

Numme: Definitioner och Satser

Definitioner		Satser	
absolutstabil	5	1.2	5
dominerande egenvektor	15	1.6	6
dominerande egenvärden	15	1.11	7
elementär radadditionsmatris	10	1.a	6
Fixpunkt	6	1.b	6
globala felet	3	2.2	10
gradient	7	2.3	10
Hessian	7	2.6	12
inducerad matrisnorm	11	2.d	11
interpolation	8	3.2: Huvudsatsen för polynom interpolation	8
Jacobimaträs	7	3.4	9
Konvergens		5.a: Trapetsregeln	9
konvergenshastighet	4/6	5.b: Sammansatta trapetsregeln	9
kvadratisk konvergens	7	6.2	3
linjär konvergens	6	6.3	3
kubisk spline	9	6.4	3
Lagrange polynom	8	6.a	4
Lipschitz kontinuerlig	3	6.b	4
lokala felet	3	6.8	5
naturlig kubisk spline	9	6.5	5
noggrannhetsordning	9	8.1	15
norm	11	8.2	15
normalekvationer	8	8.3	15
ordning	3	8.4	15
relativt bakätfel	11	8.6	15
relativt framätfel	11	8.8	15
relativt konditionstal	12	8.b	15
relativt konditionstal för matris	12	9.a	10
rot	5	9.b	10
stabilitetsområde	5	A.a: Satsen om mellanliggande värden	9
störning	5	A.b: Medelvärdessatsen för integraler	9
styr	5	A.c: Medelvärdessatsen för summor	9
undertriangulär	10	S.a	10
unimodal	8	V.a: Produktregeln för gradienten	7
överbestämd	8	V.b: Taylorutveckling i flera dimensioner	7
övertriangulär	10	12.2	16
		12.a: Rayleighkvoten	16

Metoder

Metoder	Noggrannhetsordning	Konvergensordning	Sida
Bakåt differensmetoden för värmeförlustekvationen	Tid:1 Rum:2	-	14
Bakåt Euler	1	-	4
Bakåtsubstitution med LU faktorisering	-	-	11
Bisektionsmetoden	-	1	6
Crank - Nicolson metoden	Tid:2 Rum:2	-	15
Eulers metod	1	-	3
Finita differensmetoden	2	-	12
Fixpunktiteration	-	1	6
Framåt differensmetoden för värmeförlustekvationen	Tid:1 Rum:2	-	14
Galerkinmetoden	?	-	13
Gauss - Newtons metod	-	1 (beror på)	8
Gradient sökning	-	?	8
Gyllene snittet sökning	-	1	8
Inversa potensmetoden	-	1	16
LU faktorisering	-	-	11
Monte Carlo	1/2	-	10
Newton's metod	-	2	6
Newton's metod för optimering	-	2	7
Newton's metod i flera dimensioner	-	2	7
Potensmetoden	-	1	16
Runge - Kutta ordning 4	4	-	4
Shooting	-	-	12
Trapets metoden	2	-	4

Bevis

Sats	Sida
1.6	17
1.11	17
2.5	17
3.2: Huvudsatsen för polynominterpolation	18
5.Ω	18
5.Γ	18
6.Ω	19
6.Γ	20
8.2	20
12.2	21

Numerisk Analys

Metod (Eulers metod): Betrakta begynnelsevärdesproblemet

$$\begin{cases} y'(t) = f(t, y) & : t \in [t_0, b] \\ y(t_0) = y_0 \end{cases}$$

För en given funktion f .

Definiera steglängden $h > 0$ som ett litet tal.

Från Taylors formel $y(a+h) \approx y(a) + hy'(a)$ får vi Eulers metod:

$$\begin{cases} y_{k+1} := y_k + hf(t_k, y_k) \\ t_{k+1} := t_k + h \end{cases}$$

Och får $y_n \approx y(t_n)$

Anmärkning: Approximationen $y_n \approx y(t_n)$ blir generellt bättre för mindre h .

Def (globala felet): Det globala felet definieras som skillnaden mellan den exakta och den uppskattade lösningen

$$g_k := |y(t_k) - y_k|$$

Anmärkning: Då vi oftast inte vet $y(t)$ och därför inte heller $y(t_k)$ kan denna definition sällan användas för att uppskatta det globala felet.

Def (lokala felet): Det lokala felet e_i är skillnaden mellan den uppskattade lösningen och en exakt lösning med initialvillkor y_{i-1} , dvs:

$$e_i = |z(t_i) - y_i|$$

där

$$\begin{cases} z' = f(t, z) \\ z(t_{i-1}) = y_{i-1} \\ t \in [t_{i-1}, t_i] \end{cases}$$

Def (Lipschitz kontinuerlig): En funktion $f(t, y)$ är Lipschitz kontinuerlig i y för $t \in [a, b]$ om det finns en konstant L s.a.

$$|f(t, y_1) - f(t, y_2)| \leq L|y_1 - y_2|$$

för alla $t \in [a, b]$ och y_1, y_2 .

Anmärkning: Ibland säger man endast att $f(t, y)$ är Lipschitz.

SATS 6.2: Om f är Lipschitz då existerar det precis en lösning till begynnelsevärdesproblemet

$$\begin{cases} y' = f(t, y) \\ y(a) = y_0 \\ t \in [a, b] \end{cases}$$

SATS 6.3: Antag att f är Lipschitz i y för $t \in [a, b]$. Om $Y(t)$ och $Z(t)$ löser

$$y' = f(t, y)$$

då gäller att $|Y(t) - Z(t)| \leq e^{L(t-a)} |Y(a) - Z(a)|$

SATS 6.4: Antag att f är Lipschitz kontinuerlig i variabeln y , och att det lokala felet för en ODE-lösare begränsas av $e_i \leq Ch^{k+1}$ för något C . Då begränsas det globala felet av

$$g_i \leq \frac{C}{L} (e^{L(t_i-a)} - 1) h^k$$

Def (ordning): En ODE-lösare vars globala fel är $O(h^k)$ då $h \rightarrow 0$ säges ha ordning k .

Metod (Trapetsmetoden): Betrakta begynnelsevärdesproblemet

$$\begin{cases} y'(t) = f(t, y) \\ y(t_0) = y_0 \end{cases}$$

För en given funktion f

Definiera steglängden $h > 0$ som ett litet tal

Trapetsmetoden är då:

$$t_{i+1} = t_i + h$$

$$y'_{i+1} = t_i + hf(t_i, y_i)$$

$$y_{i+1} = y_i + \frac{h}{2}(f(t_i, y_i) + f(t_{i+1}, y'_{i+1}))$$

$$\text{Och } y_n \approx y(t_n)$$

Metod (Runge-Kutta ordning 4): Betrakta begynnelsevärdesproblemet

$$\begin{cases} y'(t) = f(t, y) \\ y(t_0) = y_0 \end{cases}$$

För en given funktion f

Definiera steglängden $h > 0$ som ett litet tal

Metoden Runge-Kutta ordning 4 är då:

$$t_{i+1} = t_i + h$$

$$s_1 = f(t_i, y_i)$$

$$s_2 = f(t_i + 0.5h, y_i + 0.5hs_1)$$

$$s_3 = f(t_i + 0.5h, y_i + 0.5hs_2)$$

$$s_4 = f(t_i + h, y_i + hs_3)$$

$$y_{i+1} = y_i + \frac{h}{6}(s_1 + 2s_2 + 2s_3 + s_4) \quad \text{och } y_n \approx y(t_n)$$

SATS 6.a: • Eulers metod har ordning 1

• Trapetsmetoden har ordning 2

• Runge-Kutta har ordning 4.

Def(konvergenshastighet): Konvergenshastigheten av en given metod representerar hur snabbt det globala felet i en viss tidpunkt går mot noll då h går mot noll.

SATS 6.b: Om vi halverar steglängden h för en metod av ordning k kommer det globala felet minska med en approximativ faktor 2^k .

Anmärkning: Konvergenshastigheten beror på en metods ordning.

Anmärkning: Med g som globala felet kommer $\log(g)$ som en funktion av $\log(h)$ bilda en rät linje med lutning k .

Metod (Bakåt Euler): Betrakta begynnelsevärdesproblemet

$$\begin{cases} y'(t) = f(t, y) \\ y(t_0) = y_0 \end{cases}$$

För en given funktion f

Definiera steglängden $h > 0$ som ett litet tal

Metoden Bakåt Euler är då:

$$t_{i+1} = t_i + h$$

$$y_{i+1} = y_i + hf(t_{i+1}, y_{i+1}) \quad \text{OBS: Implicit!}$$

$$y_n \approx y(t_n)$$

Bakåt Euler har ordning 1.

