
Matematikk 1000

Øvingsoppgaver i numerikk – leksjon 9

Løsningsforslag

Oppgave 1 – Trapez-integrasjon

- a) Siden $\mathcal{P}$ skal være regulær, må vi ha at $\mathcal{P} = \{x_0, x_1, x_2\} = \{a, a + \Delta x, b\}$ med $\Delta x = (b - a)/2$. Trapez-metoden gir følgende estimat for integralet:

$$T_3 = \Delta x/2 f(x_0) + \Delta x f(x_1) + \Delta x/2 f(x_2) \quad .$$

Venstre og høgre Riemann-sum blir henholdsvis

$$\begin{aligned}\mathcal{R}_{\mathcal{P}, \mathcal{S}_v} &= f(x_0)\Delta x + f(x_1)\Delta x \quad \text{og} \\ \mathcal{R}_{\mathcal{P}, \mathcal{S}_h} &= f(x_1)\Delta x + f(x_2)\Delta x\end{aligned}$$

slik at gjennomsnittet av dem blir

$$\begin{aligned}(\mathcal{R}_{\mathcal{P}, \mathcal{S}_v} + \mathcal{R}_{\mathcal{P}, \mathcal{S}_h})/2 &= f(x_0)\Delta x/2 + f(x_1)\Delta x/2 + f(x_1)\Delta x/2 + f(x_2)\Delta x/2 = \\ &\Delta x/2 f(x_0) + \Delta x f(x_1) + \Delta x/2 f(x_2) = T_3 \quad .\end{aligned}$$

Som sagt kan denne sammenhengen generaliseres til et vilkårlig antall delintervaller n :

$$\begin{aligned}\mathcal{R}_{\mathcal{P}, \mathcal{S}_v} &= f(x_0)\Delta x + f(x_1)\Delta x + \dots + f(x_{n-1})\Delta x \quad \text{og} \\ \mathcal{R}_{\mathcal{P}, \mathcal{S}_h} &= f(x_1)\Delta x + f(x_2)\Delta x + \dots + f(x_n)\Delta x \quad .\end{aligned}$$

Gjennomsnittet blir

$$\begin{aligned}f(x_0)\Delta x/2 &+ f(x_1)\Delta x/2 + \dots + f(x_{n-1})\Delta x/2 + \\ &f(x_1)\Delta x/2 + \dots + f(x_{n-1})\Delta x/2 + f(x_n)\Delta x/2 = \\ &f(x_0)\Delta x/2 + 2 \cdot f(x_1)\Delta x/2 + \dots + 2 \cdot f(x_{n-1})\Delta x/2 + f(x_n)\Delta x/2 = \\ &\Delta x/2 f(x_0) + \Delta x f(x_1) + \dots + \Delta x f(x_{n-1}) + \Delta x/2 f(x_n) = T_n\end{aligned}$$

- b) Algoritmen kan implementeres på denne måten:

```

1  % Skript som estimerer et integral ved hjelp av
2  % trapes-metoden på en regulær partisjon. Funksjonen
3  % som skal integreres er gitt i IntFunk.m
4
5  % Input-parametre - a, b og n - antall delintervaller
6  a=0;
7  b=1;
8  n=input('Antall delintervall: ');
9
10 DeltaX=(b-a)/n;    % Bestemmer Delta x
11
12 T=0;                % Setter summen til 0
13
14 % Endepunktene:
15 T=T+DeltaX/2*IntFunk(a);
16 T=T+DeltaX/2*IntFunk(b);
17
18 % Summerer over de andre punktene
19 for i=1:(n-1)
20     x=a+i*DeltaX;
21     T=T+DeltaX*IntFunk(x);
22 end
23
24 % Skriver T til skjerm:
25 T

```

Vi bruker her funksjonfila `IntFunk.m` for hver deloppgave. For deloppgave a) kan den se slik ut:

```

function F=IntFunk(x)

% Funksjonsfil som blir brukt i forbindelse med
% numerisk integrasjon.

F=x^2;

```

Skriptet heter “`TrapezRegulaer.m`”. Vi bruker dette til å løse oppgave 1 i avsnitt 5.4 i læreboka:

a) $\int_0^1 x^2 dx, \quad n = 5.$

```

> TrapezRegulaer
Antall delintervall: 5
T = 0.34000

