Notizen der Vorlesungsreihe

Numerische Mathematik

in den Jahren 2012/2013

– Prof. Dr. Axel Klawonn –

Stand:
Montag, 13. Januar 2014

Version 1.2

Mathematisches Institut
Universität zu Köln

Inhaltsverzeichnis

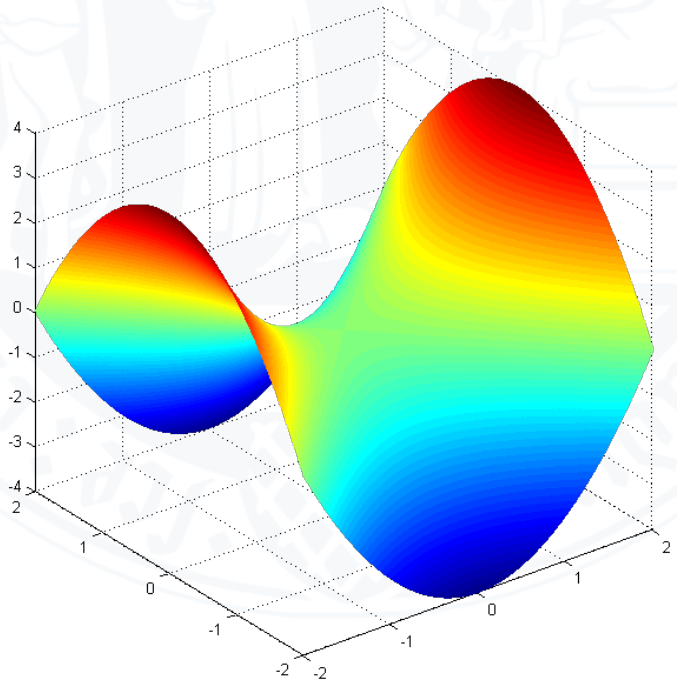
Verzeichnisse

Notizen zur Vorlesung

Numerische Mathematik I

im Sommersemester 2012

– Prof. Dr. Axel Klawonn –



Mathematisches Institut
Universität zu Köln

1. Einführung

02.04.2012
1. Vorlesung

1.1. Einige Grundaufgaben der numerischen Mathematik

- Lösung linearer und nichtlinearer Gleichungssystem
- Approximation von Funktionen durch einfache Funktionen, z. B. Polynome
- Numerische Integration und Differentiation
- Numerische Lösung von Integral- und Differenzialrechnung
- ...

Für jede dieser Grundaufgaben sind unter anderem folgende Punkte zu untersuchen:

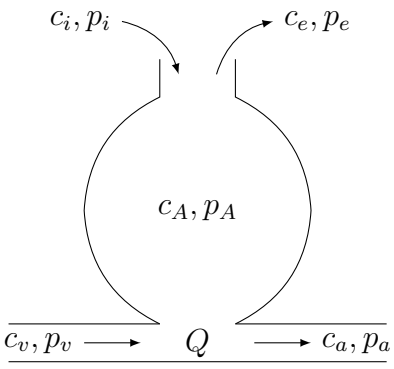
1. Entwicklung von Algorithmen zur Lösung der Grundaufgabe:
   Dabei ist besonders darauf zu achten, dass diese Algorithmen effizient auf Rechnern implementiert werden können und stabil sind.
2. Mathematische Analyse des Algorithmus:
   Es muss gezeigt werden, dass das Verfahren die Grundaufgabe löst und unter welchen Voraussetzungen.
3. Fehlerkontrolle: Meistens können Grundaufgaben nicht exakt gelöst werden, z. B. das Integral einer allgemeinen Funktion.
   Gesucht wird ein Algorithmus, der einen kontrollierbaren Fehler liefert.

2. Nichtlineare Gleichungen

05.04.2012
2. Vorlesung

2.1. Einführung

Beispiel (Lungengasaustausch):
Die Lunge hat ca. $3 \cdot 10^8$ Alveolen in denen der Austausch von Kohlendioxid und Sauerstoff stattfindet.



Notation:

c	=	Konzentration des Blutsauerstoffs
p	=	partieller Sauerstoffdruck
v	=	venös
a	=	arteriell
i	=	input
e	=	exit
A	=	in Alveole
V_A	=	Alveolenluftzufluss
Q	=	Löslichkeit Blutsauerstoff
k	=	Boltzmannkonstante
T	=	absolute Temperatur

Wir machen folgende Annahmen:

1. $V_A c_i + Q c_v = V_A c_e + Q c_a$
   (Gasmoleinfluss pro Zeiteinheit = Gasmolaustritt pro Zeiteinheit)
2. $c_e = c_A$ (ausgeatmete Luft ist Teil der Luft aus den Alveolen)
3. $p_A = kTc_A$ (Sauerstoff verhält sich wie ein ideales Gas)
4. $p_a = H(c_a)$ (partieller Druck ist Funktion der Gaskonzentration)
5. $p_a = p_A$

$\overset{\text{1. und 2.}}{\Rightarrow} V_A(c_i - c_A) = Q(c_a - c_v)$. Wir definieren $v = \frac{V_A}{Q}$. Jetzt kann man die ca. $3 \cdot 10^8$ Alveolen in Kammern mit je gleichem v zusammenfassen und einheitlich skalieren: $\mathrm{mol\,l^{-1}}$ statt $\mathrm{Molekül\,l^{-1}}$ wobei $1\,\mathrm{mol} \mathrel{\widehat{=}} 6.02 \cdot 10^{23}$ Molekülen.
Boltzmannkonstante einsetzen: $R = 6.02 \cdot 10^{23}\, k$. Wir stellen uns die Alveolen durchnummeriert vor: $i = 1, \ldots, n$ und beachten, dass c_v für alle Alveolen gleich ist.

$$\rightarrow \qquad \Big((V_A)_i, \ldots, (c_i - c_a)_i\Big) = Q_i\Big((c_a)_i - c_v\Big)$$

$$\left.\begin{array}{rcl} (p_A)_i & = & RT(c_A)_i \\ (p_a)_i & = & H\Big((c_a)_i\Big) \end{array}\right\} (p_A)_i = (p_a)_i$$

Aus experimentellen Daten ergibt sich für $H(c) = p_A \left(\frac{c}{c^\star - c}\right)^{\frac{1}{3}}$. Seien $p_v = 25$ mmHg, der partielle Druck bei dem Hämoglobin halb gesättigt ist, und c_v bekannt, so ergibt sich für die Berechnungen von $(c_a)_i$ in einem einzelnen Lungenbläschen folgende Gleichung

$$\Phi\Big[(c_a)_i, c_v, r_i\Big] = 0$$

mit $r_i = \frac{(V_A)_i}{Q_i}$ und $\Phi(c_a, c_v, r) = c_a - c_v + r\left(\frac{H(c_a)}{RT} - c_i\right)$. Für ein gegebenes c_v muss also für alle Alveolen ein nichtlineare Gleichung der Form

$$c_a - c_v + r\left(p_A\left(\frac{c_a}{(RT)^3 \cdot (c^\star - c_a)}\right)^{\frac{1}{3}} - c_i\right) = 0$$

berechnet werden.

2.2. Grundaufgabe

Gegeben seien $n \in \mathbb{N}$, $y \in \mathbb{R}^n$ und $F\colon G \subset \mathbb{R}^n \to \mathbb{R}^n$. Gesucht ist ein $x^\star \in G$, so dass

$$F(x^\star) = y$$

Dieses Problem lässt sich auf verschiedene Arten umformulieren:

1. Man betrachte die Funktion

   $$\begin{aligned} f\colon G \subset \mathbb{R}^n &\to \mathbb{R}^n \\ x &\mapsto f(x) := F(x) - y \end{aligned}$$

   Dann gilt: Das Problem hat eine Lösung $x^\star \iff f(x^\star) = 0$.
   Die Lösung der Grundaufgabe lässt sich als Nullstellensuche formulieren.

2. Man betrachte die Funktion

   $$\begin{aligned} g\colon G \subset \mathbb{R}^n &\to \mathbb{R}^n \\ x &\mapsto g(x) := f(x) + x = F(x) - y + x \end{aligned}$$

   Dann gilt: $x^\star$ löst das Problem $\iff g(x^\star) = x^\star$.
   Allgemein heißt x ein **Fixpunkt** von g, wenn $g(x) = x$.
   Die Lösung der Grundaufgabe lässt sich auch als Fixpunktsuche formulieren.

Bemerkung 2.2.1:
Offenbar sind die Formulierungen äquivalent. Es kann aber vom jeweiligen Problem, d. h. von F abhängen, welche Formulierung sich am besten lösen lässt.

Bemerkung 2.2.2:
Analytische Lösungen sind nur selten möglich bzw. sinnvoll. Aus der Algebra (Galoistheorie) ist bekannt, dass Polynome im Allgemeinen nur bis zum vierten Grad analytisch aufzulösen sind.
Gleichungen höheren Grades müssen im Allgemeinen numerisch gelöst werden.

2.3. Existenz von Lösungen

Die Untersuchung nichtlinearer Gleichungen ist ein wichtiges Teilgebiet der nichtlinearen Analysis. Wichtigstes Hilfsmittel sind hier Fixpunktsätze.

Satz 2.3.1 (Fixpunktsatz von Brouwer):
Sei $n \in \mathbb{N}$ und $\emptyset \neq G \subset \mathbb{R}^n$ sein konvex und kompakt. Weiterhin sei $g\colon G \to G$ stetig. Dann besitzt g in G mindestens einen Fixpunkt.
Erweiterung: *G ist konvex $\Leftrightarrow a, b \in G \Rightarrow x = ta + (1-t)b \in G \quad \forall t \in [0,1]$.*

Beweis:
Siehe [1].

□

Beispiel 2.3.1:
Sei $f(x) = 2\sin(x) - x$. Gesucht ist die Nullstelle in $[0, \pi]$.
$g(x) := \sin(x)$ ist stetig in $[0, \pi] =: G$, G konvex, G kompakt. Es ist $g(G) \subset G$. Nach dem Fixpunktsatz von Brouwer existiert mindestens ein Fixpunkt von g in G. Offensichtlich ist $x = 0$ ein solcher.

Gibt es noch weitere Nullstellen von f?

1. Zeichnerisch zwischen $\frac{\pi}{2}$ und π
2. Fixpunktsatz von Brouwer: $G = \left[\frac{\pi}{2}, \pi\right]$
3. Zwischenwertsatz

2. und 3. liefern nur die Existenz, es bleibt die Frage nach der Lage der Nullstelle.

2.4. Einfache Iterationsverfahren in $\mathbb{R}$

Definition 2.4.1 (Iterationsfolge):
$(x^{(k)})_{k\in\mathbb{N}}$ mit $x^{(k)} \in \mathbb{R}^n$ heißt ***Iterationsfolge*** *für eine Lösung $x^\star$ des Problems aus Abschnitt 2.2, wenn gilt:*

1. $\lim_{k\to\infty} x^{(k)} = x^\star$
2. *$x^{(k)}$ ist berechenbar für alle $k \in \mathbb{N}$*

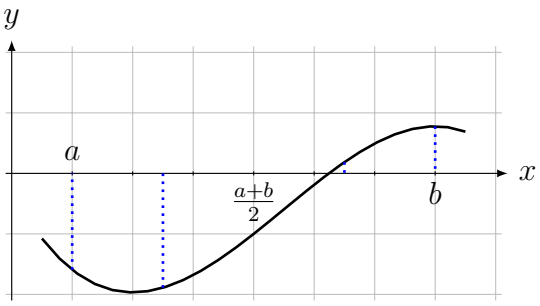
Wir betrachten nun einige **einfache Verfahren für nichtlineare Gleichungen in einer Unbekannten**.

Dazu sei

$$f\colon [a,b] \subset \mathbb{R} \to \mathbb{R}$$

eine stetige Funktion mit $\boxed{f(a) \cdot f(b) < 0}$. Dann hat f nach dem **Zwischenwertsatz** mindestens eine Nullstelle in (a, b).

2.4.1. Intervallhalbierung (Bisektion)



Algorithmus 2.4.1 Intervallhalbierung

Initialisierung:

1: $x_p^{(0)} = \begin{cases} a, & f(a) > 0 \\ b, & \text{sonst} \end{cases}$

2: $x_n^{(0)} = \begin{cases} a, & f(a) < 0 \\ b, & \text{sonst} \end{cases}$

Iteration ($k \geq 0$):

1: $x_m^{(k)} = \dfrac{x_p^{(k)} + x_n^{(k)}}{2}$ ▷ Mittelpunkt

2: $x_p^{(k+1)} = \begin{cases} x_m^{(k)}, & f(x_m^{(k)}) \geq 0 \\ x_p^{(k)}, & \text{sonst} \end{cases}$

3: $x_n^{(k+1)} = \begin{cases} x_m^{(k)}, & f(x_m^{(k)}) < 0 \\ x_n^{(k)}, & \text{sonst} \end{cases}$

Für die Iterierten $x_p^{(k)}$, $x_n^{(k)}$ mit $k \geq 0$ gelten folgende Eigenschaften:

1. $x_p^{(k)} - x_n^{(k)} = 2^{-k}(b-a)$
2. $[x_n^{(k+1)}, x_p^{(k+1)}] \subset [x_n^{(k)}, x_p^{(k)}]$
3. $f(x_n^{(k)}) \leq 0 \ \wedge \ f(x_p^{(k)}) \geq 0$

Dies kann für einen konstruktiven Beweis des Zwischenwertsatzes verwendet werden, siehe z. B. [2].

Relativer und absoluter Fehler

09.04.2012
3. Vorlesung

Sei $x^\star$ die exakte Lösung.
Als **absoluten Fehler** bezeichnet man $|x^{(k)} - x^\star| = e_{\text{abs}}$. Ist $x^\star \approx 1$, so liefert e_{abs} bei $\varepsilon = 10^{-3}$ drei genaue Dezimalstellen. Im Fall von $x^\star = 10^{-4}$ erhält man überhaupt keine genaue Dezimalstelle. Daher ist es sinnvoll den **relativen Fehler** zu betrachten:

$$\frac{|x^{(k)} - x^\star|}{|x^\star|} = e_{\text{rel}} \in (0,1)$$

Hat eine numerische Lösung $x^{(k)}$ den relativen Fehler $\varepsilon > 0$ mit $\varepsilon \ll 1$ und $|e_{\text{abs}}| < \varepsilon$ dann gilt:

$$(1-\varepsilon)|x^\star| \leq |x^{(k)}| \leq (1+\varepsilon)|x^\star|$$

z. B.: $\varepsilon = 10^{-1} = \frac{1}{10} \Rightarrow 0.9|x^\star| \leq |x^{(k)}| \leq 1.1|x^\star|$.

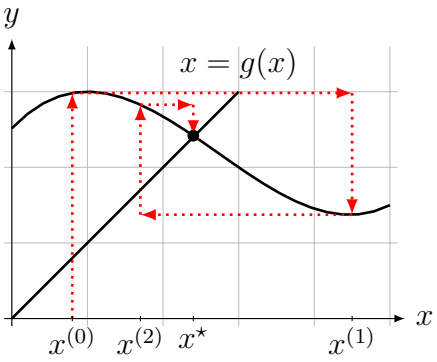
2.4.2. Fixpunktiteration

Betrachtet man die Fixpunktformulierung unserer Grundaufgabe, so ist ein naheliegender Ansatz zur Konstruktion eines Iterationsverfahrens

$$x^{(k+1)} = g(x^{(k)}) \text{ bei gegebenem } x^{(0)}$$

Ein solches Verfahren nennt man **Fixpunktiteration**.

Geometrische Interpretation



Wann konvergiert ein solches Verfahren?

2.4.3. Newtonverfahren

Das Newtonverfahren ist eine Methode zur Nullstellenbestimmung einer nichtlinearen Funktion $f(x)$. Wir nehmen an, dass $f\colon [a,b] \subset \mathbb{R} \to \mathbb{R}$ genügend glatt. Die Taylorentwicklung um $x^{(0)} \in [a,b]$ ergibt:

$$f(x) = f(x^{(0)}) + f'(x^{(0)})(x - x^{(0)}) + \frac{1}{2} f''(\xi^{(0)})(x - x^{(0)})^2$$

wobei $\xi^{(0)}$ zwischen x und $x^{(0)}$ liegt.
Ist $x^{(0)}$ nahe bei der Lösung $x^\star$ und $f''(\xi^{(0)})$ nicht zu groß, dann ist

$$\tilde{f}(x) = f(x^{(0)}) + f'(x^{(0)})(x - x^{(0)})$$

eine gute Approximation an $f(x)$ in einer Umgebung von $x^\star$.

Beispiel:

$$f''(\xi^{(0)}) \leq 2 \text{ und } |x - x^{(0)}| \leq 10^{-2}$$
$$\Rightarrow |\tilde{f}(x) - f(x)| \leq 10^{-4}$$

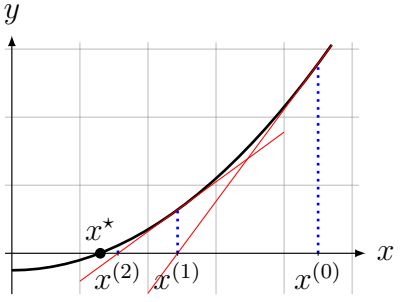
Wir benutzen nun $\tilde{f}(x)$ als Ersatz für $f(x)$ und lösen $\tilde{f}(x) = 0$.
Die Lösung verwenden wir dann als erste Näherung an $x^\star$:

$$x = x^{(1)} = x^{(0)} - \frac{f(x^{(0)})}{f'(x^{(0)})}$$

Zu einem gegebenen Startwert $x^{(0)}$ definiert man dann die Iterierten:

$$x^{(k+1)} = x^{(k)} - \frac{f(x^{(k)})}{f'(x^{(k)})}$$

Geometrische Interpretation



2.4.4. Sekantenverfahren

Im Newtonverfahren muss die Ableitung $f'(x)$ berechnet werden. Manchmal ist es wünschenswert, auch diese zu approximieren. Aus der Definition des Differenzenquotienten ergibt sich folgende Näherung:

$$f'(x) \approx \frac{f(x+\delta) - f(x)}{(x+\delta) - x} \text{ für genügend kleines } \delta$$

Dies kann man benutzen, um $f'(x^{(k)})$ zu berechnen:

$$f'(x^{(k)}) \approx \frac{f(x^{(k)}) - f(x^{(k-1)})}{x^{(k)} - x^{(k-1)}}$$

Einsetzen in das Newtonverfahren ergibt

$$x^{(k+1)} = x^{(k)} - f(x^{(k)}) \frac{x^{(k)} - x^{(k-1)}}{f(x^{(k)}) - f(x^{(k-1)})}$$

Hierzu werden zwei Startwerte benötigt. Das Sekantenverfahren wird auch als **Regula Falsi** bezeichnet.

2.5. Fixpunktiteration im $\mathbb{R}^n$

Definition 2.5.1 (Kontrahierend):
*$g\colon G \subset \mathbb{R}^n \to \mathbb{R}^n$ heißt **kontrahierend in G** bezüglich der Norm $\|\cdot\|$ im $\mathbb{R}^n$, wenn es eine Konstante $0 < q < 1$ gibt, so dass*

$$\|g(x) - g(y)\| \leq q\|x - y\| \quad \forall x, y \in G \tag{2.1}$$

*Die Konstante q heißt **Lipschitzkonstante** von g in G und eine Funktion, die (2.1) erfüllt, heißt **Lipschitzstetig**; dabei muss nicht $q < 1$ gelten.*

Unter gewissen Voraussetzungen lässt sich einfacher als mit der Definition entscheiden, ob eine Funktion kontrahierend ist.

Satz 2.5.1:
Sei $G \subset \mathbb{R}^n$ konvex und $g = \begin{pmatrix} g_1 & \cdots & g_n \end{pmatrix}^T : G \to \mathbb{R}^n$ stetig differenzierbar. Weiterhin sei

$$q := \sup_{x \in G} \|D_g(x)\| < 1$$

wobei $D_g(x)$ die Jacobimatrix sei, d. h.

$$D_g(x) = \begin{pmatrix} \frac{\partial g_1}{\partial x_1} & \cdots & \frac{\partial g_1}{\partial x_n} \\ \vdots & \ddots & \vdots \\ \frac{\partial g_n}{\partial x_1} & \cdots & \frac{\partial g_n}{\partial x_n} \end{pmatrix}$$

Dann ist g in G kontrahierend bezüglich der Norm $\|\cdot\|$.

Beweis:
Wir definieren $f\colon [0,1] \to \mathbb{R}^n$ mit $f(t) = g\Big(t \cdot x + (1-t) \cdot y\Big)$ und $x, y \in G$. Mit der Kettenregel folgt

$$D_f(t) = D_g\Big(t \cdot x + (1-t) \cdot y\Big)(x - y)$$

Seien $x, y \in G$, betrachte $\|g(x) - g(y)\| = \|f(1) - f(0)\|$

$$\begin{aligned}
\left\| \int_0^1 D_f(t)\,\mathrm{d}t \right\| &= \sup_{t\in[0,1]} \|D_f(t)\| \\
&= \sup_{t\in[0,1]} \|D_g\Big(\underbrace{t \cdot x + (1-t) \cdot y}_{\in G}\Big)(x-y)\| \\
&= \sup_{z\in G} \|D_g(z) \cdot (x-y)\| \\
&= \underbrace{\sup_{z\in G} \|D_g(z)\|}_{=q<1} \cdot \|x-y\| \\
&= q \cdot \|x-y\|
\end{aligned}$$

$\Rightarrow$ Behauptung.

