

# MAT-INF1100 Sammendrag

Øivind Gylver (oivindgy@fys.uio.no), 21. november 2013  
Med forbehold om feil.

## Differenslikninger

Simulering på datamaskin: Dersom vi skal undersøke oppførselen ved simulering, må vi bl.a. undersøke om init.bet. kan representeres eksakt. Hvis f.eks.  $x_0 = 2/3$ , får vi en liten feil  $\epsilon$  slik at  $\bar{x}_0 = 2/3 + \epsilon$ . Vi må da beregne og undersøke løsningen med  $\bar{x}_0$  som initialbetingelse.

Eksakte tall: Hvis vi har et rasjonalt tall  $a = b/c$ , kan  $a$  representeres eksakt i tallsystemet (datamaskin) dersom  $c$  kan skrives på formen  $2^n$  der  $n = 1, 2, \dots$ .

### Førsteordens (lineære)

$$x_{n+1} = r x_n \Rightarrow x_n = C r^n$$

### Annenordens, homogene med konstante koeffisienter

$$x_{n+2} + b x_{n+1} + c x_n = 0$$

der  $b, c$  er kjente reelle koeffisienter. Karakteristisk polynom

$$r^2 + br + c = 0$$

Én reell rot  $r$ :  $x_n = C r^n + D n r^n$   
 To reelle røtter  $r_1, r_2$ :  $x_n = C r_1^n + D r_2^n$   
 To komplekse røtter  $r, \bar{r}$ :  $x_n = C r^n + \bar{C} \bar{r}^n$   
 evt.  $x_n = E \rho^n \cos(n\theta) + F \rho^n \sin(n\theta)$   
 $\rho$  er modulus,  $\theta$  er argument til  $r$   
 $r = a + ib, \bar{r} = a - ib$

### Annenordens, inhomogene med konstante koeffisienter

$$x_{n+2} + b x_{n+1} + c x_n = f(n)$$

har løsning  $x_n = x_n^p + x_n^h$  der  $x_n^p$  (partikulær) og  $x_n^h$  (homogen). Gjetter  $x_n^p$ ,  
**Regel 1:**  $f(n) = \text{polynom} \Rightarrow x_n^p = \text{polynom}$  (lik grad, evt. gå opp én/to grader).  
**Regel 2:**  $f(n) = a^n P(n)$  der  $P(x)$  er et polynom  $\Rightarrow x_n^p = a^n Q(n)$  der  $Q$  er et polynom av samme grad som  $P$  (evt. gå opp én eller to grader).  
**Regel 3:** 1)  $f(n) = b^n [A \sin(an) + B \cos(an)] \Rightarrow x_n^p = b^n [C \sin(an) + D \cos(an)]$  2)  $x_n^p = n b^n [C \sin(an) + D \cos(an)]$  hvis  $x_n^h = b^n \cos(an)$  eller  $x_n^h = b^n \sin(an)$ .

## Taylorpolynom

Brukes til å tilnærme  $f(x)$  omkring et punkt  $a$  (som vi er interessert i). Dette er nyttig fordi den tilnærmede funksjonen er enklere å regne med.

$$f(x) = \underbrace{T_n(f; a)(x)}_{\text{Taylorpolynom}} + \underbrace{R_n(f; a)(x)}_{\text{restledd}}$$

Taylorpolynom  $T_n$  av grad  $n$  til  $f$  om punktet  $a$ , og restledd  $R_n$ :

$$T_n(f; a)(x) = \sum_{i=0}^n \frac{(x-a)^i}{i!} f^{(i)}(a)$$

$$R_n(f; a)(x) = \frac{(x-a)^{n+1}}{(n+1)!} f^{(n+1)}(a)$$

Feilen vi gjør følger uttrykket:

$$\overbrace{|f(x) - T_n(f; a)(x)|}^{\text{absolutt feil}} = \left| \frac{f^{(n+1)}(c)}{(n+1)!} (x-a)^{n+1} \right| \leq \epsilon \quad c \in (a, x) \quad |c - m_N| \leq \frac{b-a}{2^{N+1}}$$

hvor  $\epsilon$  er ønsket nøyaktighet. Vi trenger da et intervall for  $x$  vi ønsker å beregne feilen i. Tallet  $x$  velges fra intervallet slik at  $|(x-a)^{n+1}|$  maksimeres og  $c$  velges slik at  $|f^{(n+1)}(c)|$  maksimeres.

