Numerik für Informatiker

geTEXt von Andreas Siemer

3. August 2000

Zusammenfassung

Mitschrieb zur Vorlesung *Numerische Mathematik für die Fachrichtungen Informatik und Ingenieurwesen*, gehalten von Prof. Dr. A. Rieder an der Universität Karlsruhe (SS 2000).
Bem.: Die Numerierung der Sätze und Gleichungen entspricht nicht unbedingt der Vorlesung, da hier oft Verdoppelungen, Auslassungen oder auch verschiedene Bezeichnungen auftraten.

Inhaltsverzeichnis

1 Normen

Vektornormen

Eine *Vektornorm* ist eine Funktion $\|\cdot\| : \mathbb{C}^n \to [0,\infty[$ mit

1. $\|x\| = 0 \Leftrightarrow x = 0$
2. $\|\alpha \cdot x\| = |\alpha| \cdot \|x\| \; \forall \alpha \in \mathbb{C}$
3. $\|x+y\| \leq \|x\| + \|y\|$

Beispiel:

$$p\text{-Normen: } \|x\|_p := (\sum_{i=1}^{n} |x_i|^p)^{\frac{1}{p}}, 1 \leq p < \infty$$
$$\text{Euklidische Norm: } \|\cdot\|_2$$
$$\text{Maximumsnorm: } \|\cdot\|_\infty,\ \|x\|_\infty = \max_{1\leq i\leq n} |x_i|$$

Satz 1.1. *Zu jedem Paar* $\|\cdot\|$, $|||\cdot|||$ *von Normen existieren Konstanten* $0 < m \leq M < \infty$, *so daß*

$$m \cdot \|x\| \leq |||x||| \leq M \cdot \|x\|$$

("Vektornormen sind äquivalent")

Beispiel:

$$\|x\|_q \leq \|x\|_p \leq n^{(\frac{1}{p}-\frac{1}{q})}\|x\|_q,\ p \leq q$$

Matrixnormen

Eine *Matrixnorm* ist eine Abbildung

$$\|\cdot\| : \mathbb{C}^{n\times m} \to [0,\infty[$$

mit den drei Eigenschaften einer Vektornorm, wenn sie zusätzlich *submultiplikativ* ist, d.h.

$$\|A \cdot B\| \leq \|A\| \cdot \|B\| \; \forall A \in \mathbb{C}^{n\times r},\ B \in \mathbb{C}^{r\times m}$$

Eine Matrixnorm heißt *verträglich mit den Vektornormen* $\|\cdot\|_{\mathbb{C}^n}$ und $\|\cdot\|_{\mathbb{C}^m}$ falls gilt:

$$\|A \cdot x\|_{\mathbb{C}^n} \leq \|A\| \cdot \|x\|_{\mathbb{C}^m}$$

Beispiel: a) Zeilensummennorm:

$$\|A\| = \max_i \sum_{k=1}^{m} |a_{i,k}|$$

b) Spaltensummennorm:

$$\|A\| = \max_k \sum_{i=1}^{n} |a_{i,k}|$$

c) Frobeniusnorm:
$$\|A\| = (\sum_{i=1}^{n}\sum_{k=1}^{m}|a_{i,k}|^2)^{\frac{1}{2}}$$

d) *keine* Matrixnorm:
$$\|A\| = \max_{i,k}|a_{i,k}|$$

denn:
$$2 = \| \begin{pmatrix}1 & 1\\ 0 & 0\end{pmatrix} \cdot \begin{pmatrix}1 & 0\\ 1 & 0\end{pmatrix} \| \not\leq \| \begin{pmatrix}1 & 1\\ 0 & 0\end{pmatrix} \| \cdot \| \begin{pmatrix}1 & 0\\ 1 & 0\end{pmatrix} \| = 1$$

Vektornormen $\|\cdot\|_{\mathbb{C}^n}$, $\|\cdot\|_{\mathbb{C}^m}$ induzieren eine Matrixnorm auf $\mathbb{C}^{n\times m}$:
$$\|A\| := \max_{x\neq 0}\frac{\|Ax\|_{\mathbb{C}^n}}{\|x\|_{\mathbb{C}^m}} = \max_{\|x\|_{\mathbb{C}^m}=1}\|Ax\|_{\mathbb{C}^n}$$

Solche Matrixnormen heißen *zugeordnet* oder *induziert.*

Beispiel: p-Norm:
$$\|A\|_p := \max_{\|x\|_p=1}\|A\cdot x\|_p$$

- $p = 1:\ A = (a_1, \dots, a_m),\ a_i \in \mathbb{C}^n$
$$\|Ax\|_1 = \|\sum_{j=1}^{m} a_j x_j\|_1 \leq \sum_{j=1}^{m}|x_j|\cdot\|a_j\|_1 \leq \max_{1\leq j\leq m}\|a_j\|_1\cdot\|x\|_1$$
  Die Ungleichung ist scharf. $\Rightarrow \|\cdot\|_1$ ist die Spaltensummennorm.
- $p = \infty:\ \|\cdot\|_\infty$ ist die Zeilensummennorm
- $p = 2:\ \|Ax\|_2^2 = \langle Ax, Ax\rangle = \langle A^*Ax, x\rangle$
  Es gilt: $\|A\|_2 = \lambda_{\max}(A^*A)^{\frac{1}{2}}$ (größter Eigenwert).
  $\|\cdot\|_2$ heißt *Spektralnorm* mit den Eigenschaften:
  - $\|A\|_2 = \|A^*\|_2$
  - $\|A^*A\|_2 = \|A\|_2^2$
  - $\|Q\cdot A\|_2 = \|A\|_2\ \forall Q$ unitär $(Q^*Q = I)$

Konditionszahl κ

Sei A eine quadratische Matrix, $A \in \mathbb{C}^{n\times n}$ regulär, d.h. A^{-1} existiert.

$\kappa(A) = \|A\|\cdot\|A^{-1}\|$ heißt *Konditionszahl von A bezüglich* $\|\cdot\|$

Falls $\|\cdot\|$ induziert: $1 = \|I\| = \|A\cdot A^{-1}\| \leq \|A\|\cdot\|A^{-1}\| = \kappa(A)$
$$\kappa_p(A) = \|A\|_p\cdot\|A^{-1}\|_p,\ 1\leq p<\infty$$

Satz 1.2. *Sei $A \in \mathbb{C}^{n\times n}$ regulär. Dann gilt:*
$$\min\left\{\frac{\|A-S\|_p}{\|A\|_p}\,\middle|\, S\in\mathbb{C}^{n\times n}\ \textit{singulär}\right\} = \frac{1}{\kappa_p(A)}$$

Beispiel:
$$A = \begin{pmatrix}2 & 1\\ 1 & 0.4999\end{pmatrix},\ \kappa_2(A)\approx 31250$$

2 Lineare Gleichungssysteme: direkte Löser

Gegeben:

$$A = \{a_{i,j}\} \in \mathbb{R}^{n\times n} \text{ regulär, } b \in \mathbb{R}^n$$

Gesucht:

$$x \in \mathbb{R}^n \, : \, Ax = b \Leftrightarrow \sum_{i=1}^{n} a_{j,i}x_i = b_j, \, j = 1, \dots n \tag{2.1}$$

2.1 Auflösung gestaffelter Systeme

$Rx = z$ mit $r_{i,j} = 0$ für $i > j$ (R ist obere Δ-Matrix)

$$\begin{array}{ccccccccc}
r_{1,1}x_1 & + & r_{1,2}x_2 & + & \cdots & + & r_{1,n}x_n & = & z_1 \\
 & & r_{2,2}x_2 & + & \cdots & + & r_{2,n}x_n & = & z_2 \\
 & & & & \ddots & & \vdots & & \vdots \\
 & & & & & & r_{n,n}x_n & = & z_n
\end{array}$$

$$\begin{aligned}
\Rightarrow \det R = \prod_{i=1}^{n} r_{i,i} &\neq 0 \\
\Leftrightarrow r_{i,i} &\neq 0
\end{aligned}$$

Rückwärtssubstitution

$$\begin{aligned}
x_n &= \frac{z_n}{r_{n,n}} \\
x_{n-1} &= (z_{n-1} - r_{n-1,n}x_n)/r_{n-1,n-1} \\
&\vdots \\
x_i &= (z_i - r_{i,i+1}x_{i+1} - \cdots - r_{i,n}x_n)/r_{i,i} \\
&\vdots \\
x_1 &= (z_1 - r_{1,2}x_2 - \cdots - r_{1,n}x_n)/r_{1,1}
\end{aligned}$$

Aufwand: Berechnung von x_i benötigt $(n - i)$ Additionen und $(n - i)$ Multiplikationen sowie eine Division.

$$\Rightarrow \sum_{i=1}^{n} (2(n-1)+1) = n^2 \quad \text{Operationen}$$

Das Auflösen eines unteren Δ-Systems $Lx = z$ mit $l_{i,j} = 0$, $i < j$ verläuft analog und heißt *Vorwärtssubstitution.*

2.2 Gaußscher Algorithmus, LR-Zerlegung

Idee: Führe (2.1) über in Δ-System, das dieselbe Lösung besitzt.
Zulässige Umformungen von $(A, b) \in \mathbb{R}^{n\times(n+1)}$:

- Vertauschen von Zeilen/Spalten (Umnumerierung der Unbekannten merken) in A
- Multiplikation einer Zeile mit Zahl $\neq 0$
- Addition des Vielfachen einer Zeile zu einer anderen

$$\left(\begin{array}{cccc|c} a_{1,1} & a_{1,2} & \cdots & a_{1,n} & b_1 \\ a_{2,1} & a_{2,2} & \cdots & a_{2,n} & b_2 \\ \vdots & \vdots & \ddots & \vdots & \vdots \\ a_{n,1} & \cdots & \cdots & a_{n,n} & b_n \end{array}\right) = (A,b) = (A^{(1)}, b^{(1)})$$

1. Schritt

Produziere Nullen unterhalb von $a_{1,1}$. Es sei $a_{1,1} \neq 0$. Subtrahiere das $l_{j,1} = \frac{a_{j,1}}{a_{1,1}}$-fache der 1. Zeile von der j-ten Zeile.

$$\begin{aligned} a_{j,i}^{(2)} &= a_{j,i}^{(1)} - l_{j,1} a_{1,i} \\ b_j^{(2)} &= b_j^{(1)} - l_{j,1} b_j^{(1)} \\ i, j &= 2, \ldots, n \end{aligned}$$

$$\left(\begin{array}{cccc|c} a_{1,1}^{(1)} & a_{2,1}^{(1)} & \cdots & a_{1,n}^{(1)} & b_1^{(1)} \\ 0 & a_{2,2}^{(2)} & \cdots & a_{2,n}^{(2)} & b_2^{(2)} \\ \vdots & \vdots & \ddots & \vdots & \vdots \\ 0 & a_{n,2}^{(2)} & \cdots & a_{n,n}^{(2)} & b_n^{(2)} \end{array}\right) = (A^{(2)}, b^{(2)})$$

$A^{(2)} = L_1 A^{(1)}$, $b^{(2)} = L_1 b^{(1)}$ mit

$$L_1 = \begin{pmatrix} 1 & & & \\ -l_{2,1} & \ddots & & 0 \\ -l_{3,1} & & \ddots & \\ \vdots & 0 & & \ddots \\ -l_{n,1} & & & & 1 \end{pmatrix}$$

k-ter Schritt $(2 \leq k \leq n-1)$

$$(A^{(k)}, b^{(k)}) \to (A^{(k+1)}, b^{(k+1)})$$

$$\left(\begin{array}{cccccc|c} a_{1,1}^{(1)} & \cdots & \cdots & \cdots & \cdots & a_{1,n}^{(1)} & b_1^{(1)} \\ 0 & a_{2,2}^{(2)} & \cdots & \cdots & \cdots & a_{2,n}^{(2)} & b_2^{(2)} \\ \vdots & 0 & \ddots & & & \vdots & \vdots \\ \vdots & \vdots & & a_{k,k}^{(k)} & \cdots & a_{k,n}^{(k)} & b_k^{(k)} \\ & \vdots & & \vdots & & \vdots & \vdots \\ & & & a_{n,k}^{(k)} & \cdots & a_{n,n}^{(k)} & b_n^{(k)} \end{array}\right) = (A^{(k)}, b^{(k)}) \qquad (2.2)$$

Sei $a_{k,k}^{(k)} \neq 0$:

$$\begin{aligned} l_{j,k} &= a_{j,k}^{(k)}/a_{k,k}^{(k)},\ j = k+1,\dots,n \\ a_{j,i}^{(k+1)} &= a_{j,1}^{(k)} - l_{j,k}a_{k,i}^{(k)},\ k+1 \leq i,\ j \leq n \\ b_j^{(k+1)} &= b_j^{(k)} - l_{j,k}b_k^{(k)} \end{aligned}$$

$a_{k,k}^{(k)}$ heißt Pivot-Element.

$$A^{(k+1)} = L_k A^{(k)},\ b^{(k+1)} = L_k b^{(k)}$$

$$L_k = \begin{pmatrix} 1 & & & \\ 0 & 1 & & 0 \\ \vdots & -l_{k,k+1} & \ddots & \\ & \vdots & & \\ & -l_{n,k+1} & 0 & 1 \end{pmatrix} = I - l_k \cdot e_k^t$$

mit

$$l_k = \begin{pmatrix} 0_1 \\ \vdots \\ 0_k \\ l_{k,k+1} \\ \vdots \\ l_{n,k+1} \end{pmatrix},\ e_k = \begin{pmatrix} 0_1 \\ \vdots \\ 0_{k-1} \\ 1_k \\ 0_{k+1} \\ \vdots \\ 0_n \end{pmatrix}$$

Falls $a_{k,k}^{(k)} \neq 0$ für $k = 1,\dots,n-1$, dann

$$R = A^{(n)} = L_{n-1}A^{(n-1)} = L_{n-1}L_{n-2}\cdots L_1 A$$

$$\Rightarrow A = L \cdot R \text{ mit } L = L_1^{-1}\cdots L_{n-1}^{-1}$$

Behauptung:

$$L_k^{-1} = I + l_k e_k^t$$

Beweis:

$$L_k(I + l_k e_k^t) = I - l_k \underbrace{e_k^t l_k}_{=0} e_k = I \quad \checkmark$$

$$\begin{aligned} \Rightarrow L &= (I + l_1 e_1^t)(I + l_2 e_2^t)\cdots(I + l_{n-1}e_{n-1}^t) \\ &= I + l_1 e_1^t + \cdots + l_{n-1}e_{n-1}^t \\ &= \begin{pmatrix} 1 & & & & \\ l_{2,1} & 1 & & 0 & \\ \vdots & l_{3,2} & \ddots & & \\ \vdots & \vdots & \ddots & 1 & \\ l_{n,1} & l_{n,2} & \cdots & l_{n,n-1} & 1 \end{pmatrix} \end{aligned}$$

Definition 2.1. Die Zerlegung einer Matrix $A = LR$ in eine obere Δ-Matrix R und eine untere Δ-Matrix L, die nur Einsen auf der Diagonalen hat, heißt *Gaußsche Dreieckszerlegung* oder *LR-Zerlegung*. (engl.: LU-Decomposition, lower-upper)

Beispiel:

$$A = \begin{pmatrix} 2 & 1 & 1 \\ 4 & 3 & 3 \\ 8 & 7 & 9 \end{pmatrix}$$

$$\underbrace{\begin{pmatrix} 1 & 0 & 0 \\ -2 & 1 & 0 \\ -4 & 0 & 1 \end{pmatrix}}_{=L_1} \underbrace{\begin{pmatrix} 2 & 1 & 1 \\ 4 & 3 & 3 \\ 8 & 7 & 9 \end{pmatrix}}_{=A^{(1)}} = \begin{pmatrix} 2 & 1 & 1 \\ 0 & 1 & 1 \\ 0 & 3 & 5 \end{pmatrix} = A^{(2)}$$

$$\begin{pmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & -3 & 1 \end{pmatrix} \begin{pmatrix} 2 & 1 & 1 \\ 0 & 1 & 1 \\ 0 & 3 & 5 \end{pmatrix} = \begin{pmatrix} 2 & 1 & 1 \\ 0 & 1 & 1 \\ 0 & 0 & 2 \end{pmatrix} = A^{(3)} = R$$

$$L = L_1^{-1} \cdot L_2^{-1} = \begin{pmatrix} 1 & 0 & 0 \\ 2 & 1 & 0 \\ 4 & 3 & 1 \end{pmatrix}$$

$$\Rightarrow A = \begin{pmatrix} 1 & 0 & 0 \\ 2 & 1 & 0 \\ 4 & 3 & 1 \end{pmatrix} \cdot \begin{pmatrix} 2 & 1 & 1 \\ 0 & 1 & 1 \\ 0 & 0 & 2 \end{pmatrix}$$

Gauß-Elimination zur Lösung von (2.1)

a) $A = LR$ (LR-Zerlegung)

b) $Lz = b$ (Vorwärtssubstitution)

c) $Rx = z$ (Rückwärtssubstitution)

$\sharp$ Multiplikationen LR:

$$\sum_{k=1}^{n-1} [(n-k) + (n-k)^2] = \tfrac{1}{3}(n^3 - n)$$

$\sharp$ Multiplikationen für Gauß-Elimination:

$$\tfrac{1}{3}n^3 + n^2 - \tfrac{1}{2}n$$

Pivot-Strategie

$$A = \begin{pmatrix} \delta & 1 \\ 1 & 1 \end{pmatrix},\ 0 < \delta \ll 1$$

$$\Rightarrow L = \begin{pmatrix} 1 & 0 \\ \delta^{-1} & 1 \end{pmatrix},\ R = \begin{pmatrix} \delta & 1 \\ 0 & 1 - \delta^{-1} \end{pmatrix}$$