4

Def (absolut stabil): Betrakta begynnelsevärdesproblemet

$$\begin{cases} y' = \lambda y & : \lambda < 0 \\ y(0) = 1 \end{cases}$$

En lösning $\{y_n\}_{n \geq 0}$ sägs vara absolut stabil för ett fixt h om $\lim_{n \rightarrow \infty} y_n = 0$

Def (stabilitetsområde): Stabilitetsområdet A är de tal $h \cdot \lambda$ för vilka lösningen till begynnelsevärdesproblemet

$$\begin{cases} y' = \lambda y & : \lambda < 0 \\ y(0) = 1 \end{cases}$$

är absolut stabil.

SATS 6.8: Betrakta begynnelsevärdesproblemet

$$\begin{cases} y' = \lambda y & : \lambda < 0 \\ y(0) = 1 \end{cases}$$

Stabilitetsområdet för

- Eulers metod är $h \lambda \in (-2, 0)$
- Runge-Kutta ordning 4 är ungefärlig $\lambda h \in (-2.7853, 0)$
- Bakåt Euler har $|1 - \lambda h| > 1$ och då $\lambda h < 0$, alltså stabil för alla h .

Def (styvt): Om ett begynnelsevärdesproblem kräver små h för att undvika instabilitet av explicita metoder så säger man att problemet är styvt.

Def (störning): Antag att F representerar någon numerisk metod på ett begynnelsevärdesproblem så att $F(x) = y$ där x är initialdata och y är approximationen av lösningen av problemet i en viss tidpunkt.

Låt \tilde{x} vara indata påverkad av en liten störning ϵ_x , alltså $\tilde{x} := x + \epsilon_x$

Vi definierar nu $\tilde{y} := F(\tilde{x}) = F(x + \epsilon_x)$

Effekten, ϵ_y , av störningen definieras som $\epsilon_y := \tilde{y} - y$

SATS 6.9: Från Taylorutveckling får vi $\epsilon_y \approx \epsilon_x F'(x)$

Anmärkning: Då det oftast saknas en formel för $F(x)$ är det svårt att hitta $F'(x)$.

Def (absolutstabil): En numerisk lösning till

$$\begin{cases} y'(t) = f(t, y) \\ y(t_0) = y_0 \end{cases}$$

är absolutstabil om effekten av en liten störning försvinner då $n \rightarrow \infty$

Def (rot): En funktion $f(x)$ har en rot i $x=r$ om $f(r)=0$.

SATS 1.2: Låt f vara en kontinuerlig funktion i $[a, b]$ som uppfyller $f(a)f(b) < 0$.

Då $\exists r \in (a, b) | f(r) = 0$.

Metod (Bisektionsmetoden): För en kontinuerlig funktion $f: [a, b] \rightarrow \mathbb{R}$: $f(a)f(b) < 0$ kan vi approximera en rot för f , $r \in (a, b)$, med hjälp av bisektionsmetoden:

- 1) $c_{\min} = a$
- 2) $c_{\max} = b$
- 3) $c_{\text{mid}} = (c_{\max} + c_{\min}) / 2$
- 4) Om $f(c_{\text{mid}}) = 0$: returnera $r = c_{\text{mid}}$
- 5) Om $f(c_{\min})f(c_{\text{mid}}) < 0$:

$$c_{\max} = c_{\text{mid}}$$

Annars:

$$c_{\min} = c_{\text{mid}}$$

- 6) Gå till (3).

Repetera 3-6 tills antingen (4) returnerar r eller ett önskat antal repetitioner har gjorts, då kan vi konstatera $r \in (c_{\min}, c_{\max})$

Def (fixpunkt): Låt g vara en godtycklig funktion. r är en fixpunkt till g om $g(r) = r$.

Metod (Fixpunktsiteration): Om vi söker en fixpunkt till g kan man använda fixpunktsiteration:

- 1) Välj x_0 som startpunkt (gissning)
- 2) $x_{i+1} = g(x_i)$ för $i=1, 2, 3, \dots$

SATS 1.α: Om g är en kontinuerlig funktion och x_i konvergerar till r då är r en fixpunkt till g .

Anmärkning: huruvida x_i konvergerar beror på g'

Def (linjär konvergens, konvergenshastighet): Låt $e_i := |x_i - r|$ vara felet efter i iterationer för någon metod som söker värdet r .

Om $\lim_{i \rightarrow \infty} \frac{|e_{i+1}|}{|e_i|} = S < 1$ så har metoden linjär konvergens med konvergenshastighet (en: rate) S

Anmärkning: Mindre $S \Rightarrow$ snabbare konvergens

Anmärkning: Om vi har konvergens så är $\lim_{i \rightarrow \infty} e_i = 0$

SATS 1.β: Antag att g är en kontinuerligt deriverbar funktion och $g(r) = r$ & $|g'(r)| < 1$. Då konvergerar fixpunktsiterationer runt r givet att x_0 är tillräckligt nära r , med rate S .

SATS 1.γ: Om vi söker r för vilket $f(r) = 0$ kan man använda fixpunktsiteration på $g(x) = x + f(x)$.

Metod (Newtons metod): Om vi har en deriverbar funktion och söker en rot r för f kan vi använda Newtons metod:

- 1) Välj x_0 som startpunkt (gissning)

$$2) x_{i+1} = x_i - \frac{f(x_i)}{f'(x_i)} \quad \text{för } i=1, 2, 3, \dots$$

Anmärkning: Newtons metod sätter x_{i+1} som roten till tangentlinjen för $f(x_i)$.

Def (kvadratisk konvergens): Låt $e_i := |x_i - r|$ vara felet efter i iterationer för någon metod som söker värdet r . En metod har kvadratisk konvergens om $\lim_{i \rightarrow \infty} \frac{e_{i+1}}{e_i^2} = M < \infty$

Anmärkning: En metod med kvadratisk konvergens konvergerar snabbare än en metod med linjär konvergens.

Anmärkning: Mindre $M \Rightarrow$ metoden konvergerar generellt snabbare.

SATS 1.11: Om f är två gånger deriverbar, $f(r)=0$ och $f'(r) \neq 0$ är Newtons metod kvadratiskt konvergent och $\lim_{i \rightarrow \infty} \frac{e_{i+1}}{e_i^2} = \frac{f''(r)}{2f'(r)}$

Anmärkning: Om $f'(r)=0$ så kan Newtons metod fortfarande konvergera fast längsammare

Def (gradient): För en funktion $f: \mathbb{R}^n \rightarrow \mathbb{R}$ definierar vi gradienten som $\nabla f: \mathbb{R}^n \rightarrow \mathbb{R}^n : \nabla f(\vec{x}) = \left[\frac{\partial f}{\partial x_1}, \frac{\partial f}{\partial x_2}, \dots, \frac{\partial f}{\partial x_n} \right]^T$

Def (Jacobimatriss): För en funktion $F: \mathbb{R}^n \rightarrow \mathbb{R}^m$, där $F(\vec{x}) = [F_1(\vec{x}), F_2(\vec{x}), \dots, F_m(\vec{x})]^T$, definierar vi Jacobimatrissen som $DF: \mathbb{R}^n \rightarrow \mathbb{R}^{m \times n} : [DF]_{ij} = \frac{\partial F_i}{\partial x_j}$

Def (Hessian): För en funktion $f: \mathbb{R}^n \rightarrow \mathbb{R}$ definierar vi Hessianen som $H_f: \mathbb{R}^{n \times n} : [H_f]_{ij} = \frac{\partial^2 f}{\partial x_i \partial x_j}$. Det vill säga att Hessianen av f är Jacobimatrissen av $(\nabla f)^T$. $H_f = D(\nabla f)^T$

SATS V.α (produktsregel för gradienten): Antag $u, v: \mathbb{R}^n \rightarrow \mathbb{R}^m$, $u^T v: \mathbb{R}^n \rightarrow \mathbb{R}: x \mapsto u(x)^T v(x)$ vi har då: $\nabla(u^T v) = u^T Dv + v^T Du$

SATS V.β (Taylorutveckling i flera dimensioner): Följande påståenden är taylorutvecklingar för givna funktioner

- $f: \mathbb{R} \rightarrow \mathbb{R}: f(x+h) = f(x) + f'(x)h + \frac{1}{2}f''(x)h^2 + \dots + \frac{1}{i!}f^{(i)}(x)h^i + \dots$
- $f: \mathbb{R}^n \rightarrow \mathbb{R}: f(x+h) = f(x) + \nabla f(x)h + \frac{1}{2}h^T H_f h + \dots$
- $F: \mathbb{R}^n \rightarrow \mathbb{R}^m: F(x+h) = F(x) + DF(x)h + \dots$

Metod (Newtons metod i flera dimensioner): Låt $F: \mathbb{R}^n \rightarrow \mathbb{R}^m$ och vi söker x^* som uppfyller $F(x^*)=0$. Givet ett startvärde x^0 får vi Newtons metod i flera dimensioner:

1) Låt v vara lösning till $DF(x^k)v = -F(x^k)$

$$2) x^{k+1} = x^k + v$$

Då får vi $x^n \approx x^*$ för större n .