## Polynom interpolasjon

Det å finne et polynom  $p_n(x)$  av grad  $n$  som går gjennom  $n+1$  gitte punkter kalles polynom interpolasjon. Dette er nyttig fordi vi da kan gjøre beregninger med  $p_n(x)$ .

### Newtons form

$$p_n(x) = c_0 + c_1(x-x_0) + c_2(x-x_0)(x-x_1) + \dots + c_n(x-x_0)(x-x_1)\dots(x-x_{n-1})$$

Konstantene  $c_k$  bestemmes gjennom dividerte differanser.

### Dividerte differanser

$$f[x_0, \dots, x_k] = \frac{f[x_1, \dots, x_k] - f[x_0, \dots, x_{k-1}]}{x_k - x_0}$$

Sett opp tabell (for alle gitte punkter  $(x_k, f[x_k])$ ):

$x_0$	$f[x_0]$	$\leftarrow$	$c_0$
$x_1$	$f[x_1]$	$f[x_0, x_1]$	$\leftarrow$ $c_1$
$x_2$	$f[x_2]$	$f[x_1, x_2]$	$f[x_0, x_1, x_2]$ $\leftarrow$ $\dots$
$x_3$	$f[x_3]$	$f[x_2, x_3]$	$f[x_1, x_2, x_3]$ $f[x_0, x_1, x_2, x_3]$
	$\vdots$		

Eksempel på utregning av to dividerte differanser:

$$f[x_1, x_2, x_3] = \frac{f[x_2, x_3] - f[x_1, x_2]}{x_3 - x_1}$$

$$c_3 = f[x_0, x_1, x_2, x_3] = \frac{f[x_1, x_2, x_3] - f[x_0, x_1, x_2]}{x_3 - x_0}$$

Teller: Vi tar verdien rett til venstre og trekker fra verdien over denne.  
 Nevner:  $x_{\text{størst}} - x_{\text{minst}}$  av alle involverte  $x_k$  i teller.

## Nullpunkter til funksjoner

### Bisection metoden

Fortegnet til  $f(x)$  skrives  $\text{sgn}[f(x)]$ . Dersom  $I \subseteq [a, b]$  og  $\text{sgn}[f(a)] \neq \text{sgn}[f(b)]$ , finnes et nullpunkt  $f(c) = 0$  der  $c \in I$ . Algoritme: (1) Beregn midtpunktet  $m = (a+b)/2$  og finn  $\text{sgn}[f(m)]$ . (2) Hvis  $\text{sgn}[f(m)] = \text{sgn}[f(a)] \Rightarrow I \subset [m, b]$  eller  $\text{sgn}[f(m)] = \text{sgn}[f(b)] \Rightarrow I \subset [a, m]$ . Repetér. Metoden garanterer å finne et nullpunkt med ønsket nøyaktighet.

Feil etter  $N$  iterasjoner (feilen halvreres for hver iterasjon):

$$|c - m_N| \leq \frac{b-a}{2^{N+1}}$$

### Sekantmetoden

Ta sekanten mellom  $(x_0, f(x_0))$  og  $(x_1, f(x_1))$  og beregn dens nullpunkt  $x_2$ . Ta sekanten mellom  $(x_1, f(x_1))$  og  $(x_2, f(x_2))$  og beregn nullpunkt  $x_3$ , osv. Konvergens (ingen garanti): Antall riktige siffer øker med 62% per iteasjon.

$$x_i = x_{i-1} - \frac{x_{i-1} - x_{i-2}}{f(x_{i-1}) - f(x_{i-2})} f(x_{i-1}), \quad i = 2, 3, \dots, N$$

### Newtons metode

Ta tangenten til  $f(x_0)$  og beregn dens nullpunkt  $x_1$ . Ta tangenten til  $f(x_1)$  og beregn nullpunkt  $x_2$ , osv. Konvergens (ingen garanti): Antall riktige siffer doubles per iteasjon.