Zahlendarstellung auf dem Rechner: δ^{-1} sei darstellbar.

$$\hat{L} = \begin{pmatrix} 1 & 0 \\ \delta^{-1} & 1 \end{pmatrix}, \hat{R} = \begin{pmatrix} \delta & 1 \\ 0 & -\delta^{-1} \end{pmatrix}$$

$$\Rightarrow \|\hat{L}\hat{R} - A\|_\infty = \tfrac{1}{2}\|A\|_\infty, \text{ relativer Fehler } 50\%!$$

Grund: LR-Zerlegung ist instabil
Ausweg Spaltenpivotsuche: Vertausche die Zeilen von A:

$$PA = \begin{pmatrix} 0 & 1 \\ 1 & 0 \end{pmatrix} \begin{pmatrix} \delta & 1 \\ 1 & 1 \end{pmatrix} = \begin{pmatrix} 1 & 1 \\ \delta & 1 \end{pmatrix}$$

$$PA = LR = \begin{pmatrix} 1 & 0 \\ \delta & 1 \end{pmatrix} \begin{pmatrix} 1 & 1 \\ 0 & 1-\delta \end{pmatrix}$$

$$\hat{L} = L$$

$$\hat{R} = \begin{pmatrix} 1 & 1 \\ 0 & 1 \end{pmatrix}$$

$$\Rightarrow \|PA - \hat{L}\hat{R}\|_\infty = \frac{\delta}{2}\|PA\|_\infty$$

LR-Zerlegung mit Spaltenpivotsuche

Vertausche im k-ten Schritt (2.2) die k-te Zeile mit der Zeile $j \geq k$, für die gilt:

$$|a_{j,k}^{(k)}| = \max_{k \leq i \leq n} |a_{i,k}^{(k)}|$$

Satz 2.2. *Der Gauß-Algorithmus mit Spaltenpivotsuche liefert für jede reguläre Matrix A eine Permutationsmatrix P sowie Δ-Matrizen L und R, so daß $PA = LR$ die LR-Zerlegung von PA ist.Die Elemente von L sind betragsmäßig ≤ 1.*

Beispiel:

$$\begin{pmatrix} 2 & 1 & 1 \\ 4 & 3 & 3 \\ 8 & 7 & 9 \end{pmatrix}$$

$$\xrightarrow[(3,2,1)]{\text{Pivot}} \begin{pmatrix} 8 & 7 & 9 \\ 4 & 3 & 3 \\ 2 & 1 & 1 \end{pmatrix} \xrightarrow{\text{El.}} \begin{pmatrix} 8 & 7 & 9 \\ \frac{1}{2} & -\frac{1}{2} & -\frac{3}{2} \\ \frac{1}{4} & -\frac{3}{4} & -\frac{5}{4} \end{pmatrix}$$

$$\xrightarrow[(3,1,2)]{\text{Pivot}} \begin{pmatrix} 8 & 7 & 9 \\ \frac{1}{4} & -\frac{3}{4} & -\frac{5}{4} \\ \frac{1}{2} & -\frac{1}{2} & -\frac{3}{2} \end{pmatrix} \xrightarrow{\text{El.}} \begin{pmatrix} 8 & 7 & 9 \\ \frac{1}{4} & -\frac{3}{4} & -\frac{5}{4} \\ \frac{1}{2} & \frac{2}{3} & -\frac{2}{3} \end{pmatrix}$$

$$\Rightarrow \begin{pmatrix} 0 & 0 & 1 \\ 1 & 0 & 0 \\ 0 & 1 & 0 \end{pmatrix} \begin{pmatrix} 2 & 1 & 1 \\ 4 & 3 & 3 \\ 8 & 7 & 9 \end{pmatrix} = \begin{pmatrix} 1 & 0 & 0 \\ \frac{1}{4} & 1 & 0 \\ \frac{1}{2} & \frac{2}{3} & 1 \end{pmatrix} \begin{pmatrix} 8 & 7 & 9 \\ 0 & -\frac{3}{4} & -\frac{5}{4} \\ 0 & 0 & -\frac{2}{3} \end{pmatrix}$$

2.3 Cholesky-Zerlegung

$A \in \mathbb{R}^{n\times n}$ heißt positiv definit ($A > 0$), falls $A = A^t$ und $\langle Ax, x\rangle > 0\ \forall x \in \mathbb{R}^n \backslash \{0\}$.

Satz 2.3. *$A \in \mathbb{R}^{n\times n}$, $A = A^t$ ist positiv definit* $\Leftrightarrow$

$$\det \begin{pmatrix} a_{1,1} & \cdots & a_{1,k} \\ \vdots & \ddots & \vdots \\ a_{k,1} & \cdots & a_{k,k} \end{pmatrix} > 0 \ \forall k \in \{1,\dots,n\}$$

Satz 2.4. *Sei $A > 0$. Dann existiert eine untere Δ-Matrix L mit positiven Diagonalelementen, so daß*

$$A = LL^t$$

ist. Diese Faktorisierung heißt Cholesky-Zerlegung.

Beweis: Induktion über n:
$n = 1:$

$$A = \underbrace{a_{1,1}}_{>0} = \sqrt{a_{1,1}} \cdot \sqrt{a_{1,1}} \quad \checkmark$$

$n - 1 \to n:$

$$A_{n-1} \in \mathbb{R}^{(n-1)\times(n-1)},\ A_{n-1} > 0,\ c \in \mathbb{R}^{n-1},\ a_{n,n} \in \mathbb{R}$$

$$\begin{aligned} A &= \begin{pmatrix} A_{n-1} & c \\ c^t & a_{n,n} \end{pmatrix} \\ &= \begin{pmatrix} L_{n-1} \cdot L_{n-1}^t & c \\ c^t & a_{n,n} \end{pmatrix} \\ &\stackrel{!}{=} \begin{pmatrix} L_{n-1} & 0 \\ r^t & \alpha \end{pmatrix} \begin{pmatrix} L_{n-1}^t & r \\ 0 & \alpha \end{pmatrix} \end{aligned}$$

$$\begin{aligned} \Leftrightarrow L_{n-1} r &= c \\ \Rightarrow r &= L_{n-1}^{-1} c \end{aligned}$$

$$\begin{aligned} r^t r + \alpha^2 &= a_{n,n} \\ 0 < \det A &= \underbrace{(\det L_{n-1})}_{>0}{}^2 \alpha^2 \\ \Rightarrow \alpha^2 &> 0 \\ \Rightarrow \alpha^2 &= a_{n,n} - r^t r \\ \Rightarrow \alpha &= \sqrt{a_{n,n} - r^t r} \end{aligned}$$

$$A = LL^t \Leftrightarrow a_{i,k} = \sum_{j=1}^{k} l_{i,j} l_{j,k},\ i \geq k$$

Löse diese $\frac{n(n+1)}{2}$ Gleichungen in der Reihenfolge $(1,1)$, $(2,1)$, $\dots$, $(n,1)$, $(2,2)$,

..., $(n,2)$, ..., (n,n), d.h. spaltenweise, dann erhalten wir:

$$l_{k,k} = (a_{k,k} - \sum_{j=1}^{k-1} l_{k,j}^2)^{\frac{1}{2}} \text{ für } k = 1,\dots,n$$

$$l_{i,k} = (a_{i,k} - \sum_{j=1}^{k-1} l_{i,j} l_{k,j})/l_{k,k} \text{ für } i = k+1,\dots,n$$

3 Lineare Ausgleichsprobleme

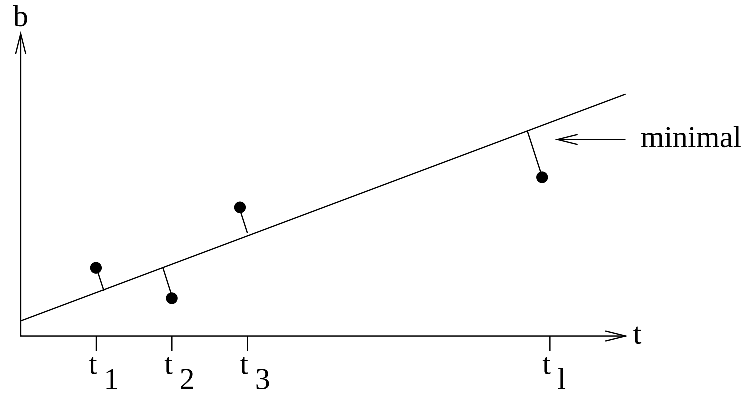
Problem:

- Meßdatensatz $(t_i, b_i) \in \mathbb{R}^d \times \mathbb{R}^k$, $i = 1,\dots,l$
- Modell $\varphi(t,x) = b$
- $x \in \mathbb{R}^n$: Modellparameter

Beispiel: Bestimmung eines Ohmschen Widerstandes
Ohmsches Gesetz:

$$b = x \cdot t = \varphi(t,x)$$

b Spannung, t Stromstärke, x Ohmscher Widerstand



Die Messungen (t_i, b_i) sind mit Meßfehlern behaftet, d.h. sie liegen nicht auf einer Geraden.
Die Steigung der Geraden, die den geringsten Abstand zu den Meßpunkten hat, ist eine vernünftige Approximation an x.

$$F(x) := \begin{pmatrix} \varphi(t_1,x) - b_1 \\ \vdots \\ \varphi(t_l,x) - b_l \end{pmatrix}, \; F : \mathbb{R}^n \to \mathbb{R}^m, \; m = l \cdot k$$

Finde $x^* \in \mathbb{R}^n$:

$$\|F(x^*)\| = \min\{\|F(x)\| \big| x \in \mathbb{R}^n\}, \; \|\cdot\| = \|\cdot\|_2$$

(Methode der kleinsten Fehlerquadrate; Gauß-Ausgleich)

φ sei linear in x. $\varphi(t,x) = x_1\psi_1(t) + \cdots + x_k\psi_k(t)$, ψ_l Modellfunktion

$$\Rightarrow F(x) = Ax - b \text{ mit einer Matrix } A \in \mathbb{R}^{m\times n}$$

Lineares Ausgleichsproblem: $A \in \mathbb{R}^{m\times n}$, $b \in \mathbb{R}^n$. Finde $x^* \in \mathbb{R}^n$:

$$\|Ax^* - b\|_2 = \min_{x\in\mathbb{R}^n} \|Ax - b\|_2 \tag{3.1}$$

$$\begin{aligned} m \geq n &: \text{ überbestimmtes Problem} \\ m < n &: \text{ unterbestimmtes Problem} \end{aligned}$$

Satz 3.1. *Es sind äquivalent:*

a) x^* löst (3.1)

b) $A^tAx^* = A^tb$ (Normalgleichung)

c) $Ax^* = P_Ab$, $P_A : \mathbb{R}^m \to \mathbb{R}^n$ Orthogonalprojektor auf Bild A

Beweis: **a)⇒b)**
- $f(x) = \|Ax - b\|_2^2$ ist differenzierbar
- x^* minimiert $f \Leftrightarrow \nabla f(x^*) = 0$
- $f = h \circ g$, $h(x) = \|x\|_2^2$, $g(x) = Ax - b$, $\nabla f(x) = \nabla h(g(x)) \cdot \mathrm{D}g(x)$
-

$$\begin{aligned} \nabla h(x) &= 2x^t, \mathrm{D}g(x) = A \\ \to \nabla f(x) &= 2(Ax - b)^t A \\ 0 = \nabla f(x^*) &= 2(Ax^* - b)^t A \\ (Ax^* - b)^t \cdot A &= 0 \\ \Leftrightarrow A^tAx^* &= A^tb \quad \surd \end{aligned}$$

b)⇒c)

$$\begin{aligned} &A^t(Ax^* - b) = 0 \\ \Rightarrow &Ax^* - b \in \mathcal{N}(A^t) = (\mathrm{Bild}\, A)^\perp \\ \Rightarrow &0 = P_A(Ax^* - b) = Ax^* - P_Ab \quad \surd \end{aligned}$$

c)⇒a) Sei $y \in \mathbb{R}^n$ beliebig, $P_Ab - b \in (\mathrm{Bild}\, A)^\perp$

$$\begin{aligned} \Rightarrow \|Ay - b\|_2^2 &= \|Ay - P_Ab\|_2^2 + \|P_Ab - b\|_2^2 \quad \text{(Pythagoras)} \\ &= \|Ay - P_Ab\|_2^2 + \|Ax^* - b\|_2^2 \\ &\geq \|Ax^* - b\|_2^2 \quad \surd \end{aligned}$$

Lemma 3.2. *Die Differenz $x^* - y^*$ zweier Lösungen von (3.1) liegt im Nullraum $\mathcal{N}(A) : A(x^* - y^*) = 0$.*
Das Ausgleichsproblem (3.1) hat genau dann eine eindeutige Lösung, wenn $\mathcal{N}(A) = \{0\}$. Das kann im unterbestimmten Fall nur für $n =$ Rang(A) *eintreten.*

Definition 3.3. Unter allen Lösungen von (3.1) nennen wir diejenige mit minimaler Norm die Minimum-Norm-Lösung (MNL) x^+ von (3.1).

Satz 3.4. *Sei $A \in \mathbb{R}^{m\times n}$. Die MNL x^+ von (3.1) ist die eindeutige Lösung der Normalgleichung im $\mathcal{N}(A)^\perp$. Ist* $\operatorname{Rang}(A) = \min\{m,n\}$, *dann:*

$$x^+ = \begin{cases} (A^tA)^{-1}A^tb, \; m \geq n \\ A^t(AA^t)^{-1}b, \; m < n \end{cases}$$

Bemerkung: Für die effiziente und stabile Berechnung von x^+ gibt es Algorithmen, die *nicht* die Normalgleichung lösen (QR-Algorithmus). Beim Übergang von A zu A^tA quadriert sich die Kondition und die Injektivität kann verloren gehen:

$$A = \begin{pmatrix} 1 & 1 \\ \delta & 0 \\ 0 & \delta \end{pmatrix}, \; 0 < \delta \ll 1$$
$$A^tA = \begin{pmatrix} 1+\cancel{\delta}^2 & 1 \\ 1 & 1+\cancel{\delta}^2 \end{pmatrix}$$

QR-Algorithmus

$$\begin{aligned} A &= QR, \; Q \text{ unitär}, \; (Q^*Q = I) \\ \|Ax-b\|_2 &= \|QRx-b\|_2 \\ &= \|Q(Rx-Q^*b)\|_2 \\ &= \|Rx-Q^*b\|_2 \end{aligned}$$

Beispiel: Welcher Punkt liegt drei Geraden im $\mathbb{R}^2$ am nächsten?

$$G_i = \{x = (x_1,x_2)^t \in \mathbb{R}^2 \,|\, b_i = a_ix_1 + x_2\}, \; i = 1,2,3; \; b_i, \, a_i \in \mathbb{R}$$

Gesucht: MNL x^+ von $Ax^+ = b$ mit

$$A = \begin{pmatrix} a_1 & 1 \\ a_2 & 1 \\ a_3 & 1 \end{pmatrix}, b = \begin{pmatrix} b_1 \\ b_2 \\ b_3 \end{pmatrix}$$

$$\operatorname{Rang}(A) = 2 \Leftrightarrow a_i \neq a_j \text{ für ein Paar } (i,j)$$
$$x^+ = (A^tA)^{-1}A^tb = \frac{1}{3d-c^2}\begin{pmatrix} 3 & c \\ c & d \end{pmatrix}\begin{pmatrix} -\sum_{i=1}^3 a_ib_i \\ b_1+b_2+b_3 \end{pmatrix}$$

mit $d = a_1^2 + a_2^2 + a_3^2$, $c = -(a_1+a_2+a_3)$.
Was passiert, falls $a_1 = a_2 = a_3$ ist ?

4 LGS: Krylov-Raum-Verfahren

4.1 Verfahren der konjugierten Gradienten: cg-Verfahren

Generell:

Gegeben: $A \in \mathbb{R}^{n\times n}$ positiv definit, $b \in \mathbb{R}^n$, *Gesucht:* $x \in \mathbb{R}^n : Ax = b$

$$\langle v, w \rangle_A := \langle Av, w \rangle; v, w \in \mathbb{R}^n \text{ neues Skalarprodukt}$$
$$v \text{ ist } A-\text{orthogonal zu } w \Leftrightarrow \langle v, w \rangle_A = 0$$
$$\|v\|_A := \langle v, v \rangle_A^{\frac{1}{2}} \text{ induzierte Norm}$$
$$V_m := \{x^0 + u | u \in U_m\}; x^0 \in \mathbb{R}^n, U_m \subset \mathbb{R}^n \ m\text{-dimensionale Teilmenge}$$

Definiere $x^m \in V_m$ durch:

$$\|x^m - x\|_A = \min\{\|v - x\|_A | v \in V_m\}$$
$$x^m = x^0 + \xi^m,\ v = x^0 + u;\ \xi^m, u \in U_m$$

Bestimme $\xi^m \in U_m$ durch:

$$\|\xi^m - (x - x^0)\|_A = \min\{\|u - (x - x^0)\|_A | u \in U_m\}$$
$$\Rightarrow \xi^m = P_m(x - x^0),\ P_m : \mathbb{R}^n \to U_m \quad A\text{-Orthogonalprojektor}$$

Sei $\{p^1, \ldots, p^m\}$ A-Orthogonalbasis (OGB) von U_m.

$$\Rightarrow x^m = x^0 + \xi^m = x^0 + P_m(x - x^0) = x^0 + \sum_{j=1}^{m} \frac{\langle p^j, x - x^0 \rangle_A}{\langle p^j, p^j \rangle_A} p^j$$
$$= x^0 + \sum_{j=1}^{m-1} \underbrace{\frac{\langle p^j, r^0 \rangle}{\langle Ap^j, p^j \rangle}}_{\alpha_j} p^j + \alpha_m p^m$$

$$r^0 = b - Ax^0$$
$$r^m = b - Ax^m \text{ Residuum von } x^m$$

$$\left.\begin{array}{rcl} \Rightarrow x^m &=& x^{m-1} + \alpha_m p^m \\ r^m &=& r^{m-1} + \alpha_m A p^m \end{array}\right\} \tag{4.1}$$

denn: $r^m = A(x - x^m) = \underbrace{A(x - x^{m-1}}_{r^{m-1}} - \alpha_m p^m)$.

