# Numerisk derivasjon og integrasjon utledning av feilestimater

Knut Mørken

6. oktober 2007

## 1 Innledning

På forelesningen 2/10 brukte vi litt tid på å repetere inhomogene differensligninger og rakk dermed ikke gjennomgå alt stoffet som var planlagt. Dette notatet oppsummerer hovedtrekkene i forelesningen, også den delen jeg ikke rakk.

## 2 Hvorfor numerisk integrasjon og derivasjon?

Mange funksjoner som kan forekomme i praksis foreligger ikke ved et symbolsk uttrykk som kan deriveres og integreres. Funksjonen kan for eksempel være resultat av målinger, eller kan bare foreligge som en funksjon programmert på datamaskin som genererer numeriske funksjonsverdier når vi sender inn et argument. Når det gjelder integrasjon er det dessuten slik at de fleste funksjoner ikke kan integreres symbolsk. Det er derfor behov for metoder som lar oss beregne tilnærminger til deriverte og integraler ved hjelp av funksjonsverdier.

## 3 Trapesmetoden

Mandag 1/10 så vi raskt på noen metoder for numerisk derivasjon og integrasjon. Tirsdag 2/10 så vi på trapesmetoden i litt mer detalj. Utledningen av trapesmetoden består av tre trinn.

**1. Grunnleggende ide.** Tilnærm en funksjon $f$ på et intervall $[x_{i-1}, x_i]$ med den rette linja som går gjennom punktene $(x_{i-1}, f(x_{i-1}))$ og $(x_i, f(x_i))$, altså sekanten $p_i$ til $f$ gjennom disse to punktene. Vi tilnærmer så integralet av $f$ med integralet av $p_i$,

$$\int_{x_{i-1}}^{x_i} f(x)\,dx \approx \int_{x_{i-1}}^{x_i} p_i(x)\,dx.$$

Tegner du en figur vil du se at området vi skal finne arealet av når vi bruker $p_i$ er et trapes. Vi har derfor

$$\int_{x_{i-1}}^{x_i} p_i(x)\,dx = \frac{f(x_{i-1}) + f(x_i)}{2}(x_i - x_{i-1}).$$

**2. Summer opp bidragene fra delintervallene.** Når vi skal integrere over et intervall $[a, b]$ deler vi dette i mindre biter,

$$a = x_0 < x_1 < x_2 < \cdots < x_{n-1} < x_n = b,$$

og summerer opp arealene over hvert delintervall,

$$\begin{aligned}\int_a^b f(x)\,dx &= \sum_{i=1}^{n} \int_{x_{i-1}}^{x_i} f(x)\,dx\\ &\approx \sum_{i=1}^{n} \int_{x_{i-1}}^{x_i} p_i(x)\,dx\\ &= \sum_{i=1}^{n} \frac{f(x_{i-1}) + f(x_i)}{2}(x_i - x_{i-1}).\end{aligned}$$

**3. Forbered formelen for programmering.** Formelen over kan i og for seg godt programmeres direkte, men det er som regel lurt å se om det er mulighet for effektivisering. For det første er det vanlig å anta at punktene $\{x_i\}_{i=0}^n$ er jevnt fordelt i intervallet $[a, b]$. Dette vil si at

$$x_i = a + ih, \quad i = 0, 1, \ldots, n, \text{ der } h = (b-a)/n = x_i - x_{i-1}.$$

Dermed har vi

$$\int_a^b f(x)\,dx \approx \frac{h}{2} \sum_{i=1}^{n} \big(f(x_{i-1}) + f(x_i)\big).$$

Vi ser at dette sparer oss for en multiplikasjon hver gang vi løper gjennom summasjonsløkken. Skriver vi ut summen som gjenstår ser vi at den svarer til

$$\begin{aligned}\sum_{i=1}^{n}\big(f(x_{i-1})+f(x_i)\big) &= \overbrace{f(x_0)+f(x_1)}^{i=1}+\overbrace{f(x_1)+f(x_2)}^{i=2}+\cdots \\ &\qquad +\overbrace{f(x_{n-2})+f(x_{n-1})}^{i=n-1}+\overbrace{f(x_{n-1})+f(x_n)}^{i=n} \\ &= f(a)+f(b)+2\sum_{i=1}^{n-1}f(x_i).\end{aligned}$$