$$x_i = x_{i-1} - \frac{f(x_{i-1})}{f'(x_{i-1})}, \quad i = 1, 2, \dots, N$$

## Numerisk derivasjon

### Newtons kvotientmetode

$$\frac{f'(x)}{f'(x)} = \frac{f(a+h) - f(a)}{h}$$

Total feil (summen av trunkerings- og avrundingsfeilen):

$$\underbrace{\left| \underbrace{f'(a)}_{\text{eksakt}} - \underbrace{f'(a)}_{\text{tilnærmet}} \right|}_{\text{absolutt feil}} \leq \underbrace{\frac{h}{2} M_1}_{\text{trunkeringsfeil}} + \underbrace{\frac{2\epsilon^*}{h} M_2}_{\text{avrundingsfeil}} \quad (\epsilon^* = 7 \cdot 10^{-17})$$

$$M_1 = \max_{x \in [a, a+h]} |f''(x)| \quad \text{og} \quad M_2 = \max_{x \in [a, a+h]} |f(x)|$$

Optimal steglengde  $h$ :

$$h^* = 2 \sqrt{\frac{\epsilon^* M_2}{M_1}} \approx 2 \sqrt{\frac{\epsilon^* |f(a)|}{|f''(a)|}}$$

### Symmetrisk versjon av Newtons kvotientmetode

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

$$\left| f'(a) - \overline{f'(a)} \right| \leq \frac{h^2}{6} M_1 + \frac{\epsilon^*}{h} M_2.$$

$$M_1 = \max_{x \in [a-h, a+h]} |f'''(x)| \quad \text{og} \quad M_2 = \max_{x \in [a-h, a+h]} |f(x)|.$$

$$h^* = \sqrt[3]{\frac{3\epsilon^* M_2}{M_1}} \approx \sqrt[3]{\frac{3\epsilon^* |f(a)|}{|f'''(a)|}}$$

### En firepunkt-metode

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

$$\left| f'(a) - \overline{f'(a)} \right| \lesssim \frac{h^4}{18} |f^{(v)}(a)| + \frac{3\epsilon^*}{h} |f(a)|$$

$$h^* = \sqrt[5]{\frac{27\epsilon^* |f(a)|}{2|f^{(v)}(a)|}}$$

### Numerisk tilnærming til den annenderiverte

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

$$\left| f''(a) - \overline{f''(a)} \right| \leq \frac{h^2}{12} M_1 + \frac{3\epsilon^*}{h^2} M_2$$

$$M_1 = \max_{x \in [a-h, a+h]} |f^{(iv)}(x)| \quad \text{og} \quad M_2 = \max_{x \in [a-h, a+h]} |f(x)|$$

$$h^* = \sqrt[4]{\frac{36\epsilon^* |f(a)|}{|f^{(iv)}(a)|}}$$

### Numerisk integrasjon

Numerisk integrasjon er lite sensitiv til avrundingsfeil, så alle feilestimater under baserer seg på trunkeingsfeil.

### Midpunktmetoden

$$\int_a^b f(x) dx \approx I_{\text{mid}}(h) = h \sum_{i=1}^n f(x_{i-1/2})$$

$$x_{i-1/2} = \frac{x_{i-1} + x_i}{2} = a + (i-1/2)h$$

Lokal feil ( $a$  og  $b$  utgjør *ett* intervall):

$$\left| \int_a^b f(x) dx - \overbrace{hf(a_{1/2})}^{\text{tilnærmet}} \right| \leq \frac{h^3}{24} M$$

$$M = \max_{x \in [a,b]} |f''(x)| \quad \text{og} \quad a_{1/2} = (a+b)/2$$

Global feil ( $a$  og  $b$  utgjør *hele* integralet,  $h$  er steglengde for ett intervall):

$$|I - I_{\text{mid}}| \leq (b-a) \frac{h^2}{24} M$$

Optimal steglengde  $h$ :

$$h \leq \sqrt{\frac{24\epsilon}{(b-a)M}}$$

der  $\epsilon$  er ønsket nøyaktighet.