Anmärkning: För x^k är k ett index, inte en potens

Metod (Newtons metod för optimering): Låt $f: \mathbb{R}^n \rightarrow \mathbb{R}$ och vi söker x^* som uppfyller $\nabla f(x^*)=0$, dvs en stationär punkt för f . Givet ett startvärde x^0 får vi då Newtons metod för optimering

1) Låt v vara lösning till $H_f(x^k)v = -(\nabla f)^T$

$$2) x^{k+1} = x^k + v$$

Då får vi $x^n \approx x^*$ för större n .

Anmärkning: Denna metod kommer från Newtons metod i flera dimensioner där vi sätter $F = (\nabla f)^T$

Anmärkning: Att lösa $H_f v = -(\nabla f)^T$ kräver m^3 operationer

Metod (Gradientssökning): Antag att vi söker ett minimum till $f: \mathbb{R}^n \rightarrow \mathbb{R}$ och vi har givet ett startvärde x^0 och en steglängd $s > 0$. Då använder vi gradientssökning för $k=0,1,\dots$

$$x^{k+1} = x^k - (\nabla f)(x^k)^T \cdot s$$

Anmärkning: För att hitta en maxpunkt tar vi istället $x^{k+1} = x^k + (\nabla f)(x^k)^T \cdot s$

Anmärkning: Denna metod är oftast lättare än Newtons metod för optimering, men den kan kräva fler iterationer

Def(unimodal): En funktion $f: \mathbb{R} \rightarrow \mathbb{R}$ är unimodal på intervallet $[a,b]$ om det finns exakt ett minimum på intervallet

Metod (Gyllene snittet sökning): Givet $f: \mathbb{R} \rightarrow \mathbb{R}$ är unimodal med ett minimum på $[a,b]$ kan vi hitta minimipunkten med hjälp av gyllene snittet sökning:

- 1) Definiera $g = (\sqrt{5}-1)/2$

- 2) $x_1 = a + (1-g)(b-a)$

- 3) $x_2 = a + g(b-a)$

- 4) Om $f(x_1) < f(x_2)$:

$$b = x_2$$

Annars

$$a = x_1$$

- 5) Repetera steg (2) till (4) önskat antal gånger

Minimipunkten kommer ligga i nya intervallet $[a,b]$.

Def(överbestämd): Ett linjärt ekvationssystem $Ax=b$ är överbestämt om det finns fler ekvationer än obekanta variabler.

Anmärkning: Detta innebär ofta att det inte finns någon lösning.

Def(normallekvationer): För ett överbestämt linjärt ekvationssystem $Ax=b$ är systemet $A^T A x = A^T b$ normal ekvationerna.

Anmärkning: Det x som löser normallekvationerna är minsta kvadratlösningen till $Ax=b$.

Metod (Gauss-Newtonss metod): Givet $r: \mathbb{R}^n \rightarrow \mathbb{R}^m$ och vi söker det $x \in \mathbb{R}^n$ som minimerar längden av $r(x)$, dvs minimerar $E(x) = \frac{1}{2} r^T r$. Med startvärde x^0 kan vi då använda Gauss-Newtonss metod för $k=0,1,2,\dots$

- 1) Låt $A = Dr(x^k)$

- 2) Bestäm V som löser $A^T A V = -A^T r(x^k)$

- 3) Sätt $x^{k+1} = x^k + V$

Def(interpolation): Funktionen $y = p(x)$ interpolerar datan $(x_1, y_1), \dots, (x_n, y_n)$ om $p(x_i) = y_i$ för $i=1,2,\dots,n$

Def(Lagrange polynom): Givet n olika punkter $(x_1, y_1), \dots, (x_n, y_n)$ så kallas polynomet

$$p(x) = \sum_{i=1}^n y_i \prod_{k \neq i} \frac{(x-x_k)}{(x_i-x_k)}$$

för Lagrange polynomet, har grad $n-1$ och interpolerar datan.

SATS 3.2 (Huvudsatsen för polynominterpolation): Låt $(x_1, y_1), \dots, (x_n, y_n)$ vara n punkter med olika värden på x . Då finns ett och endast ett polynom av grad $\leq n-1$ som interpolerar datan.

SATS 3.4: Låt $P(x)$ vara polynomet av grad $\leq n-1$ som interpolerar en funktion f i punkterna x_1, \dots, x_n . Interpolationsfelet kan skrivas som

$$f(x) - P(x) = \frac{1}{n!} (x-x_1)(x-x_2)\dots(x-x_n) f^{(n)}(c) \text{ där } c \in (\min\{x_1, \dots, x_n\}, \max\{x_1, \dots, x_n\})$$

Def(kubisk spline): Givet data $(x_1, y_1), \dots, (x_n, y_n)$ är en kubisk spline $s(x)$: en funktion $s: [x_1, x_n] \rightarrow \mathbb{R}$ så att s är definierat av $n-1$ 3:e gradens polynom:

$$s(x) = y_i + b_i(x-x_i) + c_i(x-x_i)^2 + d_i(x-x_i)^3 \text{ för } x \in [x_i, x_{i+1}]$$

Som uppfyller följande kriterier:

- 1) $s(x)$ är kontinuerlig och interpolerar datan: $s_i(x_{i+1}) = s_{i+1}(x_{i+1}) \quad i=1,2,\dots,n-2$
och $s_{n-1}(x_n) = y_n$
- 2) $s'(x)$ är kontinuerlig: $s'_{i-1}(x_i) = s'_i(x_i)$ för $i=2, \dots, n-1$
- 3) $s''(x)$ är kontinuerlig $s''_{i-1}(x_i) = s''_i(x_i)$ för $i=2, \dots, n-1$

Anmärkning: s är bestämd av $3(n-1)$ parametrar $b_1, c_1, d_1, b_2, c_2, d_2, \dots, b_{n-1}, c_{n-1}, d_{n-1}$ och villkor 1 till 3 ger $(n-1) + (n-2) + (n-2) = 3(n-1) - 2$ linjära ekvationer i b_i, c_i, d_i vilket betyder att $s(x)$ inte är entydigt bestämd.

Def(naturlig kubisk spline): Givet data $(x_1, y_1), \dots, (x_n, y_n)$ är en naturlig kubisk spline den kubiska spline som också uppfyller kriteriet:

$$4) s_i''(x_i) = 0 \text{ och } s_{n-1}''(x_n) = 0$$

Anmärkning: Villkor 1 till 4 ger $3(n-1)$ linjära ekvationer i b_i, c_i, d_i och s är då entydigt bestämd.

Def(noggrannhetsordning): Antag u är en exakt lösning till en beräkning, antag vidare att \tilde{u}_h är ett resultat av en numerisk metod för att beräkna u och \tilde{u}_h är beroende av h . Om det för tillräckligt små h finns ett C oberoende av h så att $|\tilde{u}_h - u| \leq C \cdot h^p$ så säges metoden ha noggrannhetsordning p .

SATS A.α (Satsen om mellanliggande värden): Om $f: [a, b] \rightarrow \mathbb{R}$ är kontinuerlig och $f(a) < d < f(b)$ $\exists x \in (a, b) : f(x) = d$

SATS A.β (Medelvärdessatsen för integraler): Låt $f, g: [a, b] \rightarrow \mathbb{R}$, då finns $c \in [a, b]$ s.a. $\int_a^b f(x)g(x)dx = f(c) \int_a^b g(x)dx$

SATS A.γ (Medelvärdessatsen för summor): Låt $f: \mathbb{R} \rightarrow \mathbb{R}$. Givet $\sum_{i=1}^n a_i f(x_i)$ $\exists c \in [\min\{x_1, \dots, x_n\}, \max\{x_1, \dots, x_n\}]$ så att $\sum_{i=1}^n a_i f(x_i) = f(c) \sum_{i=1}^n a_i$

SATS 5.α (Trapetsregeln): Vi har att $\int_{x_0}^{x_1} f(x)dx = \frac{h}{2} (f(x_0) + f(x_1)) - \frac{h^3}{12} f''(c)$ där $h = x_1 - x_0$ för något $c \in [x_0, x_1]$

SATS 5.β (Sammansatta trapetsregeln): Låt $u = \int_a^b f(x)dx$ och låt $a = x_0 < x_1 < \dots < x_n = b$ och definiera $\tilde{u}_h = \sum_{i=1}^n \left(\frac{h}{2} (f(x_i) + f(x_{i-1})) \right)$

Vi har då att för något $c \in [a, b]$ $u = \tilde{u}_h - \frac{f''(c)}{12} (b-a)h^2$

Anmärkning: denna metod har alltså noggrannhetsordning 2.