```

Svaret, 0.34, er ganske nær det eksakte svaret, som vi kan finne ved anti-derivasjon: $\int_0^1 x^2 dx = [1/3 x^3]_0^1 = 1/3.$

b) $\int_0^4 \sqrt{x} dx, \quad n = 8.$

Siste linje i `IntFunk.m` blir nå `'F=sqrt(x);'`. I tillegg justerer vi integrasjonsgrensene, linje 6 og 7, og får:

```
> TrapezRegulaer
Antall delintervall: 8
T = 5.2650
```

Det eksakte svaret er

$$\int_0^4 x^{1/2} dx = \left[\frac{1}{1/2+1} x^{1/2+1} \right]_0^4 = \frac{2}{3} \cdot 4^{3/2} = \frac{16}{3} \approx 5.33333 \quad .$$

c) $\int_0^\pi \cos x dx, \quad n = 6.$

Etter å ha justert funksjonsuttrykket og integrasjonsgrensene får vi:

```
> TrapezRegulaer
Antall delintervall: 6
T = 3.8858e-016
> eps
ans = 2.2204e-016
```

Vi får altså svaret $3.9 \cdot 10^{-16}$, som er omtrent maskin-nøyaktigheten (`eps`). Om vi hadde regna ut T_6 analytisk, ville vi ha funne at svaret, ikke overraskende, blir eksakt 0. Dette er også den eksakte verdien av integralet: $\int_0^\pi \cos x dx = [\sin x]_0^\pi = 0$.

- c) a) Én prosent av svaret er $1/3 \cdot 1/100 = 1/300 \approx 0.003333$. Vi prøver oss fram med høyere og høyere n -verdier (etter å ha forandra funksjonfila og skriptet tilbake til det opprinnelige).

```
> TrapezRegulaer
Antall delintervall: 3
T = 0.35185
> T-1/3
ans = 0.018519
> TrapezRegulaer
Antall delintervall: 5
T = 0.34000
> T-1/3
ans = 0.0066667
> TrapezRegulaer
Antall delintervall: 7
T = 0.33673
> T-1/3
ans = 0.0034014
> TrapezRegulaer
Antall delintervall: 8
T = 0.33594
```

```
> T-1/3
ans = 0.0026042
```

Vi finner at feilen er mindre enn 1% med $n = 8$ (og formodentlig også med $n > 8$).

- b) Her er én prosent av svaret $16/3 \cdot 1/100 = 4/75 \approx 0.05333$. Vi oppdaterer .m-filene, prøver oss fram, og finner at feilen blir mindre enn dette når $n \geq 10$:

```
> TrapezRegulaer
Antall delintervall: 9
T = 5.2759
> T-16/3
ans = -0.057481
> TrapezRegulaer
Antall delintervall: 10
T = 5.2841
> T-16/3
ans = -0.049259
```

Oppgave 2 – Uten “fasit”

- a) Som i den forrige oppgava, justerer vi funksjonsuttrykket i `IntFunk.m`, `'F=sin(x.^2);'`, og integrasjonsgrensene i linje 5 og 6 i `TrapezRegulaer.m`:

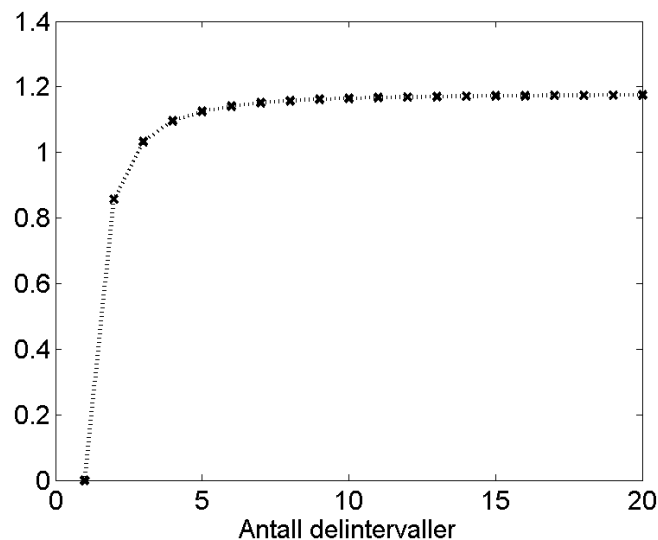
```
a=0;
b=2;
```

Vi velger, for eksempel, $n = 10$:

```
Antall delintervall: 10
T = 0.79592
```

Altså, med 10 delintervaller estimerer trapesmetoden integralet til å bli 0.79592. Dette er ikke så veldig nyttig å vite så lenge vi ikke vet noe om nøyaktigheten i dette estimatet. Men vi kan si noe om nøyaktigheten dersom vi varierer n – eller Δx . Vi regner ut T med stadig høyere n :

```
> TrapezRegulaer
Antall delintervall: 20
T = 0.80259
> TrapezRegulaer
Antall delintervall: 50
T = 0.80443
> TrapezRegulaer
Antall delintervall: 100
```



Figur 1: Denne figuren viser hvordan estimatet av integralet i oppgave 1 konvergerer når vi øker antall del-intervall. Vi har trukket stiplede linjer mellom punktene for å få et tydeligere bilde av konvergensen.

```
T = 0.80469
> TrapesRegulaer
Antall delintervall: 200
T = 0.80475
```

Vi ser at de tre første desimalene ser ut til å ligge fast; vi setter at $\int_0^2 \sin x^2 dx \approx 0.805$. Konvergensen, måten T nærmer seg et bestemt tall, er illustrert i figur 1. For å lage denne figuren, har vi brukt dette skriptet:

```
% Skript som lager et plott som viser konvergensen a
% trapes-metoden

% Vektor med n-verdier
Nvektor=1:20;
% Indeks for elementene i vektor med integral-estimer
indeks=1;
for n=Nvektor
    Tvektor(indeks)=TrapezFunksjon(n);
    indeks=indeks+1;           % indeks økes med 1
end
% Plotter:
plot(Nvektor,Tvektor,'kx:', 'linewidth',2)
set(gca,'fontsize',15)
xlabel('Antall delintervaller')
```

Skriptet refererer til funksjonsfila `TrapezFunksjon.m`, som er en funksjonsversjon av skriptet `TrapezRegulaer.m`. For kompletthetsskyld kan vi godt

ta med denne også:

```
function T=TrapesFunk(n)

% Funksjon som estimerer et integral ved hjelp av
% trapes-metoden på en regulær partisjon. Funksjonen
% som skal integreres er gitt i IntFunk.m

% Input-parametre - a, b og n - antall delintervaller
a=0;
b=1;

DeltaX=(b-a)/n;    % Bestemmer Delta x

T=0;                % Setter summen til 0

% Endepunktene:
T=T+DeltaX/2*IntFunk(a);
T=T+DeltaX/2*IntFunk(b);

% Summerer over de andre punktene
for i=1:(n-1)
    x=a+i*DeltaX;
    T=T+DeltaX*IntFunk(x);
end
```

- b) Grunnen til at vi ikke har blitt bedt om å anti-derivere her, er at det ikke er mulig; selv om den anti-deriverte til funksjonen $\sin x^2$ eksisterer, kan vi ikke skrive den som en elementær funksjon.
- c) Disse oppgavene er nå nesten ferdig løst. Det eneste vi behøver å gjøre, er å justere grensene i trapesmetode-fila, oppdatere funksjonsfila, `IntFunk.m`, og “konvergere” estimatet ved å øke n .
- i) Det at variabelen her heter t og ikke x , spiller ingen rolle. Vi endrer `a` og `b` i `TrapesRegulaer.m` til -5 og 2, endrer den siste linja i `IntFunk.m` til

```
F=x*exp(x^3)
```

og får følgende i kommandovinduet:

```
> TrapesRegulaer
Antall delintervall: 10
T = 2094.9
> TrapesRegulaer
Antall delintervall: 50
T = 637.10
> TrapesRegulaer
```

```

Antall delintervall: 100
T = 552.23
> TrapesRegulaer
Antall delintervall: 200
T = 529.84
> TrapesRegulaer
Antall delintervall: 500
T = 523.48
> TrapesRegulaer
Antall delintervall: 1000
T = 522.57
> TrapesRegulaer
Antall delintervall: 2000
T = 522.34

```

Det ser ut til at 522 er en ganske vel-konvergent verdi for integralet. Men estimatene varierer ganske kraftig – selv for relativt høye verdier av n . For dette integralet trenger vi altså langt høyere n for å oppnå konvergens enn vi gjorde i den forrige oppgava.