### Trapesmetoden

$$\int_a^b f(x) dx \approx I_{\text{trap}}(h) = h \left( \frac{f(a) + f(b)}{2} + \sum_{i=1}^{n-1} f(x_i) \right)$$

Lokal feil:

$$\left| \int_a^b f(x) dx - \frac{f(a) + f(b)}{2} h \right| \leq \frac{h^3}{6} M$$

$$M = \max_{x \in [a,b]} |f''(x)|$$

Global feil:

$$|I - I_{\text{trap}}| \leq (b-a) \frac{h^2}{6} M$$

### Simpsons metode

$$\int_a^b f(x) dx \approx I_{\text{simp}}(h) = \frac{h}{3} \left( f(a) + f(b) + 2 \sum_{i=1}^{n-1} f(x_{2i}) + 4 \sum_{i=1}^n f(x_{2i-1}) \right)$$

Lokal feil:

$$\left| \int_a^b f(x) dx - \frac{f(a) + f(b)}{2} h \right| \leq \frac{h^5}{2880} \max_{x \in [a,b]} |f^{(iv)}(x)|$$

Global feil:

$$|I - I_{\text{simp}}| \leq (b-a) \frac{h^4}{180} \max_{x \in [a,b]} |f^{(iv)}(x)|$$

### Numerisk løsning av differensiallikninger

Dette gjør oss i stand til enkelt å løse difflikninger, også de vi ikke kan løse analytisk.

### Eulers metode

For et intervall  $[a, b]$  inndelt i  $n$  tidssteg og startverdier  $(t_0, x_0)$ , kan vi iterere oss frem til løsningen  $x(t_k) \approx x_k$  av en differensiallikning med algoritmen:

$$\begin{aligned} h &= (b-a)/n \\ t_0 &= a \\ \text{for } k &= 0, 1, \dots, n-1 \\ x_{k+1} &= x_k + hf(t_k, x_k) \\ t_{k+1} &= a + (k+1)h \end{aligned}$$

hvor  $x' = f(t_n, x_n)$ .

Lokal feil (for ett steg):

$$\frac{h^2}{2} \max_{t \in [a,b]} |x''(t)| = \frac{h^2}{2} D$$

Global feil i steg  $k$ :

$$|\epsilon_k| \leq h \frac{D}{2C} (e^{(t_k-a)C} - 1) \leq h \frac{D}{2C} (e^{(b-a)C} - 1)$$

$$C = \max_{t \in [a,b], x \in \mathbb{R}} |f_x(t, x)| \quad \text{og} \quad D = \max_{t \in [a,b]} |x''(t)|$$

Global feil er av 1. orden (proporsjonal med  $h$ ), lokal feil er av 2. orden.

### Eulers midtpunktmetode

Samme som Eulers metode, men baserer seg på to steg per iterasjon.

$$\begin{aligned} h &= (b-a)/n \\ t_0 &= a \\ \text{for } k &= 0, 1, \dots, n-1 \\ x_{k+1/2} &= x_k + hf(t_k, x_k)/2 \\ x_{k+1} &= x_k + hf(t_k + h/2, x_{k+1/2}) \\ t_{k+1} &= a + (k+1)h \end{aligned}$$

Global feil er av 2. orden (proporsjonal med  $h^2$ ), lokal feil er av 3. orden.

### Differensiallikninger

#### Førsteordens (lineære)

$$y'(x) + f(x)y(x) = g(x)$$

har løsning

$$y(x) = e^{-F(x)} \left( \int e^{F(x)} g(x) dx + C \right)$$

hvor  $F(x)$  er anti-derivert til  $f(x)$  og  $e^{F(x)}$  er integrerende faktor.