SATS S.α: Låt X vara en stokastisk variabel med väntevärde $E[X]=A$ och varians $\text{Var}(X)$. Låt X_1, \dots, X_N vara N stycken oberoende stokastiska variabler med samma fördelning som X .
 Låt nu $Y := \frac{1}{N} \sum_{i=1}^N X_i$ vara en approximation av X 's medelvärde, vi har då:

- 1) $E[Y] = A$
- 2) $\text{sd}(Y) = \frac{1}{\sqrt{N}} \sqrt{\text{Var}(x)}$

SATS 9.α: Antag $V \subset \mathbb{R}^n$ är ett område med ändlig volym. Antag X är en stokastisk variabel som är likafördelad i V . Då gäller

$$\int_V f(x) dx = \text{vol}(V) \cdot E[f(X)]$$

Metod (Monte Carlo): Antag att vi vill bestämma $\int_V f(x) dx$. Vi använder Monte Carlo integration vilket är att approximera

$$\int_V f(x) dx \approx \frac{\text{vol}(V)}{N} \sum_{i=1}^N f(x_i)$$

där x_1, \dots, x_N är N oberoende utfall av X .

SATS 9.β: Låt $u = \int_V f(x) dx$ och $\tilde{u}_n = \frac{\text{vol}(V)}{N} \sum_{i=1}^N f(x_i)$ vara approximationen av u med hjälp av Monte Carlo integration där $h = \frac{1}{N}$. Monte Carlo metoden har då noggrannhetsordning $1/2$.

Anmärkning: Detta innebär att Monte Carlo är mindre noggrann än trapetsmetoden.

Def(undertriangulär): En matris $L \in \mathbb{R}^{n \times n}$ är undertriangulär om $L_{ij} = 0$ om $i < j$.

Def(övertriangulär): En matris $U \in \mathbb{R}^{n \times n}$ är övertriangulär om $U_{ij} = 0$ om $i > j$.

Def(elementär radadditionsmatris): En elementär radadditionsmatris $L_{ij}(c)$ där $c \in \mathbb{R}$ och $i > j$ är en elementär matris som utför radoperationen "addera c gånger rad j till rad i " på matrisen A när man utför beräkningen $L_{ij}(c) \cdot A$. $L_{ij}(c)$ har formen $(L_{ij}(c))_{kl} = \begin{cases} 1 & : k=l \\ c & : k=i, l=j \\ 0 & : \text{annars} \end{cases}$

SATS 2.2: $(L_{ij}(c))^{-1} = L_{ij}(-c)$

SATS 2.3: Låt L vara en undertriangulär matris $(L)_{ij} = \begin{cases} 1 & : i=j \\ l_{ij} & : i > j \\ 0 & : i < j \end{cases}$
 Vi har då $L = L_{21}(l_{21}) \cdot L_{31}(l_{31}) \cdot L_{32}(l_{32}) \cdots L_{n,n-1}(l_{n,n-1})$

$$= \prod_{i=2}^n \prod_{j=1}^{i-1} L_{ij}(l_{ij})$$

Metod (LU faktorisering): Låt $A \in \mathbb{R}^{n \times n}$ vara en matris och vi söker en undertriangulär matris L och en övertriangulär matris U som uppfyller $A = LU$. Vi kan använda **LU faktorisering** för att bestämma L och U :

- 1) Bestäm U genom Gauss-Jordan eliminering men notera de radoperationer som gjordes så man får $L_{ikjk}(c_k) \dots L_{ii,j_i}(c_i) \cdot A = U$
- 2) Definiera $L = (L_{ii,j_i}(c_i))^{-1} \dots (L_{ikjk}(c_k))^{-1} = L_{ii,j_i}(-c_i) \dots L_{ikjk}(-c_k)$

Vi får då $A = LU$

Anmärkning: LU faktorisering kräver $\sim \frac{2}{3}n^3$ operationer

Metod (Bakätsubstitution med LU faktorisering): Låt $A \in \mathbb{R}^{n \times n}$ och $b_i \in \mathbb{R}^n$ vara givna och vi söker $x_i \in \mathbb{R}^n$ som uppfyller $Ax_i = b_i$, $i=1, \dots, m$. Vi kan då använda **bakätsubstitution med LU faktorisering**:

- 1) Använd LU faktorisering för att bestämma L och U s.a. $A = LU$
- 2) Repetera följande steg för $i=1, \dots, m$:
 - a) Ansätt $Lv_i = b_i$ och bestäm v_i
 - b) Ansätt $Ux_i = v_i$ och bestäm x_i

Anmärkning: Varje iteration av (2) kräver $\sim 2n^2$ operationer vilket betyder att den fullständiga lösningen för alla $i=1, \dots, m$ kräver $\sim \frac{2}{3}n^3 + 2mn^2$ operationer.

Anmärkning: Denna metod är användbar för stora n och $m > 1$ då man allt som allt behöver färre operationer för att lösa systemen än om man skulle lösa $Ax_i = b_i$ med Gauss-Jordan eliminering för alla $i=1, \dots, m$ som kräver $\sim \frac{3}{2}mn^3$ operationer.

Def(norm): Normen av en vektor $x \in \mathbb{R}^n$, betecknat $\|x\| \in \mathbb{R}$, är ett mätt på längden av x . Det finns olika normer:

- 1) $\|x\|_2 = \sqrt{x_1^2 + x_2^2 + \dots + x_n^2}$
- 2) $\|x\|_\infty = \max_k |x_k|$

Gemensamt för alla normer är:

- 1) $\|x\| \geq 0$ med likhet om $x=0$
- 2) $\|\alpha x\| = |\alpha| \|x\|$ för $\alpha \in \mathbb{R}$
- 3) $\|x+y\| \leq \|x\| + \|y\|$

Def(inducerad matrismorm): Om $\|\cdot\|$ är en vektornorm så finns det en **inducerad matrismorm** för $A \in \mathbb{R}^{m \times n}$:

$$\|A\| = \max_{\substack{x \in \mathbb{R}^n \\ x \neq 0}} \frac{\|Ax\|}{\|x\|}$$

$\|A\|$ kan ses som ett mätt av förändringen av längden på vektorer som A appliceras på

SATS 2.2: Vi har följande sätt att bestämma inducerade matrismormer:

$$1) \|A\|_2 = \text{största singulärvärdet till } A$$

$$2) \|A\|_\infty = \max_j \sum_i |A_{ij}|$$

Def(relativt bakätfel, relativt framätfel): Låt F vara en numerisk metod för att lösa något problem så att x är indata och $y = F(x)$ är lösningen. Antag att E_x är en liten störning av indata s.a. $\tilde{x} = x + E_x$ och låt $E_y = F(\tilde{x}) - y$. E_x kallas **bakätfel** och E_y är **framätfel**.

Det relativta bakätfel och relativta framätfel definieras som:

$$\frac{\|E_x\|}{\|x\|} \text{ respektive } \frac{\|E_y\|}{\|y\|}.$$

Def(relativt konditionstal): Låt F vara en numerisk metod för att lösa något problem så att x_0 är indata och $y = F(x_0)$ vara lösningen. Om vi har baktfelet E_x och framfärdsfelet E_y definierar vi det relativt konditionstalet som den maximala förstärkningen av det relativta felet för en liten störning

$$K = \lim_{\delta \rightarrow 0} \frac{\max_{1 \leq i \leq n} \frac{\|E_{y_i}\|}{\|y_i\|}}{\|E_{x_i}\|}$$

Def(relativt konditionstal för matris): Antag A är en $n \times n$ -matris. A 's konditionstal är det relativta konditionstalet som svarar mot att lösa $Ax=b$ med b som indata. Vi betecknar A 's konditionstal som $\text{cond}(A)$

SATS 2.6: Konditionstalet för $n \times n$ -matrisen A kan beräknas som:

$$\text{cond}(A) = \|A\| \cdot \|A^{-1}\|$$

Metod(Shooting): Vi söker en lösning för randvärdesproblemet:

$$\begin{cases} y'' = f(t, y, y') \\ y(a) = y_a \\ y(b) = y_b \end{cases} \quad t \in [a, b]$$

Vi kan använda shooting-metoden:

1) Hitta lösningen $y(t, s)$ till begynnelsevärdesproblemet

$$\begin{cases} y'' = f(t, y, y') \\ y(a) = y_a \\ y'(a) = s \end{cases}$$

med hjälp av t.ex. Eulers metod.

2) Definiera $F(s) = y_b - y(b, s)$

3) Bestäm s^* så att $F(s^*) = 0$ med hjälp av t.ex. bisektionsmetoden.

