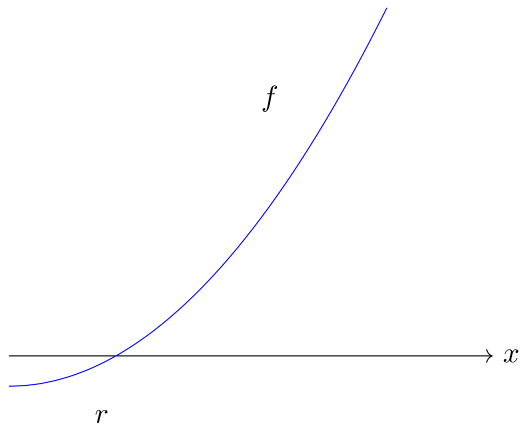
$$q(x) = x^4 - 9x^3 + 27x^2 - 31x + 12$$

har begge nullpunkt i $x = 1$. Prøv å finne nullpunktene med Newtons metode med startverdi $x_0 = 1/2$, og noter deg hvor mange iterasjoner som trengs for både p og q.

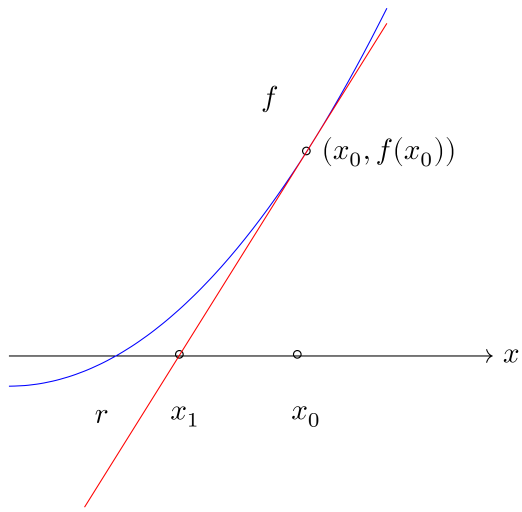
Her er til slutt en utledning av Newtons metode. Newtons metode baserer seg på at likningen er skrevet på formen

$$f(x) = 0.$$

Den leter altså etter nullpunkter til funksjoner. La oss si at nullpunktet vi leter etter kalles r.



La oss anta at vi har en tilnærming x_0 til r. Vi slår tangenten til f i x_0.



Punktet der tangenten skjærer x-aksen, kaller vi x_1. Dette punktet kan vi finne ved å sette opp likningen for tangenten til f i x_0:

$$y - f(x_0) = f'(x_0)(x - x_0)$$

og så kreve at $y = 0$ i denne likningen:

$$-f(x_0) = f'(x_0)(x_1 - x_0)$$

Løser vi denne likningen for x_1, får vi at

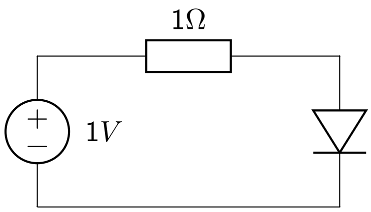
$$x_1 = x_0 - \frac{f(x_0)}{f'(x_0)}.$$

Dersom f og x_0 ser ut slik som i figuren, vil x_1 ligge nærmere nullpunktet enn x_0, og det trengs ikke så stor fantasi for å se at x_2 vil legge seg nærmere nullpunktet enn x_1. Newtons metode er definert som den rekursive følgen

$$x_{n+1} = x_n - \frac{f(x_n)}{f'(x_n)}.$$

UKENS ELEKTROTEKNIKK

Det er lett å finne frem til en krets der det vi må ty til numeriske likningsløsere. Du har allerede koblet den opp i ADE:



Vi er mest interesserte i den matematiske oppførselen til likningen, og derfor setter vi både $\frac{q}{nkT} = 1$ og $I_0 = 1$ slik at likningen blir

$$i(v) = e^v - 1$$

for det blir så greit. Lars kan ta seg av griseriet med alle de fysiske konstantene.

$\boxed{1}$ Gjør ovennevnte antagelser på dioden, og bruk Kirchhoffs spenningslov til å utlede likningen

$$1 = i + \ln(i + 1).$$

for strømmen i kretsen. Denne kan ikke løses analytisk. Bruk en numerisk likningsløser til å finne strømmen, og noter deg hvor mange iterasjoner som trengs.

Kirchhoff var forresten temmelig barsk i knolten. Han har funnet opp mye greier:
`https://no.wikipedia.org/wiki/Gustav_Kirchhoff`

UKENS KJEMI

$\boxed{1}$ Dersom man skal finne likevektskonsentrasjoner, ender man ofte opp med å måtte løse en ikkelineær likning. For eksempel må man løse likningen

$$K_c = \frac{16x^3}{(1 - 4x)^2}$$

dersom man skal finne likevektskonsentrasjonene i

$$2\,\mathrm{NOCl(g)} \rightleftharpoons 2\,\mathrm{NO(g)} + \mathrm{Cl_2(g)}$$

Nå er det på påpiret mulig å regne ut løsningen av en tredjegradslikning for hånd, men det er enklere å kjøre en numerisk likningsløser.

$\boxed{2}$ Les eksempel 4.5 i Sveins notat om vandige likevekter, og betrakt likningen

$$x = [\mathrm{HA}]_T \frac{K_a}{x + K_a} + \frac{K_w}{x}$$

der $K_w = 10^{-14}$ er dissosiasjonskonstanten til vann. Sett $[\mathrm{HA}]_T = 10^{-2}$, velg din enprotiske favorittsyre, finn K_a i tabellboken, og finn x ved Newtons metode eller fikspunktiterasjon.