□

16.04.2012
4. Vorlesung

Satz 2.5.2 (Banachscher Fixpunktsatz):
Sei $G \subset \mathbb{R}^n$ abgeschlossen und $g\colon G \to G$ kontrahierend. Dann hat g in G genau einen Fixpunkt $x^\star$. Für jeden Startwert $x^{(0)} \in G$ konvergiert das Iterationsverfahren

$$x^{(k+1)} = g(x^{(k)})$$

gegen den Fixpunkt $x^\star$.
*Dieses Verfahren bezeichnet man als **Fixpunktiteration**. Weiterhin gilt die Fehlerabschätzung:*

$$\|x^{(k)} - x^\star\| \leq \frac{q^k}{1-q} \|x^{(1)} - x^{(0)}\|$$

wobei q die Lipschitzkonstante von g in G ist.

Beweis:

1. Existenz eines Fixpunktes $x^\star$:
   Da $g(G) \subset G$ und $x^{(k+1)} = g(x^{(k)})$ gilt: $x^{(0)} \in G \Rightarrow x^{(k)} \in G$.
   Weiterhin gilt auch

$$\begin{aligned}
\|x^{(k+1)} - x^{(k)}\| &= \|g(x^{(k)}) - g(x^{(k-1)})\| \\
&\leq q\|x^{(k)} - x^{(k-1)}\| \\
&\leq q^k \|x^{(1)} - x^{(0)}\|
\end{aligned}$$

Sei nun $i, j \in \mathbb{N}$ mit $i > j$, dann folgt:

$$\begin{aligned}
\|x^{(i)} - x^{(j)}\| &= \left\| \sum_{k=j}^{i-1} x^{(k+1)} - x^{(k)} \right\| \\
&\leq \sum_{k=j}^{i-1} \|x^{(k+1)} - x^{(k)}\| \\
&\leq \left(\sum_{k=j}^{i-1} q^k \right) \|x^{(1)} - x^{(0)}\| \\
&\underset{\text{Reihe}}{\overset{\text{Geometrische}}{=}} \frac{1 - q^{i-j}}{1-q} q^j \|x^{(1)} - x^{(0)}\| \\
&\leq \frac{q^j}{1-q} \|x^{(1)} - x^{(0)}\|
\end{aligned}$$

$\Rightarrow \|x^{(i)} - x^{(j)}\| \xrightarrow{i,j \to \infty} 0$, somit ist $(x^{(k)})_{k \in \mathbb{N}}$ eine Cauchyfolge im $\mathbb{R}^n$. Da $\mathbb{R}^n$ vollständig, existiert ein $x^\star \in \mathbb{R}^n$ mit $x^\star = \lim_{k \to \infty} x^{(k)}$ bezüglich $\|\cdot\|$.

Aus der Abgeschlossenheit von G und $x^{(k)} \in G \; \forall k \in \mathbb{N}$ folgt $x \in G$. Da g stetig ist gilt

$$x^\star = \lim_{k \to \infty} x^{(k)} = \lim_{k \to \infty} g(x^{(k-1)}) = g(x^\star)$$

Somit ist $x^\star$ Fixpunkt von g.

2. Eindeutigkeit des Fixpunktes $x^\star$:
   Angenommen es existiere ein weiterer Fixpunkt $\tilde{x} \neq x^\star$ von g in G.

$$\begin{aligned}
\Rightarrow 0 < \|\tilde{x} - x^\star\| &= \|g(\tilde{x}) - g(x^\star)\| \\
&\leq q \|\tilde{x} - x^\star\|
\end{aligned}$$

   $\Rightarrow 1 \leq q < 1 \quad \lightning$

3. Fehlerabschätzung:
   Nach 1) gilt für $i > j$ und $j = k$

$$\|x^{(i)} - x^{(k)}\| \leq \frac{q^k}{1-q} \|x^{(1)} - x^{(0)}\|$$

   Betrachtet man $i \to \infty$ so folgt

$$\|x^\star - x^{(k)}\| \leq \frac{q^k}{1-q} \|x^{(1)} - x^{(0)}\|$$

□

Beispiel 2.5.1:
$f(x) = \cot(x) - x, \; x \in \left[\frac{\pi}{6}, \frac{2\pi}{5}\right] = G$
Ziel: $f(x)$ auf Nullstellen untersuchen.
Naheliegende Fixpunktformulierung: $g(x) = x$ mit $g(x) = \cot(x)$
Wir untersuchen zunächst, ob wir Satz 2.5.1 anwenden können:

$$g'(x) = -\frac{1}{\sin^2(x)} \Rightarrow \sup_{x \in G} |g'(x)| \geq 1$$

$\Rightarrow$ Satz 2.5.1 lässt sich hier nicht anwenden. Es gilt auch $g(G) \not\subset G$.

Möglicher nächster Schritt:

1. Ein anderes Intervall G wählen, so dass $g(G) \subset G$
2. Kontraktionseigenschaft mit der Definition 2.5.1 nachweisen (falls möglich)

Alternativ schauen wir uns eine äquivalente Fixpunktformulierung an:

$$x = \cot(x) \Leftrightarrow \underbrace{\operatorname{arccot}(x)}_{=:g(x)} = x$$

$\Rightarrow g'(x) = -\frac{1}{1+x^2} \Rightarrow \sup_{x\in G}|g'(x)| < 1$

$\overset{\text{Satz 2.5.1}}{\Rightarrow}$ $g(x) = \operatorname{arccot}(x)$ ist kontrahierend auf G. Es gilt auch $G \subset G$.

$\overset{\text{Satz 2.5.2}}{\Rightarrow}$ Das Fixpunktverfahren $x^{(k+1)} = \operatorname{arccot}(x^{(k)})$ mit $x^{(0)} \in G$ konvergiert gegen einen Fixpunkt $x^\star \in G$, welcher eindeutig ist.

$|g'(x)|$ ist streng monoton fallend auf $\left[\frac{\pi}{6}, \frac{2\pi}{5}\right]$.

$\Rightarrow q \overset{\text{Def.}}{=} \sup_{x\in G}|g'(x)| = \frac{36}{36+\pi^2} \approx 0.7848$

Algorithmus 2.5.1 Fixpunktiteration

```
1: function [fix] = fixpunkt(x, ε)
2: iter = 0
3: gx = g(x)
4: err_rel = |gx−x| / |x|                          ▷ |x^(1)−x^(0)| / |x^(0)|
5: while err_rel ≥ ε do
6:     x = gx
7:     gx = g(x)
8:     iter = iter + 1
9:     err_rel = |gx−x| / |x|
10:    fprintf('iter=%3i, relativer Fehler=%6.4f\n', iter, err_rel)
11: end while
12: fix = gx
```

Im Anhang befindet sich eine MATLAB-Implementierung

Satz 2.5.3 (Lokaler Konvergenzsatz):
Die Funktion $g\colon \mathbb{R}^n \to \mathbb{R}^n$ besitze einen Fixpunkt $x^\star$ und es gebe eine Umgebung von $x^\star$ in der g kontrahierend ist.
Dann existiert eine Umgebung $U(x^\star)$ von $x^\star$, so dass die Fixpunktiteration

$$x^{(k+1)} = g(x^{(k)})$$

für jedes $x^{(0)} \in U(x^\star)$ gegen $x^\star$ konvergiert.

Beweis:
Sei $\varepsilon > 0$ so gewählt, dass g kontrahierend ist auf der Kugel

$$G := \{x \in \mathbb{R}^n : \|x - x^\star\| < \varepsilon\}$$

Sei q die zugehörige Lipschitzkonstante. Dann gilt für $x \in G$:

$$\begin{aligned}\|g(x) - x^\star\| &= \|g(x) - g(x^\star)\| \\ &\leq q \cdot \|x - x^\star\| \\ &\leq q \cdot \varepsilon \\ &\underset{q<1}{<} \varepsilon\end{aligned}$$

$\Rightarrow g(x) \in G$
$\Rightarrow g(G) \subset G$
G ist abgeschlossen und g ist kontrahierend auf G.
$\overset{\text{Satz 2.5.2}}{\underset{\text{Banachscher Fixpunktsatz}}{\Rightarrow}}$ Behauptung.

□

Definition 2.5.2 (Konvergenzordnung):
Für ein Iterationsverfahren

$$x^{(k+1)} = g(x^{(k)}), \quad k \geq 0$$

mit dem Startwert $x^{(0)}$ gelte die Abschätzung

$$\|x^{(k+1)} - x^\star\| \leq c\|x^{(k)} - x^\star\|^p, \quad k \geq 0$$

mit einer Konstanten $0 < c < \infty$ und zusätzlich $c < 1$ falls $p = 1$. Dann heißt das Verfahren von mindestens ***p-ter Konvergenzordnung***.

19.04.2012
5. Vorlesung

Satz 2.5.4:
Sei $G \subset \mathbb{R}^n$ offen, $g: G \to \mathbb{R}^n$, $x^{(0)} \in G$ und $(x^{(k)})_{k\in\mathbb{N}}$ sei definiert durch

$$x^{(k+1)} = g(x^{(k)})$$

Es gebe einen Fixpunkt $x^\star \in G$ und eine Umgebung $V \subset G$ von $x^\star$ mit Konstanten $c > 0$ und $p \geq 1$ und zusätzlich $c < 1$, falls $p = 1$, so dass $\forall x \in V$ gilt

$$\|g(x) - g(x^\star)\| \leq c\|x - x^\star\|^p$$

Dann konvergiert die durch g definierte Fixpunktiteration $(x^{(k)})_{k\in\mathbb{N}}$ in einer Umgebung U von $x^\star$ gegen $x^\star$ und zwar für jeden Startwert $x^{(0)} \in U$. Dann ist das Verfahren von mindestens p-ter Konvergenzordnung.

Beweis:
Wir definieren U wie folgt:

$$U = U(x^\star) := \overline{B_\varepsilon(x^\star)} = \{x \in \mathbb{R}^n \colon \|x - x^\star\| \leq \varepsilon\}$$

Nun wählen wir $\varepsilon > 0$ so klein, dass $c \cdot \varepsilon^{p-1} =: L < 1$. Sei $x^{(k)} \in U$, dann gilt:

$$\begin{aligned}\|x^{(k+1)} - x^\star\| &= \|g(x^{(k)}) - g(x^\star)\| \\ &\leq c \cdot \|x^{(k)} - x^\star\|^p \\ &\leq c \cdot \varepsilon^p \\ &< \varepsilon\end{aligned}$$

$\Rightarrow x^{(k+1)} \in U$
Somit gilt: $x^{(0)} \in U \Rightarrow x^{(k)} \in U \; \forall k \in \mathbb{N}$. Hieraus ergibt sich sofort

$$\begin{aligned}\|x^{(k+1)} - x^\star\| &\leq c \cdot \|x^{(k)} - x^\star\|^p \\ &\leq \underbrace{c \cdot \varepsilon^{p-1}}_{=L} \cdot \|x^{(k)} - x^\star\| \\ &= L \cdot \|x^{(k)} - x^\star\|\end{aligned}$$

Die Konvergenzordnung ergibt sich aus

$$\begin{aligned}\|x^{(k+1)} - x^\star\| &= \|g(x^{(k)}) - g(x^\star)\| \\ &\leq c \cdot \|x^{(k)} - x^\star\|^p\end{aligned}$$

□

2.6. Newtonverfahren im $\mathbb{R}^n$

In diesem Abschnitt betrachten wir das Newtonverfahren zur Nullstellensuche einer nichtlinearen Funktion $f \colon G \subset \mathbb{R}^n \to \mathbb{R}^n$. $x^\star$ sei eine Nullstelle von f. Unter der Annahme, dass f stetig differenzierbar ist, ergibt die Taylorreihenentwicklung [siehe 3, Abschnitt 7]

$$f(x) = f(x^{(0)}) + D_f(x^{(0)})(x - x^{(0)}) + O\Big(\|x - x^{(0)}\|\Big)$$

für alle x in einer Umgebung von $x^{(0)}$. Hierbei steht $O\Big(\|x\|\Big)$ für eine Funktion $\varphi(t)$ mit $\varphi(0) = 0$ und $\lim_{\substack{x \to 0 \\ x \neq 0}} \frac{\varphi(\|x\|)}{\|x\|} = 0$.
Setzen wir $x = x^\star$ und nehmen an, dass $x^{(0)}$ nahe genug an der Nullstelle liegt, so erhalten wir die Näherung

$$\begin{aligned}D_f(x^{(0)})(x^\star - x^{(0)}) &\approx f(x^\star) - f(x^{(0)}) \\ &= -f(x^{(0)})\end{aligned}$$

Wie im Skalaren Fall wählt man nun als nächste Näherung $x^{(1)}$ an $x^\star$ die Lösung des linearen Gleichungssystems

$$D_f(x^{(0)})(x^{(1)} - x^{(0)}) = -f(x^{(0)})$$

wobei $D_f(x^{(0)})$ regulär sein muss.

Dies motiviert die Iterationsvorschrift für das Newtonverfahren

$$x^{(k+1)} := x^{(k)} - \left(D_f(x^{(k)})\right)^{-1} f(x^{(k)}), \quad k \geq 0$$

mit einem Startwert $x^{(0)}$.
Im Allgemeinen wird in der Numerik eine Matrix nicht invertiert, sondern ein lineares Gleichungssystem gelöst.

Algorithmus 2.6.1 Newtonverfahren

Gegeben: Startvektor $x^{(0)} \in \mathbb{R}^n$

1: $D_f(x^{(k)})\Delta x^{(k)} = -f(x^{(k)})$
2: $x^{(k+1)} = x^{(k)} + \Delta x^{(k)}$

Im Anhang befindet sich eine MATLAB-Implementierung

Beispiel 2.6.1:

$$f\colon \mathbb{R}^2 \to \mathbb{R}^2$$
$$(x, y) \mapsto \begin{pmatrix} x^3 - 3xy^2 \\ -y^3 + 3x^2y + 1 \end{pmatrix}$$

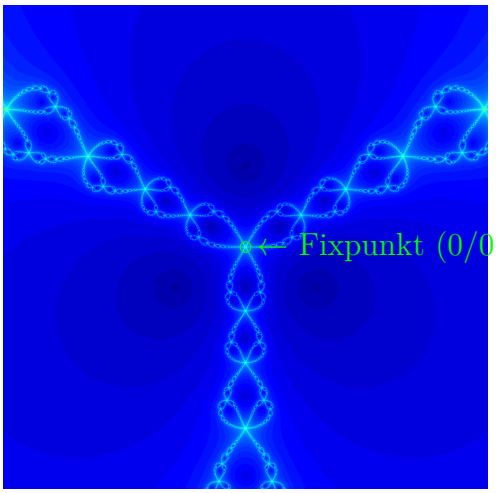


Bild der Iterationen die bei der Funktion f in jedem Startpunkt, auf dem Intervall $[-3, 3] \times [-3, 3]$ und einer jeweiligen Schrittweite von 0.005, benötigt werden um zum Fixpunkt zu gelangen. Je heller der Punkt, umso weniger Iterationen wurden benötigt.

Zu jedem Iterationsschritt ist also ein lineares Gleichungssystem mit der jeweiligen Jacobimatrix zu lösen. $\Delta x^{(k)}$ bezeichnet man als **Newtonkorrektur**.

Definition 2.6.1 (Äquivalenz der Norm):
Zwei Normen $\|\cdot\|$ und $|||\cdot|||$ eines normierten Vektorraums V heißen äquivalent, wenn es zwei Konstanten $c_1, c_2 > 0$ gibt mit

$$c_1\|x\| \leq |||x||| \leq c_2\|x\| \quad \forall x \in V$$

Lemma 2.6.1:
In endlich normierten Vektorräumen sind alle Normen äquivalent.

Beweis:
In vielen Lehrbüchern der Analysis und Numerik.

□

Satz 2.6.1:
Die Funktion $f\colon \mathbb{R}^n \to \mathbb{R}^n$ habe die Nullstelle $x^\star$ und sei in einer Umgebung von $x^\star$ stetig differenzierbar. Die Jacobimatrix $D_f(x^\star)$ sei invertierbar. Dann gibt es eine Umgebung von $x^\star$, in der das Newtonverfahren für alle Startvektoren $x^{(0)}$ gegen $x^\star$ konvergiert.
Ist die Funktion zusätzlich zweimal stetig differenzierbar in einer Umgebung von $x^\star$, so existiert eine Umgebung, in der das Verfahren für alle $x^{(0)}$ aus dieser Umgebung quadratisch konvergiert.

Beweis:

1. Zunächst beweisen wir die Konvergenz.
   Mit der Funktion
   $$g(x) := x - \Big(D_f(x)\Big)^{-1} f(x)$$
   lässt sich das Newtonverfahren als Fixpunktiteration interpretieren.
   Wir zeigen: $D_g(x^\star) = 0$. Dazu betrachten wir die Einträge der Jacobimatrix D_g. Es sei $A(x) := \Big(D_f(x)\Big)^{-1} = \Big(a_{i,j}(x)\Big)_{i,j}$.
   $$\begin{aligned}
   \Rightarrow \qquad & g(x) = \begin{pmatrix} x_1 - \sum_{i=1}^n a_{1,i}(x)\cdot f_i(x) \\ \vdots \\ x_n - \sum_{i=1}^n a_{n,i}(x)\cdot f_i(x) \end{pmatrix} \\
   \Rightarrow \qquad & \frac{\partial g_l(x)}{\partial x_j} = \delta_{l,j} - \sum_{i=1}^n \left(\frac{\partial a_{l,i}(x)}{\partial x_j} f_i(x) + a_{l,i}(x) \frac{\partial f_i(x)}{\partial x_j} \right) \\
   \overset{f(x^\star)=0}{\Rightarrow} \qquad & \frac{\partial g_l(x^\star)}{\partial x_j} = \delta_{l,j} - \sum_{i=1}^n a_{l,i}(x^\star) \frac{\partial f_i(x^\star)}{\partial x_j} = 0
   \end{aligned}$$
   da $A(x) = \Big(D_f(x)\Big)^{-1}$. Nach Voraussetzung ist $D_g(x)$ stetig. Also existiert eine Umgebung V von $x^\star$ in der gilt:
   $$\sup_{x\in V} \|D_g(x)\| \leq \frac{1}{2}$$
   Somit ist g nach Satz 2.5.1 bezüglich $\|\cdot\|$ kontrahierend. Aus Satz 2.5.3 folgt die Existenz einer Umgebung U, so dass $\forall x^{(0)} \in U$ das Newtonverfahren konvergiert.

2. Zum Nachweis der quadratischen Konvergenz genügt es nach Satz 2.5.4 zu zeigen, dass es eine Umgebung V von $x^\star$ gibt, mit
   $$\|g(x) - g(x^\star)\| \leq c\|x - x^\star\|^2 \quad \forall x \in V$$
   wobei $0 < c < \infty$ eine Konstante ist.
   Aus $g(x^\star) = x^\star$ folgt:
   $$\begin{aligned}
   g(x) - g(x^\star) &= x - \Big(D_f(x)\Big)^{-1} f(x) - x^\star \\
   &= (x - x^\star) - \Big(D_f(x)\Big)^{-1} f(x)
   \end{aligned}$$
   Aus der Taylorformel zweiter Ordnung ([1, Satz 168.4]) folgt, in einer genügend kleinen, konvexen Umgebung W von $x^\star$, dass
   $$0 = f(x^\star) = f(x) + D_f(x)(x^\star - x) + r(x, x^\star - x)$$

wobei die Komponenten des Restgliedes

$$r_j(x, x^\star - x) = \sum_{k,l=1}^{n} \left(\int_0^1 \frac{\partial^2 f_j}{\partial x_k \partial x_l}(x + th)(1-t)\,\mathrm{d}t \right) \cdot h_k h_l$$

sind. Hierbei ist $h := x^\star - x$ und $h = (h_k)_{k=1,\dots,n}$. f ist zweimal stetig in einer Umgebung von $x^\star$ woraus folgt, dass die partiellen Ableitungen von f in W beschränkt sind. Für das Restglied gilt die Abschätzung ([1, Satz 168.5])

$$\left\| r(x, x^\star - x) \right\|_\infty \leq \frac{M}{2} \|x^\star - x\|_\infty$$

mit $M := \max_{j=1,\dots,n} \sum_{k,l=1}^{n} \sup_{x \in \mathbb{N}} \left| \frac{\partial^2 f_j}{\partial x_k \partial x_l}(x) \right|$. Unter der Voraussetzung, dass alle Normen auf $\mathbb{R}^n$ äquivalent sind (Lemma 2.6.1), folgt $\forall x \in W$

$$\begin{aligned}
\|g(x) - g(x^\star)\| &\leq \hat{c} \cdot \|g(x) - g(x^\star)\|_\infty \\
&= \hat{c} \cdot \left\| x - x^\star - \left(D_f(x)\right)^{-1} f(x) \right\|_\infty \\
&\leq \hat{c} \cdot \frac{M}{2} \cdot \|x^\star - x\|_\infty^2 \\
&\leq \tilde{c} \cdot \hat{c} \cdot \frac{M}{2} \cdot \|x^\star - x\|^2 \\
&= c \cdot \|x^\star - x\|^2
\end{aligned}$$

□

23.04.2012
6. Vorlesung

Satz 2.6.2:
Sei $A \in \mathbb{R}^{n \times n}$ eine reguläre Matrix. Die Multiplikation von $f(x)$ mit A ist eine Transformation, die man auch als (offene) Skalierung bezeichnet.
Das Newtonverfahren ist invariant unter solchen affinen Abbildungen, d. h. die Iterationen $x^{(k)}$ unterscheiden sich nicht bei der Nullstellenberechnung von $f(x)$ und $Af(x)$. Man sagt auch, dass Newtonverfahren sei affin invariant.