### Annendens, homogene med konstante koeffisienter

$$y''(x) + py'(x) + q = 0$$

der  $p, q$  er kjente reelle tall. Karakteristiskisk polynom

$$r^2 + pr + q = 0 \Rightarrow r = r_1, r_2$$

Én reell rot  $r_1$ :  $y(x) = Ce^{r_1x} + Dx e^{r_1x}$   
 To reelle røtter  $r_1, r_2$ :  $y(x) = Ce^{r_1x} + De^{r_2x}$   
 To komplekse røtter:  $y(x) = e^{ax}(C \cos(bx) + D \sin(bx))$   
 $r_1 = a + ib, r_2 = a - ib$

### Annendens, inhomogene med konstante koeffisienter

$$y''(x) + py'(x) + qy = f(x)$$

har løsning  $y = y_p + y_h$  der  $y_p$  (partikulær) og  $y_h$  (homogen). Gjetter  $y_p$ ,  
Regel 1:  $f(x) = \text{polynom} \Rightarrow y_p = \text{polynom}$  (lik grad, evt. gå opp én/to grader).  
Regel 2:  $f(x) = a^x P(x)$  der  $a > 0$  og  $P(x)$  et polynom  $\Rightarrow y_p = a^x Q(x)$  der  $Q$  er et polynom av samme grad som  $P$  (evt. gå opp én eller to grader).  
Regel 3: 1)  $f(x) = a^x [A \cos(bx) + B \sin(bx)] \Rightarrow y_p = a^x [C \cos(bx) + D \sin(bx)]$   
 2)  $y_p = xa^x [C \cos(bx) + D \sin(bx)]$  hvis  $y_h = a^x \cos(bx)$  eller  $y_h = a^x \sin(bx)$ .

### Separable differensiallikninger

Separerer alle  $y$  og alle  $x$  på hver sin side og integrerer:

$$q(y)y' = p(x) \Rightarrow \int q(y)dy = \int p(x)dx$$

### Systemer av differensiallikninger

Et system av differensiallikninger kan alltid skrives som et system av førsteordens likninger. Nyttig når vi f.eks. skal bruke Eulers metode for å løse koblede difflikninger.

Skriver likningen av  $p$ 'te orden med løsning  $x(t)$  på formen:

$$x^{(p)} = g(t, x, x', \dots, x^{(p-1)})$$

Omdøp  $x \rightarrow x_1$ . Systemet av  $p$  likninger med  $p$  ukjente funksjoner  $x_1, x_2, \dots, x_p$  er:  $x'_1 = x_2, x'_2 = x_3, \dots, x'_p = g(t, x, x', \dots, x^{(p-1)})$ .

Eksempel: (sett  $x_1 = x$ )

$$\left. \begin{aligned}
 x'' + \sin(tx') - x^2 &= e^t \Rightarrow x'' = e^t - \sin(tx') + x^2 \\
 x'_1 &= x_2 \\
 x'_2 &= e^t - \sin(tx_2) + x_1^2
 \end{aligned} \right\} \text{System av 1. ordens likn.} \quad (1)$$

Eulers metode med (1):

$$\begin{aligned}
 x_{1,k+1} &= x_{1,k} + hx_{2,k} \\
 x_{2,k+1} &= x_{2,k} + hf(t_k, x_{1,k}, x_{2,k}) \\
 t_{k+1} &= a + (k+1)h
 \end{aligned}$$

hvor  $f(t_k, x_{1,k}, x_{2,k}) = e^t - \sin(tx_{2,k}) + x_{1,k}^2$

Eksempel: De koblede likningene

$$\begin{aligned}
 x''' &= y''x^2 - 3(y')^2x = (t+x')x^2 - 3(y')^2x \\
 y'' &= t+x'
 \end{aligned}$$

gir systemet av førsteordens likninger (sett  $x_1 = x, y_1 = y$ ):

$$\begin{aligned}
 x'_1 &= x_2 & y'_1 &= y_2 \\
 x'_2 &= x_3 & y'_2 &= t+x_2 \\
 x'_3 &= (t+x_2)x_1^2 - 3y_2^2x_1
 \end{aligned}$$

Vektornotasjon:

$$\begin{aligned}
 \mathbf{x}' &= \mathbf{f}(t, \mathbf{x}), \quad \mathbf{x}(a) = \mathbf{x}_0 \\
 \mathbf{f}(t, \mathbf{x}) &= (f_1(t, \mathbf{x}), \dots, f_M(t, \mathbf{x})) \\
 \mathbf{x} = \mathbf{x}(t) &= (x_1(t), \dots, x_M(t)) \\
 \mathbf{x}' = \mathbf{x}'(t) &= (x'_1(t), \dots, x'_M(t))
 \end{aligned}$$