UKENS REGTEK

Følgende passer ikke helt inn i ukens pensum, men man får bruk for det i kyb intro allerede denne uken. Det kommer som pensum for alle senere i kurset. Richard Feynman kalte det "vår juvel".

Eulers formel

$$e^{ix} = \cos x + i \sin x$$

$\boxed{1}$ Du kan utlede denne formelen ved å anta at derivasjonsreglene du lærte på skolen holder også når det står en i her og der, og derivere brøken

$$\frac{\cos x + i \sin x}{e^{ix}}.$$

Prøv. Om du ikke får det til, kan du scrolle litt nedover her:
`https://en.wikipedia.org/wiki/Euler%27s_formula`

Med Eulers formel kan vi skrive komplekse tall veldig kompakt:

Polar form

$$z = r(\cos\theta + i \sin\theta) = re^{i\theta}$$

Denne formelen er nyttig å kjenne til når man skal analysere transientresponser for andre ordens lineære differensiallikninger. Dette dukker opp i Gravdals kompendium, kap. 2.2.6. (På skolen het det homogene andre ordens differensiallikninger.)

$\boxed{2}$ Skriv $z = 1 + i$ og $w = 1 + \sqrt{3}i$ på polar form

$\boxed{3}$ Polar form er praktisk når man skal gange og dele komplekse tall for hånd. La

$$z = 1 + i = \sqrt{2}e^{i\frac{\pi}{4}}$$

og

$$w = 1 + \sqrt{3}i = 2e^{i\frac{\pi}{3}}.$$

Anta vanlige regneregler for eksponensialfunksjonen, og beregn

$$z \cdot w \quad \text{og} \quad \frac{z}{w}$$

Nå er det imidlertid sjelden at noen ganger sammen komplekse tall for hånd etter at de har landet sin første jobb. Men de geometriske tolkningene som Eulers formel gir oss, og sammenhengen med de trigonometriske funksjonene, er viktig, spesielt om du skal drive på med signalbehandling eller liknende.

$\boxed{4}$ Finn $e^{\pi i/2}$, $e^{\pi i}$, $e^{3\pi i/2}$ og $e^{2\pi i}$.

Her er noen flere nyttige ting:

$\boxed{5}$ Vis at

$$e^{-ix} = \cos x - i \sin x.$$

og

$$\cos x = \frac{e^{ix} + e^{-ix}}{2}$$

og

$$\sin x = \frac{e^{ix} - e^{-ix}}{2i}.$$

$\boxed{6}$ Kan du si noe om $\overline{z}$?

Husk ellers at

$$\cosh x = \frac{e^{x} + e^{-x}}{2}$$

og

$$\sinh x = \frac{e^{x} - e^{-x}}{2}.$$

$\boxed{7}$ Vis at

$$\cosh(ix) = \cos x$$

og

$$\sinh(ix) = i \sin x.$$

og bruk det til å vise at

$$\cos(a + bi) = \cos a \cosh b - i \sin a \sinh b$$
$$\sin(a + bi) = \sin a \cosh b + i \cos a \sinh b$$

$\boxed{8}$ Bruk alt dette til å vise at dersom $b^2 - 4c < 0$, løses

$$\ddot{x} + b\dot{x} + cx = 0$$

av

$$x(t) = Ae^{-\frac{b}{2}t} \cos\left(\frac{\sqrt{4c - b^2}}{2} t\right) + Be^{-\frac{b}{2}t} \sin\left(\frac{\sqrt{4c - b^2}}{2} t\right).$$

(Denne oppgaven er ikke helt triviell.)

UKENS KVANTEMEKANIKK

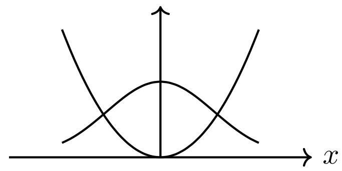
Kvantemekanikk er relevant for MTKJ og MTELSYS, men kanskje ikke for MTTK i skrivende stund.

$\boxed{1}$ Når man løser partikkel i boks med ikke helt tette vegger:
`https://en.wikipedia.org/wiki/Finite_potential_well`
støter man gjerne på likningen

$$\tan \sqrt{x} = \frac{2\sqrt{x(1 - x)}}{2x - 1}.$$

Finn x.

$\boxed{2}$ Når man jobber med den kvantemekaniske harmoniske oscillatoren, støter man visst på en situasjon der man må finne skjæringspunktet mellom kurvene gitt ved $y = \exp(-ax^2)$ og $y = bx^2$, der $a > 0$ og $b > 0$ er konstanter. Lag en funksjon som tar inn a og b, plotter disse to funksjonene i samme plot, og finner skjæringspunktene med en numerisk likningsløser. Hvorfor er det kun nødvendig å beregne det ene skjæringspunktet?



UKENS NØTTER

N1 Utled likningen

$$2\sin 2\theta + \pi - 4\theta = 0$$

i kloakkrøroppgaven over.

N2 Følgende er en modell for matauk, og kalles marginalverditeoremet (marginal value theorem): `https://en.wikipedia.org/wiki/Marginal_value_theorem`

En stær mater ungene sine ved å plukke mark fra et sted i nærheten. La $x = x_0$ være tidspunktet den forlater redet, og $x = 0$ tiden den ankommer stedet der den plukker mark. Funksjonen f beskriver hvor mye mat den får med seg. Denne kurven flater ut når tiden øker, for etterhvert som stæren fyller nebbet med mark, blir det vanskeligere og vanskeligere å fange nye mark.

Stæren ønsker å maksimere forholdet mellom total tid borte fra redet og mengde hjembrakt mat. Tidspunktet $x = x_1$ når den setter nebbet hjemover, er tidspunktet når dette forholdet er maksimert. Vis at tangenten til f i x_1 skjærer x-aksen i x_0.