Problem: Geeignete U_m, für die sich eine A-OGB *leicht* berechnen läßt.

Krylov-Räume

$$U_0 = \{0\},\ U_m = U_m(A, v) = \text{spann}\{v, Av, A^2 v, \ldots, A^{m-1} v\},\ m = 1, \ldots, n$$

Lemma 4.1. *Sei $U_m = U_m(A, r^0)$ und $r^m \neq 0$. Dann gilt:*

$$\langle r^i, r^i \rangle = \begin{cases} 0 & i \neq j \\ \|r^i\|_2^2 & i = j \end{cases} \qquad i, j \in \{0, \ldots, m\}$$

$$U_{m+1} = \text{spann}\{r^0, r^1, \ldots, r^m\}$$
$$\dim U_m = m$$

Konstruktion einer A-OGB von $U_m(A, r^0)$

Sei $m \geq 0$, $r^m \neq 0$ und $\{p^1, \ldots, p^m\}$ A-OGB von U_m.

$$\begin{aligned} &\overset{\text{Lemma(4.1)}}{\Rightarrow} && \{p^1, \ldots, p^m, r^m\} \text{ ist eine Basis in } U_{m+1} \\ &\Rightarrow && \{p^1, \ldots, p^m, p^{m+1}\} \text{ ist } A\text{-OGB in } U_{m+1} \\ & && \text{mit } p^{m+1} = r^m - \sum_{j=1}^{m} \frac{\langle r^m, Ap^j\rangle}{\langle Ap^j, p^j\rangle} p^j \\ r^m \perp_2 U_m \quad &\Rightarrow && \langle r^m, Ap^j\rangle = 0 \text{ für } j < m \end{aligned}$$

$$\Rightarrow p^{m+1} = r^m \underbrace{- \frac{\langle r^m, Ap^m\rangle}{\langle Ap^m, p^m\rangle}}_{=:\beta_m} p^m \tag{4.2}$$

Lemma 4.2. *Es gelten:*

$$\alpha_m = \frac{\|r^{m-1}\|_2^2}{\|p^m\|_A^2},\ \beta_m = \frac{\|r^m\|_2^2}{\|r^{m-1}\|_2^2}$$

cg-Algorithmus

zur Lösung von $Ax = b$, Startwert x^0:

$$\begin{aligned} & p^1 = r^0 = b - Ax^0 \\ & \texttt{fuer } m = 1, \ldots, n \\ & \{ \\ & \quad \texttt{falls } r^{m-1} = 0: \\ & \qquad \texttt{STOP} \\ & \quad a^m = Ap^m,\ \alpha_m = \|r^{m-1}\|_2^2 / \langle a^m, p^m\rangle \\ & \quad \left.\begin{aligned} x^m &= x^{m-1} + \alpha_m p^m \\ r^m &= r^{m-1} - \alpha_m a^m \end{aligned}\right\} && (4.3) \\ & \quad \beta_m = \|r^m\|_2^2 / \|r^{m-1}\|_2^2 \\ & \quad p^{m+1} = r^m + \beta_m p^m && (4.4) \\ & \} \end{aligned}$$

Konvergenzanalyse

$U_m(A, r^0) = \mathbb{R}^n \Rightarrow x^n = x$ (bei exakter Arithmetik)

cg-Verfahren: einerseits: iteratives Verfahren
andererseits: direktes Verfahren

Lemma 4.3. *Es gibt ein $Q_m \in \Pi_m^*$ (mit $\Pi_m^* = \{q_m | q_m$ ist ein Polynom vom* Grad $\leq m$, $q_m(0) = 1\}$*), so daß*

$$e^m = x^m - x = Q_m(A)e^0,\ e^0 = x^0 - x$$

ist. Weiter gilt

$$\begin{aligned}\|e^m\|_A &= \|Q_m(A)e^0\|_A \\ &= \min\{\|q_m(A)e^0\|_A \big| q_m \in \Pi_m^*\}\end{aligned}$$

sowie

$$\|e^m\|_A \leq \max_{\lambda\in\sigma(A)} |q_m(\lambda)| \cdot \|e^0\|_A \;\forall q_m \in \Pi_m^*$$

Hierbei ist $\sigma(A) = \{\lambda\big|\lambda$ *ist EW von* $A\}$.

Satz 4.4. $A > 0$ *habe nur* $k \leq n$ *verschiedene Eigenwerte. Dann gilt:*

$$x^k = x$$

Beweis: $\lambda_1, \ldots, \lambda_k$ seien die verschiedenen Eigenwerte in A.

$$\begin{aligned} q_k(t) &= \prod_{j=1}^{k}(\lambda_j - t)/\lambda_j,\; q_k(0) = 1 \Rightarrow q_k \in \Pi_k^* \\ q_k(\lambda_i) &= 0 \text{ für } i = 1, \ldots, k \\ \Rightarrow \|e^k\|_A &\leq \max_{\lambda\in\{\lambda_1,\ldots,\lambda_k\}} |q_k(\lambda)| \cdot \|e^0\|_A = 0 \\ \Rightarrow x &= x^k \quad \surd \end{aligned}$$

Satz 4.5. *Für* $A > 0$ *gilt:*

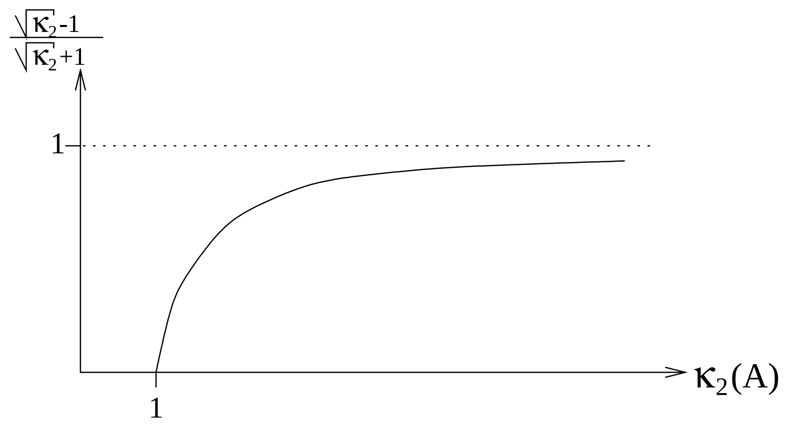
$$\begin{aligned} \|e^m\|_A &< 2\left(\frac{\sqrt{\kappa_2(A)}-1}{\sqrt{\kappa_2(A)}+1}\right)^m \cdot \|e^0\|_A \\ \kappa_2(A) &= \|A\|_2\|A^{-1}\|_2 \quad \textit{Kondition, siehe Kapitel 1} \\ &= \lambda_{\max}(A)/\lambda_{\min}(A) \end{aligned}$$

Beispiel: $A > 0$ habe nur 2 verschiedene Eigenwerte $\lambda_1 \ll \lambda_2$.

$$\Rightarrow \kappa_2(A) = \lambda_2/\lambda_1 \gg 1$$

Satz 4.5 suggeriert eine langsame Konvergenz. Aus Satz 4.4 wissen wir: $x^2 = x$.

Vorkonditionierung



Je kleiner κ_2 desto schneller konvergiert das cg-Verfahren.
Sei $B \in \mathbb{R}^{n\times n}$ positiv definit.

$$Ax = b \Leftrightarrow \overline{A}\overline{x} = b,\ \overline{A} = A \cdot B,\ x = B\overline{x}$$

cg-Algorithmus mit vorkonditioniertem B

$$\begin{array}{l}
r^0 = b - Ax^0,\ p^1 = Br^0,\ \rho_1 = \langle p^1, r^1\rangle \\
\texttt{fuer}\ m = 1, \ldots, n \\
\{ \\
\quad \texttt{falls}\ r^{m-1} = 0: \\
\quad\quad \texttt{STOP} \\
\quad a^m = Ap^m,\ \alpha_m = \rho_m / \langle a^m,\ p^m\rangle \\
\quad x^{m+1} = x^m + \alpha_m p^m,\ r^m = r^{m-1} - \alpha_m a^m \\
\quad v^m = Br^m,\ \rho_{m+1} = \langle v^m,\ r^m\rangle \\
\quad p^{m+1} = v^m + \dfrac{\rho_{m+1}}{\rho_m} p^m \\
\}
\end{array}$$

Satz 4.6. *Die Matrix B erfülle*

$$\gamma\langle B^{-1}v, v\rangle \leq \langle Av, v\rangle \leq \Gamma\langle B^{-1}v, v\rangle \quad \forall v \in \mathbb{R}^n$$

mit $0 < \gamma \leq \Gamma$. Dann gilt für das mit B vorkonditionierte cg-Verfahren:

$$\|x^m - x\|_A \leq 2\left(\frac{\sqrt{\kappa}-1}{\sqrt{\kappa}+1}\right)^m \cdot \|x^0 - x\|_A \ \textit{mit}\ \kappa = \frac{\Gamma}{\gamma}$$

Ein guter Vorkonditionierer erfüllt

a) $\kappa_2(B \cdot A) = \frac{\Gamma}{\gamma} \ll \kappa_2(A)$

b) By ist "einfach" (hat höchstens die gleiche Komplexität wie Ay)

4.2 Das GMRES-Verfahren

*G*eneralized *M*inimal *RES*idual

$$A \in \mathbb{R}^{n\times n} \text{ regulär },\ b \in \mathbb{R}^n$$

Die k-te GMRES-Iterierte x^k ist definiert durch

$$\|\underbrace{Ax^k - b}_{r^k}\|_2 = \min\{\|Av - b\|_2 \big| v \in V_k\}$$

$$V_k = x^0 + U_k(A, r^0),\ U_k(A, r^0) \text{ Krylov-Raum},\ r^0 = Ax^0 - b$$

Satz 4.7. *Es gilt*

$$\begin{aligned}\|r^k\|_2 &= \min\{\|q_k(A)r^0\|_2 \big| q_k \in \Pi_k^*\}\\ \Rightarrow \|r^k\|_2 &\leq \|q_k(A)\|_2\|r^0\|_2 \; \forall q_k \in \Pi_k^*\end{aligned}$$

Insbesondere gilt der Satz 4.4 analog für das GMRES-Verfahren.

Satz 4.8. *Sei $A = VDV^{-1}$ eine reguläre, diagonalisierbare Matrix:*

$$(D = \operatorname{diag}(\lambda_1, \ldots, \lambda_n),\ \lambda_i \in \mathbb{C},\ V \in \mathbb{C}^{n\times n})$$

Dann gilt:

$$\|r^k\|_2 \leq \kappa_2(V) \cdot \max_{1\leq i\leq n} |q_k(\lambda_i)| \cdot \|r^0\|_2 \; \forall q \in \Pi_k^*$$

Beweis:

$$\begin{aligned}\|q_k(A)\|_2 &= \|Vq_k(D)\cdot V^{-1}\|_2 \leq \|V\|_2 \cdot \|V^{-1}\|_2 \cdot \|\operatorname{diag}(q_k(\lambda_1), \ldots, q_k(\lambda_n))\|_2\\ &= \kappa_2(V) \cdot \max_{1\leq i\leq n} |q_k(\lambda_i)| \quad \checkmark\\ &\text{(Bem.: } A \text{ normal } (AA^t = A^tA) \Rightarrow \kappa_2(V) = 1)\end{aligned}$$

Satz 4.9. *Falls $\|I - A\|_2 \leq \rho < 1$:*

$$\Rightarrow \|r^k\|_2 \leq \rho^k \|r^0\|_2$$

Beweis:

$$\begin{aligned}q_k(t) &= (1-t)^k \in \Pi_k^*\\ \Rightarrow \|r^k\|_2 &\leq \|q_k(A)\|_2 \cdot \|r^0\|_2\\ &= \|(I-A)^k\|_2 \cdot \|r^0\|_2\\ &\leq \|I-A\|_2^k \cdot \|r^0\|_2\\ &\leq \rho^k \|r^0\|_2 \quad \checkmark\end{aligned}$$

Vorkonditionierung

$$Ax = b \rightsquigarrow MAx = Mb$$

Ein guter Vorkonditionierer erfüllt drei Bedingungen:

i) $\kappa_2(MA) \ll \kappa_2(A)$
(⇒vorkonditionierte Residuen der GMRES-Iteration sind gute Fehlerindikatoren)

ii) $\|I - MA\|_2 < 1$

iii) Mv "einfach"

Aspekte der Implementierung

Orthonormalisierung von $U_k(A, r^0) = \text{spann}\{r^0, Ar^0, A^2r^0, \ldots, A^{k-1}r^0\}$

1. $r^0 = b - Ax^0$, $p^1 = r^0/\|r^0\|_2$
2. für $i = 1, \ldots, k-1$

$$\tilde{p}^{i+1} := Ap^i - \sum_{j=1}^{i} \langle Ap^i, p^j \rangle_2 \cdot p^j$$

   Falls $\tilde{p}^{i+1} \neq 0$: $p^{i+1} = \tilde{p}^{i+1}/\|\tilde{p}^{i+1}\|_2$.

Satz 4.10. *$\{q^j\}_{1 \leq j \leq i}$ sei nach obigem Verfahren konstruiert.*
Falls $\tilde{p}^{i+1} = 0$ ist, dann ist

$$x = A^{-1}b \in V_i = x^0 + U_i(A, r^0), \text{ d.h. } x^i = x$$

Die Orthogonalisierung bricht nur ab, wenn x^i bereits die Lösung ist.

$$\begin{aligned} P_k &= (p^1, \ldots, p^k) \in \mathbb{R}^{n \times k} \\ x^k &= x^0 + P_k z^k \\ &\text{(wobei } z^k \in \mathbb{R}^k \text{ das lineare Ausgleichsproblem} \\ &\|\underbrace{b - A(x^0 + P_k z^k)}_{=r^0 - AP_k z^k}\|_2 = \min_{z \in \mathbb{R}^k} \|r^0 - AP_k z^k\|_2 \qquad (4.5) \\ &\text{löst.)} \qquad (4.6) \end{aligned}$$

Lemma 4.11. *$AP_k = P_{k+1}H_k$ mit $H_k \in \mathbb{R}^{(k+1) \times k}$ und*

$$(H_k)_{j,i} = \begin{cases} 0 & j > i+2 \\ \langle Ap^i, p^j \rangle & \text{sonst} \end{cases}$$

$$\Rightarrow H_k = \begin{pmatrix} * & * & * & * \\ * & * & * & * \\ 0 & * & * & * \\ 0 & 0 & * & * \\ 0 & 0 & 0 & * \end{pmatrix} \quad \textit{obere Hessenberg-Matrix}$$

Sei $e_1 = (1, 0, \ldots, 0)^t$, dann gilt $r^0 = \underbrace{\|r^0\|_2}_{=:\beta} \cdot \underbrace{P_k e_1}_{=:p^1}$

$$\begin{aligned} \Rightarrow r^k &= b - Ax^k \\ &= r^0 - A\underbrace{(x^k - x^0)}_{=P_k z^k} \\ &= \beta P_{k+1} e_1 - \underbrace{AP_{k+1} z^k}_{=P_{k+1}H_k} \\ &= P_{k+1}(\beta e_1 - H_k z^k) \\ \Rightarrow \|r^k\|_2 &= \|\beta e_1 - H_k z^k\|_2 \end{aligned}$$

Statt (4.5) müssen wir das einfachere Problem

$$\|\beta e_1 - H_k z^k\|_2 = \min_{z \in \mathbb{R}^k} \|\beta e_1 - H_k z\|_2$$

lösen.

Algorithmus GMRES

$r^0 = b - Ax^0$, $\beta = \|r^0\|_2$, $\rho = \beta$, $p^1 = r^0/\beta$, $k = 0$

`solange` $(\rho > \varepsilon \cdot \beta$ `und` $k \le k_{\max})$ | * $\varepsilon > 0$ Abbruchtoleranz, $k_{\max}$ max. Schrittzahl * |

{

  $k = k + 1$, $p^{k+1} = Ap^k$

  `fuer` $j = 1, \ldots, k$

  {

    $h_{j,k} = \langle p^{k+1}, p^j \rangle$ | * stabilisiertes Gram-Schmidt-Verfahren * |

    $p^{k+1} = p^{k+1} - h_{j,k} p^j$

  }

  $h_{k+1,k} = \|p^{k+1}\|_2$, $p^{k+1} = p^{k+1}/h_{k+1,k}$

  `Minimiere` $\|\beta e_1 - H_k z\|_2$ `ueber` $\mathbb{R}^k$, `um` z^k `zu erhalten`

  $\rho = \|\beta e_1 - H_k z^k\|_2$ | * $\rho = \|b - Ax^k\|_2$ * |

}

$x^k = x^0 + P_k z^k$

Bemerkung: Die Krylov-Raum-Basis $\{p^1, \ldots, p^k\}$ muß während der Iteration gespeichert werden. Daher legt man eine Maximalschrittzahl $k_{\max}$ fest. Hat $x^{k_{\max}}$ nicht die gewünschte Genauigkeit, so starte GMRES erneut mit $x^0 = x^{k_{\max}}$.