Beweis:
Sei $g(x) := Af(x)$ und $D_g(x) = A \cdot D_f(x)$.
Also ergibt sich für die Newtoniteration $x_g^{(k)}$ bezüglich $g(x)$:

$$\begin{aligned}
x_g^{(k+1)} &= x_g^{(k)} - \left(D_g(x_g^{(k)})\right)^{-1} g\left(x_g^{(k)}\right) \\
&= x_g^{(k)} - \left(AD_f(x_g^{(k)})\right)^{-1} Af\left(x_g^{(k)}\right) \\
&= x_g^{(k)} - \left(D_f(x_g^{(k)})\right)^{-1} f\left(x_g^{(k)}\right)
\end{aligned}$$

Ist nun der Startwert $x_f^{(0)} = x_g^{(0)}$, so gilt offensichtlich auch $x_f^{(k)} = x_g^{(k)}$, $k \geq 1$, wobei $x_f^{(k)}$ die k-te Newtoniterierte bezüglich $f(x)$ ist.

□

Abbruchkriterien für das Newtonverfahren

$$\text{Lösung von } f(x) = 0 \Longleftrightarrow \text{Minimierung des Residuums } f(x^{(k)})$$

Man könnte erwarten, dass $(x^{(k)})_{k\in\mathbb{N}}$ eine monoton fallende Folge bildet, d. h.

$$\exists L = konstant,\ 0 < L < 1,\ \text{so dass } \|f(x^{(k+1)})\| \leq L \cdot \|f(x^{(k)})\| \quad k = 0, 1, 2, \ldots$$

Dieses Kriterium heißt **Monotonietest**. Es ist leider nicht affin invariant.

$$\begin{aligned} \|f(x^{(k)})\|_2^2 &= \langle f(x^{(k)}), f(x^{(k-1)})\rangle \qquad (\text{wobei } \langle x, y\rangle = y^{\mathsf{T}}x) \\ \|Af(x^{(k)})\|_2^2 &= \langle Af(x^{(k)}), Af(x^{(k-1)})\rangle \end{aligned}$$

Es bietet sich ein alternativer Monotonietest an:

$$\left\|\left(D_f(x^{(k)})\right)^{-1} f(x^{(k+1)})\right\| \leq L \cdot \left\|\left(D_f(x^{(k)})\right)^{-1} f(x^{(k)})\right\|$$

Dieser Test heißt **natürlicher Monotonietest**. Er ist offensichtlich **affin invariant**.

Da wir die rechte Seite $\Delta x^{(k)} = \left(D_f(x^{(k)})\right)^{-1} f(x^{(k)})$ bereits kennen (Newtonkorrektur), müssen wir nur für die linke Seite das zusätzliche Gleichungssystem

$$D_f(x^{(k)})\overline{\Delta x^{(k+1)}} = -f(x^{(k+1)}) \text{ lösen.}$$

Im Kapitel über lineare Gleichungssysteme werden wir sehen, dass man dies mit $\mathcal{O}(n^2)$ zusätzlichem Aufwand berechnen kann. Der natürliche Monotonietest lautet dann:

$$\|\overline{\Delta x^{(k+1)}}\| \leq L \cdot \|\Delta x^{(k)}\|$$

Deuflhard [siehe 4] schlägt für einen weiten Anwendungsbereich $L = 0.5$ vor.

Man bricht also ab, sobald

$$\|\Delta x^{(k)}\| \leq \varepsilon$$

wobei $\varepsilon > 0$ eine vorgegebene Genauigkeit sei. Gilt während der Iteration einmal

$$\|\overline{\Delta x^{(k+1)}}\| > L \cdot \|\Delta x^{(k)}\|$$

so ist die Iteration abzubrechen und mit einem neuen Startvektor $x^{(0)}$ zu wiederholen.

Unsere bisherigen Konvergenzkriterien ergeben nun gute Konvergenz, wenn die Startvektoren in der Nähe der Lösung $x^\star$ liegen. Dies ist ohne Kenntnis von $x^\star$ meistens nicht einfach zu erreichen. Wir betrachten daher jetzt eine Strategie mit der sich die globalen Konvergenzeigenschaften manchmal verbessern lassen:

$$x^{(k+1)} = x^{(k)} + \Delta x^{(k)}$$

Die Idee ist es, eine Dämpfung der Newtonkorrektur zu verwenden. Dazu wählen wir für jedes $k \in \mathbb{N}$ einen Dämpfungsparameter $\lambda_k \in (0, 1]$ und definieren die Newtoniterierte im k-ten Schritt als $x^{(k+1)} = x^{(k)} + \lambda_k \Delta x^{(k)}$.
Zur Auswahl der Dämpfungsparameter kann man wieder den natürlichen Monotonietest verwenden und zwar mit $L := 1 - \frac{\lambda_k}{2}$, d. h.

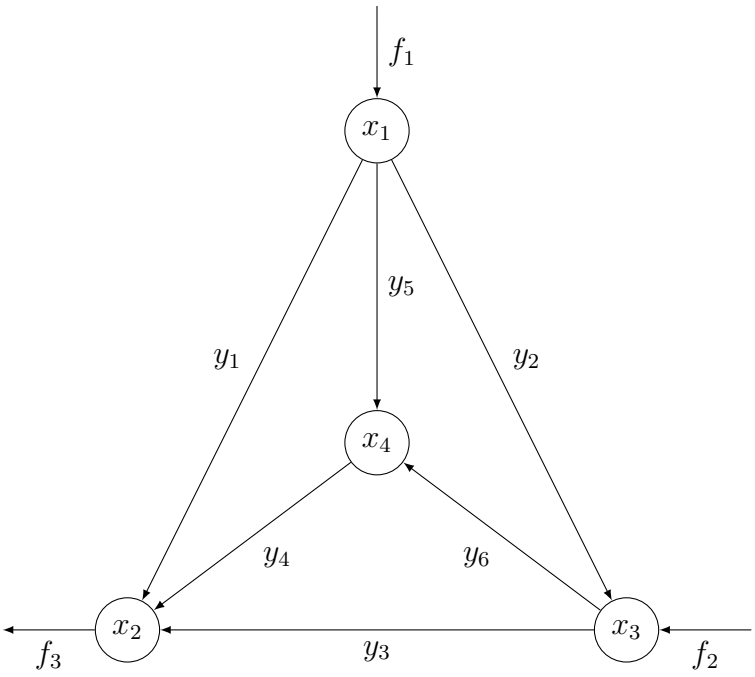
$$\|\overline{\Delta x^{(k+1)}(\lambda_k)}\| \leq \left(1 - \frac{\lambda_k}{2}\right) \|\Delta x^{(k)}\|$$

wobei $\overline{\Delta x^{(k+1)}(\lambda_k)} = -\Big(D_f(x^{(k)})\Big)^{-1} f\Big(\underbrace{x^{(k)} + \lambda_k \Delta x^{(k)}}_{x^{(k+1)}}\Big)$.

Die Dämpfungsparameter λ_k wählt man aus einer vorgegebenen Folge $1, \frac{1}{2}, \frac{1}{4}, \ldots, \lambda_{\min}$ und bricht die Iteration ab, falls man $\lambda_k < \lambda_{\min}$ benötigt. Hat man im k-ten Schritt ein λ_k gefunden, so versucht man es im $(k+1)$-ten Schritt mit $\lambda_{k+1} = \min(1, 2\lambda_k)$.
Theoretische Untersuchungen zu guten Dämpfungsstrategien finden sich im Buch von Deuflhard [4].

3. Klassische Iterationsverfahren für lineare Gleichungssysteme

3.1. Einführung: Stromnetze und Graphen



x_1, x_2, x_3, x_4 seien Haushalte oder Verteiler. In den Haushalten kann Strom entnommen werden (Senken) und in den Verteilern kann Strom eingespeist werden (Quellen) oder er wird nur weitergeleitet bzw. verteilt.
Die Knoten x_i bezeichnen wir auch als Potenziale, die Leitungen zwischen den Haushalten und Verteilern mit y_i (Kanten) und die Differenzen der Potenziale entlang der Kanten werden als **Potenzialdifferenzen** (**Spannungen**) $\hat{y}_i$ bezeichnet. Strom fließt von höherem zu niedrigerem Potenzial. Dies kann durch eine lineare Beziehung $Ax = \hat{y}$ dargestellt werden mit $x = \begin{pmatrix} x_1 & x_2 & x_3 & x_4 \end{pmatrix}^\mathsf{T}$ und $\hat{y} = \begin{pmatrix} \hat{y}_1 & \hat{y}_2 & \cdots & \hat{y}_6 \end{pmatrix}^\mathsf{T}$.

Dabei bekommt A den Eintrag 1 oder -1, wenn zwischen zwei Knoten eine Verbindung durch eine Kante besteht, das Vorzeichen bestimmt sich durch die Richtung des Stroms.

$$\Rightarrow Ax = \begin{pmatrix} -1 & 1 & 0 & 0 \\ -1 & 0 & 1 & 0 \\ 0 & 1 & -1 & 0 \\ 0 & 1 & 0 & -1 \\ -1 & 0 & 0 & 1 \\ 0 & 0 & -1 & 1 \end{pmatrix} \begin{pmatrix} x_1 \\ x_2 \\ x_3 \\ x_4 \end{pmatrix} = \begin{pmatrix} \hat{y}_1 \\ \hat{y}_2 \\ \hat{y}_3 \\ \hat{y}_4 \\ \hat{y}_5 \\ \hat{y}_6 \end{pmatrix} = \hat{y}$$

Die Stromleitungen haben in der Regel verschiedene Widerstände, aus denen sich die Leitfähigkeit c durch Kehrwertbildung ergibt, d. h.

$$c = \frac{1}{\text{Widerstand}}$$

Es seien $c_i,\ i = 1, \ldots, 6$ die Leitfähigkeiten der verschiedenen Kanten $y_i,\ i = 1, \ldots, 6$.

Ohmsche Gesetz:

Strom in einer Kante = Leitfähigkeit · Potenzialdifferenz $\Leftrightarrow y_i = c_i \cdot \hat{y}_i, \quad i = 1, \ldots, 6$ mit $y = \begin{pmatrix} y_1 & \ldots & y_6 \end{pmatrix}^\mathsf{T}$ bezeichnen wir auch den Strom im Stromnetz. Das Ohmsche Gesetz lässt sich durch folgenden linearen Zusammenhang beschreiben:

$$y = CAx$$

mit einer positiv definiten Diagonalmatrix $C = \operatorname{diag}(c_i)$.

Kirchhoffsche Knotenregel:
Der Fluss in einen Knoten hinein ist gleich dem Fluss aus diesem Knoten heraus.

$$\Rightarrow \quad \begin{array}{cccc|c} -y_1 & -y_5 & -y_2 & = & f_1 & x_1 \\ y_1 & +y_3 & +y_4 & = & -f_3 & x_2 \\ y_2 & -y_3 & -y_6 & = & f_2 & x_3 \\ -y_4 & +y_5 & +y_6 & = & 0 & x_4 \end{array}$$

Nachrechnen ergibt hieraus $A^\mathsf{T} y = f$ mit $f = \begin{pmatrix} f_1 & -f_3 & f_2 & 0 \end{pmatrix}^\mathsf{T}$. Durch einsetzen ergibt sich

$$A^\mathsf{T} C A x = f$$

Ohne Einschränkung: $C = I$ (Widerstand gleich in allen Knoten)

$$A^\mathsf{T} C A = A^\mathsf{T} A = \begin{pmatrix} 3 & -1 & -1 & -1 \\ -1 & 3 & -1 & -1 \\ -1 & -1 & 3 & -1 \\ -1 & -1 & -1 & 3 \end{pmatrix}$$

$\Rightarrow A^\mathsf{T} A$ ist nicht invertierbar.
Dies liegt daran, dass man zu den Potenzialen jeweils dieselbe Konstante c addieren kann, ohne die Potenzialdifferenzen zu verändern.

„Ein Knoten muss geerdet werden“.

Wir setzen dazu $x_4 = 0$ und streichen die letzte Zeile und Spalte der Matrix $A^\mathsf{T} A$.

$$\Rightarrow \begin{pmatrix} 3 & -1 & -1 \\ -1 & 3 & -1 \\ -1 & -1 & 3 \end{pmatrix} \begin{pmatrix} x_1 \\ x_2 \\ x_3 \end{pmatrix} = \begin{pmatrix} f_1 \\ -f_3 \\ f_2 \end{pmatrix}$$

$$\Rightarrow x = \begin{pmatrix} x_1 \\ x_2 \\ x_3 \end{pmatrix} = \begin{pmatrix} \frac{f_1}{2} + \frac{f_2}{4} - \frac{f_3}{4} \\ \frac{f_1}{4} + \frac{f_2}{2} - \frac{f_3}{4} \\ \frac{f_1}{4} + \frac{f_2}{4} - \frac{f_3}{2} \end{pmatrix}$$

Aus dem Ohmschen Gesetz $y = CAx \overset{C=I}{=} Ax$ erhalten wir nun die Ströme y_i:

$$y = \begin{pmatrix} y_1 \\ y_2 \\ y_3 \\ y_4 \\ y_5 \\ y_6 \end{pmatrix} = \begin{pmatrix} \frac{f_2}{4} - \frac{f_1}{4} \\ -\frac{f_1}{4} - \frac{f_3}{4} \\ \frac{f_2}{4} + \frac{f_3}{4} \\ \frac{f_1}{4} + \frac{f_2}{2} - \frac{f_3}{4} \\ -\frac{f_1}{2} - \frac{f_2}{4} + \frac{f_3}{4} \\ -\frac{f_1}{4} - \frac{f_2}{4} + \frac{f_3}{2} \end{pmatrix}$$

3.2. Gauß-Seidel-, Jacobi- und SOR-Verfahren

26.04.2012
7. Vorlesung

Betrachte das lineare Gleichungssystem $Ax = b,\ A \in \mathbb{R}^{n\times n},\ x, b \in \mathbb{R}^n,\ A$ invertierbar. Gesucht: x.
Additive Zerlegung von A: $A = D + L + R,\ D$ invertierbar.

$$D = \begin{pmatrix} a_{1,1} & 0 & \cdots & 0 \\ 0 & \ddots & \ddots & \vdots \\ \vdots & \ddots & \ddots & 0 \\ 0 & \cdots & 0 & a_{n,n} \end{pmatrix} \quad L = \begin{pmatrix} 0 & \cdots & \cdots & 0 \\ a_{2,1} & \ddots & & \vdots \\ \vdots & \ddots & \ddots & \vdots \\ a_{n,1} & \cdots & a_{n,n-1} & 0 \end{pmatrix} \quad R = \begin{pmatrix} 0 & a_{1,2} & \cdots & a_{1,n} \\ \vdots & \ddots & \ddots & \vdots \\ \vdots & & \ddots & a_{n-1,n} \\ 0 & \cdots & \cdots & 0 \end{pmatrix}$$

Jacobi-Verfahren (Einzelschrittverfahren)

$$Dx^{(k+1)} + Lx^{(k)} + Rx^{(k)} = b, \quad k \geq 0$$

mit gegebenem Startwert $x^{(0)} \in \mathbb{R}^n$.

Komponentenweise geschrieben ergibt sich hier:

$$x_i^{(k+1)} = \frac{b_i - \sum_{j\neq i} a_{i,j} x_j^{(k)}}{a_{i,i}}, \quad \text{für } i = 1, \ldots, n$$

Gauß-Seidel-Verfahren (Gesamtschrittverfahren)

$$(D + L)x^{(k+1)} + Rx^{(k)} = b, \quad k \geq 0$$

mit gegebenem Startwert $x^{(0)} \in \mathbb{R}^n$.

Komponentenweise geschrieben ergibt sich hier:

$$x_i^{(k+1)} = \frac{b_i - \sum_{j=1}^{i-1} a_{i,j} x_j^{(k+1)} - \sum_{j=i+1}^{n} a_{i,j} x_j^{(k)}}{a_{i,i}}, \quad \text{für } i = 1, \ldots, n$$

SOR-Verfahren (Successive-Overrelaxation)

Mit Hilfe des Gauß-Seidel-Zwischenwerts

$$\tilde{x}_i^{(k+1)} = \left(\frac{b_i - \sum_{j=1}^{i-1} a_{i,j} x_j^{(k+1)} - \sum_{j=i+1}^{n} a_{i,j} x_j^{(k)}}{a_{i,i}} \right)$$

erhält man als Verallgemeinerung des Gauß-Seidel-Verfahrens, den nächsten Wert $x_i^{(k+1)}$ als Linearkombination

$$x_i^{(k+1)} = (1 - \omega) x_i^{(k)} + \omega \cdot \tilde{x}_i^{(k+1)}, \quad \text{für } i = 1, \ldots, n$$

wobei $\omega \in \mathbb{R}$. Für das Gauß-Seidel-Verfahren gilt $\omega = 1$.
ω heißt hier **Relaxationsparameter** und führt uns auf das sogenannte Successive-Overrelaxation-Verfahren (SOR-Verfahren). Ist der Relaxationsparameter kleiner eins spricht man von **Unterrelaxation**, ist er größer als eins von **Überrelaxation**. Beim SOR-Verfahren gilt demnach $\omega \geq 1$.

Alle Verfahren lassen sich in folgender Form darstellen:

$$x^{(k+1)} = Bx^{(k)} + C, \text{ mit } x^{(0)} \in \mathbb{R}$$

Jacobi-Verfahren: $B = -D^{-1}(L+R), \quad C = D^{-1}b$

Gauß-Seidel-Verfahren: $B = -(D+L)^{-1}R, \quad C = (D+L)^{-1}b$

SOR-Verfahren: $B = (D+\omega L)^{-1}\Big((1-\omega)D - \omega R\Big), \quad C = \omega(D+\omega L)^{-1}b$

3.3. Konvergenzaussagen

Definition 3.3.1 (Matrixnorm):
Sei $\mathbb{K} \in \{\mathbb{R}, \mathbb{C}\}$, $\|\cdot\| : \mathbb{K}^n \to \mathbb{R}$ eine Norm und $A \in \mathbb{R}^{m\times n}$ eine Matrix. Dann heißt

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

die zugeordnete ***Matrixnorm****.*

Bemerkung 3.3.1:

$$\|A\| = \sup_{\substack{x\in\mathbb{R}^n \\ x\neq 0}} \frac{\|Ax\|}{\|x\|} = \sup_{x\in\mathbb{R}^n} \left\| A\frac{x}{\|x\|} \right\| = \sup_{\substack{y\in\mathbb{R}^n \\ \|y\|=1}} \|Ay\|$$

Beispiel (Maximumsnorm):
Sei $x \in \mathbb{K}^n$ und $\|x\|_\infty = \max_{i=1,\dots,n}\Big\{|x_1|,\dots,|x_n|\Big\} = \max_{i=1,\dots,n}\Big\{|x_i|\Big\} = 1$. Sei $A \in \mathbb{K}^{n\times n}$, $A \neq 0$.

$$\begin{aligned}
\|Ax\|_\infty &= \max_{i=1,\dots,n} \left| \sum_{j=1}^{n} a_{i,j}x_j \right| \\
&\leq \max_{i=1,\dots,n} \sum_{j=1}^{n} |a_{i,j}||x_j| \\
&\leq \max_{i=1,\dots,n} \sum_{j=1}^{n} |a_{i,j}|
\end{aligned}$$

Da $x \in \mathbb{K}^n$ mit $\|x\|_\infty = 1$ beliebig, folgt:

$$\|A\|_\infty \leq \max_{i=1,\dots,n} \sum_{j=1}^{n} |a_{i,j}|$$

Das Maximum wird für einen Index $i \in \{1,\dots,n\}$ angenommen, es sei $i = j$. Wir definieren folgenden Vektor $y = \begin{pmatrix} y_1 & \dots & y_n \end{pmatrix}^\mathsf{T} \in \mathbb{K}^n$ mit

$$y_k = \begin{cases} \frac{\overline{a_{j,k}}}{|a_{j,k}|}, & a_{j,k} \neq 0 \\ 0, & \text{sonst} \end{cases}$$

Nach Konstruktion gilt $\|y\|_\infty = 1$.

$$\begin{aligned}\|Ay\|_\infty &= \max_{i=1,\dots,n} \left|\sum_{k=1}^{n} a_{i,k} y_k\right| \\ &\geq \left|\sum_{k=1}^{n} a_{j,k} y_k\right| \\ &= \sum_{k=1}^{n} |a_{j,k}| \\ &= \max_{i=1,\dots,n} \sum_{k=1}^{n} |a_{j,k}|\end{aligned}$$

Also gilt:

$$\|A\|_\infty = \max_{i=1,\dots,n} \sum_{k=1}^{n} |a_{i,k}|$$

Hieraus ergibt sich der Name **maximale Zeilensummennorm**. Für $A = 0$ gilt dies trivialerweise.

