Nullpunkter for funksjoner

Numeriske metoder for $f(x) = 0$

Forelesning uke 39, 2006

MAT-INF1100

Løsning i formel

Eksempler

$ax + b = 0$	ja
$ax^2 + bx + c = 0$	ja
$ax^4 + .. + e = 0$	jo, men..
$a \sin x + b \cos x - c = 0$	ja
$(\sin x - 0.3)(\mathrm{e}^{2x} - 7) = 0$	ja
$ax^5 + .. + ex + f = 0$	nei
$e^{-x} + \sin x = 0$	nei
....	nei

Bare et fåtall algebraiske likninger kan løses i formel

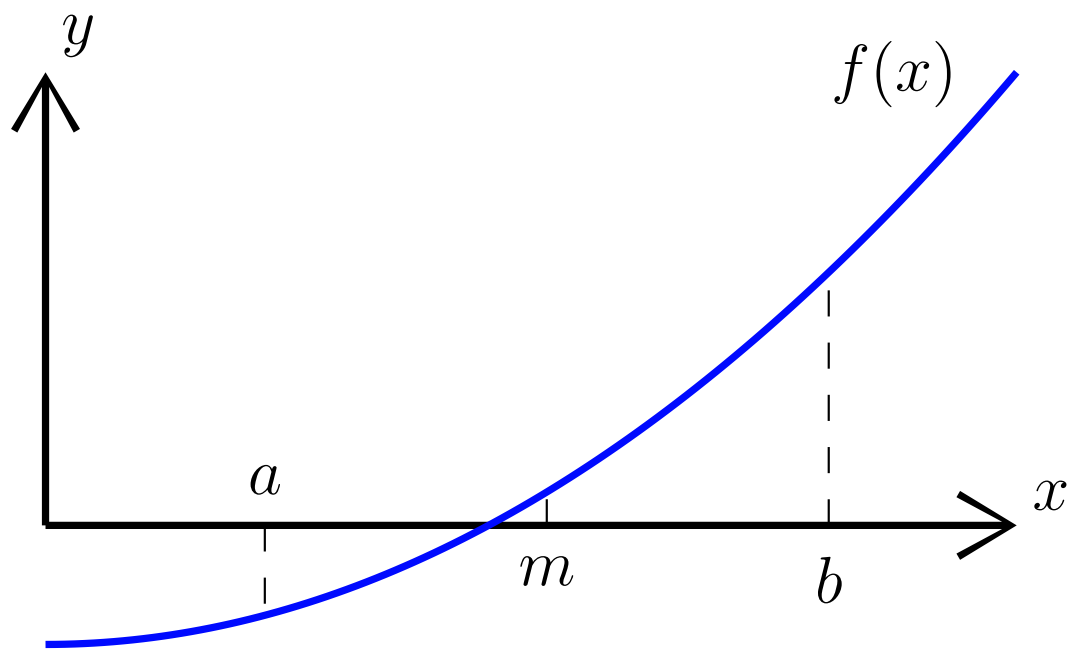
Viktig om $f(x) = 0$

Ofte finnes ingen formel for f selv, men bare en algoritme, slik at f beregnes numerisk. Da må nullpunkter også finnes numerisk.

Numerisk metode

- Metode baseres på utregninger av f, og evt. f'
- Har preg av prøving og feiling satt i system
- Typisk iterasjon (repeterte operasjoner):
  Har tilnærmelse x_n til nullpunkt
  Forbedret verdi x_{n+1} finnes vha. $f(x_n)$ (kalles ofte residuet) og $f'(x_n)$ el. $f(x_{n-1})$ etc.

Halveringsmetode; kompendium 5.3



f kontinuerlig og $f(a) < 0 < f(b) \Rightarrow$
det er et nullpunkt, c, i (a, b) (skjæringssetningen)

Deler intervall i 2 ved $m = \frac{1}{2}(a + b)$.
Beholder det intervall der f bytter tegn; på tegning: (a, m).
Gjentar...
Metode virker like bra når $f(a) > 0 > f(b)$

Halveringsmetode; algoritme

Definerer følger a_n, b_n ved

1. Initialisering: $a_0 = a$, $b_0 = b$
2. Rekursjon: $m = \frac{1}{2}(a_n + b_n)$

$$\text{hvis} \begin{cases} f(m) = 0 & \rightarrow \quad c = m, \text{ stopp} \\ f(m)f(a_n) < 0 & \rightarrow \quad b_{n+1} = m,\ a_{n+1} = a_n \\ f(m)f(b_n) < 0 & \rightarrow \quad b_{n+1} = b_n,\ a_{n+1} = m \end{cases}$$

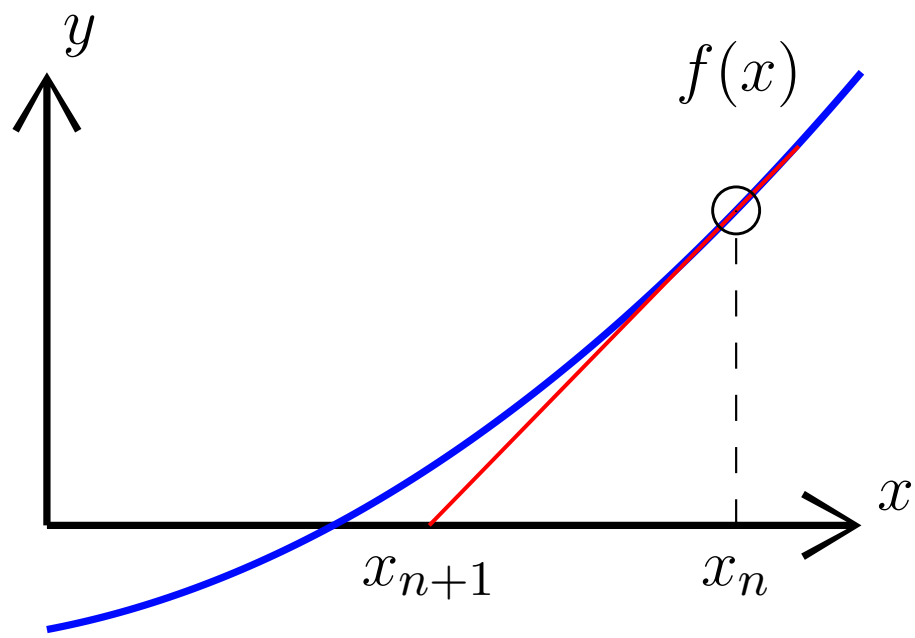
Intervall halveres hver gang: $|c - a_n|,\ |c - b_n| < (b - a)2^{-n}$
Følger konverger mot c