5 Nichtlineare Gl.-Systeme: Newton-Verfahren

Abstraktes Problem: $F : D \to \mathbb{R}^n, D \subset \mathbb{R}^n$ offen. Finde

$$x^* \in D : \; F(x^*) = 0 \tag{5.1}$$

Sei $x^{\text{alt}} \in D$ eine Näherung an $x^* \in D$.
Ziel: Verbesserung von x^{alt}.

$$F(x^{\text{alt}}) = F(x^{\text{alt}}) - \underbrace{F(x^*)}_{=0} \approx F'(x^{\text{alt}})(x^{\text{alt}} - x^*)$$

$$\text{mit } F'(x) = \left\{ \frac{\partial F_i}{\partial x_j} \right\}_{i,j} \text{ Jacobi-Matrix}$$

Bestimme x^{neu} daher durch

$$\begin{aligned} &F(x^{\text{alt}}) = F'(x^{\text{alt}})(x^{\text{alt}} - x^{\text{neu}}) \\ \Leftrightarrow\ & x^{\text{neu}} = x^{\text{alt}} - F'(x^{\text{alt}})^{-1} F(x^{\text{alt}}) \end{aligned}$$

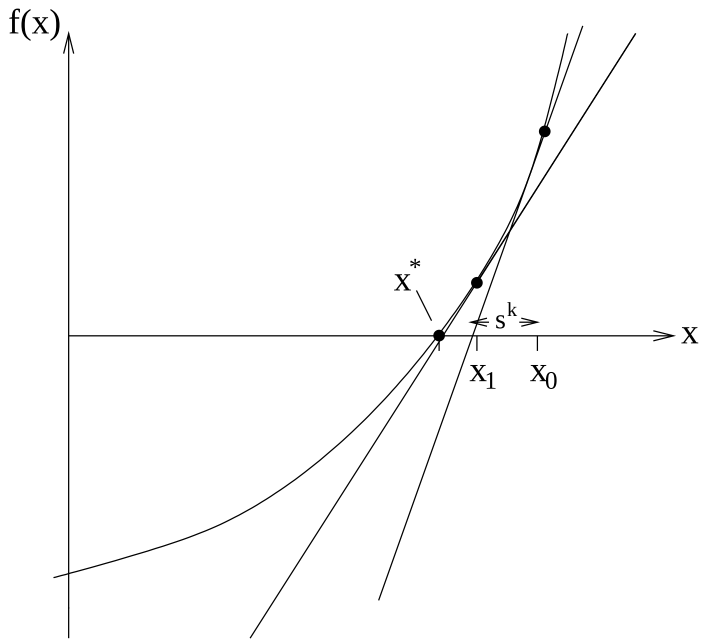
Newton-Verfahren

Sei x^0 beliebig. Für $k = 0, 1, \ldots$:

$$\begin{aligned} x^{k+1} &= x^k + s^k \\ F'(x^k)s^k &= -F(x^k) \end{aligned} \tag{5.2}$$

s^k: k-ter Newton-Schritt (oder -korrektur)

Graphische Interpretation ($n = 1$)



Beispiel:

$$\begin{aligned} f(t) &= t^2 - a,\ a > 0 \\ f(t^*) = 0 &\Leftrightarrow t^* = \pm\sqrt{a} \\ x^{k+1} &= \frac{x^k - [(x^k)^2 - a]}{f'(x^k) = 2x^k} \\ &= \tfrac{1}{2}\left(x^k + \frac{a}{x^k}\right) \qquad \text{Verfahren von Heron} \end{aligned}$$

Falls $x^0 > 0 : \lim_{k\to\infty} x^k = \sqrt{a}$. Sei $a = 2$:

k	x^k	$\lvert x^k - \sqrt{2}\rvert$
0	2	0.58
1	1.5	0.086
2	1.4166	0.0025
3	1.4142157	$2.4 \cdot 10^{-6}$
4	1.4142136	exakt bis auf TR-Genauigkeit
$\vdots$	$\vdots$	$\vdots$

Voraussetzungen an F

a) Gleichung (5.1) hat eine Lösung $x^* \in D$

b) $F' : D \to \mathbb{R}^{n\times n}$ ist Hölderlin-stetig der Ordnung $\alpha \in]0,1]$:

$$\|F'(x) - F'(y)\| \leq \gamma \|x - y\|^\alpha$$

c) $F'(x^*)$ ist regulär

$B(\delta) = \{x \in \mathbb{R}^n \,\big|\, \|x - x^+\| \leq \delta\}$

Satz 5.1. *F erfülle die Voraussetungen. $\{x^k\}$ sei die Newtonfolge (5.2). Dann gibt es ein $\delta > 0$, so daß $\lim_{k\to\infty} x^k = x^*$, falls $x^0 \in B(\delta)$. Außerdem ist x^* die einzige Nullstelle von F in $B(\delta)$. Weiter gilt:*

$$\|x^{k+1} - x^*\| \leq c_N \cdot \|x^k - x^*\|^{1+\alpha},\ k = 0, 1, \ldots$$

mit $c_N = 2\gamma \cdot \|F'(x^*)^{-1}\| / (1+\alpha)$ ($\alpha = 1$: quadratische Konvergenz; $0 < \alpha < 1$: superlineare Konvergenz).

Beweis: Wir zeigen nur die Fehlerabschätzung.

$$\begin{aligned}
e^k &:= x^k - x^*,\ x^k \in B(\delta) \subset D \\
e^{k+1} &= e^k - F'(x^k)^{-1} \underbrace{(F(x^k) - F(x^*))}_{=\int_0^1 F'(x^*+te^k)e^k\,dt} \\
\Rightarrow e^{k+1} &= F'(x^k)^{-1} \cdot \int_0^1 (F'(x^k) - F'(x^* - te^k))e^k\,dt \\
\Rightarrow \|e^{k+1}\| &\leq \|F'(x^k)^{-1}\| \cdot \gamma \cdot \|e^k\|^{1+\alpha} \cdot \underbrace{\int_0^1 (1-t)^\alpha\,dt}_{=\frac{1}{1+\alpha}}
\end{aligned}$$

Für δ hinreichend klein gilt:

$$\begin{aligned}
&\|F'(y)^{-1}\| \leq 2\|F'(x^*)^{-1}\|\ \forall y \in B(\delta) \\
\Rightarrow\ &\|e^{k+1}\| \leq \underbrace{\frac{2\gamma}{1+\alpha}\|F'(x^*)^{-1}\|}_{=c_N} \cdot \|e^k\|^{1+\alpha}
\end{aligned}$$

Hieraus folgen induktiv die anderen Aussagen des Satzes. $\surd$

$$A \in \mathbb{R}^{n\times n} \text{ regulär } \Rightarrow F(x^*) = 0 \Leftrightarrow \underbrace{AF(x^*)}_{G(x^*)} = 0$$

$$x - G'(x)^{-1}G(x) = x - F'(x)F(x)$$

$\Rightarrow$ Die Newtonfolge $\{x^k\}$ ändert sich nicht, wenn F mit einer regulären Matrix multipliziert wird. Das Newtonverfahren ist *affin-invariant.*
Die Konvergenz von $\{x^k\}$ hängt von x^0 ab. Wie kann man während der Iteration feststellen, ob das Newtonverfahren konvergiert?

$$F(x^*) = 0 \Leftrightarrow \|F(x)\| \rightarrow \text{ minimieren}$$

Idee: Setze Iteration fort, solange

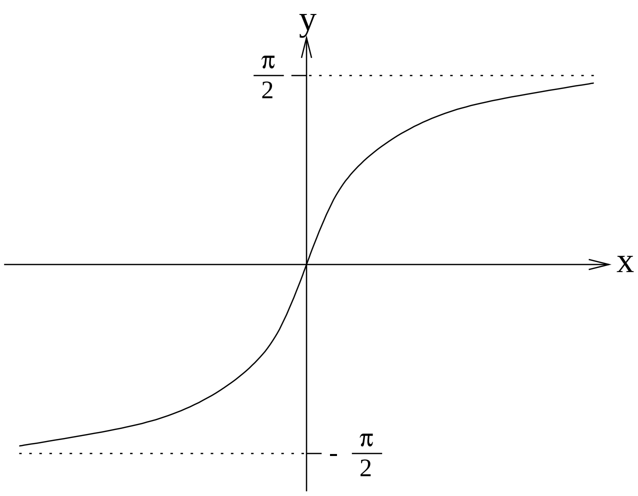
$$\|F(x^{k+1})\| \leq \vartheta \|F(x^k)\| \tag{5.3}$$

für ein $\vartheta \in]0,1]$.
Der Monotonietest ist nicht affin-invariant. Ersetze (5.3) durch

$$\|F'(x^k)^{-1}F(x^{k+1})\| \leq \vartheta \| \underbrace{F'(x^k)^{-1}F(x^k)}_{=:-s^k} \| \tag{5.4}$$

Stoppe das Newton-Verfahren, falls (5.4) verletzt ist, oder falls $\underbrace{\|s^k\|}_{\approx\|e^k\|} \leq$ tol ist.

Beispiel: Betrachte $f(x) = \arctan x$, $x^* = 0$. Newton-Folge mit $x^0 = 10$:



$$x^1 = -138,\ x^2 = 2.9 \cdot 10^4,\ x^3 = -1.5 \cdot 10^9, \ldots$$

Woran scheitert die Konvergenz?

$$s^k = -f(x^k)/f'(x^k),\ |x^k| \gg 1$$

$$f'(x^k) = (1 + (x^k)^2)^{-1} \ll 1,\ |f(x^k)| \approx \frac{\pi}{2}$$

$\Rightarrow |s^k|$ ist zu groß.
Beobachtung: s^k zeigt von x^k in Richtung von x^*.
Idee: Dämpfe das Newton-Verfahren:

$$x^{k+1} = x^k + \lambda s^k,\ 0 < \lambda \leq 1$$

Globales Newton-Verfahren

$k = 0$, $x^0 \in D$
`while` $(\|F(x^k)\| > \varepsilon \|F(x^0)\|)$
$\{$
  $F'(x^k)s^k = -F(x^k)$, $\lambda = 1$
  `(*)` $y = x^k + \lambda s^k$
  `if` $(\|F(y)\| < (1 - \beta\lambda) \cdot \|F(x^k)\|)$
    $x^{k+1} = y$
  `else`
  $\{$
    `verkleinere` λ (z.B. $\lambda = \lambda/2$)
    `goto (*)`
  $\}$
  $k = k + 1$
$\}$

Fixpunktiteration

Newton-Verfahren:

$$x^{k+1} = \Phi(x^k) \text{ mit } \Phi(x) = x - F'(x)^{-1} F(x)$$

$$F(x^*) = 0 \Leftrightarrow \Phi(x^*) = x^*,\ x^* \text{ ist Fixpunkt von } \Phi$$

Sei nun $\Phi : D \to \mathbb{R}^n$, $D \subset \mathbb{R}^n$ eine beliebige Abbildung. *Fixpunktproblem:* Finde

$$x^* \in D : \ \Phi(x^*) = x^*$$

Fixpunktiteration:

$$x^{k+1} = \Phi(x^k),\ k = 0, 1, \ldots;\ x^0 \in D \tag{5.5}$$

Lemma 5.2. *Sei Φ stetig und $\{x^k\}$ aus (5.5) konvergiere gegen ξ. Dann gilt $\Phi(\xi) = \xi$.*

Beweis:

$$\xi = \lim_{k\to\infty} x^{k+1} = \lim_{k\to\infty} \Phi(x^k) \overset{\text{Stetigkeit}}{=} \Phi(\xi) \quad \surd$$

Wann konvergiert (5.5)?

Definition 5.3. $g : D \to \mathbb{R}^n$, $D \subset \mathbb{R}^n$ heißt *Kontraktion bezüglich* $\|\cdot\|$ falls ein $0 \le q < 1$ existiert mit

$$\|g(x) - g(y)\| \le q\|x - y\| \ \forall x, y \in D$$

q heißt *Kontraktionszahl.*

Lemma 5.4. *D konvex, offen; $g : D \to \mathbb{R}^n$ stetig, differenzierbar.*
Ist $q = \sup_{x \in D} \|g'(x)\| < 1$, dann ist g eine Kontraktion bezüglich $\|\cdot\|$ mit Kontraktionszahl q.

Beweis: Mittelwertsatz in Integralform.

Beispiel:

$$\Phi(x_1,x_2) = \left(\mathrm{e}^{-\frac{1}{2}(x_1+x_2)}, \tfrac{1}{3}\sin(x_1+x_2)\right)^t$$
$$\Phi'(x_1,x_2) = \begin{pmatrix} -\frac{1}{2}\mathrm{e}^{-\frac{1}{2}(x_1+x_2)} & -\frac{1}{2}\mathrm{e}^{-\frac{1}{2}(x_1+x_2)} \\ \frac{1}{3}\cos(x_1+x_2) & \frac{1}{3}\cos(x_1+x_2) \end{pmatrix}$$
$$\|\Phi'(x_1,x_2)\|_\infty = \max\left\{\mathrm{e}^{-\frac{1}{2}(x_1+x_2)}, \tfrac{2}{3}|\cos(x_1+x_2)|\right\}$$

$$D = \{x \in \mathbb{R}^2 \,|\, x_1 + x_2 > a > 0\} \text{ konvex offen}$$

$\Rightarrow \Phi : D \to \mathbb{R}^n$ ist Kontraktion bezüglich $\|\cdot\|_\infty$.

Beispiel: $\Phi(x) = x - F'(x)^{-1}F(x)$, F stetig differenzierbar.

$$\frac{\partial \Phi_i}{\partial x_k} = \delta_{i,k} - \sum_{j=1}^n \frac{\partial}{\partial x_k}(F'(x)^{-1})_{i,j} \cdot F_j(x) - \sum_{j=1}^n (F'(x)^{-1})_{i,j} \frac{\partial}{\partial x_k} F_j(x)$$

Sei x^* eine Nullstelle von F.

$$\Rightarrow \Phi'(x^*) = I - F'(x^*)^{-1} \cdot F'(x^*) = 0$$
$$\Rightarrow \Phi \text{ ist Kontraktion in der Umgebung von } x^*.$$

Satz 5.5 (Banachscher Fixpunktsatz). *Sei $D \subset \mathbb{R}^n$ abgeschlossen, $\Phi : D \to D$ eine Kontraktion bezüglich $\|\cdot\|$ mit Kontraktionszahl q. Dann hat Φ genau einen Fixpunkt in $x^* \in D$.*
Die Fixpunktiteration (5.5) konvergiert gegen x^* für jedes $x^0 \in D$. Es gelten folgende Fehlerabschätzungen:

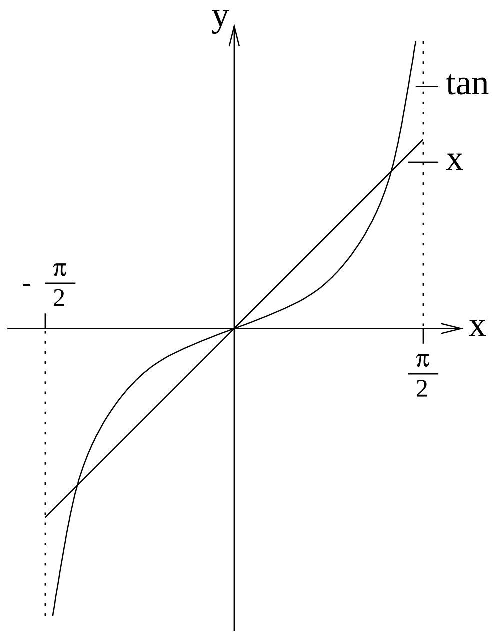
$$\|x^* - x^k\| \le \frac{q^k}{1-q}\|x^* - x^0\|$$
$$\|x^* - x^k\| \le \frac{q}{1-q}\|x^k - x^{k-1}\|$$

Beweis: Annahme: Es gibt zwei Fixpunkte $x^*, \overline{x} \in D$ von Φ.

$$\|x^* - \overline{x}\| = \|\Phi(x^*) - \Phi(\overline{x})\| \le q\|x^* - \overline{x}\| \Rightarrow 1 \le q \quad \lightning$$

Rest siehe Literatur. $\surd$

Beispiel: Finde $x^* \in [\frac{\pi}{2}, \frac{3\pi}{2}]: \tan x^* = x^*$.



$$\Phi(x) = \tan x,\ \Phi'(x) = 1 + \tan^2 x \geq 1 \Rightarrow \Phi \text{ ist keine Kontraktion.}$$

Ausweg: Formuliere Problem mit Φ^{-1}.

$$x = \tan x \Leftrightarrow x = \tan(x - \pi) \Leftrightarrow x = \pi + \arctan x$$

Finde $x^* \in [\frac{\pi}{2}, \frac{3\pi}{2}]: \pi + \arctan x^* = x^*$:

$$\begin{aligned}
&\Phi(x) = \pi + \arctan x \\
\Rightarrow &\Phi' = (1 + x^2)^{-1} \\
\Rightarrow &\max_{x \in [\frac{\pi}{2}, \frac{3\pi}{2}]} |\Phi'| < 1 \\
\Rightarrow &\Phi \text{ ist Kontraktion} \\
&\Phi(\mathbb{R}) = [\tfrac{\pi}{2}, \tfrac{3\pi}{2}] \\
\Rightarrow &\Phi([\tfrac{\pi}{2}, \tfrac{3\pi}{2}]) \subset [\tfrac{\pi}{2}, \tfrac{3\pi}{2}] \\
\Rightarrow &\text{Die Voraussetzungen von Satz 5.5 sind erfüllt:} \\
&x^0 = \pi \Rightarrow x^4 = 4.4934
\end{aligned}$$

Satz 5.6 (lokaler Konvergenzsatz). $\Phi : \mathbb{R}^n \to \mathbb{R}^n$ *stetig differenzierbar und habe einen Fixpunkt* x^*, $\|\Phi'(x^*)\| < 1$.
Dann gibt es eine abgeschlossende Umgebung D*, in der gilt:*

a) Φ *ist eine Kontraktion*

b) $\Phi(D) \subset D$

d.h. die Fixpunktiteration (5.5) konvergiert gegen x^* *für jedes* $x^0 \in D$.