Korollar 3.3.0.a:
Die zugeordnete Matrixnorm $\|\cdot\|$ hat folgende Eigenschaften:

1. *$\|\cdot\|$ ist eine Norm auf dem Vektorraum der Matrizen*
2. *Sei λ ein Eigenwert der Matrix A, dann gilt $\|A\| \geq |\lambda|$*
3. *Für beliebige $A \in \mathbb{K}^{m\times n}$, $B \in \mathbb{K}^{n\times k}$ gilt $\|A \cdot B\| \leq \|A\| \cdot \|B\|$*

Beweis:
Übung.

□

Definition 3.3.2 (Spektralradius):
Sei $A \in \mathbb{K}^{n\times n}$ eine Matrix und λ_i, $i = 1, \dots, n$ bezeichne die (nicht notwendigerweise verschiedenen) Eigenwerte von A. Dann heißt

$$\rho(A) = \max_{i=1,\dots,n} |\lambda_i|$$

*der **Spektralradius** von A.*

Satz 3.3.1:
Für jede Matrix $A \in \mathbb{K}^{n\times n}$ und jedes $\varepsilon > 0$ existiert eine Norm $\|\cdot\|_{A,\varepsilon}$ auf $\mathbb{K}^n$, so dass für die zugeordnete Matrixnorm gilt:

$$\|A\|_{A,\varepsilon} \leq \rho(A) + \varepsilon$$

Beweis:

Sei $D = \begin{pmatrix} 1 & 0 & \cdots & \cdots & 0 \\ 0 & \varepsilon & \ddots & & \vdots \\ \vdots & \ddots & \varepsilon^2 & \ddots & \vdots \\ \vdots & & \ddots & \ddots & 0 \\ 0 & \cdots & \cdots & 0 & \varepsilon^{n-1} \end{pmatrix} \in \mathbb{R}^{n\times n}$. Für eine beliebige Matrix $B \in \mathbb{K}^{n\times n}$ ergibt

BD, dass die k-te Spalte von B mit ε^{k-1} multipliziert wird. Entsprechend wird bei $D^{-1}B$ die k-te Zeile mit ε^{1-k} multipliziert. Sei nun $J = T^{-1}AT$ die Jordansche Normalform von A mit $J = \begin{pmatrix} \lambda_1 & \mu_1 & 0 & \cdots & 0 \\ 0 & \ddots & \ddots & \ddots & \vdots \\ \vdots & \ddots & \ddots & \ddots & 0 \\ \vdots & & \ddots & \ddots & \mu_{n-1} \\ 0 & \cdots & \cdots & 0 & \lambda_n \end{pmatrix}$, wobei λ_i Eigenwert von A und $\mu_i \in \{0, 1\}$. Dann gilt:

$$C = D^{-1}JD = \begin{pmatrix} \lambda_1 & \varepsilon\mu_1 & 0 & \cdots & 0 \\ 0 & \ddots & \ddots & \ddots & \vdots \\ \vdots & \ddots & \ddots & \ddots & 0 \\ \vdots & & \ddots & \ddots & \varepsilon\mu_{n-1} \\ 0 & \cdots & \cdots & 0 & \lambda_n \end{pmatrix}$$

Wir definieren $\|x\|_{A,\varepsilon} := \left\|(TD)^{-1}x\right\|_\infty$.

$$\begin{aligned}
\|A\|_{A,\varepsilon} &= \sup_{x\in\mathbb{K}^n} \frac{\left\|(TD)^{-1}Ax\right\|_\infty}{\left\|(TD)^{-1}x\right\|_\infty} \\
&= \sup_{(TD)^{-1}y\in\mathbb{K}^n} \frac{\left\|(TD)^{-1}ATDy\right\|_\infty}{\left\|(TD)^{-1}TDy\right\|_\infty} \\
&= \sup_{y\in\mathbb{K}^n} \frac{\left\|(TD)^{-1}ATDy\right\|_\infty}{\|y\|_\infty} \\
&= \sup_{y\in\mathbb{K}^n} \frac{\left\|D^{-1}JDy\right\|_\infty}{\|y\|_\infty} \\
&= \sup_{y\in\mathbb{K}^n} \frac{\|Cy\|_\infty}{\|y\|_\infty} \\
&\overset{\text{Definition}}{=} \|C\|_\infty \\
&\leq \max_{i=1,\ldots,n} \left(|\lambda_i| + \varepsilon|\mu_i|\right) \\
&\leq \max_{i=1,\ldots,n} |\lambda_i| + \varepsilon \\
&= \rho(A) + \varepsilon
\end{aligned}$$

□

30.04.2012
8. Vorlesung

Definition 3.3.3 (Konvergenz):
Es seinen $B \in \mathbb{K}^{n\times n}$, $c \in \mathbb{K}^n$ und $x^{(k)} \in \mathbb{K}^n$, $k \in \mathbb{N}$. Das iterative Verfahren

$$x^{(k+1)} = Bx^{(k)} + c$$

*heißt **konvergent**, wenn für alle $c \in \mathbb{K}^n$ ein vom Startwert $x^{(0)} \in \mathbb{K}^n$ unabhängiger Grenzwert $x^\star = x^\star(c) \in \mathbb{K}^n$ existiert gegen den $x^{(k)}$ konvergiert.*

Satz 3.3.2:
Das Iterationsverfahren

$$x^{(k+1)} = Bx^{(k)} + c$$

ist genau dann konvergent, wenn $\rho(B) < 1$. In diesem Fall konvergiert es gegen die eindeutig bestimmte Lösung $x^\star$ von $x^\star = Bx^\star + c$.

Beweis:

„⇐“: Sei $\rho(B) < 1$, dann existiert nach Satz 3.3.1 eine Norm $\|\cdot\|$, so dass $\|B\| < 1$. Wir definieren nun die Funktion

$$g(x) := Bx + c$$

und suchen einen Fixpunkt von g. Dazu betrachten wir die Fixpunktiteration

$$x^{(k+1)} = g(x^{(k)}) = Bx^{(k)} + c$$

Offensichtlich gilt:

$$\begin{aligned} \|g(x) - g(y)\| &= \|B(x-y)\| \\ &\leq \underbrace{\|B\|}_{=:L} \|x-y\| \end{aligned}$$

Da $L := \|B\| < 1$, folgt aus dem Banachschen Fixpunktsatz (Satz 2.5.2) die Konvergenz des Fixpunktverfahrens für g gegen den eindeutig bestimmten Fixpunkt $x^\star$. Nach Konstruktion gilt auch

$$x^\star = g(x^\star) = Bx^\star + c$$

Streng genommen benötigen wir hier gegebenenfalls eine komplexe Version des Banachschen Fixpunktsatzes. Diese lässt sich aber analog beweisen.

„⇒“: Sei nun das Verfahren

$$x^{(k+1)} = Bx^{(k)} + c$$

konvergent. Es genügt zu zeigen, dass $\rho(B) < 1$. Da das Verfahren konvergent ist folgt $\forall c \in \mathbb{K}^n \; \exists$ ein vom Startwert $x^{(0)} \in \mathbb{K}^n$ unabhängigem Grenzwert $x^\star \in \mathbb{K}^n$. Wir können also $c := 0 \in \mathbb{K}^n$ wählen, dann konvergiert das Verfahren für $x^{(0)} := 0 \in \mathbb{K}^n$ gegen die Lösung $x^\star = 0 \in \mathbb{K}^n$. Sei nun y Eigenvektor von B zum Eigenvektor λ. Wir wählen nun als Startvektor $x^{(0)} := y$.

$$\begin{aligned} \Rightarrow x^{(k+1)} &= Bx^{(k)} \\ &= B^{k+1}x^{(0)} \\ &\underset{\substack{x^{(0)} \\ \text{Eigenvektor}}}{=} \lambda^{k+1}x^{(0)} \end{aligned}$$

Die so erhaltene Iteration konvergiert nur gegen $x^\star = 0 \in \mathbb{K}^n$, wenn $|\lambda| < 1$. Da λ ein beliebiger Eigenvektor ist, folgt $\rho(B) < 1$.

□

3.4. Konvergenzkriterien

Satz 3.4.1:
Sei $A \in \mathbb{K}^{n\times n}$ eine hermitesche positiv definite Matrix, dann konvergiert das SOR-Verfahren zur Lösung von $Ax = b$ für jede Wahl von $x^{(0)} \in \mathbb{K}^n$ und jede rechte Seite $b \in \mathbb{K}^n$, falls $0 < \omega < 2$.

Beweis:
Die Iterationsmatrix des SOR-Verfahrens lautet:

$$B_\omega = (D + \omega L)^{-1}\Big((1-\omega)D - \omega R\Big)$$

Es genügt zu zeigen $\rho(B_\omega) < 1$, wenn $0 < \omega < 2$. Sei λ betragsgrößter Eigenwert von B_ω, d. h. $B_\omega x = \lambda x$ mit $|\lambda| = \rho(B_\omega) \Leftrightarrow \Big((1-\omega)D - \omega R\Big)x = \lambda(D + \omega L)x$. Bilden wir daraus das innere Produkt mit $x^{\mathsf{H}} = \overline{x}^{\mathsf{T}}$ und $\langle x, y\rangle := y^{\mathsf{H}}x$

$$(1-\omega)\langle Dx, x\rangle - \omega\langle Rx, x\rangle = \lambda\Big(\langle Dx, x\rangle + \omega\langle Lx, x\rangle\Big) \tag{3.1}$$

Sei nun $d := \langle Dx, x\rangle,\ l := \langle Lx, x\rangle$. Da A hermitesch ist, gilt:

$$\begin{aligned}
\langle Rx, x\rangle &= x^{\mathsf{H}}Rx \\
&= (R^{\mathsf{H}}x)^{\mathsf{H}}x \\
&= (Lx)^{\mathsf{H}}x \\
&= \langle x, Lx\rangle \\
&= \overline{\langle Lx, x\rangle} \\
&= \bar{l}
\end{aligned}$$

$$\begin{aligned}
&\overset{(3.1)}{\Leftrightarrow} (1-\omega)d - \omega\bar{l} = \lambda(d + \omega l) \\
&\Leftrightarrow\ \lambda = \frac{(1-\omega)d - \omega\bar{l}}{d + \omega l}
\end{aligned}$$

Unter Benutzung von $l = a + \imath b,\ a, b \in \mathbb{R}$ und $\imath^2 = -1$ erhalten wir daraus

$$|\lambda|^2 = \frac{\Big((1-\omega)d - \omega a\Big)^2 + \omega^2 b^2}{(d + \omega a)^2 + \omega^2 b^2}$$

Also gilt:

$$\begin{aligned}
|\lambda| < 1 \ &\Leftrightarrow\ |(1-\omega)d - \omega a| < |d + \omega a| \\
&\overset{\hat{a}:=\frac{a}{d}}{\Leftrightarrow} |(1-\omega)d - \omega\hat{a}| < |d + \omega\hat{a}|
\end{aligned}$$

Da A positiv definit ist gilt:

$$\begin{aligned}
0 < (Ax, x) &\overset{A=D+L+R}{=} d + l + \bar{l} \\
&= d + 2a \\
&= d(1 + 2\hat{a})
\end{aligned}$$

$$\begin{aligned}
A \text{ positiv definit } &\Rightarrow d > 0 \\
&\Rightarrow \hat{a} > -\tfrac{1}{2}
\end{aligned}$$

Mit $0 < \omega < 2$ gilt dann:

$$\begin{aligned} |1 - \omega - \omega\hat{a}| &< 1 + \omega\hat{a} \\ &= |1 + \omega\hat{a}| \end{aligned}$$

□

Die Bedingung $0 < \omega < 2$ ist nicht nur hinreichend für die Konvergenz des SOR-Verfahrens, sondern auch notwendig.

Satz 3.4.2:
Sei B_ω die SOR-Iterationsmatrix für eine beliebige Matrix A mit nicht verschwindendem Diagonalelement (von A). Dann gilt:

$$\rho(B_\omega) \geq |\omega - 1| \quad \forall \omega \in \mathbb{C}$$

Beweis:

Sei $B_\omega = (D + \omega L)^{-1}\Big((1-\omega)D - \omega R\Big) \in \mathbb{K}^{n\times n}$
$= (I + \omega D^{-1}L)^{-1}\cancel{D^{-1}}\cancel{D}\Big((1-\omega)I - \omega D^{-1}R\Big)$

Nach Konstruktion sind $I + \omega D^{-1}L$ und $(1-\omega)I - \omega D^{-1}R$ Dreiecksmatrizen. Daher gilt:

$$\det(I + \omega D^{-1}L) = 1$$
$$\det\Big((1-\omega)I - \omega D^{-1}R\Big) = (1-\omega)^n$$

$\Rightarrow \det(B_\omega) = (1-\omega)^n$
Aus der Jordanschen Normalform folgt auch:

$$\det(B_\omega) = \prod_{i=1}^{n} \lambda_i$$

wobei λ_i die Eigenwerte von B_ω bezeichne.
$\Rightarrow \prod_{i=1}^{n} |\lambda_i| = |1-\omega|^n$
$\Rightarrow$ Es muss mindestens ein Eigenwert $|\lambda_i| \geq |1-\omega|$ existieren.

□

Definition 3.4.1 (starkes Zeilensummenkriterium):
Eine Matrix erfüllt das ***starke Zeilensummenkriterium****, wenn*

$$|a_{i,i}| > \sum_{i \neq j} |a_{i,j}| \quad \forall i = 1, \ldots, n$$

Beispiel:

$$A = \begin{pmatrix} 4 & 1 & 2 & 0 \\ 1 & 4 & 1 & 0 \\ 2 & 0 & 4 & 0 \\ 0 & 0 & 2 & 4 \end{pmatrix}, \quad A = \begin{pmatrix} 9 & 8 & 0 \\ -5 & 10 & 4 \\ 1 & -6 & 8 \end{pmatrix}$$

Satz 3.4.3:
Sei $A \in \mathbb{K}^{n\times n}$ invertierbar und erfülle das starke Zeilensummenkriterium. Dann konvergiert das Jacobi-Verfahren zur Lösung von $Ax = b$ für jede rechte Seite $b \in \mathbb{K}^n$ und jedem Startvektor $x^{(0)} \in \mathbb{K}^n$ gegen die eindeutig bestimmte Lösung $x^\star$ von $Ax = b$.

Beweis:
Das Jacobi-Verfahren lautet

$$x^{(k+1)} = Bx^{(k)} + C$$

mit $B = -D^{-1}(L+R)$, $C = D^{-1}b$. Nach Satz 3.3.2 genügt es zu zeigen, dass $\rho(B) < 1$ ist.

$$\begin{aligned} \|B\|_\infty &= \|D^{-1}(L+R)\|_\infty \\ &= \max_{i=1,\dots,n} \frac{\sum_{j\neq i} |a_{i,j}|}{|a_{i,i}|} \\ &\overset{\substack{\text{starkes} \\ \text{Zeilensummen-} \\ \text{kriterium erfüllt}}}{<} 1 \end{aligned}$$

Mit $\rho(B) \leq \|B\|_\infty < 1$ folgt die Behauptung.

□

Optimale Wahl des Relaxationsparameter

Exkurs: Rademacher

Definition 3.4.2 (konsistent geordnet):
Man nennt die Matrix $A \in \mathbb{R}^{n\times n}$ mit der additiven Aufspaltung $A = D + L + R$ ***konsistent geordnet****, wenn die Eigenwerte der Matrizen*

$$B(\alpha) = -D^{-1}\{\alpha L + \alpha^{-1}R\}, \quad \alpha \in \mathbb{C}$$

unabhängig vom Parameter α also stets gleich denen der Jacobi-Iterationsmatrix $B = B(1)$ sind.

Satz 3.4.4 (Eigenwertbeziehung):
Sei J die Jacobi-Iterationsmatrix und B_ω die SOR-Iterationsmatrix.
Die Matrix $A \in \mathbb{R}^{n\times n}$ sei konsistent geordnet und $0 \leq \omega \leq 2$. Dann besteht zwischen den Eigenwerten $\mu \in \sigma(J)$ und $\lambda \in \sigma(B_\omega)$ die Beziehung

$$\sqrt{\lambda}\omega\mu = \lambda + \omega - 1$$

Definition 3.4.3 (optimaler Relaxationsparameter):
Der ***optimale Relaxationsparameter*** *$\omega_{opt} \in (0,2)$ ist durch die Bedingung*

$$\rho(B_{\omega_{opt}}) \leq \rho(B_\omega), \quad \omega \in (0,2)$$

charakterisiert, wobei B_ω die SOR-Iterationsmatrix beschreibt.

Satz 3.4.5 (optimaler Parameter):
Sei J die Jacobi-Iterationsmatrix und B_ω die SOR-Iterationsmatrix.
Es sei $A \in \mathbb{R}^{n\times n}$ konsistent geordnet. Weiter seien die Eigenwerte von J reell und es gelte $\rho := \rho(J) < 1$. Dann gilt

$$\omega_{opt} = \frac{2}{1+\sqrt{1-\rho^2}}$$

und

$$\rho(B_{\omega_{opt}}) = \omega_{opt} - 1 = \frac{1-\sqrt{1-\rho^2}}{1+\sqrt{1-\rho^2}} < 1$$

Allgemein ist dabei für $0 < \omega < 2$:

$$\rho(B_\omega) = \begin{cases} \omega - 1 & \omega_{opt} \leq \omega \\ \frac{1}{4}\left(\rho\omega + \sqrt{\rho^2\omega^2 - 4(\omega-1)}\right)^2 & \omega \leq \omega_{opt} \end{cases}$$

3.5. Abstiegsverfahren

Exkurs: Rademacher

Definition 3.5.1 (A-Norm):
Sei $A \in \mathbb{R}^{n\times n}$ eine symmetrische positiv definite Matrix.

$$\langle Ax, y\rangle = \langle x, Ay\rangle \quad \forall x, y \in \mathbb{R}^n$$
$$\langle Ax, x\rangle > 0 \quad \forall x \in \mathbb{R}^{n\times n} \setminus \{0\}$$

*Sei $\langle\cdot,\cdot\rangle$ das euklidische Skalarprodukt auf $\mathbb{R}^n$ und $\|\cdot\|$ die euklidische Vektornorm. Dann wird für die Matrix A zusätzlich die sogenannte **A-Norm** definiert*

$$\|x\|_A = \langle Ax, x\rangle^{\frac{1}{2}}$$

Die Matrix A sei (symmetrisch) positiv definit. Die eindeutige Lösung des Gleichungssystems $Ax = b$ ist charakterisiert, durch die Eigenschaft

$$Q(x) < Q(y) \quad \forall y \in \mathbb{R}^n \setminus \{x\}$$

mit dem quadratischen Funktional

$$Q(y) := \frac{1}{2}\langle Ay, y\rangle - \langle b, y\rangle$$

Definition 3.5.2 (Abstiegsverfahren):
Die ***Abstiegsverfahren*** *bestimmen ausgehend von einem geeigneten Startvektor* $x^{(0)} \in \mathbb{R}^n$ *eine Folge von Iterierten* $x^{(k)}$, $k \in \mathbb{N}$, *durch*

$$x^{(k+1)} = x^{(k)} + \alpha_k r^{(k)}$$

Dabei sind die $r^{(k)}$ *vorgegebene oder auch erst im Verlauf der Iteration berechnete Abstiegsrichtung, und die Schrittweiten* $\alpha_k \in \mathbb{R}$ *sind durch die folgende Vorschrift bestimmt (****line search-Methode****):*

$$Q(x^{(k+1)}) = \min_{\alpha \in \mathbb{R}} Q(x^{(k)} + \alpha r^{(k)})$$

Für das Gradientenverfahren gilt die Fehlerabschätzung

$$\|x^{(k)} - x\|_A \leq \left(\frac{1 - \frac{1}{\kappa}}{1 + \frac{1}{\kappa}} \right)^k \|x^{(0)} - x\|_A \quad k \in \mathbb{N}$$

mit der Spektralkonditionszahl $\kappa := \text{cond}_2(A) = \frac{\lambda_{\max}}{\lambda_{\min}}$ von A.