- ii) Som før justerer vi integransjonsgrensene og integranden. I kommandovinduet får vi:

```

> TrapesRegulaer
Antall delintervall: 10
T = 0.97305
> TrapesRegulaer
Antall delintervall: 20
T = 0.97148
> TrapesRegulaer
Antall delintervall: 40
T = 0.97109

```

Altså ser 0.971 ut til å være et rimelig nøyaktig svar.

- iii) Her ser vi ut til å få 0 til svar uansett hvilken n vi velger:

```

> TrapesRegulaer
Antall delintervall: 5
T = 1.1102e-016
> TrapesRegulaer
Antall delintervall: 10
T = 3.3307e-016
> TrapesRegulaer
Antall delintervall: 20
T = 4.1633e-017
> TrapesRegulaer
Antall delintervall: 50
T = 6.9389e-018

```

(Husk at maskina ikke er i stand til å skille mellom 0 og tall som er mindre enn 10^{-16} i absoluttverdi.)

- d) Siden intervallet vi skal integrere over, er symmetrisk omkring 0, og $\sin x^3$ er en odde funksjon, $\sin(-x)^3 = -\sin x^3$, kan vi med sikkerhet si at integralet er 0 (se s. 231 i læreboka).
- e) Vi forandrer den øvre integrasjonsgrensa, b , fra $1/2$ til 1 og prøver å få konvergente svar:

```
> TrapezRegulaer
Antall delintervall: 5
T = -1.2136
> TrapezRegulaer
Antall delintervall: 10
T = 0.87002
> TrapezRegulaer
Antall delintervall: 20
T = 6.9894
> TrapezRegulaer
Antall delintervall: 50
T = -1.2181
> TrapezRegulaer
Antall delintervall: 100
T = 0.88779
> TrapezRegulaer
Antall delintervall: 500
T = -0.92567
> TrapezRegulaer
Antall delintervall: 1000
T = 1.1546
```

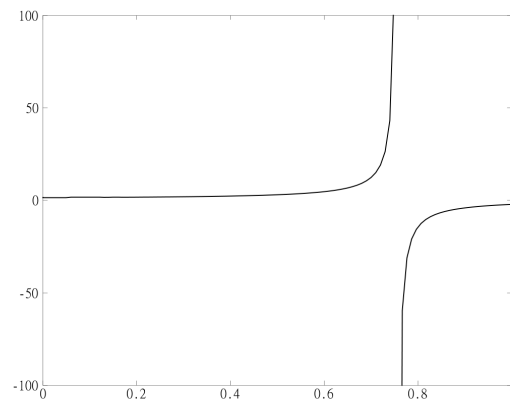
Her har vi tydeligvis alvorlige problem med å få konvergent svaret. Hva kan grunnen være til dette? Vel, vi vet jo at tangens-funksjonen, $\tan x$, har loddrette asymptoter for $x = \pi/2 + n\pi$, $n \in \mathbb{Z}$. Vår integrand går mot pluss eller minus uendelig når x går mot $\sqrt{\pi/2 - 1} \approx 0.76$. Se figur 2. Følgelig er ikke integralet veldefinert, og vi kan ikke regne med å få noen konvergens her¹.

Oppgave 3 – Taket på et tårn

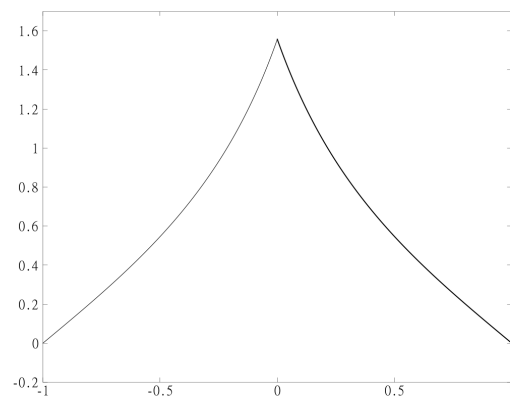
$$f(x) = \tan(1 - x) \quad , \quad D_f = [0, 1] \quad .$$

- a) Vi får en viss idé om høyden om vi plotter grafen til funksjonen. Et slikt plott er vist i figur 3. Det kan se ut til at $f(x)$ er maksimal for $x = 0$.