Dermed har vi

$$\int_a^b f(x)\,dx \approx \frac{h}{2}\Big(f(a)+f(b)+2\sum_{i=1}^{n-1}f(x_i)\Big)$$

som nesten halverer antall beregninger av funksjonsverdier. Når $f$ er en enkel funksjon betyr ikke dette så mye, men hvis det er krevende å finne $f(x)$ kan regnetiden med et slikt enkelt grep kanskje reduseres fra to dager til en dag! Uansett er det en god regel å ikke gjøre unødige beregninger. Du kan lese mer om trapesmetoden og hvordan den kan programmeres på sidene 104–107 i det norske kompendiet og i seksjon 8.7 i Kalkulus.

# 4 Simpsons formel

Simpsons formel er analog med trapesregelen, men er basert på å tilnærme $f$ med parabler i stedenfor rette linjer. En parabel er bestemt av verdien til $f$ i tre punkter så vi må da dele opp $[a, b]$ i et like antall delintervaller (odde antall punkter) for å få det hele til å gå opp. Som for trapesregelen er det viktig å finpusse formelen slik at den kan programmeres effektivt, se seksjon 7.2.2 i det norske kompendiet.

# 5 Valg av $h$ ved numerisk integrasjon

Når numerisk integrasjon brukes i praksis er det viktig å vite hvor stor $h$ må være. To strategier er da vanlig å bruke.

## 5.1 Numerisk bestemmelse av $h$

Anta at vi skal integrere $f$ numerisk på intervallet $[a, b]$ ved hjelp av trapesmetoden. Vi kan da først beregne en tilnærming $T_0$ til integralet der vi bare tilnærmer $f$ med en rett linje. Vi kan så halvere $h$ og beregne en tilnærming $T_1$ med to linjesegmenter. Vi halverer igjen og beregner en tilnærming $T_2$ med fire linjesegmenter også videre. Vi generer altså en følge

$$T_0, T_1, T_2, \ldots, T_j, \ldots$$

av tilnærminget til integralet. Et vanlig stoppkriterium er å fortsette dette inntil forskjellen på to påfølgende tilnærminger er mindre enn en passende toleranse, altså $|T_j - T_{j+1}| < \epsilon$, helt analogt til det som ofte gjøres for for eksempel Newtons metode. Dette er ingen garanti for konvergens, men i mangel av noe bedre er det vanlig. Som alltid bør vi passe oss for absolutt feil og heller bruke

$$\frac{|T_j - T_{j+1}|}{|T_{j+1}|}.$$

Et helt tilsvarende kriterium kan brukes med Simpsons metode. Som nevnt over kan du lese mer om hvordan trapesmetoden kan programmeres på sidene 104–107 i det norske kompendiet.

### 5.2 Bestemmelse av $h$ ut fra feilestimat

Det kan vises at om vi deler $[a, b]$ i $n$ delintervaller så er feilen i trapesmetoden begrenset ved

$$\left| \int_a^b f(x)\,dx - \frac{h}{2}\Big(f(a) + f(b) + \sum_{i=1}^{n} f(x_i)\Big) \right| \leq \frac{(b-a)^3}{12n^2} M$$

der $M = \max_{x\in[a,b]} \big|f''(x)\big|$. Hvis vi ønsker å bestemme integralet slik at den asbolutte feilen er mindre enn $\epsilon$ kan vi ut fra dette finne $n$. Den absolutte feilen vil bli mindre enn $\epsilon$ hvis

$$\frac{(b-a)^3}{12n^2} M < \epsilon.$$

Løser vi denne ulikheten med hensyn på $n$ får vi

$$n > \sqrt{\frac{(b-a)^3 M}{12\epsilon}}.$$

Velger vi $n$ som det minste heltallet som er større enn uttrykket på høyre side er vi altså garantert at feilen vi gjør ved å bruke trapesmetoden er mindre enn $\epsilon$. Leggg merke til at vi da må kunne finne $M$ som er maksimumsverdien til $|f''|$ på intervallet $[a, b]$. Dette vil som regel være vanskelig, men legg merke til at vi kan erstatte $M$ med en øvre grense. Dette vil øke $n$, men garantere at feilen blir mindre enn $\epsilon$. Hvis for eksempel $f(x) = \sin x$ er $\bigl|f''(x)\bigr| = |\sin x|$ slik at $M \leq 1$, uansett hva $[a, b]$ er.