Bemerkung: Mit diesem Satz kann man die lokale Konvergenz des Newton-Verfahrens zeigen.

Beweis: a) $x \mapsto \|\Phi'(x)\|$ stetig

$$\Rightarrow \exists \delta > 0 : \ \|\Phi'(x)\| \leq q < 1 \quad \forall x \in D = \{x \in \mathbb{R}^n \,\big|\, \|x - x^*\| \leq \delta\}$$
$$\Rightarrow \Phi \text{ ist Kontraktion in } D$$

b) Sei $y \in D$. Zu zeigen: $\Phi(y) \in D$.

$$\|\Phi(y) - x^*\| = \|\Phi(y) - \Phi(x^*)\| \leq q\|y - x^*\| \leq q \cdot \delta < \delta$$

$$\Rightarrow \Phi(y) \in D \quad \surd$$

6 Interpolation und Approximation

Situation: Von $f : \mathbb{R} \to \mathbb{R}$ sind nur $f(t_i)$ und eventuell Ableitungen $f^{(j)}(t_i)$ bekannt. $f, t_i \in \mathbb{R}$, $0 \leq i \leq n$.
Problem: Was ist eine vernünftige Approximation an $f(t)$ für $t \notin \{t_1, \ldots, t_n\}$?
Anwendung:

- Logarithmentafeln (historisch)
- Computergrafik (Repräsentation geometrischer Objekte)
  - CAD
  - CAM
  - ...

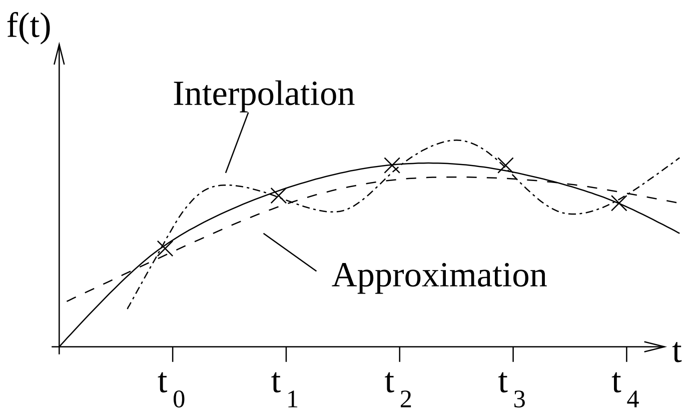
Aufgabe: Konstruiere $\varphi : \mathbb{R} \to \mathbb{R}$ einfach (z.B. Linearkombinationen von stückweisen Polynomen, Exponentialfunktionen oder rationalen Funktionen), die entweder

$$\varphi^{(j)}(t_i) = f^{(j)}(t_i) \qquad \text{(Interpolation)}$$

oder

$$\|\varphi - f\| \text{ “klein”} \qquad \text{(Approximation)}$$

erfüllt. t_i heißen *Knoten* oder *Stützstellen*, $f^{(j)}(t_i)$ heißen *Stützwerte*.



6.1 Polynominterpolation

Gegeben:

$$f_i = f(t_i),\ i = 0, \dots, n;\ t_0 < t_1 < \dots < t_n$$

Gesucht: $D \in \Pi_n = \{\text{Polynome vom Grad } n\}$ mit

$$P(t_i) = f_i,\ 0 \le i \le n$$

Satz 6.1. $(t_i, f_i) \in \mathbb{R}^2$, $0 \le i \le n$ *mit* $t_i \neq t_j\ \forall j \in \{0, \dots, n\} \backslash \{i\}$
$\Rightarrow$ *Es gibt genau ein* $P = P(f|t_0, \dots, t_n) \in \Pi_n$ *mit* $P(t_i) = f_i$, $i = 0, \dots, n$.

Beweis: a) Eindeutigkeit: $P, Q \in \Pi_n$ mit $P(t_i) = f_i = Q(t_i)$

$$\Rightarrow \Delta = P - Q \in \Pi_n \text{ und } \Delta(t_i) = 0,\ i = 0, \dots, n$$
$$\Rightarrow \Delta = 0$$

b) Existenz: $\Pi_n = \operatorname{spann}\{1, t, t^2, \dots, t^n\}$ monomiale Basis

$$P(t) = a_n t^n + a_{n-1} t^{n-1} + \dots + a_1 t + a_0$$
$$P(t_i) = f_i,\ i = 0, \dots, n$$
$$\Leftrightarrow \underbrace{\begin{pmatrix} 1 & t_0 & t_0^2 & \cdots & t_0^n \\ \vdots & t_1 & \vdots & & \vdots \\ \vdots & \vdots & \vdots & & \vdots \\ 1 & t_n & t_n^2 & \cdots & t_n^n \end{pmatrix}}_{=:V \in \mathbb{R}^{(n+1)\times(n+1)}} \begin{pmatrix} a_0 \\ \vdots \\ a_n \end{pmatrix} = \begin{pmatrix} f_0 \\ \vdots \\ f_n \end{pmatrix}$$

V heißt Vandermonde-Matrix
aus a) folgt $\mathcal{N}(V_n) = \{0\}$

$$\Rightarrow V_n^{-1} \text{ ex.} \Rightarrow (a_0, \dots, a_n)^t = V_n^{-1}(f_0, \dots, f_n)^t \quad \checkmark$$

Die Bestimmung von $P(f|t_0, \dots, t_n)$ aus obigem System ist zu aufwendig.

$\Pi_n = \operatorname{spann}\{L_{n,0}, \dots, L_{n,n}\}$, $L_{n,i}$ Lagrange-Basis bzgl. $t_0, \dots, t_n$

$$L_{n,k}(t) = \prod_{\substack{j=0 \\ j \neq k}}^{n} \frac{t - t_j}{t_k - t_j},\ L_{n,k}(t_i) = \begin{cases} 1 & i = k \\ 0 & \text{sonst} \end{cases}$$

$$\Rightarrow P = P(f|t_0, \dots, t_n) = \sum_{k=0}^{n} f_k L_{n,k}(t), \text{ denn}$$

$$P(t_i) = \sum_{k=0}^{n} f_k L_{n,k}(t_i) = f_i,\ 0 \le i \le n$$

Lemma 6.2 (Lemma von Aitken). $(t_i, f_i) \in \mathbb{R}^2$, $t_0 < t_1 < \dots < t_n$. *Dann gilt*

$$P(f|t_0, \dots, t_n)(t) = \frac{(t_o - t)P(f|t_1, \dots, t_n)(t) - (t_n - t)P(f|t_0, \dots, t_{n-1})(t)}{t_0 - t_n}$$

Beweis: Sei φ rechte Seite von oben $\Rightarrow \varphi \in \Pi_n$

$$\varphi(t_0) = \frac{-(t_n - t_0)}{t_0 - t_n} f_0 = f_0, \text{ analog } \varphi(t_n) = f_n$$

Sei $1 \leq i \leq n-1$:

$$\begin{aligned}\varphi(t_i) &= \frac{(t_0 - t_i)f_i - (t_n - t_i)f_i}{t_0 - t_n} = f_i \\ &\overset{\text{Satz 6.1}}{\Rightarrow} \varphi = P(f|t_0, \ldots, t_n) \quad \checkmark\end{aligned}$$

Sei $t \in \mathbb{R}$ fest. $P_{i,k} := P(f|t_{i-k}, \ldots, t_i)$, $i \geq k$. Es ist $P_{i,0} = f_i$ und $P_{n,n} = P(f|t_0, \ldots, t_n)(t)$. $P_{n,n}$ läßt sich rekursiv berechnen.

Schema von Neville

$$\begin{aligned}P_{i,0} &= f_i, \; i = 0, \ldots, n \\ P_{i,k} &= \frac{(t_{i-k} - t)P_{i,k-1} - (t_i - t)P_{i-1,k-1}}{t_{i-k} - t_i} \\ &= P_{i,k-1} + \frac{t - t_i}{t_i - t_{i-k}}(P_{i,k-1} - P_{i-1,k-1}), \; i \geq k\end{aligned}$$

$$\begin{array}{ccccccccccc}
f_0 & = & P_{0,0} & & & & & & & & \\
 & & & \searrow & & & & & & & \\
f_1 & = & P_{1,0} & \to & P_{1,1} & & & & & & \\
 & & & & & \searrow & & & & & \\
 & & \vdots & & \vdots & & \ddots & & & & \\
 & & & \searrow & & \searrow & & & \searrow & & \\
f_{n-1} & = & P_{n-1,0} & \to & \cdots & \cdots & & \to & P_{n-1,n-1} & & \\
 & & & \searrow & & & & & & \searrow & \\
f_n & = & P_{n,0} & \to & P_{n,1} & \to & \cdots & & P_{n,n-1} & \to & P_{n,n}
\end{array}$$

Satz 6.3. *Sei $f \in C^{n-1}(a,b)$, $a < t_0 < \cdots < t_n < b$.*
Für $t \in [a,b]$ gibt es ein $\tau = \tau(t) \in]a,b[$, so daß

$$f(t) - P(f|t_0, \ldots, t_n)(t) = \frac{f^{(n-1)}(\tau)}{(n-1)!}\omega(t)$$

mit $\omega = \prod_{i=0}^{n}(t - t_i)$ ist Newton-Polynom.

Beweis: Sei $t \in \{t_0, \ldots, t_n\}$

$$F(x) := f(x) - P(f|t_0, \ldots, t_n)(x) - k \cdot \omega(x)$$

$k = k(t)$ sei so bestimmt, daß $F(t) = 0$.

$$\begin{aligned}&\Rightarrow F(t_i) = 0, \; i = 0, \ldots, n \text{ und } F(t) = 0 \\ &\Rightarrow F \text{ hat } n+2 \text{ Nullstellen} \\ &\Rightarrow F' \text{ hat } n+1 \text{ Nullstellen} \\ &\Rightarrow F'' \text{ hat } n \text{ Nullstellen} \\ &\Rightarrow F^{(n+1)} \text{ hat mindestens eine Nullstelle, diese sei } \tau\end{aligned}$$

Wegen $P^{(n+1)}(f|t_0,\dots,t_n)=0$ folgt

$$0=F^{(n+1)}(\tau)=f^{(n+1)}(\tau)-k\cdot\underbrace{(n+1)!}_{\omega(n+1)}$$

$$\Rightarrow k=\frac{f^{(n+1)}(\tau)}{(n+1)!}$$

Korollar 6.4.

$$\|f-P(f|t_0,\dots,t_n)\|_\infty\le\sup_{\tau\in[0,n]}\frac{|f^{(n+1)}(\tau)|}{(n+1)!}\cdot\|\omega\|_\infty$$

Für festes n hängt der Fehler noch von ω ab, das heißt von der Lage der Stützstellen.

Satz von Faber. *Zu jeder Folge $\{t_0^{(n)},\dots,t_n^{(n)}\}_{n\in\mathbb{N}}$ von Knoten in $[a,b]$ existiert ein $f\in(a,b)$ so, daß*

$$P(f|t_0^{(n)},\dots,t_n^{(n)})$$

nicht gleichmässig gegen f konvergiert für $n\to\infty$.

6.2 Splines und Splineinterpolation

Nachteile der Polynominterpolation:

- starke Oszillation (bei hoher Knotenzahl)
- keine Konvergenz

Wunsch: Die Interpolation sei glatt und durchlaufe die Stützstellen ohne starke Oszillation

Definition 6.5. Sei $\Delta=\{t_0,\dots,t_{l+1}\}$ ein Gitter von $l+2$ Knoten mit $a=t_0<t_1<\dots<t_{l+1}=b$. Ein Spline der Ordnung k bezüglich Δ ist eine Funktion $s\in C^{k-2}[a,b]$ mit

$$s\big|_{[t_i,t_{i+1}]}=p_i,\ i=0,\dots,n$$

wobei $p_i\in\Pi_{k-1}$ ist.
$S_{k,\Delta}$: Raum aller Splines der Ordnung k bezüglich Δ. Es gilt $\Pi_{k-1}\big|_{[a,b]}\subset S_{k,\Delta}$.
Abgebrochene Potenzen vom Grad k:

$$(t-t_i)_+^k=\begin{cases}(t-t_i)^k\text{ für }t\ge t_i\\ 0\text{ sonst}\end{cases}$$

Satz 6.6.

$$B=\{1,t,t^2,\dots,t^{(k-1)},(t-t_1)_+^{k-1},(t-t_2)_+^{k-1},\dots,(t-t_l)_+^{(k-1)}\}$$

ist eine Basis von $S_{k,\Delta}$. Insbesondere gilt

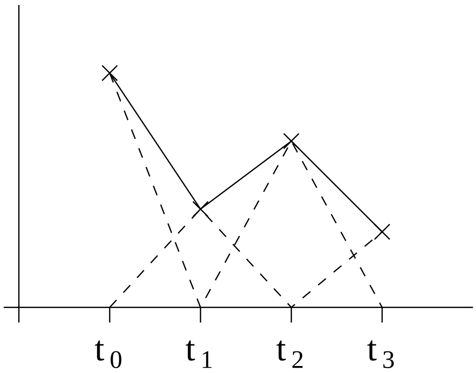
$$\dim S_{k,\Delta}=k+l$$

Die Basis B ist für numerische Anwendungen nicht *geeignet.*

Spline-Interpolation

Interpoliere $f: \mathbb{R} \to \mathbb{R}$ bezüglich $\Delta = \{t_0, \ldots, t_{l+1}\}$ durch einen Spline aus $S_{k,\Delta}$.
$k = 2$: lineare Spline

$$\dim S_{2,\Delta} = 2 + l = \sharp\{t_0, \ldots, t_{l+1}\}$$



In den Anwendungen (CAD, CAM) spielen *kubische* Splines der Ordnung 4 (Ordnung=Grad+1) jedoch eine gewichtige Rolle.
Ab jetzt: $k = 4$

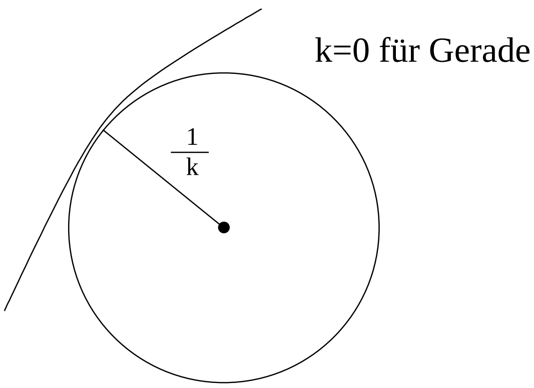
$$S_{4,\Delta} \subset C^2[a,b]$$

$$\dim S_{4,\Delta} - \sharp\text{Knoten} = l + 4 - (l + 2) = 2$$

$\Rightarrow$ 2 Freiheitsgrade bleiben durch die Interpolationsbedingungen unbestimmt.
Forderung: Der interpolierende kubische Spline minimiere die Krümmung unter allen interpolierenden C^2-Funktionen.
Sei $y: [a,b] \to \mathbb{R}$, $y \in C^2$:

$$k(t) = \frac{y''(t)}{(n + y'(t)^2)^{\frac{1}{2}}} \quad \text{Krümmung}$$

$\frac{1}{k}$ Radius des Krümmungskreises an $(t, y(t))$.



Für $|y'(t)| \ll 1$ gilt $k(t) \approx y''(t)$

$$\Rightarrow \|y'(t)\|_{L^2}^2 = \int y''(t)^2 dt = \langle y'', y'' \rangle_{L^2}$$

Satz 6.7. *$s \in S_{k,\Delta}$ interpoliere f in den Knoten $\Delta = \{t_0, \dots, t_{l+1}\}$.*
Sei $y \in C^2(a,b)$ eine beliebige andere Funktion die f interpoliert in Δ so, daß

$$s''(t)\,(y'(t) - s'(t))\,\Big|_{t=a}^{t=b} = 0$$

ist. Dann gilt

$$\|s''\|_{L^2} < \|y''\|_{L^2}$$

Beweis:

$$\begin{aligned}
\|y''\|^2_{L^2(a,b)} &= \|s'' + (y'' - s'')\|^2_{L^2} \\
&= \|s''\|^2_{L^2} + \underbrace{2\int_a^b s''(y'' - s'')dt}_{=:A} + \underbrace{\|y'' - s''\|^2_{L^2}}_{>0 \text{ falls } A=0} \\
&> \|s''\|^2_{L^2}
\end{aligned}$$

$$A = \underbrace{s''(y' - s')}_{0 \text{ nach Vor.}}\Big|_a^b - \int_a^b s'''(y' - s')dt \quad \text{partielle Integration}$$

s''' ist stückweise konstant auf $]t_i, t_{i+1}[$, $i = 0, \dots, l$.

$$s'''(t) = d_i \ \forall t \in]t_i, t_{i+1}[,\ i = 0, \dots, l$$

$$\Rightarrow A = -\sum_{i=0}^{l} d_i \int_{t_i}^{t_{i+1}} (y - s)'dt = -\sum_{i=0}^{l} d_i (y - s)\Big|_{t_i}^{t_{i+1}} = 0$$

dann $y(t_i) = f(t_i) = s(t_i)$, $i = 0, \dots, l+1$ $\surd$.