Definition 3.5.3 (A-Orthogonal):
Wir nennen zwei Vektoren $y, z \in \mathbb{R}^n$ ***A-Orthogonal*** *oder* ***A-Konjugiert****, wenn gilt*

$$\langle y, Az \rangle = 0$$

4. Rechnerarithmetik

4.1. Einführung

„Verrechnet“ (Deutschlandradio, `http://www.dradio.de/aktuell/791580/`)

4.2. Zahldarstellung

03.05.2012
9. Vorlesung

[Die Darstellung in diesem Kapitel folgt 5]

Definition 4.2.1 (d-adische Zahldarstellung):
Sei $d \in \mathbb{Z}$, $|d| > 1$, $x \in \mathbb{R}$. Die d-näre oder d-adische Darstellung von x ist

$$x = \pm \sum_{k=-l}^{\infty} a_k d^{-k},\ a_k \in \mathbb{N},\ 0 \leq a_k < |d|$$

*Man nennt d die **Basis**. Wenn die Basis d gewählt ist, so kann man die dargestellte Zahl durch aneinanderreihen der a_k darstellen: $\pm a_{-l} a_{-l+1} \dots a_{-1} a_0 a_1 \dots$.*

Beispiel:

1. $d = 10$:
   $\pi \approx 3.14 = 3 \cdot 10^0 + 1 \cdot 10^{-1} + 4 \cdot 10^{-2}$
2. $d = 2$:

$$\begin{aligned}(3.14)_{10} &\approx 11.001000111101\dots \\ &= 1 \cdot 2^1 + 1 \cdot 2^0 + 0 \cdot 2^{-1} + 0 \cdot 2^{-2} + 1 \cdot 2^{-3} + 0 \cdot 2^{-4} + \dots\end{aligned}$$

Die Darstellung in verschiedenen Zahlensystemen kann verschieden viele Ziffern benötigen. Beim numerischen rechnen wird die (normierte) **Gleitkommadarstellung** (floating point representation) benutzt:

$$x = \pm \left(\sum_{k=1}^{l} a_k d^{-k} \right) \cdot d^e, \quad a_1 \neq 0$$

wobei $m = \sum_{k=1}^{l} a_k d^{-k}$ **Mantisse** und e **Exponent** heißen.

$$\begin{aligned}(3.14)_{10} &= (3 \cdot 10^{-1} + 1 \cdot 10^{-2} + 4 \cdot 10^{-3}) \cdot 10^1 \\ &= (0.314) \cdot 10^1\end{aligned}$$

Die normierte Gleitkommadarstellung ($a_1 \neq 0$) hat den Vorteil, dass man im dualen System a_1 nicht abspeichern muss. Zum speichern einer binären Zahl benötigt man also nur $l - 1$ Ziffern der Mantisse. a_1 wird auch als „hidden Bit“ bezeichnet (Bit = binary digit). Die Gleitkommazahlen (floating point numbers) bilden eine endliche Teilmenge der

reellen Zahlen:

$$\begin{aligned}\mathbb{F} &= \mathbb{F}_d \\ &:= \Big\{ x \in \mathbb{R} \;\Big|\; \exists a_i \in \mathbb{N} \text{ mit } 0 \le a_i < d, \quad \exists e \in \{e_{\min}, \dots, e_{\max}\} \subset \mathbb{Z}, \text{ so dass } x = a \cdot d^e \\ &\qquad \text{mit } a = v \cdot \sum_{i=1}^{l} a_i d^{-i}, \quad l \in \mathbb{N}, \quad v \in \{-1, 1\} \Big\} \cup \{0\}\end{aligned}$$

$e_{\min}$ und $e_{\max}$ bestimmen die betragsmäßig kleinste und größte Zahl, die sich auf dem Rechner (mit $\mathbb{F}_d$) darstellen lassen. Dies ist abhängig vom Rechner und Compiler.

Definition 4.2.2 (Maschinenzahl):
Eine Maschinenzahl ist eine Zahl, die unter Berücksichtigung der vorhandenen Vereinbarungen in einer Maschine (Rechner) exakt darstellbar ist.

4.3. Gleitkommaarithmetik

Im Folgenden beschränken wir uns auf $d = 2$. Die **relative Genauigkeit** der Gleitkommazahlen wird durch die Länge l der Mantisse bestimmt. Genauer gesagt, definiert man die **Maschinengenauigkeit *eps*** als Betrag der Differenz zwischen Eins und der kleinsten Maschinenzahl, die größer ist als Eins, d. h.

$$eps := \left| 1 - \left(1 + 2^{-(l-1)} \right) \right| = 2^{-(l-1)}$$

Jede Zahl $x \in \mathbb{R}$ lässt sich durch runden auf die nächste Maschinenzahl als Gleitkommazahl darstellen. Wir führen dazu die Notation $rd(x) \in \mathbb{F}$ ein, wobei

$$\frac{|x - rd(x)|}{|x|} \le \frac{eps}{2} = 2^{-l}$$

gelten soll (round to nearest). Es gibt also ein $\delta \in \mathbb{R}$ mit $|\delta| < eps$, so dass $rd(x) = x(1+\delta)$.

WICHTIG: Es garantiert, dass eine reelle Zahl, obwohl sie nicht exakt im Rechner dargestellt werden kann, immerhin exakt ist **bis auf einen Faktor $1 + eps$**.

Bei Maschinenoperationen, d. h. Rechenoperationen mit Gleitkommazahlen, entstehen im Allgemeinen nicht wieder Gleitkommazahlen. Daher müssen hierfür die Grundrechenarten neu definiert werden:

$$\begin{aligned}&\text{Seien } x, y \in \mathbb{F} : \\ &x \oplus y \;:= \mathrm{rd}(x + y) \\ &x \ominus y \;:= \mathrm{rd}(x - y) \\ &x \odot y \;:= \mathrm{rd}(x \cdot y) \\ &x \oslash y \;:= \mathrm{rd}(x / y), \quad y \neq 0\end{aligned}$$

4.4. IEEE-Arithmetik

Es gibt zwei Implementierungen verschiedener Genauigkeit (single und double precision). [Für weitere Details siehe 5]

4.4.1. Einfache Genauigkeit (single precision)

Bei einfacher Genauigkeit wird jede Gleitkommazahl in einem Speicherplatz der Länge 32 gespeichert. Einem sogenannten 32-Bit-Wort. Gespeichert werden:

1. 1 Vorzeichenbit v
2. Ein Exponent e der Länge 8
3. Eine Mantisse m der Länge 23

Die erste Ziffer a_1 (hidden Bit) wird in der IEEE-Arithmetik nicht gespeichert. Weiterhin sind $e_{\min} = -126$ und $e_{\max} = 127$. Die kleinste, darstellbare, positive (normierte) Gleitkommazahl ist somit

$$N_{\min} = (1 + \underbrace{0.00\ldots0}_{23}) \cdot 2^{-126} \approx 1.18 \cdot 10^{-38}$$

und die entsprechend größte Zahl ist

$$N_{\max} = (1 + \underbrace{0.11\ldots1}_{23}) \cdot 2^{127} = (2 - 2^{-23}) \cdot 2^{127} \approx 2^{128} \approx 3.40 \cdot 10^{38}$$

Die Zahlen $e_{\min}$ und $e_{\max}$ ergeben sich aus der Länge 8 des Exponenten, da $2^8 = 256$ ist. Somit haben wir theoretisch 256 verschiedene Möglichkeiten für den Exponenten $e = (e_1|e_2|\ldots|e_8)$:

$$\begin{aligned}
(00\ldots0)_2 = (0)_{10} &\equiv \pm(0.a_1a_2\ldots a_{23}) \cdot 2^{-126} \\
(00\ldots1)_2 = (1)_{10} &\equiv \pm(1.a_1a_2\ldots a_{23}) \cdot 2^{-126} \\
(00\ldots10)_2 = (2)_{10} &\equiv \pm(1.a_1a_2\ldots a_{23}) \cdot 2^{-125} \\
\vdots \quad\quad\quad \vdots &\quad\quad\quad \vdots \\
(011\ldots11)_2 = (127)_{10} &\equiv \pm(1.a_1a_2\ldots a_{23}) \cdot 2^{0} \\
(10\ldots00)_2 = (128)_{10} &\equiv \pm(1.a_1a_2\ldots a_{23}) \cdot 2^{1} \\
\vdots \quad\quad\quad \vdots &\quad\quad\quad \vdots \\
(11\ldots10)_2 = (254)_{10} &\equiv \pm(1.a_1a_2\ldots a_{23}) \cdot 2^{127}
\end{aligned}$$

Die Zahl null kann nur durch $e(00\ldots0)$ und $(a_1|a_2|\ldots|a_{23}) = (\underbrace{00\ldots0}_{23})$ dargestellt werden, d. h. durch

$$\pm(00\ldots0) \cdot 2^{-126}$$

Der Faktor 2^{-126} mag zunächst verwirren, er erlaubt uns jedoch noch kleinere Zahlen darzustellen, als die kleinste normierte Gleitkommazahl:
$\Rightarrow$ subnormale Zahlen.
Bei diesen Darstellungen haben wir nur 255 Möglichkeiten ausgenutzt. Theoretisch könnte

man auch bei 2^{-127} beginnen. Die Darstellung

$$(\underbrace{11\ldots1}_{8}) = (255)_{10} \equiv \pm(1.a_1a_2\ldots a_{23}) \cdot 2^{-127}$$

ermöglicht in der IEEE-Arithmetik die Darstellung von $\pm\infty$ und NaN (später mehr dazu).

Subnormale Zahlen

Die Darstellung

$$(00\ldots0)_2 = (0)_{10} \equiv \pm(0.a_1a_2\ldots a_{23}) \cdot 2^{-126} \tag{4.1}$$

mit $(a_1|a_2|\ldots|a_{23}) = (\underbrace{00\ldots0}_{23})$ erscheint zunächst überraschend, erlaubt uns aber noch kleinere Zahlen als 2^{-126} darzustellen, allerdings mit geringerer Genauigkeit:
subnormale Gleitkommazahlen.
Ist die Mantisse in (4.1) ungleich null, so können wir Zahlen zwischen 2^{-127} und 2^{-149} darstellen.

$$(0.1\underbrace{0\ldots0}_{22}) \cdot 2^{-126} = 2^{-127}$$
$$(0.\underbrace{0\ldots0}_{22}1) \cdot 2^{-126} = 2^{-149}$$

Die Maschinengenauigkeit *eps* in einfacher Genauigkeit ist:

$$\begin{aligned} eps &= 2^{-(l-1)} \\ &= 2^{-23} \\ &\approx 1.19 \cdot 10^{-7} \end{aligned}$$

4.4.2. Doppelte Genauigkeit (double precision)

Jede Gleitkommazahl in doppelter Genauigkeit wird in einem 64-Bit-Wort gespeichert:

1. 1 Vorzeichenbit v
2. Ein Exponent e der Länge 11
3. Eine Mantisse m der Länge 52

Prinzipiell ist die Vorgehensweise analog zur einfachen Genauigkeit:

$$\begin{aligned} & e_{\min} = -1022 \\ & e_{\max} = 1023 \\ \Rightarrow\quad & N_{\min} = 2^{-1022} && \approx 2.33 \cdot 10^{-308} \\ & N_{\max} = (2 - 2^{-52}) \cdot 2^{1023} && \approx 1.80 \cdot 10^{308} \end{aligned}$$

Die Maschinengenauigkeit ist dann:

$$eps = 2^{-52} \approx 2.22 \cdot 10^{-16}$$

07.05.2012
10. Vorlesung

Beispiel:
$a = \frac{4}{3}$, $b = a - 1$, $c = 3b$, $e = 1 - c$.
Analytisch ist $e = 0$, in der IEEE-Arithmetik ist $e = eps$.

4.4.3. Erweitertes Format (extended format)

80-Bit-Wörter für die Darstellung von Gleitkommazahlen:

1. 1 Vorzeichenbit v
2. Ein Exponent e der Länge 15
3. Eine Mantisse m der Länge 64

Es wird **kein** hidden Bit verwendet. Die Maschinengenauigkeit im extended format ist

$$eps = 2^{-63} \approx 1.08 \cdot 10^{-19}$$

Die Implementierung des extended format ist hard-/softwareabhängig:

- INTEL: Hardware
- SUN Workstations: Software

4.4.4. Runden (rounding)

Standardeinstellung im IEEE:

– Runden auf die nächstgelegene Gleitkommazahl (round to nearest)

Zusätzlich gilt noch:

– Liegt x genau zwischen zwei Gleitkommazahlen, so wird auf die nächste gerade Zahl gerundet

Im IEEE gibt es vier verschiedene Rundungsmodi [siehe 5, Kapitel 5 und 6].

4.4.5. Ausnahmen (exceptions)

Division durch null:
Vor dem IEEE-Standard gab es zwei (Standard-)Möglichkeiten:

1. Ergebnis wird auf die größte darstellbare Gleitkommazahl gesetzt.
   Nachteil: $\frac{1}{0} - \frac{1}{0} = 0$
2. Abbruch mit Fehlermeldung
   $\Rightarrow$ Programmierer weiß, dass ein bzw. welcher Fehler aufgetreten ist

In der IEEE-Arithmetik wird der Wert auf ∞ gesetzt. In MATLAB wird dafür das Symbol Inf verwendet. Das Programm rechnet dann mit sinnvollen Regeln für Inf weiter.

Ungültige Operatoren (invalid operations, NaN)

Da es in MATLAB möglich ist mit Inf zu rechnen, können ungültige Operationen bzw. Ergebnisse auftauchen: $0 \cdot Inf$, $\frac{0}{0}$.
In der IEEE-Arithmetik erhalten sie den Wert NaN (Not a Number).

Überlauf (overflow)

Wir sprechen vom Überlauf (overflow), wenn das Ergebnis einer Operation endlich, aber größer als $N_{\max}$ ist. Bei dem Roundingmode „round to nearest“ wird als Ergebnis ∞ angegeben.

Unterlauf (underflow)

Wir sprechen vom Unterlauf (underflow), falls das exakte Resultat einer Rechnung ungleich null, aber betragsmäßig kleiner als $N_{\min}$ ist, der kleinsten, normierten, positiven Gleitkommazahl.
Vor dem IEEE-Standard wurde das Ergebnis auf null gesetzt (flush to zero). In der IEEE-Arithmetik wird das Ergebnis korrekt gerundet und möglichst als subnormale Gleitkommazahl dargestellt. Man spricht von schrittweisem Unterlauf („gradual underflow“). Lässt sich das Ergebnis auch so nicht darstellen, wird es auf null gesetzt.

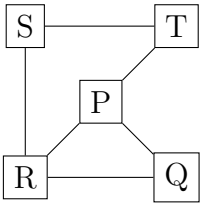
Inexakte Ergebnisse

Eigentlich keine Ausnahme, da Ergebnisse, die nicht als Gleitkommazahl dargestellt werden können, entsprechend gerundet werden.

5. Lineare Ausgleichsprobleme

5.1. Einführung: CARL FRIEDRICH GAUSS und die Landesvermessung des Königreichs Hannover (1821-1844)

Beispiel: Höhenmessung aus einer Vorlesung von Gauss, nach Richard Dedekind. Gesammelte Werke: „Gauß in seiner Vorlesung über die Methode der kleinsten Quadrate“, Seite 293 - 306, insbesondere Seite 299.

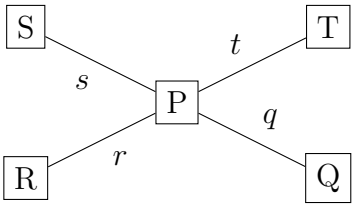


$$\begin{aligned} S &= \text{Ammensen} \\ T &= \text{Brocken} \\ Q &= \text{Meridianzeichen (Wehrider Papiermühle)} \\ R &= \text{Hohenhagen} \\ P &= \text{Göttinger Sternwarte} \end{aligned}$$

Natürlich können keine absoluten, sondern nur relative Höhen gemessen werden. Als Referenzpunkt wählen wir die Sternwarte. Die relativen Höhenmessungen ergaben:

$$\begin{aligned} Q &= P + 66.334 \\ R &= P + 349.366 \\ R &= Q + 283.596 \\ S &= Q + 206.58 \\ S &= R - 76.108 \\ T &= R + 648.427 \\ T &= S + 719.612 \end{aligned}$$

Da wir die Göttinger Sternwarte als Referenzpunkt definiert haben, führen wir nun die Höhendifferenzen bezüglich P als neue Variable ein.



$$\begin{aligned} q &= Q - P \\ r &= R - P \\ s &= S - P \\ t &= T - P \end{aligned}$$

Einsetzen der neuen Variablen in die Messungen ergibt das Gleichungssystem:

$$\begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ -1 & 1 & 0 & 0 \\ -1 & 0 & 1 & 0 \\ 0 & -1 & 1 & 0 \\ 0 & -1 & 0 & 1 \\ 0 & 0 & -1 & 1 \end{pmatrix} \cdot \begin{pmatrix} q \\ r \\ s \\ t \end{pmatrix} = \begin{pmatrix} 66.334 \\ 349.366 \\ 283.596 \\ 206.58 \\ -76.108 \\ 648.427 \\ 719.612 \end{pmatrix}$$

Aus den Messungen erhalten wir also ein **überbestimmtes lineares Gleichungssystem**: $Ax = b$, dessen **Lösung** $x = \begin{pmatrix} q & r & s & t \end{pmatrix}^{\mathsf{T}}$ die richtigen Höhenunterschiede sind.

5.2. Überbestimmte lineare Gleichungssysteme

Definition 5.2.1 (über- und unterbestimmte Gleichungssysteme):
Sei $A \in \mathbb{K}^{m\times n}$ und $b \in \mathbb{K}^m$. Das lineare Gleichungssystem $Ax = b$ heißt

***überbestimmt**, falls $m > n$*
***unterbestimmt**, falls $m < n$*

Überbestimmte Gleichungssysteme sind im Allgemeinen nicht lösbar, unterbestimmte Gleichungssysteme im Allgemeinen nicht eindeutig lösbar.

Satz 5.2.1:
Sei $A \in \mathbb{K}^{m\times n}$ und $b \in \mathbb{K}^m$, dann gilt:

$$Ax = b \text{ ist lösbar} \iff b \in \operatorname{range}(A)$$

Beweis:
Klar, siehe Lineare Algebra.

□

Man sieht leicht, dass im Gaußschen Beispiel $b \notin \operatorname{range}(A)$.

1. $\ker(A) = \{x \in \mathbb{K}^n | Ax = 0\}$
2. $\operatorname{range}(A) = \operatorname{Im}(A) = \{y \in \mathbb{K}^m | \exists z \in \mathbb{K}^n : y = Az\}$
3. $Ax = b$ ist lösbar $\iff b \perp \ker(A^{\mathsf{H}})$
4. Sei $W \subset \mathbb{K}^m$ ein Untervektorraum und sei $W^\perp := \{x \in \mathbb{K}^m : x \perp y \quad \forall y \in W\}$ das orthogonale Komplement, wobei $x \perp y \Leftrightarrow 0 = y^{\mathsf{H}}x \overset{\text{Definition}}{=} \langle x, y\rangle$
5. $\mathbb{K}^m = W \oplus W^\perp$

Satz 5.2.2:
Sei $A \in \mathbb{K}^{m\times n}$ eine beliebige Matrix, dann gilt:

$$\mathbb{K}^m = \ker(A^{\mathsf{H}}) \oplus \operatorname{range}(A)$$

Beweis:
Es genügt zu zeigen: $\ker(A^{\mathsf{H}}) = (\operatorname{range}(A))^\perp$.
Sei $x \in \ker(A^{\mathsf{H}})$ und $y \in \operatorname{range}(A)$ beliebig.
$\Rightarrow \exists z \in \mathbb{K}^n$ mit $y = Az$. Dann gilt:

$$\begin{aligned}\langle x, y\rangle &= \langle x, Az\rangle \\ &= \langle \underbrace{A^{\mathsf{H}}x}_{=0}, z\rangle = 0\end{aligned}$$

$\Rightarrow x \in (\operatorname{range}(A))^\perp$
Rückrichtung analog.

□

5.3. Abschwächung des Lösungsbegriff

10.05.2012
11. Vorlesung

Betrachte $Ax = b$ und definiere $r(y) := Ay - b$.
$r := r(y)$ bezeichnen wir als Residuum. r können wir auch als Fehler betrachten, wenn $y \in \mathbb{K}^n$ beliebig. Ist $r = 0$, so haben wir eine Lösung, sonst nicht. Als Fehlermaß können wir

$$\|r(y)\|_2 = \|r\|_2 = \sqrt{r^{\mathsf{H}} r}$$

nehmen. Eine Lösung ist genau dann gegeben, wenn $\|r\|_2 = 0$. Die Idee, den Lösungsbegriff abzuschwächen, ist nun nicht mehr zu verlangen, dass $\|r\|_2 = 0$ ist, sondern dass $\|r\|_2$ minimal wird.

$$\min_{y \in \mathbb{K}^n} \|r(y)\|_2 = \min_{y \in \mathbb{K}^n} \|Ay - b\|_2$$

Satz 5.3.1:
Sei $A \in \mathbb{K}^{m \times n}$, $b \in \mathbb{K}^m$ und $x_0 \in \mathbb{K}^n$. Dann gilt:

$$\|Ax_0 - b\|_2 = \min_{x \in \mathbb{K}^n} \|Ax - b\|_2 \Leftrightarrow A^H A x_0 = A^H b$$

Beweis:
Aus Satz 5.2.2 $\left(\mathbb{K}^m = \ker(A^{\mathsf{H}}) \oplus \operatorname{range}(A)\right)$ folgt $\exists b_1 \in \operatorname{range}(A),\ b_2 \in \ker(A^{\mathsf{H}})$, so dass $b = b_1 + b_2$. Dann gilt: $Ax - b_1 \perp b_2$.

$$\begin{aligned} \Rightarrow \|Ax - b\|_2^2 &= \|Ax - b_1 - b_2\|_2^2 \\ &= \|Ax - b_1\|_2^2 + \|b_2\|_2^2 \end{aligned}$$

$\Rightarrow \|Ax - b\|_2$ ist minimal
$\Leftrightarrow Ax - b_1 = 0$
$\Rightarrow A^{\mathsf{H}} A x = A^{\mathsf{H}} b_1 \Leftrightarrow (A^{\mathsf{H}}(Ax - b_1)) = 0$
$\Rightarrow \underbrace{Ax - b_1}_{\in \operatorname{range}(A)} \in \ker(A^{\mathsf{H}}) \underset{\text{Satz 5.2.2}}{\rightarrow} Ax - b_1 = 0$
$\underset{b = b_1 + b_2}{\overset{b_2 \in \ker(A^{\mathsf{H}})}{\Leftrightarrow}} A^{\mathsf{H}} A x = A^{\mathsf{H}} b$

□

1. Die Aufgabe, ein $x_0 \in \mathbb{K}^n$ zu finden, so dass

   $$\|Ax_0 - b\|_2 = \min_{x \in \mathbb{K}^n} \|Ax - b\|_2$$

   heißt **lineares Ausgleichsproblem**.