¹Desverre er det ikke dette en “idiotsikker” metode; man *kan* lage implementeringer som ser ut til å konvergere for integraler som ikke er veldefinerte.



Figur 2: Plott av funksjonen $\tan(x^2 + 1)$.



Figur 3: Plott av grafen til $f(x) = \tan(x-1)$ for $x \in [0, 2]$ – sammen med grafen til $f(-x)$.

Dette får vi stadfesta om vi ser på $f'(x)$:

$$f'(x) = \frac{1}{\cos^2(1-x)} \cdot (-1) < 0.$$

Siden $f'(x)$ alltid er negativ når $x \in D_f$, og f er kontinuerlig, vil f være maksimal for den laveste x -verdien – altså $x = 0$. $f(0) = \tan 1 \approx 1.56$. Høyden er altså ca. 1.56 m.

b) Lengda skal, i følge den første formelen, være

$$L = \int_0^1 \sqrt{1 + (f'(x))^2} dx = \int_0^1 \sqrt{1 + \frac{1}{\cos^4(1-x)}} dx \quad .$$

Vi oppdaterer integrasjonsgrensene i `TrapezRegulaer.m` og endrer den siste linja i `IntFunk.m` til

```
F=sqrt(1+1/(cos(1-x))^4);
```

I MATLAB får vi:

```
> TrapezRegulaer
Antall delintervall: 5
T = 1.9094
> TrapezRegulaer
Antall delintervall: 10
T = 1.8846
> TrapezRegulaer
Antall delintervall: 20
T = 1.8782
> TrapezRegulaer
Antall delintervall: 50
T = 1.8764
```

Vi kan nok relativt trygt konkludere at lengda er 1.88 m.

c) Vi bestemmer arealet ved å regne ut integralet

$$2\pi \int_0^1 x f(x) dx = \int_0^1 2\pi x \tan(1-x) dx \quad .$$

Vi gjør de sedvanlige justeringene og får²:

```
> TrapezRegulaer
Antall delintervall: 5
T = 1.1252
> TrapezRegulaer
Antall delintervall: 10
T = 1.1650
> TrapezRegulaer
Antall delintervall: 20
T = 1.1750
> TrapezRegulaer
Antall delintervall: 50
T = 1.1778
```

Volumet blir 1.178 m³.

²Her har vi valgt å legge faktoren 2π inn i integranden.

Ekstra: Oppgave 4 – Monte Carlo-integrering

- a) Siden begge uttrykkene skal gi $\bar{f}$, kan vi helt enkelt sette dem (tilnærma) lik hverandre:

$$\begin{aligned}\frac{1}{b-a} \int_a^b f(x) dx &\approx \frac{1}{n} \sum_{i=1}^n f(x_i) \Leftrightarrow \\ \int_a^b f(x) dx &\approx \frac{b-a}{n} \sum_{i=1}^n f(x_i) \quad (q.e.d.)\end{aligned}$$

- b) Hvis '`>rand`' gir et tilfeldig tall mellom 0 og 1, må '`a+(b-a)*rand`' gi et tilfeldig tall mellom `a` og `b`. Vi tester:

```
> a=-1;
> b=1;
> a+(b-a)*rand
ans = -0.51153
> a+(b-a)*rand
ans = -0.040910
> a+(b-a)*rand
ans = -0.75354
> a+(b-a)*rand
ans = -0.73380
> a+(b-a)*rand
ans = 0.50765
> a+(b-a)*rand
ans = 0.92631
> a+(b-a)*rand
ans = -0.60546
```

Dette ser ut til å fungere. (Husk å bruke pil-tastene når du repeterer kommandoer på denne måten.)