Korollar 6.8. *$s \in S_{4,\Delta}$ interpoliere f bezüglich $\Delta = \{t_0, \dots, t_{l+1}\}$. Außerdem erfülle s noch* eine *der folgenden Randbedingungen:*

i) $s'(a) = f'(a)$, $s'(b) = f'(b)$ *(vollständige Spline-Interpolation)*

ii) $s''(a) = s''(b) = 0$ *(natürliche Spline-Interpolation)*

iii) falls f periodisch mit Periode $b - a$:

$$s'(a) = s'(b),\ s''(a) = s''(b)$$

(periodische Spline-Interpolation)

Dann ist s eindeutig bestimmt. Für jede andere interpolierende Funktion $y \in C^2(a,b)$, die im Falle von i) und iii) denselben Randbedingungen genügt, gilt ferner:

$$\|s''\|_{L^2(a,b)} < \|y''\|_{L^2(a,b)}$$

Satz 6.9. *Sei $s \in S_{4,\Delta}$ der vollständig interpolierende Spline der Funktion $f \in C^4(a,b)$ bezüglich $\Delta = \{t_0, \dots, t_{l+1}\}$. Dann gilt*

$$\|f - s\|_\infty \le \tfrac{5}{384} h^4 \|f^{(4)}\|_\infty$$

mit $h = \max_{0 \le i \le l} |t_{i+1} - t_i|$.

Berechnung der interpolierenden Spline

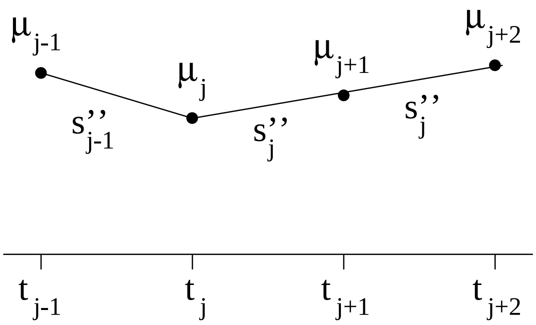
$$\Delta = \{t_0, \dots, t_{l+1}\},\ a = t_0 < t_1 < \dots < t_{l+1} = b$$

$$s \in S_{4,\Delta} \text{ interpoliert } f:\ s(t_j) = f(t_j) = f_j,\ j = 0, \dots, l+1$$

$$h_{j+1} = t_{j+1} - t_j,\ M_j = s''(t_j),\ j = 0, \dots, l+1 \quad \text{(Momente des Splines)}$$

Aus den Momenten kann der Spline rekonstruiert werden.

$$s_j = s\big|_{[t_j, t_{j+1}]} \in \Pi_3 \Rightarrow s_j'' = s''\big|_{[t_j, t_{j+1}]} \in \Pi_1 \quad \text{(affine lineare Funktion)}$$



$$\Rightarrow s_j''(t) = \frac{t_{j+1} - t}{h_{j+1}} + M_{j+1} \frac{t - t_j}{h_{j+1}}$$

Integration liefert:

$$\begin{aligned} s_j'(t) &= -M_j \frac{(t_{j+1} - t)^2}{2h_{j+1}} + M_{j+1} \frac{(t - t_j)^2}{2h_{j+1}} + A_j \\ s_j(t) &= M_j \frac{(t_{j+1} - t)^3}{6h_{j+1}} + M_{j+1} \frac{(t - t_j)^3}{6h_{j+1}} + A_j(t - t_j) + B_j \end{aligned} \tag{6.1}$$

für Integrationskonstanten A_j, B_j.
Es gelten:

$$\begin{aligned} f_j = s(t_j) &= M_j \frac{h_{j+1}^2}{6} + B_j \\ f_{j+1} = s(t_{j+1}) &= M_{j+1} \frac{h_{j+1}^2}{6} + A_j h_{j+1} + B_j,\ j = 0, \dots, l \end{aligned}$$

$$\left.\begin{array}{r} \Rightarrow B_j = f_j - M_j \frac{h_{j+1}^2}{6} \\ A_j = \frac{f_{j+1} - f_j}{h_{j+1}} - \frac{h_{j+1}}{6}(M_{j+1} - M_j) \end{array}\right\} \tag{6.2}$$

s_j hat nun die Darstellung

$$s_j(t) = \alpha_j + \beta_j(t - t_j) + \gamma_j(t - t_j)^2 + \delta_j(t - t_j)^3,\ j = 0, \dots, l$$

(Taylor-Entwicklung von s_j um t_j)

$$
\begin{aligned}
\Rightarrow \alpha_j &= s_j(t_j) = f_j \\
\beta_j &= s_j'(t_j) \overset{(6.1)}{=} -M_j \frac{h_{j+1}}{2} + A_j \\
&= \frac{f_{j+1} - f_j}{h_{j+1}} - \frac{2M_j + M_{j+1}}{6} h_{j+1} \\
\gamma_j &= \tfrac{1}{2} s_j''(t_j) = \tfrac{1}{2} M_j \\
\delta_j &= \tfrac{1}{6} s_j'''(t_j) = \tfrac{1}{6}(s'')'(t_j) \\
&= \tfrac{1}{6} \frac{M_{j+1} - M_j}{h_{j+1}}
\end{aligned}
$$

Damit ist s durch M_j und f_j vollständig bestimmt.
Die M_j erhalten wir aus

$$
s_{j-1}'(t_j) = s_j'(t_j) \tag{6.3}
$$

$$
\begin{aligned}
\text{(6.1) und (6.2)} \;\Rightarrow s_j'(t) &= -M_j \frac{(t_{j+1} - t)^2}{2h_{j+1}} + M_{j+1} \frac{(t - t_j)^2}{2h_{j+1}} \\
&\quad + \frac{f_{j+1} - f_j}{h_{j+1}} - \frac{h_{j+1}}{6}(M_{j+1} - M_j),\ j = 0, \ldots, l \\
\Rightarrow s_{j-1}'(t_j) &= \frac{f_j - f_{j-1}}{h_j} + \frac{h_j}{3} M_j + \frac{h_j}{6} M_{j-1} \\
= s_j'(t_j) &= \frac{f_{j+1} - f_j}{h_{j+1}} - \frac{h_{j+1}}{3} M_j - \frac{h_{j+1}}{3} M_{j+1}
\end{aligned}
$$

$$
\begin{aligned}
\text{(6.3)} \;\Rightarrow\; \frac{h_j}{6} M_{j-1} + \frac{h_j + h_{j+1}}{3} M_j + \frac{h_{j+1}}{6} M_{j+1} \\
= \frac{f_{j+1} - f_j}{h_{j+1}} - \frac{f_j - f_{j-1}}{h_j},\ j = 1, \ldots, l
\end{aligned} \tag{6.4}
$$

Das sind l Bedingungen für die $l+2$ Unbekannten $M_0, \ldots, M_{l+1}$. Je zwei weitere Bedingungen erhalten wir aus Korollar 6.8.

i) *vollständige Spline-Interpolation*

$$
\begin{aligned}
s'(a) &= f'(a) = f_0' \\
s'(b) &= f'(b) = f_{l+1}' \\
\Rightarrow s_0'(a) &= -M_0 \frac{h_0}{2} + \frac{f_1 - f_0}{h_1} - \frac{h_1}{6}(M_1 - M_0) \overset{!}{=} f_0'
\end{aligned}
$$

$$
\Rightarrow \frac{h_1}{3} M_0 + \frac{h_1}{6} M_1 = \frac{f_1 - f_0}{h_1} - f_0'
$$

analog:

$$
\frac{h_{l+1}}{6} M_l + \frac{h_{l+1}}{3} M_{l+1} = f_{l+1}' - \frac{f_{l+1} - f_l}{h_{l+1}}
$$

ii) *natürliche Spline-Interpolation*

$$
\begin{aligned}
s''(a) &= s''(b) = 0 \\
\Rightarrow M_0 &= M_{l+1} = 0
\end{aligned}
$$

Führen wir die Abkürzungen

$$\lambda_j = \frac{h_{j+1}}{h_j + h_{j+1}},\ \mu_j = 1 - \lambda_j = \frac{h_j}{h_j + h_{j+1}}$$

sowie

$$d_j = \frac{6}{h_j + h_{j+1}} \left(\frac{f_{j+1} - f_j}{h_{j+1}} - \frac{f_j - f_{j+1}}{h_j} \right),\ j = 1, \ldots, l$$

ein, dann wird aus (6.4)

$$\mu_j \cdot M_{j-1} + 2M_j + \lambda_j M_j = d_j,\ j = 1, \ldots, l$$

Definieren wir zusätzlich

im Fall i)

$$\begin{aligned} \lambda_0 &= 1 \\ d_0 &= \frac{6}{h_1} \left(\frac{f_1 - f_0}{h_1} - f_0' \right) \\ \mu_{l+1} &= 1 \\ d_{l+1} &= \frac{6}{h_{l+1}} \left(f_{l+1}' - \frac{f_{l+1} - f_l}{h_{l+1}} \right) \end{aligned}$$

im Fall ii)

$$\begin{aligned} \lambda_0 &= d_0 = 0 \\ \mu_{l+1} &= d_{l+1} = 0 \end{aligned}$$

dann erhalten wir das Gleichungssystem:

$$\underbrace{\begin{pmatrix} 2 & \lambda_0 & & & \\ \mu_1 & \ddots & \ddots & 0 & \\ & \ddots & \ddots & \ddots & \\ & 0 & \ddots & \ddots & \lambda_l \\ & & & \mu_{l+1} & 2 \end{pmatrix}}_{A_l} \begin{pmatrix} M_0 \\ \vdots \\ M_{l+1} \end{pmatrix} = \begin{pmatrix} d_0 \\ \vdots \\ d_{l+1} \end{pmatrix}$$

A_l ist regulär (kubische Spline-Interpolation ist eindeutig). Außerdem gilt $\lambda_i \geq 0$, $\mu_i \geq 0$.

$$\begin{aligned} &\lambda_i + \mu_i = 1 \\ &\Rightarrow A_l \text{ ist diagonaldominant} \\ &\Rightarrow LR\text{-Zerlegung ist stabil} \end{aligned}$$

6.3 Freiformkurven, Beziertechnik

Anwendung:

- Entwurf am Computer (CAD)

- Geometrische Aspekte einer Kurve spielen eine entscheidende Rolle.

Zentrale Frage: Welche Parameter beeinflußen die Gestalt einer Kurve?
$P: [a,b] \overset{a \leq b}{\to} \mathbb{R}^d$ ist polynomiale Kurve vom Grad n, falls

$$P(t) = \sum_{i=0}^{n} a_i t^i \text{ mit } a_i \in \mathbb{R}^d,\, a_n \neq 0$$

Π_k^d ist die Menge aller polynomialen Kurven im $\mathbb{R}^d$ vom Grad n.

Bernstein-Polynome

$$\varphi: [a,b] \to [0,1]$$
$$\varphi(t) = \frac{t-a}{b-a} \text{ bijektiv}$$

Definition 6.10. Das i-te Bernstein-Polynom $B_i^n \in \Pi_n^1$ bezüglich $[0,1]$ ist definiert durch

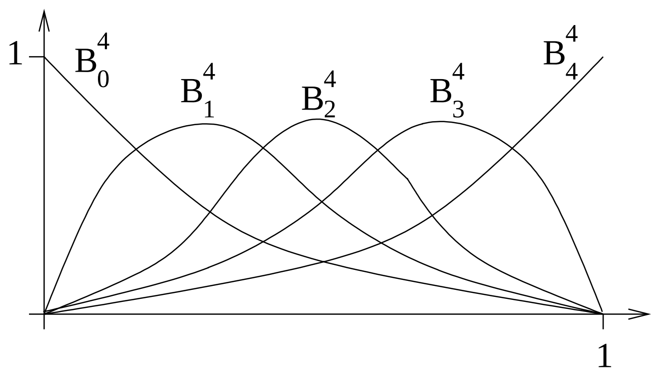
$$B_i^n(\lambda) := \binom{n}{i}(1-\lambda)^{n-i}\lambda^i,\, i = 0,\ldots,n$$

$B_i^n(\cdot; a,b)$, das i-te Bernstein-Polynom bezüglich $[a,b]$, ist definiert durch

$$B_i^n(t;a,b) := B_i^n(\varphi(t)) = \frac{1}{(b-a)^n}\binom{n}{i}(t-a)^i(b-t)^{n-i},\, i = 0,\ldots,n$$

Es gilt:

$$\begin{aligned}1 = ((1-\lambda)+\lambda)^n &= \sum_{i=0}^{n}\binom{n}{i}(1-\lambda)^{n-i}\lambda^i \\ &= \sum_{i=0}^{n} B_i^n(\lambda)\ \forall\lambda \in \mathbb{R}\end{aligned}$$



Satz 6.11. *Es gelten folgende Aussagen:*

1. $\lambda = 0$ *ist* i*-fache Nullstelle von* B_i^n
2. $\lambda = 1$ *ist* $(n-i)$*-fache Nullstelle von* B_i^n
3. $B_i^n(\lambda) = B_{n-i}^n(1-\lambda)$

4. $(1-\lambda)B_0^n(\lambda) = B_0^{n-i}(\lambda)$, $\lambda B_n^n(\lambda) = B_{n+1}^{n-1}(\lambda)$

5. $B_i^n(\lambda) \geq 0$, $\lambda \in [0,1]$, $\sum_{i=0}^n B_i^n(\lambda) = 1$

6. B_i^n *hat in* $[0,1]$ *genau ein Maximum, und zwar bei* $\lambda = \frac{i}{n}$

7. $B_i^n(\lambda) = \lambda \cdot B_{i-1}^{n-1}(\lambda) + (1-\lambda)B_i^{n-i}(\lambda)\ \forall \lambda \in \mathbb{R}$, $i = 1, \ldots, n$

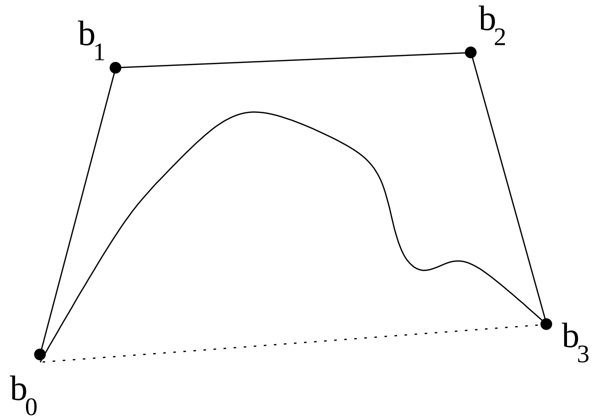
8. $\{B_0^n, \ldots, B_n^n\}$ *ist Basis von* Π_n^1

Analoge Aussagen gelten für $B_i(i; a, b)$*. Die Maxima liegen bei* $a + \frac{i}{n}(b-a)$*.*

$$P \in \Pi_n^d \Rightarrow P(t) = \sum_{i=0}^n b_i B_i^n(t; a, b),\ b_i \in \mathbb{R}^d$$

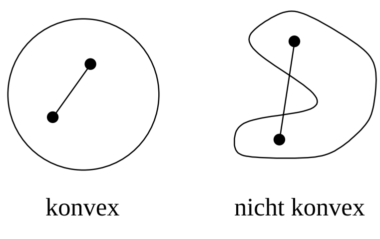
b_i Kontroll-/Bezier-Punkte von P

Der Streckenzug $b_0, b_1, \ldots, b_n$ im $\mathbb{R}^d$ heißt Bezier-Polygon.



Eine Menge $K \subset \mathbb{R}^d$ heißt *konvex*, falls gilt:

$$x, y \in K \Rightarrow \mu \cdot x + (1-\mu)y \in K\ \forall \mu \in [0,1]$$



Sei $A \subset \mathbb{R}^d$ beliebig:

$$\mathrm{co}(A) = \bigcap \{B \subset \mathbb{R}^d \,\big|\, B \text{ konvex}, A \subset B\}$$

$\mathrm{co}(A)$ heißt *konvexe Hülle* von A.