2. $A^{\mathsf{H}} A x = A^{\mathsf{H}} b$ heißt **System der Normalgleichungen** oder einfach **Normalgleichungen**. Die Lösung x_0 heißt **Kleinste-Quadratische-Lösung**; man spricht auch von der **Methode der kleinsten Quadrate**.

3. Geometrisch besagt $A^{\mathsf{T}}(Ax - b) = 0$, dass $Ax - b$ eine Normale auf $\operatorname{range}(A)$ bildet. Daher der Name Normalgleichung.

4. Die Normalgleichungen sind immer lösbar. Es gilt: $\ker(A) = \ker(A^{\mathsf{H}} A)$. Somit haben wir

   $$A^{\mathsf{H}} b \in \operatorname{range}(A) \overset{\text{Satz 5.2.2}}{\perp} \ker(A) = \ker(A^{\mathsf{H}} A)$$

   $\Rightarrow A^{\mathsf{H}} A x = A^{\mathsf{H}} b$ ist lösbar.
   Gilt zusätzlich $\operatorname{rang}(A) = n$, dann ist $A^{\mathsf{H}} A$ positiv definit.

5.4. Pseudoinverse

Da die zu einem Gleichungssystem gehörigen Normalgleichungen immer lösbar sind, auch wenn das Gleichungssystem selbst (im klassischen Sinne) nicht lösbar ist, führen wir jetzt den Begriff der verallgemeinerten Lösung eines Gleichungssystems ein.

Definition 5.4.1 (Moore-Penrose-Lösung):
Sei $A \in \mathbb{K}^{m\times n}$, $b \in \mathbb{K}^m$, dann heißt $x^+ \in \mathbb{K}^n$ genau dann ***verallgemeinerte*** *oder* ***Moore-Penrose-Lösung*** *von $Ax = b$, wenn gilt:*

1. *x^+ ist Kleinste-Quadrate-Lösung, d. h. $A^H Ax^+ = A^H b$*
2. *Unter allen Kleinste-Quadrate-Lösungen y von $Ax = b$ hat x^+ die kleinste Norm, d. h. $\|x^+\|_2 \leq \|y\|_2$*

Satz 5.4.1:
Sei $A \in \mathbb{K}^{m\times n}$, $b \in \mathbb{K}^m$, dann gilt:

1. *$x^+ \in \mathbb{K}^n$ ist genau dann verallgemeinerte Lösung von $Ax = b$, wenn $A^H Ax^+ = A^H b$ und $x^+ \in \text{range}(A)$*
2. *Die verallgemeinerte Lösung $x^+ \in \mathbb{K}^n$ von $Ax = b$ existiert und ist eindeutig bestimmt.*

Beweis:

1. „$\Leftarrow$“ Sei $x^+ \in \mathbb{K}^n$ mit $A^H Ax^+ = A^H b$ und $x^+ \in \text{range}(A^H)$ gegeben. Damit x^+ verallgemeinerte Lösung ist, muss x^+ minimal sein.
   Angenommen y ist weitere Kleinste-Quadrate-Lösung, dann

   $$A^H Ax^+ = A^H b = A^H Ay \Rightarrow (x^+ - y) \in \ker(A^H A)$$

   Es existiert also ein $y_0 \in \ker(A^H A) = \ker(A)$, so dass $y = x^+ + y_0$
   $\overset{\text{Satz 5.2.2}}{\underset{\text{Pythagoras}}{\Rightarrow}}$ $\|y\|_2^2 = \|x^+\|_2 + \|y_0\|_2^2 \geq \|x^+\|_2^2$. Also ist x^+ minimal und somit verallgemeinerte Lösung.

   „$\Rightarrow$“ Sei x^+ verallgemeinerte Lösung, dann gilt nach Definition 5.4.1: $A^H Ax^+ = A^H b$ und $\|x^+\|_2 \leq \|x\|_2 \quad \forall x \in$ Kleinste-Quadrate-Lösung
   $\overset{\text{Satz 5.2.2}}{\Rightarrow}$ $\exists x_0 \in \ker(A)$ und $x_1 \in \text{range}(A^H)$ mit $x^+ = x_0 + x_1$.
   Weiterhin gilt: $A^H Ax_1 = A^H Ax^+ = A^H b$. Also ist $x_1 \in \text{range}(A^H)$ auch Kleinste-Quadrate-Lösung von $Ax = b$.

   1. Fall: $x_0 = 0 \Rightarrow x^+ = x_1 \in \text{range}(A^H)$
   2. Fall: $x_0 \neq 0$, dann folgt mit $x_1 \perp x_0$, dass $\|x^+\|_2^2 = \|x_1\|_2^2 + \|x_0\|_2^2 > \|x_1\|_2^2$.
      Dies ist ein Widerspruch zur Minimalität von x^+. $\Rightarrow x^+ \in \text{range}(A^H)$.

2. Genügt zu zeigen:
   $A^H Ax = A^H b$ mit $x \in \text{range}(A^H)$ ist eindeutig lösbar. Da die Normalgleichungen immer lösbar sind, existiert immer eine Kleinste-Quadrate-Lösung x^+ von $Ax = b$. Wie schon im ersten Teil des Beweises, haben wir die Zerlegung

   $$x^+ = x_0 + x_1, \quad x_0 \in \ker(A), \quad x_1 \in \text{range}\, A^H \text{ und } A^H Ax_1 = A^H b$$

   Somit existiert immer eine Lösung der Normalgleichungen mit $x_1 \in \text{range}(A^H)$.

Angenommen, es existiere eine weitere Kleinste-Quadrate-Lösung $\tilde{x}_1 \in \text{range}(A^H)$, dann gilt: $A^H A\tilde{x}_1 = A^H b = A^H A x_1$
$\Rightarrow (x_1 - \tilde{x}_1) \in \ker(A^H A) = \ker(A)$
$\Rightarrow \underbrace{(x_1 - \tilde{x}_1)}_{\in \text{range}(A^H)} \in \ker(A) \cap \text{range}(A^H) = \{0\}$
$\Rightarrow x_1 = \tilde{x}_1$

□

Definition 5.4.2 (Pseudoinverse):
Sei $A \in \mathbb{K}^{m\times n}$ eine gegebene Matrix. Dann existiert für jedes $b \in \mathbb{K}^m$ genau eine verallgemeinerte Lösung $x^+ \in \mathbb{K}^n$. Diese definiert eine Abbildung

$$\begin{aligned} A^+ \colon \mathbb{K}^m &\to \mathbb{K}^n \\ b &\mapsto x^+ \end{aligned}$$

Alternativ können wir auch $x^+ = A^+ b$ schreiben. Die Abbildung A^+ heißt ***verallgemeinerte Inverse****,* ***Pseudoinverse*** *oder* ***Moore-Penrose-Inverse****.*

Satz 5.4.2:
Die Moore-Penrose-Inverse $A^+ \colon \mathbb{K}^m \to \mathbb{K}^n$ ist eine lineare Abbildung.

Beweis:
Gegeben seien $b, c \in \mathbb{K}^m$ und $\lambda, \mu \in \mathbb{K}$. Weiterhin seien $x, y, z \in \mathbb{K}^n$ verallgemeinerte Lösungen der Gleichungssysteme

$$\begin{aligned} Ax &= (\lambda b + \mu c) \\ Ay &= b \\ Az &= c \end{aligned}$$

d. h. es gilt: $x = A^+(\lambda b + \mu c), \quad y = A^+ b, \quad z = A^+ c.$
Weiterhin haben wir:

$$\begin{aligned} A^H A x &= A^H(\lambda b + \mu c) \\ &= \lambda A^H b + \mu A^H c \\ &= \lambda A^H A y + \mu A^H A z \\ &= A^H A(\lambda y + \mu z) \end{aligned}$$

$\Rightarrow \Big(x - (\lambda y + \mu z)\Big) \in \ker(A^H A) = \ker(A)$ und da x, y, z verallgemeinerte Lösungen sind, gibt es auch $\Big(x - (\lambda y + \mu z)\Big) \in \text{range}(A^H)$
$\Rightarrow \Big(x - (\lambda y + \mu z)\Big) \in \ker(A) \cap \text{range}(A^H) = \{0\}$
$\Rightarrow x = (\lambda y + \mu z)$
$\Rightarrow A^+(\lambda y + \mu z) = \lambda A^+ b + \mu A^+ c$

□

A^+ kann mit Hilfe der Singulärwertzerlegung berechnet werden, siehe Abschnitt 5.4.1. Unter gewissen Voraussetzungen an A kann man A^+ aber auch explizit angeben:

1. Ist A quadratisch und invertierbar, so ist $A^+ = A^{-1}$

2. Ist $A \in \mathbb{K}^{m\times n}$ eine Matrix mit vollem Rang, dann gilt:
   2.1. $m \geq n$: $A^+ = (A^{\mathsf{H}}A)^{-1}A^{\mathsf{H}}$
   2.2. $m < n$: $A^+ = A^{\mathsf{H}}(AA^{\mathsf{H}})^{-1}$

Beweis (zu 2.2.):
Zu einem beliebigen $b \in \mathbb{K}^m$ sei $x^+ \in \mathbb{K}^n$ die verallgemeinerte Lösung von $Ax = b$.
Es genügt zu zeigen: $x^+ = A^{\mathsf{H}}(AA^{\mathsf{H}})^{-1}b$.

$$
\begin{aligned}
x^+ \text{ verallgemeinerte Lösung} &\Rightarrow x^+ \in \operatorname{range}(A^{\mathsf{H}}) \\
&\Rightarrow \exists y \in \mathbb{K}^m, \text{ so dass } x^+ = A^{\mathsf{H}}y \\
&\Rightarrow A^{\mathsf{H}}A\underbrace{A^{\mathsf{H}}y}_{=x^+} = A^{\mathsf{H}}Ax^+ = A^{\mathsf{H}}b \\
&\Rightarrow (AA^{\mathsf{H}}y - b) \in \ker(A^{\mathsf{H}}) \underset{\substack{A \text{ voller} \\ \text{Rang}}}{=} \{0\} \\
&\Rightarrow AA^{\mathsf{H}}y = b
\end{aligned}
$$

$\operatorname{rang}(A) = \operatorname{rang}(A^{\mathsf{H}}) \Rightarrow A^{\mathsf{H}}$ hat ebenfalls vollen Rang

$\Rightarrow AA^{\mathsf{H}}$ ist hermitesch und positiv definit, also insbesondere invertierbar
$\Rightarrow y = (AA^{\mathsf{H}})^{-1}b$
$\Rightarrow x^+ = A^{\mathsf{H}}y = A^{\mathsf{H}}(AA^{\mathsf{H}})^{-1}b$

□

Satz 5.4.3:
Sei A^+ die verallgemeinerte Inverse von A. Dann gilt:

1. $AA^+A = A$
2. *Ist A eine normale $(n \times n)$-Matrix, d. h. $AA^T = A^TA$, so gilt $AA^+ = A^+A$*

Beweis:
Übung.

□

Mit der Pseudoinversen A^+ haben wir eine Verallgemeinerung der inversen Matrix A^{-1} (für quadratische Matrizen) gefunden, die auch eine elegante Darstellung der Kleinste-Quadrate-Lösung mit minimaler Norm darstellt.

5.4.1. Die Singulärwertzerlegung

Exkurs: Rademacher

Definition 5.4.3 (Singulärwertzerlegung):
Sei $A \in \mathbb{R}^{m\times n}$. Dann nennt man ein Matrixprodukt der Gestalt

$$A = U\Sigma V^T$$

*wobei $U \in \mathbb{R}^{m\times m}$ und $V \in \mathbb{R}^{n\times n}$ orthogonale Matrizen sind und $\Sigma = (s_{i,j}) \in \mathbb{R}^{m\times n}$ mit $s_{i,j} = 0\ \forall i \neq j$ und $s_{1,1} \geq s_{2,2} \geq \cdots \geq 0$ **Singulärwertzerlegung** von A. Die positiven Diagonaleinträge $\sigma_i := s_{i,i},\ i = 1, 2, \ldots$ von Σ heißen **Singulärwerte** von A.*

Algorithmus 5.4.1 Bestimmung der Singulärwertzerlegung

1: - Bestimme ein Orthonormalsystem von Eigenvektoren $v_1, \ldots, v_n$ zur Matrix $A^\mathsf{T}A$
   - $A^\mathsf{T}A$ berechnen
   - Die Eigenwerte λ_i zu $A^\mathsf{T}A$ berechnen
   - Eigenvektor $\tilde{v}_i$ zu λ_i berechnen
   - Normierung des Eigenvektors: $v_i = \|\tilde{v}_i\|^{-1}\tilde{v}_i$
2: - Setze $\sigma_i = \sqrt{\lambda_i}$
3: - Setze $u_i = \sigma_i^{-1}Av_i \; \forall i$ mit $\sigma_i \neq 0$
4: - Ergänze u_i zu einer Orthonormalbasis des $\mathbb{R}^n$ mit $u_j \in \ker(AA^\mathsf{T})$

Im Anhang befindet sich eine MATLAB-Implementierung

Mit Hilfe der Singulärwertzerlegung lässt sich auch die Pseudoinverse berechnen.
Es sei $A = U\Sigma V^\mathsf{T}$ die Singulärwertzerlegung der Matrix $A \in \mathbb{R}^{m\times n}$ und es sei $r = \operatorname{rang}(A)$.
Wir definieren die Matrix

$$\Sigma^+ = \operatorname{diag}(\sigma_1^{-1}, \ldots, \sigma_r^{-1}, 0, \ldots, 0) \in \mathbb{R}^{n\times m}$$

Damit lässt sich dann die Pseudoinverse A^+ berechnen: $A^+ = V\Sigma^+U^\mathsf{T}$.

5.5. Lösen der Normalgleichung

14.05.2012
12. Vorlesung

$$\begin{pmatrix} 1 & 1 \\ \varepsilon & 0 \\ 0 & \varepsilon \end{pmatrix} \in \mathbb{R}^{3\times 2}, \quad \varepsilon > 0, \quad b \in \mathbb{R}^3 \text{ beliebig}$$

Ziel: Finde die Kleinste-Quadrate-Lösung von $Ax = b$
Löse $A^\mathsf{T}Ax = A^\mathsf{T}b$

Da $\varepsilon > 0$ gilt $\operatorname{rang}(A) = 2$

$$A^\mathsf{T}A = \begin{pmatrix} 1+\varepsilon^2 & 1 \\ 1 & 1+\varepsilon^2 \end{pmatrix} \Rightarrow A^\mathsf{T}A \text{ symmetrisch positiv definit}$$

Angenommen $\varepsilon = 10^{-4}$, dann $\varepsilon^2 = 10^{-8}$ und bei Rechnungen mit einfacher Genauigkeit ($eps \approx 1.19 \cdot 10^{-7}$) erhalten wir:

$$\begin{aligned} A^\mathsf{T}A &= \begin{pmatrix} \operatorname{rd}(1+\varepsilon^2) & 1 \\ 1 & \operatorname{rd}(1+\varepsilon^2) \end{pmatrix} \\ &= \begin{pmatrix} \operatorname{rd}(1+10^{-8}) & 1 \\ 1 & \operatorname{rd}(1+10^{-8}) \end{pmatrix} \\ &= \begin{pmatrix} 1 & 1 \\ 1 & 1 \end{pmatrix} \end{aligned}$$

$A^\mathsf{T}A$ hat offensichtlich nicht mehr vollen Rang!
$\Rightarrow$ Die Kleinste-Quadrate-Lösung ist nicht mehr eindeutig.

Problem ist die Größe von ε und das Rechnen in Gleitkommaarithmetik bzw. der entsprechenden Maschinengenauigkeit.

Faustregel:

Die Normalgleichungen sind in der Regel eindeutig lösbar, wenn A vollen Rang hat und für die (verallgemeinerte) Konditionszahl $\kappa(A) = \|A\| \cdot \|A^+\|$ gilt:

$$\kappa \ll \frac{1}{\sqrt{eps}}$$

Kleinste-Quadrate-Lösungen werden in der Statistik oft über die Normalgleichung gelöst. Dies ist sinnvoll, solange die Messfehler der Daten größer sind, als die Rundungsfehler. Dann ist der Einfluss der Rundungsfehler wahrscheinlich gering im Vergleich zu den Messfehlern, insbesondere wenn mit doppelter Genauigkeit und IEEE-Arithmetik gerechnet wird. Einfache Genauigkeit ist nicht zu empfehlen.

5.6. Die QR-Zerlegung

In dem Abschnitt betrachten wir einen anderen Zugang, die Kleinste-Quadrate-Lösung von $Ax = b$ zu berechnen.

Hier: $\mathbb{K} = \mathbb{R}, \quad m \geq n$

Definition 5.6.1 (QR-Zerlegung):
Sei $A \in \mathbb{R}^{m\times n}$ eine beliebige Matrix, $R \in \mathbb{R}^{m\times n}$ obere Dreiecksmatrix mit $R^T = \begin{pmatrix} R_1^T & 0 \end{pmatrix}$ und $R_1 \in \mathbb{R}^{n\times n}$. Weiterhin sei $Q \in \mathbb{R}^{m\times m}$ eine Orthogonalmatrix. Wenn

$$A = QR$$

*gilt, so nennt man dieses Matrixprodukt **QR-Zerlegung** von A.*

Bilden Q, R und A eine QR-Zerlegung, so gilt:

$$A = Q_1 R_1$$

wobei $Q = \begin{pmatrix} Q_1 & Q_2 \end{pmatrix}$ mit $Q_1 \in \mathbb{R}^{m\times n}, \quad Q_2 \in \mathbb{R}^{m\times(m-n)}$. Diese Zerlegung bezeichnen wir als reduzierte QR-Zerlegung von A. Da Q Orthogonalmatrix, hat $Q_1 = \begin{pmatrix} q_1 & \cdots & q_n \end{pmatrix}$ orthogonale Spaltenvektoren, d. h. $q_i^\mathsf{T} q_j = \delta_{i,j}$ mit $\delta_{i,j} = \begin{cases} 1, & \text{falls } i = j \\ 0, & \text{sonst} \end{cases}$ (Kronecker-Delta).

Satz 5.6.1:
Sei $A \in \mathbb{R}^{m\times n}$ mit $\operatorname{rang}(A) = n$. Dann existiert eine reduzierte QR-Zerlegung von A.

Beweis (mit Gram-Schmidt-Orthogonalisierung):
Seien $a_1, \ldots, a_n$ die Spaltenvektoren von A, d. h. $A = \begin{pmatrix} a_1 & \ldots & a_n \end{pmatrix}$. Gesucht sind Spaltenvektoren $q_1, \ldots, q_n$.

1. $\langle a_1, \ldots, a_n \rangle = \langle q_1, \ldots, q_n \rangle$
2. $q_i^\mathsf{T} q_j = \delta_{i,j}, \; i, j = 1, \ldots, n$

Definieren wir nun $R := (r_{i,j})_{i,j} \in \mathbb{R}^{n \times n}$, so gilt:

$$\begin{aligned} A &= \begin{pmatrix} a_1 & \dots & a_n \end{pmatrix} \\ &= \begin{pmatrix} q_1 & \dots & q_n \end{pmatrix} \begin{pmatrix} r_{1,1} & r_{1,2} & \dots & r_{1,n} \\ 0 & r_{2,2} & \dots & r_{2,n} \\ \vdots & \ddots & \ddots & \vdots \\ 0 & \dots & 0 & r_{n,n} \end{pmatrix} \\ &= QR \end{aligned}$$

□

5.6.1. Klassisches und modifiziertes Gram-Schmidt-Verfahren

Algorithmus 5.6.1 Klassisches Gram-Schmidt-Verfahren

1: **for** $i \leftarrow 1$ **to** n **do**
2: $\quad \hat{q}_i = a_i$
3: $\quad$ **for** $j \leftarrow 1$ **to** $i-1$ **do**
4: $\quad\quad r_{i,j} = q_j^{\mathsf{T}} \cdot a_i$
5: $\quad\quad \hat{q}_i = \hat{q}_i - r_{i,j} \cdot q_j$
6: $\quad$ **end for**
7: $\quad r_{i,i} = \|\hat{q}_i\|$
8: $\quad q_i = \dfrac{\hat{q}_i}{r_{i,i}}$
9: **end for**

Im Anhang befindet sich eine MATLAB-Implementierung

ACHTUNG:

Das klassische Gram-Schmidt-Verfahren ist **nicht** immer stabil. !