4) Vi får lösningen $y(t) = y(t, s^*)$.

Metod(Finita differensmetoden): Vi söker en lösning för randvärdesproblemet:

$$\begin{cases} y'' = f(t, y, y') \\ y(a) = y_a \\ y(b) = y_b \end{cases} \quad t \in [a, b]$$

Vi kan använda finita differensmetoden:

1) Approximera y' och y'' med finit differens

$$y'(t) = \frac{y(t+h) - y(t-h)}{2h} \quad y''(t) = \frac{y(t+h) - 2y(t) + y(t-h)}{h^2}$$

2) Låt $y_i = y(t_i)$ och $t_{i+1} = t_i + h$ för en steglängd h

3) Lös ekvationssystemet $\frac{y_{i+1} - 2y_i + y_{i-1}}{h^2} = f(t_i, y_i, \frac{y_{i+1} - y_{i-1}}{2h})$

$$t_0 = a, \quad t_N = b, \quad h = \frac{b-a}{N}$$

$$y_0 = y_a, \quad y_N = y_b$$

Metod (Galerkinmetoden): Betrakta randvärdesproblemet

$$\begin{cases} y'' = f(t, y, y') \\ y(a) = y_a \\ y(b) = y_b \end{cases} \quad (*)$$

Låt $r(t; y) := y'' - f(t, y, y')$. Att lösa (*) innebär att man hittat y sådan att $r(t; y) = 0$ för alla t .

Galerkinmetoden består av följande steg:

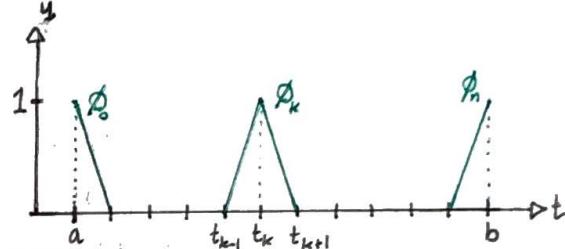
1) Välj en bas $\mathcal{B} = \{\phi_0, \phi_1, \dots, \phi_n\}$ till ett delrum $V \subset L^2([a, b])$:

Låt $h = \frac{b-a}{n}$; $t_k = a + h \cdot k$

$$\phi_k(t) := \begin{cases} (t-t_{k-1}) \frac{1}{h} & : t_{k-1} \leq t \leq t_k \\ (t_{k+1}-t) \frac{1}{h} & : t_k \leq t \leq t_{k+1} \\ 0 & : \text{annars} \end{cases} \quad k=1, 2, \dots, n-1$$

$$\phi_0(t) := \begin{cases} (t-t_0) \frac{1}{h} & : t_0 \leq t \leq t_1 \\ 0 & : \text{annars} \end{cases}$$

$$\phi_n(t) := \begin{cases} (t-t_{n-1}) \frac{1}{h} & : t_{n-1} \leq t \leq t_n \\ 0 & : \text{annars} \end{cases}$$



Vi får $\phi_i(t_j) = \delta_{ij}$ och $\phi_i \notin C^2([a, b])$ Vi, j

2) Antag att lösningen y till (*) ligger i V

$$\text{Alltså } y(t) = \sum_{k=0}^n c_k \phi_k(t)$$

$$\text{Vi får då } y_a = y(a) = y(t_0) = \sum_{k=0}^n c_k \phi_{k,0} = c_0 \Rightarrow c_0 = y_a$$

$$y_b = y(b) = y(t_n) = \sum_{k=0}^n c_k \phi_{k,n} = c_n \Rightarrow c_n = y_b$$

$$y(t_i) = c_i$$

3) Minimera $r(t; y)$ genom att välja c_0, \dots, c_n s.a. $r \perp \mathcal{B}$
dvs.

$$\langle r, \phi_i \rangle = \int_a^b r(t) \phi_i(t) dt = 0 \quad i=0, 1, 2, \dots, n$$

$$\Rightarrow \int_a^b y'' \phi_i dt = \int_a^b f \phi_i dt \quad (**)$$

4) Använd partiella derivator för att eliminera y'' i (**)

$$\int_a^b y'' \phi_i dt = [y'(t) \phi_i(t)]_a^b - \int_a^b y' \phi_i' dt$$

$$\text{Vilket ger: } \int_a^b f \phi_i + y' \phi_i' dt = [y' \phi_i]_a^b \quad i=0, \dots, n$$

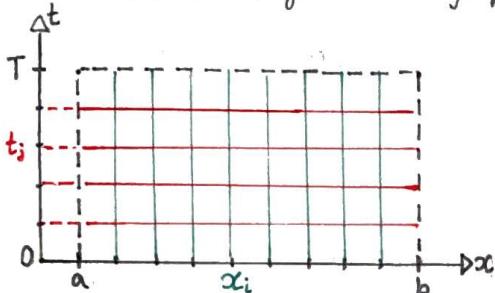
Lös för c_0, \dots, c_n med:

$$\int_a^b \phi_i(t) \phi_j(t) dt = \begin{cases} h/6 & : j=i+1 \text{ eller } j=i-1 \\ 2h/3 & : i=j \\ 0 & : \text{annars} \end{cases}$$

$$\int_a^b \phi_i'(t) \phi_j'(t) dt = \begin{cases} -1/h & : j=i+1 \text{ eller } j=i-1 \\ 2/h & : i=j \\ 0 & : \text{annars} \end{cases}$$

Metod (Framåt differensmetoden för värmelämningsekvationen): Värmelämningsekvationen
 Lyder: $\begin{cases} u_t = Du_{xx} & \text{för } u: [a,b] \times [0,T] \rightarrow \mathbb{R} \\ u(x,0) = f(x) \\ u(a,t) = l(t) \\ u(b,t) = r(t) \end{cases}$

Vi söker lösningar med hjälp av (finita) framåt differensmetoden:



Låt $u(x_i, t_j)$ beteckna den exakta lösningen med:

$$x_i = a + ih \quad \text{för steglängd } h = \frac{b-a}{n}$$

$$t_j = j \cdot k \quad \text{för steglängd } k = T/m$$

Låt u_{ij} beteckna den uppskattade lösningen i (x_i, t_j) d.v.s $u_{ij} \approx u(x_i, t_j)$

Vi använder finit differens:

$$u_{xx} \approx \frac{1}{h^2} (u(x+h, t) - 2u(x, t) + u(x-h, t))$$

$$u_t \approx \frac{1}{k} (u(x, t+k) - u(x, t))$$

Vi får då istället ekvationssystemet:

$$\begin{cases} \frac{D}{h^2} (u_{i+1,j} - 2u_{ij} + u_{i-1,j}) = \frac{1}{k} (u_{i,j+1} - u_{ij}) \\ u_{i,0} = f(x_i) \\ u_{0,j} = l(t_j) \\ u_{n,j} = r(t_j) \end{cases}$$

Om vi känner u_{ij} kan $u_{i,j+1}$ beräknas:

$$u_{i,j+1} = u_{i,j} + \frac{Dk}{h^2} (u_{i-1,j} - 2u_{ij} + u_{i+1,j})$$

$$\text{Vi får då } u_{i,j+1} = \sigma_{i-1,j} + (1-2\sigma)u_{ij} + \sigma u_{i+1,j} \quad \text{för } \sigma := \frac{Dk}{h^2}$$

Och på matrisform:

$$\begin{bmatrix} u_{1,j+1} \\ u_{2,j+1} \\ \vdots \\ u_{n-2,j+1} \\ u_{n-1,j+1} \end{bmatrix} = \begin{bmatrix} 1-2\sigma & \sigma & & & \\ \sigma & \ddots & \ddots & & \\ & \ddots & \ddots & \ddots & \\ & & \ddots & \ddots & \sigma \\ & & & \sigma & 1-2\sigma \end{bmatrix} \begin{bmatrix} u_{1,j} \\ u_{2,j} \\ \vdots \\ u_{n-2,j} \\ u_{n-1,j} \end{bmatrix} + \sigma \begin{bmatrix} u_{0,j} \\ 0 \\ \vdots \\ 0 \\ u_{n,j} \end{bmatrix}$$

Anmärkning: Metoden är explicit och har ordning 1 i tiden och 2 i rummet.