Satz 6.12. $P = \sum_{i=0}^n b_i B_i^n(\cdot; a, b) \in \Pi_n^d$

$$\Rightarrow P(t) \in \mathrm{co}(\{b_0, \ldots, b_n\})\ \forall t \in [a, b]$$

Definition 6.13. Zu $P = \sum_{i=0}^n b_i B_i^n$ definieren wir Teilpolynome $b_i^k \in \Pi_k^d$, $i = 0, \ldots, n-k$ durch

$$b_i^k(t; a, b) = \sum_{j=0}^k b_{i+j} B_j^k(t; a, b)$$

d.h. b_i^k wird durch die Bezierpunkte $b_i, \ldots, b_{i+k}$ definiert. ($b_i^0 = b_i$)

Satz 6.14. *Sei* $P = \sum_{i=0}^n b_i B_i^n(\cdot; a, b)$

$$\Rightarrow P^{(k)}(t) = \frac{1}{b-a} \frac{n!}{(n-k)!} \Delta^k b_0^{n-k}(t),\ k = 0, \ldots, n$$
$$\Delta b_i^k = b_{i+1}^k - b_i^k,\ \Delta^k = \Delta(\Delta^{k-1}),\ k = 1, \ldots, n$$

Beispiel:

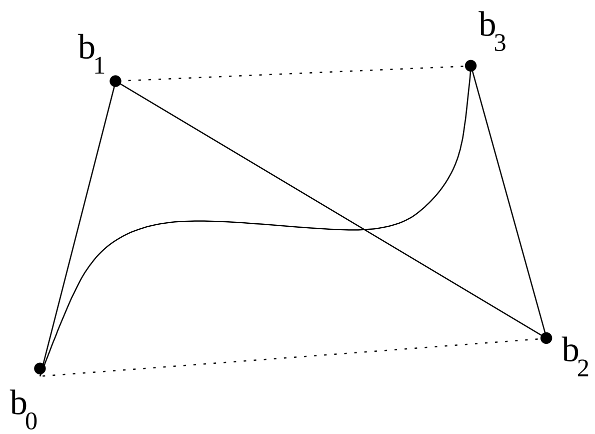
$$\Delta^2 b_i^k = \Delta(b_{i+1}^k - b_i^k) = b_{i+2}^k - 2b_{i+1}^k + b_i^k$$

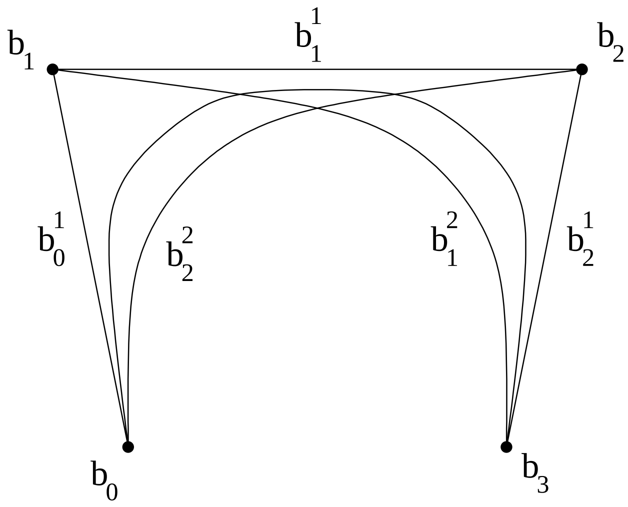
Folgerung:

$$P^{(k)}(a) = \frac{1}{b-a} \frac{n!}{(n-k)!} \Delta^k b_0$$
$$P^{(k)}(b) = \frac{1}{b-a} \frac{n!}{(n-k)!} \Delta^k b_{n-k}$$

speziell:

$$P(a) = b_0,\ P(b) = b_n$$
$$P'(a) = n(b_1 - b_0),\ P'(b) = n(b_n - b_{n-1})$$





Siehe z.B.

`http://i33www.ira.uka.de/`
`applets/mocca/html/noplugin/inhalt.html`

Algorithmus von Casteljau

- schnelle Approximation von $P(t)$
- Approximation von $\{P(t) \big| t \in [a,b]\}$ durch Polygonzüge ("Konstruktion von P")

Lemma 6.15. *Die Teilpolynome erfüllen*

$$b_i^k(t) = (1-\varphi(t))b_i^{k-1}(t) + \varphi(t)b_{i+1}^{k-1}(t)$$

für $k = 0, \dots, n;\ i = 0, \dots, n-k.$

Bemerkung: Für b_i^k bzgl. $[0,1]$ gilt natürlich $\varphi(t) = t$.

Wegen $b_i^0(t) = b_i$ können wir $P(t)$ aus $\{b_0, \dots, b_n\}$ berechnen:

$$
\begin{array}{ccccccccccc}
b_n^0 & & & & & & & & & & \\
 & \overset{\varphi(t)}{\searrow} & & & & & & & & & \\
b_{n-1}^0 & \overset{1-\varphi(t)}{\rightarrow} & b_{n-1}^1 & & & & & & & & \\
 & \overset{\varphi(t)}{\searrow} & & \overset{\varphi(t)}{\searrow} & & & & & & & \\
b_{n-2}^0 & \overset{1-\varphi(t)}{\rightarrow} & b_{n-2}^1 & \overset{1-\varphi(t)}{\rightarrow} & b_{n-2}^2 & & & & & & \\
\vdots & & & & \vdots & \ddots & & & & & \\
 & & & & & & b_2^{n-2} & & & & \\
 & & & & & & & \searrow & & & \\
b_1^0 & \rightarrow & & \rightarrow & \vdots & & b_1^{n-2} & \rightarrow & b_1^{n-1} & & \\
 & \overset{\varphi(t)}{\searrow} & & & \vdots & & & \searrow & & \searrow & \\
b_0^0 & \overset{1-\varphi(t)}{\rightarrow} & b_0^1 & \rightarrow & \cdots & \cdots & b_0^{n-2} & \rightarrow & b_0^{n-1} & \rightarrow & b_0^n = P(t)
\end{array}
$$

7 Diskrete Fouriertransformation (DFT)

Die DFT $D_n : \mathbb{C}^n \to \mathbb{C}^n$ ist definiert durch:

$$(D_n f)_j = \frac{1}{n} \sum_{k=0}^{n-1} f_k \omega_n^{-jk}$$

mit $\omega_n = e^{\mathrm{i}\frac{2\pi}{n}} = \cos(\frac{2\pi}{n}) + \mathrm{i} \cdot \sin(\frac{2\pi}{n})$. $\mathrm{i}^2 = -1$

Bemerkung: In diesem Abschnitt beginnt der Index von Vektoren und Matrizen bei 0.

Matrix-Darstellung

$$D_n = \frac{1}{n} \begin{pmatrix} 1 & 1 & \cdots & 1 \\ 1 & \omega_0^{-1} & \cdots & \omega_n^{-(n-1)} \\ \vdots & \vdots & \ddots & \vdots \\ 1 & \omega_n^{-(n-1)} & \cdots & w_n^{-(n-1)(n-1)} \end{pmatrix}$$

Satz 7.1.

$$\begin{aligned} D_n^{-1} &= n\overline{D_n} \\ \Rightarrow f &= D_n^{-1} D_n f = \sum_{j=0}^{n-1} (D_n f)_j \omega_j \\ \textit{mit } \omega_j &= \left(1, \omega_n^j, \omega_n^{2j}, \ldots, \omega_n^{(n-1)j}\right) \end{aligned}$$

Mit j nimmt die Anzahl der Oszillationen (Vorzeichenwechsel) zu, das heißt die Frequenz erhöht sich.

Interpretation: $(D_n f)_j$ mißt, wie stark die Schwingung ω_j in f enthalten ist. Die DFT führt eine Spektralanalyse durch. Die DFT ist zentral für die moderne Kommunikations- und Nachrichtentechnik, z.B. zur Bildkompression (JPEG).

FFT

Die direkte Auswertung von $D_n f$ erfordert n^2 komplexe Multiplikationen. Wir werden D_n faktorisieren:

$$D_n = \frac{1}{n} A_p \cdot \ldots \cdot A_1 P_n^t \text{ für } n = 2^p$$

P_n Permutation, $A_j \in \mathbb{C}^{n \times n}$: 2 Einträge pro Zeile $\neq 0$. Auswertung von $D_n f$ über Faktorisierung: $n \cdot \overbrace{\log_2 n}^{p}$ komplexe Multiplikationen.
Sei $\tilde{D_n} = n \cdot D_n$. Betrachte $n = 4$:

$$\tilde{D_4} = \begin{pmatrix} 1 & 1 & 1 & 1 \\ 1 & -\mathrm{i} & -1 & \mathrm{i} \\ 1 & -1 & 1 & -1 \\ 1 & \mathrm{i} & -1 & -\mathrm{i} \end{pmatrix}, \pi_4 = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix}$$

π_4 ist eine Permutation, die zuerst die geraden und dann die ungeraden Indizes anordnet.

$$\pi_4 \cdot \begin{pmatrix} v_0 \\ v_1 \\ v_2 \\ v_3 \end{pmatrix} = \begin{pmatrix} v_0 \\ v_2 \\ v_1 \\ v_3 \end{pmatrix}$$

$$\tilde{D_4} \cdot \pi_4 = \left(\begin{array}{cc|cc} 1 & 1 & 1 & 1 \\ 1 & -1 & -\mathrm{i} & \mathrm{i} \\ \hline 1 & 1 & -1 & -1 \\ 1 & -1 & \mathrm{i} & -\mathrm{i} \end{array}\right), \ \tilde{D_2} = \begin{pmatrix} 1 & 1 \\ 1 & -1 \end{pmatrix}$$

$$\tilde{D_4} = \begin{pmatrix} I_2 & \Omega_2 \\ I_2 & \Omega_2 \end{pmatrix} \cdot \begin{pmatrix} \tilde{D_2} & 0 \\ 0 & \tilde{D_2} \end{pmatrix} \cdot \pi_4^t \tag{7.1}$$

mit $I_2 = \operatorname{diag}(1, \ldots, 1) \in P^{l \times l}$, $\Omega_2 = \begin{pmatrix} 1 & 0 \\ 0 & -\mathrm{i} \end{pmatrix} = \operatorname{diag}(1, \omega_4^{-1})$.

Die allgemeine Version von (7.1) drückt $\tilde{D_n}$ durch $\tilde{D_{\frac{n}{2}}}$ aus.

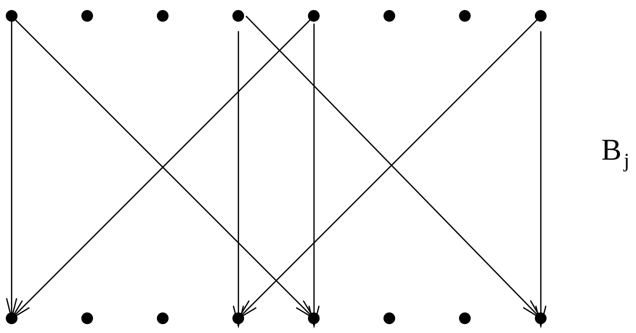
$$\pi_n : \mathbb{C}^n \to \mathbb{C}^n, \, n = 2m$$

$$\omega = \pi^t V \Leftrightarrow \omega = (V_0, V_2, \ldots, V_{n-2}, V_1, \ldots, V_{n-1})$$

Satz 7.2. *Sei $n = 2m$ und $\Omega_m = \operatorname{diag}\left(1, \omega_n^{-1}, \omega_n^{-2}, \ldots, \omega_n^{-(n-1)}\right)$, $\omega_n = e^{\mathrm{i}\frac{2\pi}{n}}$. Dann gilt*

$$\tilde{D_n} \cdot \pi_n = \left(\begin{array}{c|c} \tilde{D_m} & \Omega_m \tilde{D_m} \\ \hline \tilde{D_m} & -\Omega_m \tilde{D_m} \end{array}\right) = \left(\begin{array}{c|c} I_m & \Omega_m \\ \hline I_m & \Omega_m \end{array}\right) \begin{pmatrix} \tilde{D_m} & 0 \\ 0 & \tilde{D_m} \end{pmatrix}$$

Die Matrizen $B_n = \begin{pmatrix} I_m & \Omega_m \\ I_m & -\Omega_m \end{pmatrix}$ heißen "Butterfly"-Matrizen.



Satz 7.3 (Cooley-Tuckey-Basis - 2. Faktorisierung). *Sei $n = 2^p$. Dann ist*

$$\tilde{D_n} = A_p^{(p)} \cdot \ldots \cdot A_1^{(p)} P_n^t \tag{7.2}$$

mit $P_2 = I_2$, $P_n = \pi_n \begin{pmatrix} P_{n/2} & 0 \\ 0 & P_{n/2} \end{pmatrix}$.

$$A_j = \operatorname{blockdiag}(\underbrace{B_{2j}, \ldots, B_{2^j}}_{2^{p \cdot j}\ Blöcke}) = \begin{pmatrix} B_{2^j} & & 0 \\ & \ddots & \\ 0 & & B_{2^j} \end{pmatrix}$$

B_{2^j} *Butterfly-Matrix,* $j = 1, \ldots, p$.
Die Realisierung von $D_n f$ *basierend auf (7.2) heißt* Cooley-Tuckey-Algorithmus.

Aspekte des CT-Algorithmus

2 Phasen:

(a) Permutationsphase ($f = P_n^t f$)

(b) Multiplikationsphase ($f = A_p \cdot A_{p-1} \cdots A_1 f$)

(a) Permutationsphase:

$$P_8^t f = (f_0, f_4, f_2, f_6, f_1, f_5, f_3, f_7)^t$$
$$P_{16}^t f = (f_0, f_8, f_4, \underbrace{f_{12}}_{\substack{12_{10}=1100_2 \\ 3_{10}=0011_2}}, f_2, f_{10}, f_6, f_{14}, f_1, f_9, \underbrace{f_5}_{\substack{5_{10}=0101_2 \\ 10_{10}=1010_2}}, f_{13}, f_{11}, f_7, f_{15})^t$$

Satz 7.4. *Sei* $n = 2^p$. *Die Funktion* $r_n = \{0, \ldots, n-1\} \to \{0, \ldots, n-1\}$ *sei definiert durch*

$$(P_n^t f)_k = f_{r_n(k)}, \; k = 0, \ldots, n-1$$

Dann gilt

$$r_n(2^{p-1}b_{p-1} + 2^{p-2}b_{p-2} + \cdots + 2b_1 + b_0) = 2^{p+1}b_0 + 2^{p-2}b_1 + \cdots + 2b_{p-2} + b_{p-1}$$

für $b_j \in \{0, 1\}$, $j = 0, \ldots, p-1$. *Deshalb heißt* P_n Bitspiegelung.

(b) Multiplikationsphase:
B_l Butterfly-Matrix der Ordnung $l = 2l_*$:

$$y = B_l z \Leftrightarrow \begin{pmatrix} y_0 \\ y_n \end{pmatrix} = \begin{pmatrix} z_0 + \Omega_{l_*} z_n \\ z_0 - \Omega_{l_*} z_n \end{pmatrix}$$

mit $y = \begin{pmatrix} y_0 \\ y_n \end{pmatrix}$, $z = \begin{pmatrix} z_0 \\ z_n \end{pmatrix}$; $y_0, y_n, z_0, z_n \in \mathbb{C}^{l_*}$.

Wir betrachten nun die Anwendung von $A_j^{(p)}$ auf $x \in \mathbb{C}^n$.

$$y = A_j^{(p)} x \Leftrightarrow y_{l+r} = B_{l+r}; \; l = 2^j, \; r = 2^{p-j}$$

mit $y_{l+r}, x_{l+r} \in \mathbb{C}^{l+r}$.

$$(x_{l+r})_{j,k} = x_{kl+j}, \; x_{l+r} = \begin{pmatrix} x_0 & \cdots & x_l & \cdots & x_{(r-1)l} \\ \vdots & & \vdots & & \vdots \\ x_{l-1} & \cdots & x_{2l-1} & \cdots & x_{rl-1} \end{pmatrix}$$

CT-Algorithmus

$f \in \mathbb{C}^n$, $n = 2^p$, $f = \frac{1}{n} P_n^t f$ (Bitspiegelung)

```
for j = 1,...,p
{
  l = 2^j, r = l/2, l_* = l/2
  for q = 0,...,l_* - 1
  {
    ω = cos(2πq/l) - i · sin(2πq/l)
    for k = 0,...,r - 1
    {
      τ = ω · f_{kl+j+l_*}
      f_{kl+j+l_*} = f_{kl+j} + τ
      f_{kl} = f_{kl+j} - τ
    }
  }
}
```

Aufwand (ohne Berechnung von ω, ohne reelle Multiplikation):

$$\frac{1}{2} n \log_2 n \text{ komplexe Multiplikationen}$$

8 Numerische Quadratur

Ziel: Berechnung des Riemann-Integrals

$$I(f) := I_a^b(f) = \int_a^b f(x)dx,$$

wenn keine Stammfunktion bekannt ist. Die zahlenmäßige Bestimmung von $I(f)$ heißt *numerische Integration* oder *numerische Quadratur.*

Quadraturformeln

$I : C(a,b) \to \mathbb{R}$, $f \mapsto I(f)$ ist eine *positive Linearform.*

linear:

$$I(\alpha f + \beta g) = \alpha \cdot I(f) + \beta \cdot I(g) \; \forall f, g \in C(a,b),\; \alpha, \beta \in \mathbb{R}$$

positiv:

$$f \geq 0 \Rightarrow I(f) \geq 0$$

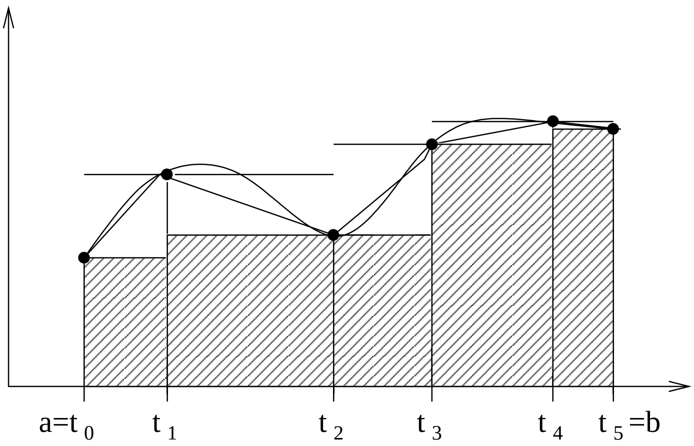
additiv: Sei $\tau \in [a,b]$:

$$I(f) = I_a^\tau(f) + I_\tau^b(f)$$

Ziel der numerischen Quadratur ist die Konstruktion positiver Linearformen $\hat{I} : C(a,b) \to \mathbb{R}$, $f \mapsto \hat{I}(f)$ mit

$$\hat{I}(f) - I(f) \text{ “klein”}$$

Beispiel: f stetig:



Zerlege $[a, b]$ in n Teilintervalle $[t_i, t_{i+1}]$, $i = 0, \ldots, n-1$ mit $a = t_0 < t_1 < \cdots < t_n = b$:

$$\hat{I}_n(f) = \sum_{i=0}^{n-1} \tfrac{h_i}{2}(f(t_i) + f(t_{i+1})),\ h_i = t_{i+1} - t_i$$