Das Gram-Schmidt-Verfahren lässt sich auch mit Hilfe von Orthogonalprojektionen schreiben. Dabei heißt eine Abbildung $P\colon \mathbb{R}^m \to \mathbb{R}^m$ **Orthogonalprojektion**, wenn

1. $P^2 = P$
2. $P^{\mathsf{T}} = P$

Mit Hilfe einer Orthogonalprojektion lässt sich ein Vektor u in eine orthogonale Summe zweier Vektor u_1, u_2 zerlegen.

$$u = u_1 + u_2, \quad u_1 := Pu, \quad u_2 := (I - P)u, \quad u_1 \perp u_2$$

Sei nun für $i = 1, \dots, n$

$$P_i := q_i q_i^{\mathsf{T}} \in \mathbb{R}^{m \times m}$$

Dann ist P_i eine Orthogonalprojektion, wie man leicht mit Hilfe der Orthonormalität der q_i nachrechnet.
Das Produkt uv^{T} heißt **dyadisches Produkt** oder **Dyade** und hat die Eigenschaft

$$u^{\mathsf{T}} v w = w u^{\mathsf{T}} v, \quad u, v, w \in \mathbb{R}^m$$

Beweis durch Nachrechnen mit $uv^\mathsf{T} = (u_i v_j)_{i,j}, \quad u, v \in \mathbb{R}^n$.

Es folgt sofort, dass sich die Berechnung der $\hat{q}_i$ mit Hilfe der Projektion P_i schreiben lässt als

$$\begin{aligned}\hat{q}_i &= q_i - \sum_{j=1}^{i-1} \underbrace{\underbrace{(a_j^\mathsf{T} a_i)}_{=q_j q_j^\mathsf{T}}}_{=P_j} a_j \\ &= a_i - \sum_{j=1}^{i-1} P_j a_i = \left(I - \sum_{j=1}^{i-1} P_j\right) a_i \end{aligned} \tag{5.1}$$

Das Gram-Schmidtsche-Orthogonalisierungsverfahren ist durch folgende Rekursionsvorschrift definiert:

$\hat{q}_1 := a_1$
$q_1 := \frac{\hat{q}_1}{\|\hat{q}_1\|_2}$
for $i \leftarrow 2$ **to** n **do**
  $\hat{q}_i := a_i - \sum_{j=1}^{i-1} (q_j^\mathsf{T} a_i) q_j$
  $q_i := \frac{\hat{q}_i}{\|\hat{q}_i\|_2}$
end for

$q_i^\mathsf{T} q_j = \delta_{i,j}$ rechnet man sofort nach und es gilt $\langle a_1, \ldots, a_n \rangle = \langle q_1, \ldots, q_n \rangle$.

Wir definieren für $i, j = 1, \ldots, n$: $r_{i,j} = \begin{cases} \|\hat{q}_i\|_2, & i = j \\ q_j^\mathsf{T} a_i, & i < j \\ 0, & i > j \end{cases}$

Dann gilt nach Konstruktion (Auflösen von (5.1) nach a_i):

$$\begin{aligned} a_1 &= r_{1,1} q_1 \\ a_2 &= r_{1,2} q_1 + r_{2,2} q_2 \\ a_3 &= r_{1,3} q_1 + r_{2,3} q_2 + r_{3,3} q_3 \\ &\vdots \\ a_n &= r_{1,n} q_1 + r_{2,n} q_2 + \cdots + r_{n,n} q_n \end{aligned}$$

Man kann leicht durch vollständige Induktion zeigen:

$$(I - P_{i-1})(I - P_{i-2}) \cdot \ldots \cdot (I - P_1) = I - \sum_{j=1}^{i-1} P_j$$

Somit gilt für die $\hat{q}_i$ aus dem Gram-Schmidtsche-Orthogonalisierungsverfahren

$$\begin{aligned} \hat{q}_i &= (I - P_{i-1}) \cdot \ldots \cdot (I - P_1) a_i \\ &= P_{i-1}^\perp \cdot P_{i-2}^\perp \cdot \ldots \cdot P_1^\perp a_i \end{aligned} \tag{5.2}$$

mit $P_j^\perp = (I - P_j)$.

Mathematisch sind (5.1) und (5.2) gleich, (5.2) ist allerdings numerisch stabieler, d. h. Rundungsfehler haben einen geringeren Einfluss auf das Ergebnis.

21.05.2012
13. Vorlesung

Um $\hat{q}_i$ zu berechnen geht man wie folgt vor:

$$
\begin{aligned}
\hat{q}_i^{(1)} &:= a_i \\
\hat{q}_i^{(2)} &:= P_1^{\perp}\hat{q}_i^{(1)} && = \hat{q}_i^{(1)} - q_1 q_1^{\mathsf{T}}\hat{q}_i^{(1)} && \overset{u^{\mathsf{T}}vw = wu^{\mathsf{T}}v}{=} && \hat{q}_i^{(1)} - q_1^{\mathsf{T}}\hat{q}_i^{(1)} q_1 \\
\hat{q}_i^{(3)} &:= P_2^{\perp}\hat{q}_i^{(2)} && = \hat{q}_i^{(2)} - q_2 q_2^{\mathsf{T}}\hat{q}_i^{(2)} && = && \hat{q}_i^{(2)} - q_2^{\mathsf{T}}\hat{q}_i^{(2)} q_2 \\
&\vdots && \vdots && \vdots \\
\hat{q}_i = \hat{q}_i^{(i)} &:= P_{i-1}^{\perp}\hat{q}_i^{(i-1)} && = \hat{q}_i^{(i-1)} - q_{i-1} q_{i-1}^{\mathsf{T}}\hat{q}_i^{(i-1)} && = && \hat{q}_i^{(i-1)} - q_{i-1}^{\mathsf{T}}\hat{q}_i^{(i-1)} q_{i-1}
\end{aligned}
$$

Algorithmus 5.6.2 Modifiziertes Gram-Schmidt-Verfahren

1: **for** $i \leftarrow 1$ **to** n **do**
2: $\quad \hat{q}_i = a_i$
3: **end for**
4: **for** $i \leftarrow 1$ **to** n **do**
5: $\quad r_{i,i} = \|\hat{q}_i\|_2$
6: $\quad q_i = \dfrac{\hat{q}_i}{r_{i,i}}$
7: $\quad$ **for** $j \leftarrow i+1$ **to** n **do**
8: $\quad\quad r_{i,j} = q_i^{\mathsf{T}} q_j$
9: $\quad\quad \hat{q}_j = \hat{q}_j - r_{i,j} q_i$
10: $\quad$ **end for**
11: **end for**

Im Anhang befindet sich eine MATLAB-Implementierung

In den Übungen wird ein Beispiel behandelt, aus dem man sieht, dass das modifizierte Gram-Schmidt-Verfahren wesentlich weniger anfällig gegenüber Rundungsfehlern ist. Der Einfluss von Rundungsfehlern kann beim klassischen Gram-Schmidt-Verfahren dazu führen, dass die Spalten der Matrix Q nicht mehr orthogonal sind. [Für mehr Details siehe 6]

Berechnung der Kleinste-Quadrate-Lösung mit der QR-Zerlegung

$Ax = b,\ A \in \mathbb{R}^{n\times n}$ invertierbar, $b \in \mathbb{R}^n,\ x \in \mathbb{R}^n$.
Gegeben sei eine QR-Zerlegung $A = QR$:

$$
\begin{aligned}
Ax = b &\Leftrightarrow QRx = b \\
&\Leftrightarrow Rx = Q^{\mathsf{T}}b \quad (R = \text{obere Dreiecksmatrix})
\end{aligned}
$$

Die rechte Seite $Q^{\mathsf{T}}b$ lässt sich einfach berechnen (Matrix-Vektor-Produkt). Das so entstandene lineare Gleichungssystem $Rx = Q^{\mathsf{T}}b$ lässt sich durch Rückwärtssubstitution einfach berechnen.

Es bietet sich folgende Vorgehensweise zum lösen des linearen Gleichungssystems an:

1. Berechne die reduzierte QR-Zerlegung $A = QR$
2. Berechne $y = Q^{\mathsf{T}}b$
3. Löse durch rückwärtseinsetzen $Rx = y$ nach x auf

Dies ist ein gutes Verfahren, aber die Gauß-Elimination (siehe Abschnitt 7.2) ist im Allgemeinen schneller (im Sinne der Anzahl arithmetischer Operationen).

Zurück zur Berechnung der Kleinste-Quadrate-Lösung:
Um den Einfluss von Rundungsfehlern (Verlust der Orthogonalität von Q) zu verringern, berechnen wir nicht $x = R^{-1}Q^{\mathsf{T}}b$, sondern wenden das modifizierte Gram-Schmidt-Verfahren auf das erweiterte System $\begin{pmatrix} A & b \end{pmatrix}$ an und erhalten:

$$\begin{pmatrix} A & b \end{pmatrix} = \begin{pmatrix} Q & q_{n+1} \end{pmatrix} \begin{pmatrix} R & y \\ 0 & \eta \end{pmatrix}$$

mit $q_{n+1} \in \mathbb{R}^m,\ y \in \mathbb{R}^n,\ \eta \in \mathbb{R},\ A \in \mathbb{R}^{m \times n}$.

$$\begin{aligned} \Rightarrow Ax - b &= \begin{pmatrix} A & b \end{pmatrix} \begin{pmatrix} x \\ -1 \end{pmatrix} \\ &= \begin{pmatrix} Q & q_{n+1} \end{pmatrix} \begin{pmatrix} R & y \\ 0 & \eta \end{pmatrix} \begin{pmatrix} x \\ -1 \end{pmatrix} \\ &= \begin{pmatrix} Q & q_{n+1} \end{pmatrix} \begin{pmatrix} Rx - y \\ -\eta \end{pmatrix} \\ &= Q(Rx - y) - \eta q_{n+1} \end{aligned}$$

Da nach Konstruktion q_{n+1} orthogonal zu den Spalten von Q ist, folgt:

$$\|Ax - b\|_2^2 = \|Q(Rx - y)\|_2^2 + \eta^2 \|q_{n+1}\|_2^2 \overset{\substack{Q^{\mathsf{T}}Q = I \\ q_{n+1}^{\mathsf{T}} q_{n+1} = 1}}{=} \|Rx - y\|_2^2 + \eta^2$$

Somit erhalten wir die Kleinste-Quadrate-Lösung aus

$$Rx = y \text{ (Lösen durch rückwärtseinsetzen)}$$

5.6.2. Householder-Transformationen

Ein anderer Ansatz, die QR-Zerlegung einer Matrix $A \in \mathbb{R}^{m \times n}$ zu berechnen, besteht darin, die Matrix A sukzessive (durch orthogonale Transformationen) auf obere Dreiecksgestalt zu bringen. Dazu soll schrittweise jeweils die Spalte unterhalb des Diagonalelements auf null transformiert werden. Nach n Schritten soll das Ziel erreicht sein.

Fragen:

1. Geht das?
2. Wie sehen die Transformationen aus?

Zur Konstruktion der orthogonal Matrizen verwenden wir sogenannte **Householder-Transformationen**. Eine Householder-Transformation P ist von der Form:

$$P := I - \frac{2}{v^{\mathsf{T}}v} v v^{\mathsf{T}}, \quad 0 \neq v \in \mathbb{R}^n$$

Eine Householder-Transformation P hat folgende Eigenschaften:

1. $PP^{\mathsf{T}} = P^{\mathsf{T}}P = I$
2. $P^{\mathsf{T}} = P$
3. $P^2 = I$

$\Rightarrow P$ ist symmetrische Orthogonalmatrix.
Mit den Eigenschaften des dyadischen Produkts ergibt sich:

$$Px = x - 2\left(\frac{v^\mathsf{T}x}{v^\mathsf{T}v}\right)\cdot v, \quad x \in \mathbb{R}^m$$

Um das Matrix-Vektor-Produkt Px zu berechnen, muss die Householdermatrix P nicht explizit aufgestellt werden.
Householdermatrizen sind geeignet, um Nullen unterhalb der Diagonalen einer Matrix einzuführen. Dazu betrachten wir folgendes Problem:
Zu gegebenen Vektoren $x, y \in \mathbb{R}^m$ ist eine Householdermatrix P zu finden, so dass

$$Px = y \tag{5.3}$$

P orthogonal $\Rightarrow \|x\|_2 = \|Px\|_2 = \|y\|_2$.
x und y müssen also dieselbe Länge haben:

$$(5.3) \Leftrightarrow x - \frac{2v^\mathsf{T}x}{v^\mathsf{T}v}\cdot v = y$$

und einem geeigneten Vektor $v \in \mathbb{R}^m$. Durch nachrechnen sehen wir, dass

$$v := x - y$$

diese Gleichung erfüllt.
Für die QR-Zerlegung wählen wir spezielle Vektoren y, nämlich $y = \sigma e_1$ mit $\sigma := \pm\|x\|_2$ und $e_1 \in \mathbb{R}^m$ sei der erste Einheitsvektor $\begin{pmatrix}1 & 0 & \dots & 0\end{pmatrix}^\mathsf{T}$. Dann gilt:

$$v = x - y = x - \sigma e_1$$

Um Auslöschung zu vermeiden, wählt man für die Implementierung:

$$\sigma = -\operatorname{sign}(x_1)\|x\|_2$$

Auslöschung:

Subtraktion fast gleich großer Zahlen bzw. von Zahlen bei denen die führenden Ziffern gleich sind. Auslöschung ist schlecht konditioniert (siehe Kapitel 6).

Um die QR-Zerlegung von A zu berechnen, sucht man nun orthogonale Matrizen $Q_1, \dots, Q_n \in \mathbb{R}^{m\times m}$, so dass

$$A_j := Q_j Q_{j-1} \dots Q_1 A, \quad j = 1, \dots, n$$

in den ersten j Spalten obere Dreiecksgestalt hat.
Unter oberer Dreiecksgestalt verstehen wir hier, dass die Elemente unterhalb der Diagonalen null sind. Schematisch gehen wir wie folgt vor:

$$A = \begin{pmatrix}* & * & *\\ * & * & *\\ * & * & *\\ * & * & *\end{pmatrix} \xrightarrow{Q_1} \underbrace{\begin{pmatrix}* & * & *\\ 0 & * & *\\ 0 & * & *\\ 0 & * & *\end{pmatrix}}_{A_1} \xrightarrow{Q_2} \underbrace{\begin{pmatrix}* & * & *\\ 0 & * & *\\ 0 & 0 & *\\ 0 & 0 & *\end{pmatrix}}_{A_2} \xrightarrow{Q_3} \underbrace{\begin{pmatrix}* & * & *\\ 0 & * & *\\ 0 & 0 & *\\ 0 & 0 & 0\end{pmatrix}}_{A_3} = R$$

Die Matrix A_j hat also die Form

$$A_j := \begin{pmatrix} R_{j-1} & Z_j & M_j \\ 0 & x_j & N_j \end{pmatrix}$$

wobei $R_j \in \mathbb{R}^{(j-1)\times(j-1)}$, $x_j \in \mathbb{R}^{m-j+1}$.
Nun wählen wir die Householdermatrix P_j, so dass

$$P_j x_j = \sigma e_1$$

und definieren $Q_j := \begin{pmatrix} I_{j-1} & 0 \\ 0 & P_j \end{pmatrix}$. Offensichtlich gilt $A_{j+1} = Q_j A_j$.
A_{j+1} hat dann die gewünschte Eigenschaft, dass die ersten $j+1$ Spalten eine Matrix von oberer Dreiecksgestalt bilden. Definieren wir $Q := Q_1 Q_2 \dots Q_n$, so gilt:

$$R := Q_n Q_{n-1} \dots Q_1 A = Q^\mathsf{T} A$$

wobei R obere Dreiecksgestalt hat und Q orthogonal ist.

Algorithmus 5.6.3 QR-Zerlegung mit Householder-Transformationen

```
function [Q,R] = QR_Householder(A)
[m,n] = size(A); % Bestimmung der Dimension von A
% Test ob m groesser oder gleich n, sonst Fehlermeldung
if m < n
 error('m muss groesser als n sein!');
end
% Initialisierung von R tilde und Q tilde
Rtilde = A;
Qtilde = eye(m,m);
% Schleife ueber alle Spalten von A, dabei sukzessive Erzeugung
   der oberen Dreiecksstruktur
for i = 1:n
 % Kopie der ersten Spalte der Untermatrix im i-ten Schritt
 a = zeros(m,1);
 a(i:m) = Rtilde(i:m,i);
 % Bestimmung von alpha
 alpha = sign(a(i))*norm(a,2);
 % Setze w = a + alpha*e_i
 w = a;
 w(i) = w(i) + alpha;
 % Normierung
 no = norm(w,2);
 w = w/no;
 % Berechnung der Spiegelungsmatrix
 S = eye(m)-2*w*(w');
 % Update von Q tilde und R tilde mittels S
 Qtilde = Qtilde*S;
 Rtilde = S*Rtilde;
end
% Reduktion von Q tilde auf Q und R tilde auf R
Q = zeros(m,n);
Q(1:m,1:n) = Qtilde(1:m,1:n);
R = zeros(n,n);
R(1:n,1:n) = Rtilde(1:n,1:n);
end
```

5.6.3. Givens-Rotationen

Exkurs: Rademacher

Eine weitere Alternative zur Berechnung der QR-Zerlegung einer Matrix $A \in \mathbb{R}^{m\times n}$ ist, neben Projektionen (Abschnitt 5.6.1) und Spiegelungen (Abschnitt 5.6.2), eine Methode die auf Rotationen basiert.

Definition 5.6.2 (Givens-Rotation):
Unter einer ***Givens-Rotation*** *im $\mathbb{R}^n$ versteht man eine Drehung in der durch zwei Einheitsvektoren e_i und e_k aufgespannten Ebene. Die Transformationsmatrix ist gegeben durch:*

$$G_{i,k} = \begin{pmatrix} 1 & 0 & \cdots & 0 & \cdots & \cdots & \cdots & 0 & \cdots & \cdots & 0 \\ 0 & \ddots & & \vdots & & & & \vdots & & & \vdots \\ \vdots & & 1 & 0 & 0 & & 0 & 0 & 0 & & \vdots \\ 0 & \cdots & 0 & c & 0 & \cdots & 0 & s & 0 & \cdots & 0 \\ \vdots & & 0 & 0 & 1 & & 0 & 0 & 0 & & \vdots \\ \vdots & & & \vdots & & \ddots & & \vdots & & & \vdots \\ \vdots & & 0 & 0 & 0 & & 1 & 0 & 0 & & \vdots \\ 0 & \cdots & 0 & -s & 0 & \cdots & 0 & c & 0 & \cdots & 0 \\ \vdots & & 0 & 0 & 0 & & 0 & 0 & 1 & & \vdots \\ \vdots & & & \vdots & & & & \vdots & & \ddots & 0 \\ 0 & \cdots & \cdots & 0 & \cdots & \cdots & \cdots & 0 & \cdots & 0 & 1 \\ & & & \uparrow & & & & \uparrow & & & \\ & & & i\text{-te Spalte} & & & & k\text{-te Spalte} & & & \end{pmatrix} \in \mathbb{R}^{n\times n}$$

wobei $c, s \in \mathbb{R}$ und $c^2 + s^2 = 1$.

Die QR-Zerlegung auf der Basis von Givens-Rotationen transformiert die Matrix A schrittweise in eine obere rechte Dreiecksmatrix R. Durch Anwenden einer Givens-Rotation kann jedoch nur ein einzelnes Unterdiagonalelement eliminiert werden und nicht eine ganze Spalte. Schematisch gehen wir wie folgt vor:

$$A = \begin{pmatrix} * & * & * \\ * & * & * \\ * & * & * \\ * & * & * \end{pmatrix} \xrightarrow{G_{1,2}} \underbrace{\begin{pmatrix} * & * & * \\ 0 & * & * \\ * & * & * \\ * & * & * \end{pmatrix}}_{A_1} \xrightarrow{G_{1,3}} \underbrace{\begin{pmatrix} * & * & * \\ 0 & * & * \\ 0 & * & * \\ * & * & * \end{pmatrix}}_{A_2} \xrightarrow{G_{1,4}} \underbrace{\begin{pmatrix} * & * & * \\ 0 & * & * \\ 0 & * & * \\ 0 & * & * \end{pmatrix}}_{A_3}$$

$$\xrightarrow{G_{2,3}} \underbrace{\begin{pmatrix} * & * & * \\ 0 & * & * \\ 0 & 0 & * \\ 0 & * & * \end{pmatrix}}_{A_4} \xrightarrow{G_{2,4}} \underbrace{\begin{pmatrix} * & * & * \\ 0 & * & * \\ 0 & 0 & * \\ 0 & 0 & * \end{pmatrix}}_{A_5} \xrightarrow{G_{3,4}} \underbrace{\begin{pmatrix} * & * & * \\ 0 & * & * \\ 0 & 0 & * \\ 0 & 0 & 0 \end{pmatrix}}_{A_6} = R$$

Um die Werte c und s zu bestimmen machen wir folgenden Ansatz:

$$\begin{aligned} & s \cdot a_{i,i} + c \cdot a_{j,i} = 0 \quad \text{und} \quad c^2 + s^2 = 1 \\ \Rightarrow \quad & c := \frac{a_{i,i}}{\sqrt{a_{i,i}^2 + a_{j,i}^2}}, \quad s := -\frac{a_{i,i}}{\sqrt{a_{i,i}^2 + a_{j,i}^2}} \end{aligned}$$

Bei dieser Wahl von c und s wird das Element $a_{j,i}$ eliminiert. Zur Elimination der i-ten Spalte (unterhalb der Diagonale) sind $n-i-1$ Givens-Rotationen notwendig. Hieraus lässt sich leicht abschätzen, dass der Aufwand zum Erstellen der QR-Zerlegung nach Givens größer ist als der Aufwand beim Verwenden der Householder-Transformationen. Genaue Analyse und effiziente Durchführung führt zu $\frac{4n^3}{3} + \mathcal{O}(n^2)$ arithmetische Operationen, also den doppelten Aufwand verglichen mit der Householder-Methode.