Metod (Bakåt differensmetoden för värmelämningsekvationen): Vi kan lösa problemet ovan med bakåt differensmetoden som är identisk med framåt differens men man ansätter istället: $u_t \approx \frac{1}{k} (u(x, t) - u(x, t-k))$

Vi får då ekvationerna $-\sigma u_{i+1,j} + (1+2\sigma)u_{ij} - \sigma u_{i-1,j} = u_{i,j-1}$ och på matrisform:

$$\begin{bmatrix} 1+2\sigma & -\sigma & & & \\ -\sigma & \ddots & \ddots & & \\ & \ddots & \ddots & \ddots & \\ & & \ddots & \ddots & -\sigma \\ & & & -\sigma & 1+2\sigma \end{bmatrix} \begin{bmatrix} u_{1,j} \\ u_{2,j} \\ \vdots \\ u_{n-2,j} \\ u_{n-1,j} \end{bmatrix} = \begin{bmatrix} u_{1,j-1} \\ u_{2,j-1} \\ \vdots \\ u_{n-2,j-1} \\ u_{n-1,j-1} \end{bmatrix} + \sigma \begin{bmatrix} u_{0,j} \\ 0 \\ \vdots \\ 0 \\ u_{n,j} \end{bmatrix}$$

Anmärkning: Vi behöver lösa ett linjärt ekvationssystem i varje tidssteg.
 Metoden är implicit med samma ordning som framåt, men är stabilare.

SATS 8.1: Låt $T = \begin{bmatrix} 1 & -1 & & \\ -1 & 1 & & \\ & & 1 & -1 \\ & & 0 & 1 \\ & & & -1 & 1 \end{bmatrix}$ vara en $n \times n$ matris. T har di egenvärden

$$\lambda_i = 1 - 2\cos\left(\frac{\pi i}{n+1}\right)$$

SATS 8.2: Antag att $n \times n$ matrisen M har egenvärde λ med tillhörande egenvektor v . Då gäller att $(\alpha M + \beta I)$ har egenvärde $\alpha\lambda + \beta$ men tillhörande egenvektor v , givet $\alpha, \beta \in \mathbb{R}$.

SATS 8.2: Låt h vara steglängden i rummet och k vara steglängden i tiden för framåt finitdifferensmetoden för värmeförädlingsekvationen med $D > 0$. Metoden är då stabil om $\frac{Dk}{h^2} < \frac{1}{2}$

SATS 8.3: Låt h vara steglängden i rummet och k vara steglängden i tiden för bakåt finitdifferensmetoden för värmeförädlingsekvationen med $D > 0$. Metoden är då oavhängigt stabil, det vill säga stabil för alla positiva h och k .

Metod (Crank-Nicolson metoden): Vi kan lösa värmeförädlingsekvationen

$$\begin{cases} u_t = Du_{xx} \\ u(x, 0) = f(x) \\ u(a, t) = l(t) \\ u(b, t) = r(t) \end{cases}$$

Med hjälp av Crank-Nicolson metoden som följer samma struktur som framåt finit differensmetod men man ansätter istället

$$u_t = \frac{i}{k}(u_{ij} - u_{i,j-1}) \text{ och } u_{xx} = \frac{1}{2} \left(\frac{u_{i+1,j} - 2u_{ij} + u_{i-1,j}}{h^2} \right) + \frac{1}{2} \left(\frac{u_{i+1,j-1} - 2u_{ij-1} + u_{i-1,j-1}}{h^2} \right)$$

Med $\sigma = Dk/h^2$ får vi då ekvationssystemet

$$2u_{ij} - 2u_{i,j-1} = \sigma(u_{i+1,j} - 2u_{ij} + u_{i-1,j} + u_{i+1,j-1} - 2u_{ij-1} + u_{i-1,j-1})$$

Alltså

$$-\sigma u_{i-1,j} + (2+2\sigma)u_{ij} - \sigma u_{i+1,j} = \sigma u_{i-1,j-1} + (2-2\sigma)u_{ij-1} + \sigma u_{i+1,j-1}$$

Med $u_j = [u_{1j}, \dots, u_{nj}]^T$ får vi på matrisform:

$$Au_j = Bu_{j-1} + \sigma(S_{j-1} + S_j)$$

där

$$A = \begin{bmatrix} 2+2\sigma & -\sigma & & & \\ -\sigma & 2-2\sigma & \sigma & & 0 \\ & 0 & -\sigma & 2+2\sigma & \\ & & & 0 & -\sigma \\ & & & & 2+2\sigma \end{bmatrix}; B = \begin{bmatrix} 2-2\sigma & \sigma & & & 0 \\ 0 & -\sigma & 2-2\sigma & \sigma & \\ & 0 & 0 & -\sigma & 2+2\sigma \\ & & & 0 & -\sigma \\ & & & & 2-2\sigma \end{bmatrix}; S_j = \begin{bmatrix} u_{0j} \\ 0 \\ \vdots \\ 0 \\ u_{nj} \end{bmatrix}$$

SATS 8.4: Crank-Nicolson metoden applicerad på värmeförädlingsekvationen med $D > 0$ är stabil för alla steglängder h, k .

SATS 8.5: Crank-Nicolson metoden applicerat på värmeförädlingsekvationen har ordning 2 i både tiden och rummet.

Def(dominerande egenvärde, dominerande egenvektor): Låt A vara en $m \times m$ -matris. Ett dominerande egenvärde av A är ett egenvärde λ vars absolutvärde är större än alla andra egenvärde till A . Om den existerar så kallas en egenvektor till det dominerande egenvärdet för en dominerande egenvektor.

SATS 12.1 (Rayleighkvoten): Låt λ vara ett egenvärde till matrisen A med egenvektor x . Vi får då

$$\lambda = \frac{x^T A x}{x^T x}$$

Anmärkning: Givet en approximativ egenvektor x ger Rayleighkvoten den bästa approximationen av det associerade egenvärdet. Denna approximation kallas **Rayleighmetoden**.

Metod (Potensmetoden): Antag att vi söker det dominerande egenvärdet λ och en associerad dominerande egenvektor u till $m \times m$ matrisen A . Vi kan då använda **potensmetoden**:

1) Ansätt en startgissning x_0 till u

2) För $j=1, 2, 3, \dots, k+1$:

$$u_{j-1} = x_{j-1} / \|x_{j-1}\|_2$$

$$x_j = A u_{j-1}$$

3) $u = u_k$

$$4) \lambda \approx \frac{u^T A u}{u^T u}$$

SATS 12.2: Låt A vara en $m \times m$ -matris med reella egenvärden $\lambda_1, \dots, \lambda_m$ som uppfyller $|\lambda_1| \geq |\lambda_2| \geq |\lambda_3| \geq \dots \geq |\lambda_m|$. Anta att egenvektorerna till A spänner upp \mathbb{R}^m . För nästan varje startgissning x_0 konvergerar potensmetoden linjärt mot en egenvektor associerad till λ_1 med en konstant faktor $S = |\lambda_2|/\lambda_1|$

Metod (Inversa potensmetoden): Antag att vi söker det egenvärde λ till A vars absolutvärde är närmast 0, och en associerad egenvektor u . Vi kan då använda **inversa potensmetoden**:

1) Ansätt en startgissning x_0 till u

2) För $j=1, 2, 3, \dots, k+1$:

$$u_{j-1} = x_{j-1} / \|x_{j-1}\|_2$$

Lös $A x_j = u_{j-1}$

3) $u = u_k$

$$4) \lambda \approx \frac{u^T A u}{u^T u}$$

Anmärkning: Om vi söker ett egenvärde till A som är närmast ett reellt tal s (och dess associerade egenvektor) kan man utföra inversa potensmetoden på matrisen $(A - sI)$, egenvärdet blir då $\lambda + s$ och egenvektorn är u .

Bevis:

SATS 1.11: Låt f vara två gånger deriverbar och $f(x^*) = 0$. Om $f'(x^*) \neq 0$ har Newtons metod kvadratisk konvergens lokalt.