- c) Vi kan velge oss integralet

$$\int_{-1}^1 (x^2 + e^x) dx = \frac{2}{3} + e - \frac{1}{e} \quad .$$

Nedenfor er et forslag til hvordan metoden kan implementeres. Uttrekket av x er gjort i linje 14. Vi har valgt å lage en vektor med sekvenser av Monte Carlo-estimerer inni `for`-løkka slik at vi kan bruke dem til å plote konvergensens etterpå. Dette er også gjort i skriptet fra og med linje 19.

```
1 % Skript som regner ut integralet av funksjonen gitt ved
2 % IntFunk.m fra a til b ved Monte Carlo metoden
3
4 % Integrasjonsgrenser
```

```

5  a=-1;
6  b=1;
7  % Maksimalt antall treknings
8  nMax=100;
9
10 % Setter summen til 0:
11 S=0;
12 for i=1:nMax
13     x=a+(b-a)*rand;    % Trekker tilfeldig x i intervallet
14     y=IntFunk(x);      % Bestemmer funksjonsverdi
15     S=S+y;             % Summerer
16     IntEst(i)=(b-a)*S/i; % Bestemmer estimat av integralet
17 end
18
19 % Plotter estimatet som funksjon av antall trekk
20 plot(1:n,IntEst,'bx','linewidth',2)
21 % Plotter også eksakt svar
22 IntEksakt=2/3+exp(1)-exp(-1);
23 hold on
24 plot([1 nMax],[1 1]*IntEksakt,'k-')
25 hold off
26 % Justerer opp skriftstørrelsen på aksene
27 set(gca,'fontsize',20)
28 xlabel('Antall trekk')

```

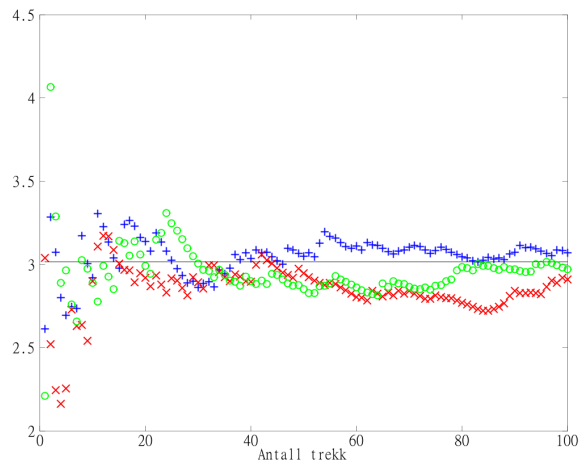
Vi har kalt skriptet 'MonteCarlo.m'. Som vi ser, plotter det integral-estimatet som funksjon av antall trekk. Når vi kjører det i MATLAB, ser det slik ut:

```

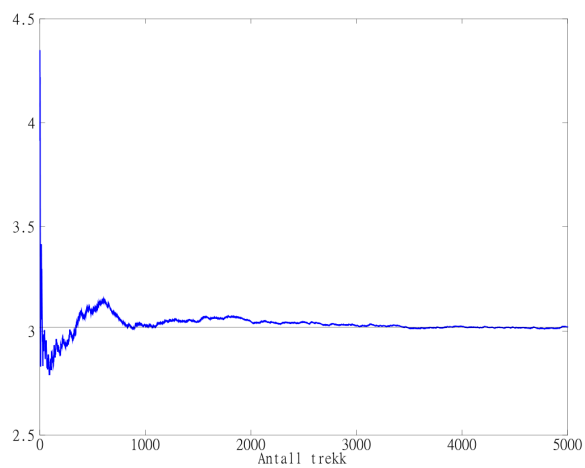
> MonteCarlo
Nedre integrasjonsgrense: -1
Ovre integrasjonsgrense: 1
Maksimalt antall trekk: 100

```

Selvsagt vil vi ikke få de samme sekvensene av x -verdier hver gang vi kjører skriptet. Derfor vil estimatet gå mot integralet på ulike måter hver gang vi kjører skriptet. I figur 4 har vi vist hvordan tre ulike sekvenser konvergerer. Selv om disse resultatene ser ganske ulike ut for hver gang, ser alle tre ut til å konvergere mot riktig svar – men ganske sakte. At konvergensens går sakte ser vi enda tydeligere i figur 5. Først etter flere tusen trekk begynner kurva å plassere seg oppå verdien av integralet (på denne skalaen). Like fullt er Monte Carlo-integrering en svært nyttig og mye brukt teknikk – ikke minst når man skal beregne fler-dimensjonale integraler. (Det skal vi ikke gjøre i dette kurset.)



Figur 4: Tre sekvenser med Monte Carlo-estimat av integralet fra oppgave 1a) Den svarte linja er den eksakte verdien av integralet.



Figur 5: Konvergensplott for en lengre sekvens av en Monte Carlo-beregning.