Legg merke til likheten mellom det å estimere $n$ her og det å estimere antall ledd vi må ta med i Taylorpolynomet for å få feilen i en eller annen tilnærming liten nok. Dette kan du lese mer om i seksjonene 8.7 og 11.2 i Kalkulus.

## 6 Numerisk derivasjon

Den enkleste formelen for numerisk derivasjon er basert direkte på definisjonen av den deriverte,

$$f'(a) \approx \frac{f(a+h) - f(a)}{h} \tag{1}$$

der $h$ er 'lite' tall. I seksjon 9.6 i det norske kompendiet kan du finne andre tilnærminger til den deriverte, for eksempel

$$f'(a) \approx \frac{f(a+h) - f(a-h)}{2h}.$$

En annen derivasjonsformel som framkommer ved å interpolere $f$ i punktene $a - 2h$, $a - h$, $a$, $a + h$ og $a + 2h$ med et polynom $p_4$ av grad 4 og bruke tilnærmingen $f'(a) \approx p_4'(a)$ er

$$f'(a) \approx \frac{f(a-2h) - 8f(a-h) + 8f(a+h) - f(a+2h)}{12h}.$$

Når vi tilnærmer den deriverte som i (1) gjør vi åpenbart en feil, så sant $f$ ikke er en rett linje. På den annen side vil vi forvente at denne feilen blir mindre når vi lar $h$ bli mindre. Dette er i og for seg riktig, men hvis vi programmerer formelen over og prøver på en konkret funksjon vil vi se at feilen først avtar med $h$, for så å øke igjen når $h$ blir liten. Dette kommer av at en annen feilkilde enn den rent matematiske kommer inn i bildet, nemlig avrundingsfeil. Når $h$ blir liten vil $f(a+h)$ og $(a)$ bli omtrent like store og vi har allerede sett at subtraksjon av to nesten like store tall vil kunne gi stor avrundingsfeil. Ja, velger vi $h$ liten nok vil $f(a)$ og $f(a + h)$ bli avrundet til det samme flyttallet slik at tilnærmingen til den deriverte gitt ved (1) blir 0. Dette kan du lese mer om i seksjon 6.2 i det norske kompendiet.

Det samme skjer om vi bruker de andre tilnærmingene over, men forskjellen er at feilen blir mindre før den begynner å øke igjen. Du kan teste dette selv ved hjelp av programmet `numder.py` som du finner på kurshjemmesida.

# 7 Generelt om numerisk derivasjon og integrasjon

Det fins mange alternative metoder for numerisk derivasjon og integrasjon. En standard måte å finne fram til slike metoder er ved å tilnærme funksjonen som skal deriveres eller integreres med et interpolasjonspolynom (se seksjon 9.2.1 i det norske kompendiet) og så derivere eller integrere dette polynomet i steden for den opprinnelige funksjonen, se innledningen til seksjon 9.6 på sidene 169–170 i det norske kompendiet. Eksempler på dette er vist i seksjonene 9.6.1 og 9.6.2 i kompendiet. Ved integrasjon er det så vanlig å dele opp intervallet i mindre deler og bruke en slik tilnærming på hvert delintervall.

Ved hjelp av Taylorpolynomer er det mulig å finne uttrykk for feilen i slike tilnærminger. Flere eksempler på dette er vist i seksjon 9.6.3. Teknikken for å utlede feilestimater kan virke komplisert til å begynne med, men følger en fast oppskrift som er oppsummert på side 178 i seksjon 9.6.4 i det norske kompendiet.