Bevis:

$$\text{Låt } g(x) = x - \frac{f(x)}{f'(x)} \Rightarrow g(x^*) = x^*$$

$$\text{Vi får då } g'(x) = 1 - \frac{f'(x)^2 - f(x)f''(x)}{f'(x)^2} \Rightarrow g'(x^*) = 1 - 1 = 0 < 1$$

Detta visar lokal konvergens

$$0 = f(x^*) = f(x_n + (x^* - x_n)) = f(x_n) + f'(x_n)(x^* - x_n) + f''(\theta) \frac{(x^* - x_n)^2}{2}$$

$$\begin{aligned} e_{n+1} &= |x_{n+1} - x^*| = |g(x_n) - x^*| = |x_n - \frac{f(x_n)}{f'(x_n)} - x^*| \\ &= \left| x_n + \frac{f'(x_n)(x^* - x_n) + f''(\theta) \frac{(x^* - x_n)^2}{2}}{f'(x_n)} - x^* \right| \\ &= \underbrace{|(x^* - x_n)^2 \frac{f''(\theta)}{2f'(x_n)}|}_{e_n^2} \\ &\Rightarrow \frac{e_{n+1}}{e_n^2} = \left| \frac{f''(\theta)}{2f'(x_n)} \right| \xrightarrow{n \rightarrow \infty} \left| \frac{f''(x^*)}{2f'(x^*)} \right| \quad \text{då } \underset{\theta \in [x_n, x^*]}{x_n \rightarrow x^*} \text{ och } \theta \in [x_n, x^*]. \quad \text{Q.E.D.} \end{aligned}$$

SATS 1.6: Antag att g är kontinuerligt deriverbar och att $g(x^*) = x^*$ samt att $S = |g'(x^*)| < 1$. Då är fixpunktiteration lokalt konvergent med linjär konvergensordning

Bevis:

$$\begin{aligned} e_{n+1} &= |x_{n+1} - x^*| = |g(x_n) - x^*| = |g(x_n) - g(x^*)| = |g(x_n) - g(x_n + (x^* - x_n))| \\ &= |g(x_n) - g(x_n) - (x^* - x_n)g'(\theta)| \quad \theta \in [x_n, x^*] \\ &= \underbrace{|(x^* - x_n)g'(\theta)|}_{e_n} \\ &\Rightarrow \frac{e_{n+1}}{e_n} = |g'(\theta)| \xrightarrow{n \rightarrow \infty} |g'(x^*)| = S < 1 \quad \text{då } \theta \in [x_n, x^*] \quad \text{Q.E.D.} \end{aligned}$$

SATS 2.5: Konditionstalet för en lösning av ett linjärt ekvationssystem $Ax = b$, betecknat $\text{cond}(A)$, där b är indata och x är utdata, kallas även konditionstalet för matrisen A och beräknas som $\text{cond}(A) = \|A\| \cdot \|A^{-1}\|$.

Bevis: $\tilde{x} = x + \epsilon_x$ $\tilde{b} = b + \epsilon_b$ och $A\tilde{x} = \tilde{b} \Rightarrow \tilde{x} = A^{-1}(b + \epsilon_b) \Rightarrow x + \epsilon_x = x + A^{-1}\epsilon_b$

$$\epsilon_x = A^{-1}\epsilon_b$$

$$\begin{aligned} \text{cond}(A) &:= \max_{\epsilon_b, x \neq 0} \frac{\|\epsilon_x\| / \|x\|}{\|\epsilon_b\| / \|b\|} = \max_{\epsilon_b, x \neq 0} \frac{\|A^{-1}\epsilon_b\| / \|x\|}{\|\epsilon_b\| / \|A^{-1}x\|} = \max_{x \neq 0} \frac{\|A^{-1}\epsilon_b\|}{\|x\|} \cdot \max_{\epsilon_b \neq 0} \frac{\|\epsilon_b\|}{\|A^{-1}x\|} \\ &= \{ \text{per definition} \} = \|A\| \cdot \|A^{-1}\| \quad \text{Q.E.D.} \end{aligned}$$

SATS 3.2 (Huvudsatsen för polynominterpolation): Låt $(x_1, y_1), \dots, (x_n, y_n)$ vara n punkter med olika värden på x . Det finns ett unikt polynom av grad $\leq n-1$ som interpolerar datan.

Bewis:

Vi bevisar existens genom att definiera Lagrange-polynomet:

$$p(x) = \sum_{i=1}^n y_i \frac{\prod_{j \neq i} (x - x_j)}{\prod_{j \neq i} (x_i - x_j)}$$

vilket är ett polynom av grad $\leq n-1$ som interpolerar datan.

Unikhet: Antag p_1 och p_2 är två olika polynom av grad $\leq n-1$ som interpolerar datan $\Rightarrow p_1(x_i) = p_2(x_i) \quad \forall i=1, 2, \dots, n$

$$\text{Låt } Q := p_1 - p_2 \Rightarrow Q(x_i) = 0 \quad \forall i=1, 2, \dots, n$$

Alltså är x_i rötter till Q . Faktoriseringssatsen ger
 $Q(x) = \hat{Q}(x) \prod_{i=1}^n (x - x_i)$ där $\hat{Q}(x)$ är ett polynom

Detta är en motsägelse då $Q(x) = p_1(x) - p_2(x)$ är grad $\leq n-1$.

Alltså finns inte två olika polynom av grad $\leq n-1$ som interpolerar datan

Q.E.D.

SATS 5.Ω: Derivatan av reellvärda funktionen f kan approximeras med finit framtidifferens som

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

och har noggrannhetsordning 1.

Bewis:

$$\text{Låt } u_h(x) = \frac{f(x+h) - f(x)}{h}$$

$$\text{Vi får felet } e_h = |f'(x) - u_h(x)| = \left| f'(x) - \frac{f(x+h) - f(x)}{h} \right|$$

$$\{\text{Taylor}\} = \left| f'(x) - \frac{f(x) + h f'(x) + \frac{h^2}{2} f''(c) - f(x)}{h} \right| \quad c \in [x, x+h]$$

$$= \left| \frac{h}{2} f''(c) \right| \leq Ch \quad \text{för något } C \geq \left| \frac{f''(c)}{2} \right|$$

Alltså har vi noggrannhetsordning 1.

Q.E.D.

SATS 5.Γ: Sammansatta trapetsregeln har noggrannhetsordning 2.

Bewis:

Vi vill lösa $\int_a^b f(x) dx$. Vi delar upp intervallet $[a, b]$ i n

delintervall $a = x_0 < x_1 < \dots < x_n = b : x_i - x_{i-1} = h$

och får

$$\int_a^b f(x) dx = \sum_{i=1}^n \int_{x_{i-1}}^{x_i} f(x) dx$$

Beweis forts.

Vi betraktar $\int_{x_{i-1}}^{x_i} f(x) dx$. Vi interpolerar $f(x)$ i punkterna $(x_{i-1}, f(x_{i-1}))$ och $(x_i, f(x_i))$:

$$f(x) = f(x_{i-1}) \frac{x - x_{i-1}}{x_i - x_{i-1}} + f(x_i) \frac{x - x_i}{x_i - x_{i-1}} + E(x)$$

$$\text{Med interpoleringsfelet } E(x) = \frac{(x - x_{i-1})(x - x_i)}{2} f''(c(x))$$

$$\text{Vi får } \int_{x_{i-1}}^{x_i} f(x) dx = \frac{f(x_{i-1})}{x_i - x_{i-1}} \int_{x_{i-1}}^{x_i} (x - x_{i-1}) dx + \frac{f(x_i)}{x_i - x_{i-1}} \int_{x_{i-1}}^{x_i} (x - x_i) dx + \int_{x_{i-1}}^{x_i} E(x) dx$$

$$\{x_i - x_{i-1} = h\} = \frac{h}{2} (f(x_i) + f(x_{i-1})) + \int_{x_{i-1}}^{x_i} \frac{(x - x_{i-1})(x - x_i)}{2} f''(c(x)) dx$$

{Medelvärdessats för integrator}

$$\begin{aligned} &= \frac{h}{2} (f(x_i) + f(x_{i-1})) + f''(c_i) \int_{x_{i-1}}^{x_i} \frac{(x - x_{i-1})(x - x_i)}{2} dx \\ &= \frac{h}{2} (f(x_i) + f(x_{i-1})) - \frac{f''(c_i)}{12} h^3 \end{aligned}$$

$$\text{Vi får då: } \int_a^b f(x) dx = \sum_{i=1}^n \frac{h}{2} (f(x_i) + f(x_{i-1})) - \sum_{i=1}^n \frac{f''(c_i)}{12} h^3$$

{Medelvärdessats för summa}

$$\begin{aligned} &= \sum_{i=1}^n \frac{h}{2} (f(x_i) + f(x_{i-1})) - \frac{f''(c)}{12} h^2 \sum_{i=1}^n h \\ &= \sum_{i=1}^n \frac{h}{2} (f(x_i) + f(x_{i-1})) - (b-a) \frac{f''(c)}{12} h^2 \end{aligned}$$

Fellet för sammansatta trapetsmetoden:

$$e_h = \left| \int_a^b f(x) dx - \sum_{i=1}^n \frac{h}{2} (f(x_i) + f(x_{i-1})) \right| = (b-a) \frac{f''(c)}{12} h^2 \leq Ch^2$$

$$\text{för något } C \geq (b-a) \frac{f''(c)}{12}$$

Q.E.D.