Programeksempel for $f = x^2 - 2$: **halv.java**

Halveringsmetode; egenskaper

- Robust når den kan brukes
- Kjent, men langsom konvergensrate
- Grunn til ineffektivitet: bruker bare fortegn på residu, $f(m)$ ikke størrelse
- Halveringsmetoden brukes i en del andre sammenhenger, bla.a bevisføring.

Newtons metode; Kalkulus 7.3



1. Gjetter på x_0
2. Rekursjon (iterasjon):
   Tilnærmer $f(x_{n+1}) \approx f(x_n) + f'(x_n)(x_{n+1} - x_n) = 0$

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

Newtons metode; Egenskaper

Konvergens; Kalkulus, sats 7.3.3

$f(c) = 0$, $f'(c) \neq 0$ og f'' eks. i omegn om $c \Rightarrow$
Finnes en $\delta > 0$ slik at $x_0 \in (c - \delta, c + \delta)$ medfører konvergens.
Dvs: går bra når en starter nær nok til c.

Konvergensrate

For store n gjelder (vises ikke her)

$$|x_{n+1} - c| \leq B(x_n - c)^2$$

Antall korrekte siffere fordobles hver gang!

Ulempe

Krever kjennskap til f'. Kan være problem når f bare finnes som en numerisk algoritme.

Implementasjon, eksempel

Stoppkriterier

Vi stopper når ett av følgene skjer

1. Toleranse for x: $|x_n - x_{n-1}| < \epsilon_x$
2. Toleranse for residu: $|f(x_n)| < \epsilon_f$
3. Max antall iterasjoner: $n \leq n_{\max}$

ϵ_x og ϵ_f bør ikke være mindre enn feil i utregninger (avrunding mm.)

Eksempel $f(x) = x^2 - 2 = 0$, $c = \sqrt{2}$

Rekursjonen blir $x_{n+1} = x_n/2 + 1/x_n$; kan anlyseres.

Numerisk løsning gir desimalutvikling av $\sqrt{2}$.

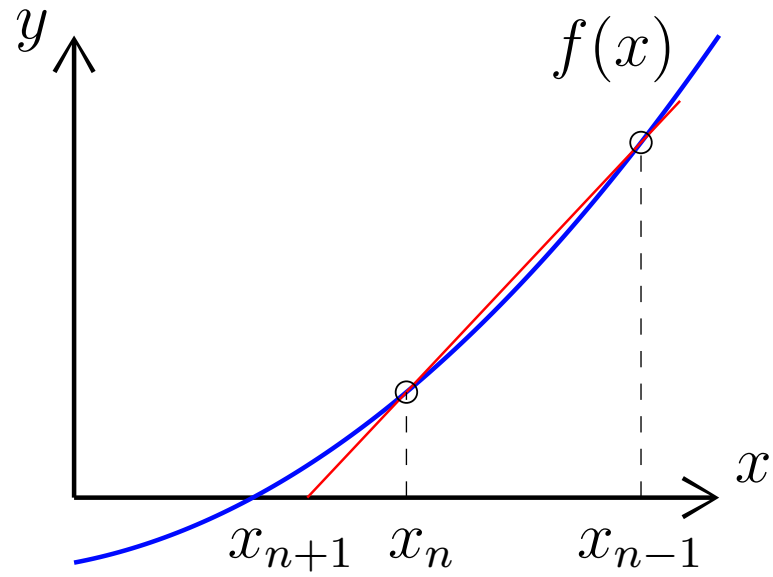
Program **Newton.java**

linux>java Newt
gi xstart
6.0
gi max antall iterasjoner
12
gi avbruddskriterium for x og f
1.0E–8 1.0E–8

n	xn	feil	f(xn)
0	6.00000000	–4.58578644	34.00000000
1	3.16666667	–1.75245310	8.02777778
2	1.89912281	–0.48490924	1.60666744
3	1.47612029	–0.06190673	0.17893113
4	1.41551171	–0.00129815	0.00367340
5	1.41421416	–0.00000060	0.00000168
6	1.41421356	–0.00000000	0.00000000

Hvorfor er f(xn)/feil ca. -2.8 for n=3, 4 og 5 ?

Sekantmetode; ikke i pensum



Newtons metode: $f(x_n) + \beta(x_{n+1} - x_n) = 0$, $\beta = f'(x_n)$
For sekantmetoden brukes sekanten for endringsraten, β

$$\beta = \frac{f(x_n) - f(x_{n-1})}{x_n - x_{n-1}}$$

Differenstilnærmelse til f'. Sekantmetode nesten like god som Newtons metode. Programeksempel **Secant.java**

Viktige temaer vi ikke tar opp

- Hvordan løse sett av likninger i flere ukjente
- Hvordan finne gode startverdier
- Det er mange nullpunkter som vi vil finne
- Knep for å sikre eller forbedre konvergens for vanskelige problemer