$\hat{I}_n(f)$ heißt *Trapezsumme.*

$$R_{\mathrm{u}}^{(n)}(f) := \sum_{i=0}^{n-1} h_i \cdot \min_{t\in[t_i,t_{i+1}]} f(t) \quad \text{Riemannsche Untersumme}$$

$$R_{\mathrm{o}}^{(n)}(f) := \sum_{i=0}^{n-1} h_i \cdot \max_{t\in[t_i,t_{i+1}]} f(t) \quad \text{Riemannsche Obersumme}$$

Sei $h = \max\{h_i\}$.

$$\begin{array}{ccccc} R_{\mathrm{u}}^{(n)}(f) & \leq & \hat{I}_n(f) & \leq & R_{\mathrm{o}}^{(n)}(f) \\ \downarrow h \to 0 & & & & \downarrow h \to 0 \\ I(f) & & & & I(f) \end{array}$$

$$\Rightarrow \lim_{\substack{h\to 0 \\ (n\to\infty)}} \hat{I}_n(f) = I(f)$$

Definition 8.1. Eine Quadraturformel $\hat{I}$ zur Berechnung von $\int_a^b f(x)dx$ hat die Form:

$$\hat{I}(f) := (b-a) \sum_{i=0}^{n} \lambda_i f(t_i)$$

mit den Knoten $t_0, \ldots, t_n$ und den Gewichten $\lambda_0, \ldots, \lambda_n$, wobei $\sum_{i=0}^n \lambda_i = 1$ ist. Es gelten:

- $\hat{I}(1) = b - a = I(1)$ (konstante Funktionen werden exakt integriert)
- $\hat{I}$ positiv $\Leftrightarrow \lambda_i \geq 0$; $i = 0, \ldots, n$

Newton-Cotes-Formeln

$$\hat{I}_n(f) := \int_a^b P(f|t_0,\dots,t_n)(t)dt,\ a = t_0 < t_1 < \dots < t_n = b$$

$$P(f|t_0,\dots,t_n) = \sum_{i=0}^{n} f(t_i)L_{n,i}(t)$$

$L_{n,i}$ sind die Lagrange-Polynome zu $t_0,\dots,t_n$.

$$\Rightarrow \hat{I}_n(f) = (b-a)\cdot \sum_{i=0}^{n} \underbrace{\tfrac{1}{b-a}\int_a^b L_{n,i}(t)dt}_{=:\lambda_{n,i}} \cdot f(t_i)$$

$\lambda_{n,i}$ hängen von den Knoten ab. $\hat{I}_n$ ist exakt für Polynome bis zum Grad n:

$$\hat{I}(P) = I(P)\ \forall P \in \Pi_n$$

Lemma 8.2. *Zu $n+1$ paarweise verschiedenen Knoten $t_0,\dots,t_n$ gibt es genau eine Quadraturformel*

$$\hat{I}(f) = (b-a)\sum_{i=0}^{n} \lambda_i f(t_i),$$

die für alle $P \in \Pi_n$ exakt ist.

Beweis: Die Existenz haben wir oben gezeigt.
Eindeutigkeit: $L_{n,i}$ Lagrange-Polynome zu $t_0,\dots,t_n$.

$$I(L_{n,i}) = \hat{I}(L_{n,i}) = (b-a)\cdot\sum_{j=0}^{n} \lambda_i L_{n,i}(t_i) = (b-a)\cdot\lambda_i$$

$$\Rightarrow \lambda_i = \tfrac{1}{b-a}\cdot I(L_{n,i}) = \tfrac{1}{b-a}\cdot\int_a^b L_{n,i}(t)dt = \lambda_{n,i} \quad \checkmark$$

Bei äquidistanten Knoten $t_i = a + ih$; $i = 0,\dots,n$, $h = \frac{b-a}{n}$ heißen die oben konstruierten Quadraturformeln *Newton-Cotes-Formeln*. Ihre Gewichte sind

$$\begin{aligned}\lambda_{n,i} &= \tfrac{1}{b-a}\cdot\int_a^b \prod_{\substack{j=0\\ j\neq i}}^{n} \frac{t-t_j}{t_i-t_j}dt \\ &= \tfrac{1}{n}\int_0^n \prod_{\substack{j=0\\ j\neq i}}^{n} \frac{s-j}{i-j}ds \quad \text{[Subst.: } s = (t-a)/h]\end{aligned}$$

$$\Rightarrow \lambda_{n,i} \in \mathbb{Q}$$

n	Gewichte	Fehler	Namen
1	$\frac{1}{2}\ \frac{1}{2}$	$\frac{h^3}{12}f''(\tau)$	Trapezregel
2	$\frac{1}{6}\ \frac{4}{6}\ \frac{1}{6}$	$\frac{h^5}{90}f^{(4)}(\tau)$	Simpson-Regel/Keplersche Faßregel
3	$\frac{1}{8}\ \frac{3}{8}\ \frac{3}{8}\ \frac{1}{8}$	$\frac{3h^5}{80}f^{(4)}(\tau)$	Newtonsche $\frac{3}{8}$-Regel
4	$\frac{7}{90}\ \frac{32}{90}\ \frac{12}{90}\ \frac{32}{90}\ \frac{7}{90}$	$\frac{8h^7}{945}f^{(6)}(\tau)$	Milne-Regel

Für $n \leq 7$ sind die Gewichte positiv.

Lemma 8.3. *Sei $f \in C^2(a,b)$. Dann gilt für die Trapezregel*

$$T(f) = \tfrac{b-a}{2}(f(a) + f(b))$$

die Fehlerdarstellung

$$T(f) - I(f) = \tfrac{h^3}{12} \cdot f''(\tau)$$

für ein $\tau \in [a,b]$.

Beweis: Sei $P = P(f\big|a,b) \in \Pi_1$:

$$\begin{aligned}
f(t) &= P(t) + \tfrac{1}{2} f''(\tau(t)) \cdot (t-a)(t-b) \quad \text{(Satz 6.3)} \\
\Rightarrow I(f) &= I(P) + \tfrac{1}{2} \int_a^b f''(\tau(t)) \cdot \underbrace{(t-a)(t-b)}_{\leq 0} \, dt \\
&= T(f) + \tfrac{1}{2} f''(\xi) \underbrace{\int_a^b (t-b)(t-a) \, dt}_{= -\frac{(b-a)^3}{6}} \quad \text{(MWS d. I.)}
\end{aligned}$$

$$\Leftrightarrow T(f) - I(f) = \tfrac{(b-a)^3}{12} f''(\xi) \quad \checkmark$$

Mittelwertsatz der Integralrechnung. *Seien f, φ stetig, $\varphi \geq 0$ (oder $\varphi \leq 0$):*

$$\Rightarrow \int_a^b f(a)\varphi(x)dx = f(\xi) \cdot \int_a^b \varphi(x)dx$$

für ein $\xi \in [a,b]$.

Lemma 8.4 (Die Keplersche Faßregel).

$$S(f) = \tfrac{b-a}{6} \left(f(a) + 4f(\tfrac{a+b}{2}) + f(b) \right)$$

ist auch für Polynome vom Grad 3 exakt. Für $f \in C^4(a,b)$ gilt mit $h = (b-a)/2$:

$$S(f) - I(f) = \frac{f^{(4)}(\tau)}{90} \cdot h^5$$

Beweis: Sei $Q \in \Pi_3$ beliebig. Sei

$$\begin{aligned}
P &= P(Q\big|a, \tfrac{a+b}{2}, b) \in \Pi_2 \\
\Rightarrow Q(t) &= P(t) + \underbrace{\frac{Q'''(\tau(t))}{6}}_{\gamma = \text{const.}} \underbrace{(t-a)(t-\tfrac{a+b}{2})(t-b)}_{=\omega(t)} \quad \text{(Satz 6.3)}
\end{aligned}$$

$$\Rightarrow \int_a^b Q(t)dt = \underbrace{\int_a^b P(t)dt}_{=S(Q)} + \gamma \int_a^b \omega(t)dt$$

$$\int_a^b \omega(t)dt = K(a,b) \underbrace{\int_{-1}^1 (x-1)x(x+1)dx}_{=0} \quad \left[\text{Subst.: } x = \tfrac{2t-a-b}{b-a}\right]$$

$$\Rightarrow \int_a^b Q(t)dt = S(Q) \; \forall Q \in \Pi_3 \quad \checkmark$$

Zusammengesetzte Formeln

Lemma 8.5. *Seien $h = (b-a)/n$, $n \in \mathbb{N}$, $t_i = a + ih$, $i = 0, \ldots, n$. Dann hat die Trapezsumme*

$$T(h) = h \cdot \left(\tfrac{1}{2}(f(a) + f(b)) + \sum_{i=1}^{n-1} f(t_i) \right)$$

für $f \in C^2(a,b)$ den Fehler

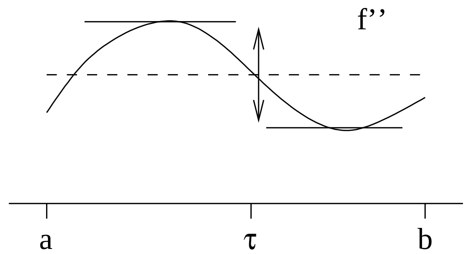
$$T(h) - \int_a^b f(x)dx = \frac{(b-a)h^2}{12} f''(\tau)$$

für ein $\tau \in (a,b)$.

Beweis: $T_i = \frac{h}{2}(f(t_i) + f(t_{i+1}))$ (Trapezregel bzgl. $[t_i, t_{i+1}]$)

$$T(h) = \sum_{i=0}^{n-1} T_i$$

$$\begin{aligned} T(h) - \int_a^b f(x)dx &= \sum_{i=0}^{n-1} \left(T_i - \int_{t_i}^{t_{i+1}} f(x)dx \right) \\ [\text{Lemma 8.3}] &= \sum_{i=0}^{n-1} \tfrac{h^3}{12} f''(\tau_i) \\ &= \frac{(b-a)h^2}{12} \cdot \tfrac{1}{n} \cdot \sum_{i=0}^{n-1} f''(\tau_i) \end{aligned}$$



$$\min_{t \in [a,b]} f''(t) \le \tfrac{1}{n} \cdot \sum_{i=0}^{n-1} f''(\tau_i) \le \max_{t \in [a,b]} f''(t)$$

Nach dem Zwischenwertsatz für stetige Funktionen gilt:

$$\exists \tau \in (a,b): \; f''(\tau) = \frac{1}{n} \sum_{i=0}^{n-1} f''(\tau_i) \quad \surd$$

Seien n gerade und S_i die zusammengesetzte Simpson-Regel bzgl. $[t_{2i}, t_{2i+2}]$, $i = 0, \ldots, \frac{n}{2} - 1$.

$$\begin{aligned} S(h) = \sum_{i=0}^{\frac{n}{2}-1} S_i = \sum_{i=0}^{\frac{n}{2}-1} \tfrac{2h}{6} \left(f(t_{2i}) + 4f(t_{2i+1}) + f(t_{2i+2}) \right) \\ = \tfrac{h}{3} \left[f(a) + 4f(a+h) + 2f(a+2h) + \cdots \right. \\ \left. \cdots + 2f(b-2h) + 4f(b-h) + f(b) \right] \end{aligned}$$

$$S(h) - \int_a^b f(x)dx = \frac{b-a}{180} h^4 f^{(4)}(\xi) \text{ für ein } \xi \in (a,b),\ f \in C^4(a,b)$$

Gauß-Quadratur

$I(f) = \int_a^b f(x)w(x)dx$, $w(x) > 0$ stetig, $\int_a^b x^k w(x)dx < \infty\ \forall k \in \mathbb{N}$. $a = -\infty$ und / oder $b = \infty$ ist zugelassen.
Ziel: Quadraturformeln

$$\hat{I}_n(f) = \sum_{i=0}^{n} \lambda_{n,i} f(\tau_{n,i})$$

die Polynome möglichst hohen Grades exakt integrieren.
Freiheitsgrade: $n+1$ Gewichte + $n+1$ Stützstellen.
Hoffnung: Es gibt $\hat{I}_n$ mit $\hat{I}_n(p) = I(f)\ \forall p \in \Pi_{2n+1}$.

Lemma 8.6. *Sei $\hat{I}_n$ exakt für alle $p \in \Pi_{2n+1}$. Dann gilt für*

$$p_{n+1}(t) = (t-\tau_{n,0})(t-\tau_{n,1})\cdots(t-\tau_{n,n})$$

die Orthogonalitätsrelation

$$\int_a^b q(t)p_{n+1}(t)w(t)dt = 0\ \forall q \in \Pi_n$$

Beweis: $q(t)p_{n+1}(t) \in \Pi_{2n+1}$

$$\begin{aligned}\Rightarrow \int_a^b q(t)p_{n+1}(t)w(t)dt &= \sum_{i=0}^{n} \lambda_{n,i} q(\tau_{n,i}) \overbrace{p_{n+1}(\tau_{n,i})}^{=0\ \forall i} \\ &= 0 \quad \checkmark\end{aligned}$$

Folgerung: p_{n+1} steht senkrecht auf Π_n bzgl. des Skalarprodukts

$$\langle f,g\rangle_w = \int_a^b f(t)g(t)w(t)dt$$

Konstruiere daher $\{p_k\}_{k\in\mathbb{N}_0}$, $p_k \in \Pi_k$, die Orthogonalpolynome bzgl. w sind, d.h.

$$\langle p_k, p_j\rangle_w = \begin{cases} \langle p_k,p_k\rangle_w, & k=j \\ 0, & k\neq j\end{cases}$$

Satz 8.7. *Zu jedem Gewicht w von oben gilt es eindeutig bestimmte Orthogonalpolynome $\{p_k\}_{k\in\mathbb{N}_0}$, $p_k \in \Pi_k$ mit führendem Koeffizienten 1, d.h. $p_k(t) = t^k + \alpha_{k-1}t^{k-1} + \ldots$ zu finden. Sie genügen der Rekursion*

$$p_k(t) = (t-a_k)p_{k-1}(t) + b_k p_{k-2}(t),\ k = 1,2,\ldots$$

mit $p_{-1} = 0$ und $p_0 = 1$ und

$$a_k = \frac{-\langle tp_{k-1}, p_{k-1}\rangle_w}{\langle p_{k-1}, p_{k-1}\rangle_w},\ b_k = -\frac{\langle p_{k-1}, p_{k-1}\rangle_w}{\langle p_{k-2}, p_{k-2}\rangle_w}$$

Satz 8.8. *Seien $\{p_k\}_k$ die Orthogonalpolynome von oben bzgl. w. Dann hat p_k genau k einfache Nullstellen in $]a,b[$.*

Lemma 8.9. *Seien $\tau_{n,0}, \ldots, \tau_{n,n}$ die Nullstellen des $(n+1)$-ten Orthodonalpolynoms bzgl. w. Dann gilt für $\hat{I}_n(f) = \sum_{i=0}^n \lambda_{n,i} f(\tau_{n,i})$:*

$$\hat{I}_n(f) \text{ ist exakt auf } \Pi_n \Leftrightarrow \hat{I}_n \text{ ist exakt auf } \Pi_{2n+1}.$$

Seien $L_{n,i}$ die Lagrange-Polynome bzgl. $\tau_{n,0}, \ldots, \tau_{n,n}$

$$\Rightarrow \lambda_{n,i} = \int_a^b w(t) L_{n,i}(t) dt$$

Lemma 8.10. $\lambda_{n,i} > 0 \; \forall i \in \{0, \ldots, n\}$.

$$\Rightarrow \hat{I}_n \text{ ist positive Linearform}$$

Die so konstruierten Quadraturformeln heißen Gauß-Formeln zur Gewichtsfunktion w.

$$\int_a^b f(t)w(t)dt - \hat{I}_n(f) = \langle p_{k+1}, p_{k+1} \rangle_w \frac{f^{(2n+2)}(\tau)}{(2n+2)!}$$

Beispiel: $a = -1$, $b = 1$, $w = 1$. $\{p_k\}_k$ sind die Legendre-Polynome

$$p_0 = 1,\ p_1(t) = t,\ p_2(t) = t^2 - \tfrac{1}{3}$$

$$p_3(t) = tp_2(t) - \tfrac{12}{45} \cdot p_1(t) = t(t^2 - \tfrac{3}{5})$$

$\Rightarrow \{-\sqrt{\frac{3}{5}}, 0, \sqrt{\frac{3}{5}}\}$ sind die Nullstellen von p_3.

$$\begin{aligned} L_{2,0}(t) &= \tfrac{5}{6} t (t - \sqrt{\tfrac{3}{5}}) & \Rightarrow \lambda_{2,0} &= \tfrac{5}{9} \\ L_{2,1}(t) &= -(\tfrac{5}{3} t^2 - 1) & \Rightarrow \lambda_{2,1} &= \tfrac{8}{9} \\ L_{2,2}(t) &= \tfrac{5}{6} t (t + \sqrt{\tfrac{3}{5}}) & \Rightarrow \lambda_{2,2} &= \tfrac{5}{9} \end{aligned}$$

$$\hat{I}_2(f) = \tfrac{5}{9} f(-\sqrt{\tfrac{3}{5}}) + \tfrac{8}{9} f(0) + \tfrac{5}{9} f(\sqrt{\tfrac{3}{5}})$$

exakt auf Π_5.

$$\int_a^b f(x)dx = \tfrac{b-a}{2} \int_{-1}^1 f(a + \tfrac{t+1}{2}(b-a))dt,\ -\infty < a < b < \infty$$

$$\Rightarrow \hat{I}_2(f) = \tfrac{b-a}{18} \Bigg[5f(a + \tfrac{1-\sqrt{\frac{3}{5}}}{2}(b-a)) + 8f(\tfrac{1}{2}(a+b)) + 5f(a + \tfrac{1+\sqrt{\frac{3}{5}}}{2}(b-a)) \Bigg]$$

(-: ENDE :-)