SATS 6.Ω: Lokala felet $e_k = |y_k - z(t_k)|$ där $\begin{cases} z'(t) = f(t, z) \\ z(t_{k-1}) = y_{k-1} \\ z(t_0) = y_0 \end{cases}$ och $\begin{cases} y'(t) = f(t, y) \\ y(t_0) = y_0 \end{cases}$ för Eulers metod uppfyller $e_{k+1} \leq \frac{h^2}{2} C$

Beweis:

$$\begin{aligned} e_{k+1} &= |y_{k+1} - z(t_{k+1})| = |y_{k+1} - z(t_k + (t_{k+1} - t_k))| \\ &= \left| y_k + hf(t_k, y_k) - \underbrace{z(t_k)}_{y_k} + \underbrace{(t_{k+1} - t_k)}_h \underbrace{z'(t_k)}_{f(t_k, z_k)} + \frac{(t_{k+1} - t_k)^2}{2} z''(\theta_k) \right| \quad \theta_k \in [t_k, t_{k+1}] \\ &= \frac{h^2}{2} |z''(\theta_k)| \quad \text{anta det finns ett } C \text{ s.a. } C \geq |z''(\theta_k)| \end{aligned}$$

$$\Rightarrow e_{k+1} \leq C \frac{h^2}{2}$$

SATS 6.1: Låt $g_k = |y(t_k) - y_k|$ vara det globala felet för Eulers metod på problemet

$$\begin{cases} y' = f(t, y) \\ y(a) = y_a \\ t \in [a, b] \end{cases}$$

där z löser

$$\begin{cases} z' = f(t, z) \\ z(t_{k-1}) = y_{k-1} \\ t \in [t_{k-1}, t_k] \end{cases}$$

$$\text{Vi har då } g_k \leq e_k + e^{Lh} e_{k-1} + e^{2Lh} e_{k-2} + \dots + e^{(k-1)Lh} e_1 \leq \frac{C}{L} h^p (e^{Lhk} - 1)$$

Hjälpsats 6.3: Antag att f är Lipschitz i y för $t \in [a, b]$. Om $Y(t)$ och $Z(t)$ löser $y' = f(t, y)$ så gäller att

$$|Y(t) - Z(t)| \leq e^{L(t-a)} |Y(a) - Z(a)|$$

Beweis: $g_k = |y_k - y(t_k)| = |y_k - z(t_k) + z(t_k) - y(t_k)| \leq \underbrace{|y_k - z(t_k)|}_{e_k} + |z(t_k) - y(t_k)|$

{Sats 6.3. $a = t_{k-1}$ }

$$\leq e_k + e^{L(t_k - t_{k-1})} |z(t_{k-1}) - y(t_{k-1})| = e_k + e^{Lh} |y_{k-1} - y(t_{k-1})| = e_k + e^{Lh} g_{k-1}$$

{pi samma sätt}

$$\leq e_k + e^{Lh} (e_{k-1} + e^{Lh} g_{k-2}) \leq \dots \leq e_k + e^{Lh} (e_{k-1} + e^{Lh} (e_{k-2} + e^{Lh} (\dots + e^{Lh} g_1, \dots)))$$

$$\leq e_k + e^{Lh} e_{k-1} + e^{2Lh} e_{k-2} + \dots + e^{(k-1)Lh} g_1$$

$$g_1 = |y_1 - y(t_1)| = |y_1 - z(t_1)| = e_1$$

$$\Rightarrow g_k \leq e_k + e^{Lh} e_{k-1} + e^{2Lh} e_{k-2} + \dots + e^{(k-1)Lh} e_1$$

Antag $e_i \leq Ch^{p+1}$

$$\leq Ch^{p+1} [e^{Lh} + e^{2Lh} + \dots + e^{(k-1)Lh}] = Ch^{p+1} \frac{e^{Lhk} - 1}{e^{Lh} - 1} \leq Ch^{p+1} \frac{e^{Lhk} - 1}{Lh}$$

$$\Rightarrow g_k \leq \frac{C}{L} h^p (e^{Lhk} - 1)$$

SATS 8.2: Framt finitdifferensmetoden för värmeförädlingsekvationen

$$\begin{cases} u_t = Du_{xx} \\ u(x, 0) = f(x) \\ u(a, t) = l(t) \\ u(b, t) = r(t) \end{cases}$$

är stabil om $\frac{Dk}{h^2} < \frac{1}{2}$ där k är steg längden i tiden och h är steg längden i rummet

Beweis: Vi har $u_{n+1} = Au_n + \sigma V_n$: $\sigma = \frac{Dk}{h^2}$

där $A = \begin{bmatrix} 1-2\sigma & \sigma & & 0 \\ \sigma & \ddots & \ddots & 0 \\ & \ddots & \ddots & \sigma \\ 0 & \ddots & \ddots & 1-2\sigma \end{bmatrix} = (1-\sigma)I - \sigma T$

där $T = \begin{bmatrix} 1 & -1 & & 0 \\ -1 & \ddots & \ddots & 0 \\ & \ddots & \ddots & -1 \\ 0 & \ddots & -1 & 1 \end{bmatrix}$ som har egenvärden på formen $1 - 2 \cos \theta_k$

Bevis forts.

Låt $\tilde{u}_0 = u_0 + \varepsilon_0$ där ε_0 är ett litet fel

$$\Rightarrow \varepsilon_{n+1} = \tilde{u}_{n+1} - u_{n+1} = A\tilde{u}_n + \sigma v_n - (A\tilde{u}_n + \sigma v_n) = A(\tilde{u}_n - u_n)$$

$$= A^2(\tilde{u}_{n-1} - \tilde{u}_n) = \dots = A^{n+1}\varepsilon_0$$

Lösningen är stabil om $\varepsilon_n \rightarrow 0$ då $n \rightarrow \infty \Rightarrow A^n \rightarrow 0$

A är symmetrisk $\Rightarrow A$ är diagonalisierbar

$$\Rightarrow A^n = P^\top \Lambda^n P \quad \text{Alltså} \quad \lim_{n \rightarrow \infty} A^n = 0 \Rightarrow \lim_{n \rightarrow \infty} \lambda_i^n = 0 \quad \forall i=1,2,\dots,m.$$

$$\text{då } \Lambda^n = \begin{bmatrix} \lambda_1^n & & & \\ & \lambda_2^n & & 0 \\ & & \ddots & \\ 0 & & & \lambda_m^n \end{bmatrix} \Rightarrow |\lambda_i| < 1 \quad \forall i=1,2,\dots,m$$

$$\text{Då } A = (1-\sigma)I - \sigma T \text{ har } A \text{ egenvärden } \lambda_k = (1-\sigma) - \sigma(1 - 2\cos(\theta_k)) \in [-1, 1]$$
$$1 - \lambda_1 \sigma < \lambda_k < 1 \quad |\lambda_k| < 1 \Rightarrow \sigma < \frac{1}{2} \Rightarrow \frac{Dk}{h^2} < \frac{1}{2} \quad \text{Q.E.D.}$$

SATS 12.2: Låt A vara en $m \times m$ -matris med reella egenvärden $\lambda_1, \dots, \lambda_m$ som uppfyller $|\lambda_1| > |\lambda_2| \geq \dots \geq |\lambda_m|$. Antag vidare att egenvektorerna till A spänner upp \mathbb{R}^m .

För nästan alla startvektorer kommer potensmetoden konvergera linjärt till en egenvektor associerad till λ_1 med en konstant faktor $S = \lambda_2/\lambda_1$.

Bevis:

Låt v_1, \dots, v_m vara egenvektorer med respektive egenvärden $\lambda_1, \dots, \lambda_m$ och som spänner upp \mathbb{R}^m . Bestäm startvektor $x_0 = c_1 v_1 + c_2 v_2 + \dots + c_m v_m$.

Då detta ska gälla för "nästan" alla x_0 ansätter vi $c_1 \neq 0$ och $c_2 \neq 0$

Applicering av potensmetoden ger:

$$Ax_0 = c_1 \lambda_1 v_1 + c_2 \lambda_2 v_2 + \dots + c_m \lambda_m v_m$$

$$A^2x_0 = c_1 \lambda_1^2 v_1 + c_2 \lambda_2^2 v_2 + \dots + c_m \lambda_m^2 v_m$$

$$A^3x_0 = c_1 \lambda_1^3 v_1 + c_2 \lambda_2^3 v_2 + \dots + c_m \lambda_m^3 v_m$$

med normering i varje steg. Då antalet steg $k \rightarrow \infty$ kommer första termen i högerled dominera då

$$\frac{A^k x_0}{\lambda_1^k} = c_1 v_1 + c_2 \left(\frac{\lambda_2}{\lambda_1}\right)^k v_2 + \dots + c_m \left(\frac{\lambda_m}{\lambda_1}\right)^k v_m$$

Antagandet att $|\lambda_1| > |\lambda_i|$ för $i > 1$ gör att alla utom första termen i högerled kommer konvergera mot 0 med faktor $S \leq \lambda_2/\lambda_1$, och exakt den faktorn då $c_2 \neq 0$. Det följer att metoden konvergerar mot en multipel av den dominanta egenvektorn v_1 , med egenvärde λ_1 .