Die QR-Zerlegung nach Givens ist aber sehr stabil und eine Pivotisierung ist nicht erforderlich. Das Verfahren gewinnt auch an Bedeutung, wenn die Matrix A bereits dünn besetzt ist. Nur Unterdiagonalelemente $a_{j,i} \neq 0$ müssen gezielt eliminiert werden. Bei sogenannten Hessenberg-Matrizen (das sind rechte obere Dreiecksmatrizen, die zusätzlich noch eine linke Nebendiagonale besitzen) kann die QR-Zerlegung mit Givens-Rotationen in nur $\mathcal{O}(n^2)$ Operationen durchgeführt werden. Die QR-Zerlegung von Hessenberg-Matrizen spielt eine entscheidende Rolle bei dem wichtigsten Verfahren zur Berechnung von Eigenwerten einer Matrix.

Algorithmus 5.6.4 QR-Zerlegung mit Givens-Rotationen

1: **for** $i \leftarrow 1$ **to** n **do**
2: **for** $j \leftarrow i+1$ **to** m **do**
3: **if** $A_{j,i} \neq 0$ **then**
4: $G = \text{GIVENS}(m, i, j, A_{i,i}, A_{j,i})$
5: $Q = QG^{\mathsf{T}}$
6: $A = GA$
7: **end if**
8: **end for**
9: **end for**
10: $R = A$

Die Routine $G = \text{GIVENS}(n, i, k, x, y)$ erzeugt dabei die Givens-Matrix der Dimension n, welche an den Zeilen und Spalten mit den Indizes i und k die Werte $c = \frac{x}{\rho}$ und $s = \frac{y}{\rho}$ setzt, wobei $\rho = \text{sign}(x)\sqrt{x^2 + y^2}$.

Im Anhang befindet sich eine MATLAB-Implementierung

---

6. Kondition und Stabilität

24.05.2012
14. Vorlesung

6.1. Einführung

Die Kondition eines mathematischen Problems beschreibt die Abhängigkeit der Lösung dieses Problems von Störungen, während die Stabilität eine Eigenschaft von Algorithmen ist und deren Abhängigkeit von Störungen beschreibt.

6.2. Kondition

Gegeben sei das abstrakte Problem

$$f\colon X \to Y$$

X, Y normierte Vektorräume, X enthält Daten, Y enthält Lösungen.
Im Allgemeinen ist f nicht linear, aber oft stetig. Man betrachtet das Verhalten von f in einem Datenpunkt $x \in X$.
Ein **gut konditioniertes Problem** liegt vor, wenn kleine Änderungen (Störungen), der Daten nur kleine Änderungen in $f(x)$ hervorrufen. Andererseits heißt ein Problem **schlecht konditioniert**, wenn kleine Änderungen in $x \in X$ zu großen Änderungen in $f(x)$ führen.

Sei δ_x eine kleine Störung der Daten in x und $\delta_f = f(x + \delta_x) - f(x)$. Sei $D \subset X$ eine Umgebung von Null, die alle zulässigen Störungen von x enthält; zulässig ist eine Störung, wenn $f(x + \delta_x) \in Y$.

Definition 6.2.1 (Konditionszahl):

1. *Die **absolute Konditionszahl** κ_{abs} von $f(x)$ ist definiert als*

$$\kappa_{abs} := \kappa_{abs}(x) := \sup_{\delta_x \in D} \frac{\|\delta_f\|}{\|\delta_x\|} = \sup_{\delta_x \in D} \frac{\|f(x - \delta_x) - f(x)\|}{\|\delta_x\|}$$

2. *Die **relative Konditionszahl** κ von $f(x)$ ist definiert als*

$$\kappa := \kappa(x) := \sup_{\delta_x \in D} \frac{\|\delta_f\| \div \|f(x)\|}{\|\delta_x\| \div \|x\|} = \sup_{\delta_x \in D} \frac{\|f(x + \delta_x) - f(x)\|\|x\|}{\|f(x)\|\|\delta_x\|}$$

Annahme: f zweimal stetig differenzierbar.

$\overset{\text{Taylorentwicklung}}{\Rightarrow}$ $\delta_f \approx J(x)\delta_x$, wobei $J(x)$ die Jacobimatrix von $f(x)$ sei.
Daraus folgt:

1. $\kappa_{\text{abs}}(x) \approx \|J(x)\|$
2. $\kappa \approx \frac{\|J(x)\|\|x\|}{\|f(x)\|}$

Die relative Konditionszahl kann zur Charakterisierung der Kondition benutzt werden.

κ klein, z. B. $1, 10^{-1}, 10^{-2}$ $\Rightarrow$ gut konditioniert
κ groß, z. B. $10^{5}, 10^{10}, 10^{16}$ $\Rightarrow$ schlecht konditioniert

Beispiel 6.2.1:
Zu gegebenem $x \in \mathbb{C}$ soll $\frac{x}{2}$ berechnet werden:
$f\colon \mathbb{C} \to \mathbb{C},\ x \mapsto \frac{x}{2}$
$J(x) = \frac{1}{2},\ \|x\| = \sqrt{\overline{x}^{\mathsf{T}} x}$
$\Rightarrow \kappa \approx \frac{\|J(x)\|\|x\|}{\|f(x)\|} = \frac{\frac{1}{2}\cancel{|x|}}{\frac{\cancel{|x|}}{2}} = 1$
$\Rightarrow$ Das Problem ist gut konditioniert

Beispiel 6.2.2:
$x \in \mathbb{R},\ x > 0,\ f(x) := \sqrt{x}$
$J(x) = \frac{1}{2\sqrt{x}}$
$\kappa \approx \frac{\|J(x)\|\|x\|}{\|f(x)\|} = \frac{\frac{1}{2}\frac{1}{|\sqrt{x}|}|x|}{|\sqrt{x}|} = \frac{1}{2}$
$\Rightarrow$ Quadratwurzelziehen ist gut konditioniert

Beispiel 6.2.3:
$x, y \in \mathbb{C}$ gegeben, $f\colon \mathbb{C}^2 \to \mathbb{C},\ (x, y) \mapsto x - y$
$(\mathbb{C}^2, \|\cdot\|_\infty),\ (\mathbb{C}, |\cdot|)$
$J(x,y) = \begin{pmatrix} \frac{\partial f(x,y)}{\partial x} & \frac{\partial f(x,y)}{\partial y} \end{pmatrix} = \begin{pmatrix} 1 & -1 \end{pmatrix} \Rightarrow \|J(x,y)\|_\infty = 2$
Für die relative Konditionszahl ergibt sich:

$$\kappa \approx \frac{\|J(x,y)\|\|x,y\|}{\|f(x,y)\|} = \frac{2\max\{|x|,|y|\}}{|x-y|}$$

Die Konditionszahl ist also groß, wenn $|x - y| \approx 0$, also wenn x und y etwa gleich groß sind bzw. viele gleiche führende Ziffern haben. Subtraktion zweier gleich großer Zahlen (**Auslöschung**) ist somit ein **schlecht konditioniertes Problem**.

Kondition von Matrix-Vektor-Multiplikationen

$A \in \mathbb{C}^{m\times n}$ beliebig aber fest.

$$\begin{aligned} f\colon \mathbb{C}^n &\to \mathbb{C}^m \\ x &\mapsto Ax \end{aligned}$$

Mit der Definition 6.2.1 2. der relative Konditionszahl folgt:

$$\kappa(x) = \sup_{\delta_x \in D} \frac{\|A(x+\delta_x - Ax\|\|x\|}{\|Ax\|\|\delta_x\|} = \sup_{\delta_x \in D} \left(\frac{\|A\delta_x\|}{\|\delta_x\|}\right) \frac{\|x\|}{\|Ax\|} \leq \|A\| \cdot \frac{\|x\|}{\|Ax\|}$$

Sei nun zunächst A quadratisch und invertierbar, dann:

$$\|A^{-1}\| = \sup_{y\in\mathbb{C}^n} \frac{\|A^{-1}y\|}{\|y\|} = \sup_{z\in\mathbb{C}^n} \frac{\|z\|}{\|Az\|} \geq \frac{\|x\|}{\|Ax\|} \Rightarrow \kappa(x) \leq \|A\| \cdot \|A^{-1}\|$$

Ist A nicht mehr quadratisch, hat aber vollen Rang, dann

$$\kappa(x) \leq \|A\| \cdot \|A^{+}\|$$

Ist A quadratisch und invertierbar, so gelten diese Aussagen auch für $A^{-1}b$.

Daraus folgt Satz 6.2.1.

Satz 6.2.1:
Gegeben sei eine invertierbare Matrix $A \in \mathbb{C}^{n\times n}$ und ein Vektor $x \in \mathbb{C}^n$. Für die relative Konditionszahl des Matrix-Vektor-Produkts $Ax = b$ gilt bezüglich kleiner Störungen von x

$$\kappa(x) \leq \|A\| \frac{\|x\|}{\|b\|} \leq \|A\| \cdot \|A^{-1}\|$$

Analog gilt für das Problem, zu einem gegebenen Vektor $b \in \mathbb{C}^n$, das Matrix-Vektor-Produkt $x = A^{-1}b$ zu berechnen, die Abschätzung der relativen Konditionszahl bezüglich kleiner Störungen von b

$$\kappa(b) \leq \|A^{-1}\| \frac{\|b\|}{\|x\|} \leq \|A\| \cdot \|A^{-1}\|$$

Definition 6.2.2 (Konditionszahl einer Matrix):

1. *Für eine invertierbare Matrix $A \in \mathbb{K}^{n\times n}$ definiert man die* ***Konditionszahl*** *als*

$$\kappa := \kappa(A) := \|A\| \cdot \|A^{-1}\|$$

2. *Für eine Matrix $A \in \mathbb{K}^{m\times n}$ definiert man die* ***verallgemeinerte Konditionszahl*** *als*

$$\kappa(A) := \|A\| \cdot \|A^{+}\|$$

Die Kondition des Matrix-Vektor-Produkts hängt von der Kondition der Matrix A ab. Ist sie groß, so ist das Problem schlecht konditioniert.
Welchen Einfluss hat die Störung von A, x und b auf die Lösung von $Ax = b$?

Satz 6.2.2:
Sei $A \in \mathbb{K}^{n\times n}$ invertierbar und sei $\delta_A \in \mathbb{K}^{n\times n}$ eine Matrix, so dass $\|A^{-1}\|\|\delta_A\| < 1$.

1. *Dann ist $(A + \delta_A)$ invertierbar und es gilt*

$$\|(A + \delta_A)^{-1}\| \leq \frac{\|A^{-1}\|}{1 - \|A^{-1}\|\|\delta_A\|}$$

2. *Es seien $b \in \mathbb{K}^n \setminus \{0\}$, $\delta_b \in \mathbb{K}^n$ und $x, \delta_x \in \mathbb{K}^n$ seien Lösungen von $Ax = b$ bzw. $(A + \delta_A)(x + \delta_x) = (b + \delta_b)$, dann gilt:*

$$\frac{\|\delta_x\|}{\|x\|} \leq \frac{\kappa(A)}{1 - \kappa(A)\frac{\|\delta_A\|}{\|A\|}} \left(\frac{\|\delta_A\|}{\|A\|} + \frac{\|\delta_b\|}{\|b\|} \right)$$

Beweis:

1. Sei $y \in \mathbb{K}^n,\ y \neq 0$

$$\begin{aligned} \Rightarrow \left\|(I + A^{-1}\delta_A)y\right\| &\geq \|y\| - \|A^{-1}\delta_A y\| \\ &\geq \|y\| - \|A^{-1}\delta_A\|\|y\| \\ &= (1 - \|A^{-1}\delta_A\|)\|y\| \\ &\geq \underbrace{(1 - \|A^{-1}\|\|\delta_A\|)}_{>0 \text{ nach Voraussetzung}}\|y\| \end{aligned} \tag{6.1}$$

$\Rightarrow I + A^{-1}\delta_A$ ist invertierbar
Weiterhin gilt: $A + \delta_A = A(I + A^{-1}\delta_A) \Rightarrow A + \delta_A$ invertierbar
Setzen wir $x = (I + A^{-1}\delta_A)y$

$$\begin{aligned} \overset{(6.1)}{\Rightarrow} \|x\| &= \left\|(I + A^{-1}\delta_A)y\right\| \\ &\overset{(6.1)}{\geq} \left(1 - \|A^{-1}\|\|\delta_A\|\right)\|y\| \\ &= \left(1 - \|A^{-1}\|\|\delta_A\|\right)\|(I + A^{-1}\delta_A)^{-1}x\| \end{aligned}$$

Für beliebiges $x \in \mathbb{K}^n$ gilt:

$$\begin{aligned} &\left\|(I + A^{-1}\delta_A)^{-1}x\right\| \leq \frac{1}{1 - \|A^{-1}\|\|\delta_A\|}\|x\| \\ \Rightarrow\ &\left\|(I + A^{-1}\delta_A)^{-1}\right\| \leq \frac{1}{1 - \|A^{-1}\|\|\delta_A\|} \\ \Rightarrow\ &\left\|(A + \delta_A)^{-1}\right\| = \left\|(I + A^{-1}\delta_A)^{-1}A^{-1}\right\| \leq \frac{\|A^{-1}\|}{1 - \|A^{-1}\|\|\delta_A\|} \end{aligned}$$

2. Wir subtrahieren $Ax = b$ von $(A + \delta_A)(x + \delta_x) = b + \delta_b = b$
   $\Rightarrow (A + \delta_A)\delta_x = \delta_b - \delta_A x$
   Somit gilt: $\delta_x = (A + \delta_A)^{-1}(\delta_b - \delta_A x)$

$$\begin{aligned} \Rightarrow \frac{\|\delta_x\|}{\|x\|} &= \frac{\left\|(A + \delta_A)^{-1}(\delta_b - \delta_A x)\right\|}{\|x\|} \\ &\leq \frac{\left\|(A + \delta_A)^{-1}\right\|\|\delta_b - \delta_A x\|}{\|x\|} \\ &\overset{\text{Satz 6.2.2 1.}}{\leq} \frac{\|A^{-1}\|}{1 - \|A^{-1}\|\|\delta_A\|}\left(\frac{\|\delta_b\|}{\|x\|} + \frac{\|\delta_A x\|}{\|x\|}\right) \\ &= \frac{\|A^{-1}\|\|A\|}{1 - \underbrace{\|A^{-1}\|\|\delta_A\|}_{=\kappa(A)\frac{\|\delta_A\|}{\|A\|}}}\left(\underbrace{\frac{\|\delta_b\|}{\|A\|\|x\|}}_{\leq \frac{\|\delta_b\|}{\|b\|}} + \underbrace{\frac{\|\delta_A x\|}{\|A\|\|x\|}}_{\leq \frac{\|\delta_A\|}{\|A\|}}\right) \\ &\leq \frac{\kappa(A)}{1 - \kappa(A)\frac{\|\delta_A\|}{\|A\|}}\left(\frac{\|\delta_b\|}{\|b\|} + \frac{\|\delta_A\|}{\|A\|}\right) \end{aligned}$$

□

Aus diesem Satz folgt, wie der relative Fehler des Lösungsvektors von den Störungen in A und b abhängt. Wir sehen, dass sich der relative Fehler $\frac{\|\delta_x\|}{\|x\|}$ bei kleinem Anfangsfehler $\frac{\|\delta_A\|}{\|A\|}$ und $\frac{\|\delta_b\|}{\|b\|}$ um den Faktor $\kappa(A)$ verstärkt. Ein Gleichungssystem $Ax = b$ heißt **gut** bzw. **schlecht konditioniert** wenn $\kappa(A)$ klein bzw. groß ist.

6.3. Stabilität

04.06.2012
15. Vorlesung

Um ein abstraktes Problem $f\colon X \to Y$ zu lösen, benötigen wir einen Algorithmus, den wir mit $\tilde{f}\colon X \to Y$ bezeichnen. Als relativen Fehler haben wir

$$\frac{\|\tilde{f}(x) - f(x)\|}{\|f(x)\|}$$

Wir nennen einen Algorithmus **genau**, wenn

$$\frac{\|\tilde{f}(x) - f(x)\|}{\|f(x)\|} = \mathcal{O}(eps)$$

wobei *eps* die Maschinengenauigkeit ist.

$\mathcal{O}$-Notation (Landau-Notation)

$f(x) = \mathcal{O}\big(g(x)\big)$ bedeutet, dass es eine Konstante $c > 0$ gibt, so dass

$$f(x) \leq c \cdot g(x)$$

für $x \to \infty$ oder $x \to 0$.

Beispiel:
$\sin(t^2) = \mathcal{O}(t^2)$ für $t \to 0$.

Im Hinblick auf Rundungsfehler ist Genauigkeit oft schwer zu erfüllen. Stattdessen führt man den Begriff der **Stabilität** an.

Definition 6.3.1 (stabil):
*Ein Algorithmus $\tilde{f}$ für ein Problem f heißt **stabil** für $x \in X$, wenn für ein $\tilde{x} \in X$ mit*

$$\frac{\|\tilde{x} - x\|}{\|x\|} = \mathcal{O}(eps) \text{ gilt: } \frac{\|\tilde{f}(x) - f(\tilde{x})\|}{\|f(\tilde{x})\|} = \mathcal{O}(eps)$$

Ein weiterer wichtiger Stabilitätsbegriff ist der der **Rückwärtsstabilität**.

Definition 6.3.2 (rückwärtsstabil):
*Ein Algorithmus $\tilde{f}$ für ein Problem f heißt **Rückwärtsstabil**, wenn es für jedes $x \in X$ ein $\tilde{x}$ mit*

$$\frac{\|\tilde{x} - x\|}{\|x\|} = \mathcal{O}(eps) \text{ gibt, so dass } \tilde{f}(x) = f(\tilde{x})$$

„Ein stabiler Algorithmus liefert fast die richtige Antwort auf fast die richtige Frage."

„Ein rückwärtsstabiler Algorithmus liefert die exakte Antwort auf fast die richtige Frage."

Zitat aus [6]

Frage: Wie genau ist ein rückwärtsstabiler Algorithmus?

Satz 6.3.1:

$$\frac{\|\tilde{f}(x) - f(x)\|}{\|f(x)\|} = \mathcal{O}\big(\kappa(x) \cdot eps\big)$$

wobei $\kappa(x)$ die Konditionszahl des Problems $f(x)$ ist.

Beweis:
[Siehe 6, Theorem 15.1]

□

Welchen Einfluss hat die Konditionszahl auf die Genauigkeit der berechneten Lösung eines linearen Gleichungssystems $Ax = b$ und zwar unabhängig vom verwendeten Algorithmus?

Sei *eps* die Maschinengenauigkeit in IEEE-Arithmetik (einfach: $1.19 \cdot 10^{-7}$, doppelt: $2.22 \cdot 10^{-16}$). Sei $\tilde{A} = \text{rd}(A)$ und $\tilde{b} = \text{rd}(b)$, wobei $\text{rd}(\cdot)$ die Rundung auf die nächste Maschinenzahl ist. Als Störungen betrachten wir: $\delta_A = A - \tilde{A}$ und $\delta_b = b - \tilde{b}$.

$$\Rightarrow \quad \frac{\|\delta_A\|}{\|A\|} = \frac{\|A - \tilde{A}\|}{\|A\|} \approx eps$$
$$\frac{\|\delta_b\|}{\|b\|} = \frac{\|b - \tilde{b}\|}{\|b\|} \approx eps$$

Um Satz 6.2.2 2. anwenden zu können, benötigen wir

$$1 > \gamma \geq \kappa(A)\frac{\|\delta_A\|}{\|A\|} = \mathcal{O}\big(\kappa(A) \cdot eps\big)$$

Ist dies der Fall, so folgt

$$\begin{aligned}\frac{\|\delta_x\|}{\|x\|} &\leq \frac{\kappa(A)}{1 - \kappa(A) \cdot eps}(c \cdot eps + c \cdot eps) \\ &\leq \frac{c \cdot 2 \cdot eps \cdot \kappa(A)}{1 - \gamma} \\ &= \mathcal{O}\big(eps \cdot \kappa(A)\big)\end{aligned}$$

Diesen Fehler nennt man **unvermeidlich**, da er nicht vom verwendeten Algorithmus zur Lösung des linearen Gleichungssystems $Ax = b$ abhängt.

$$\kappa(A) = \|A\| \cdot \|A^{-1}\| \geq \|AA^{-1}\| = 1$$

Die Konditionszahl ist immer größer als Eins und daher ein Verstärkungsfaktor. Angenommen die Maschinengenauigkeit ist $eps = 10^{-t}$ und die Konditionszahl $\kappa(A) = 10^k$ dann kann die Lösung einen relativen Fehler der Ordnung 10^{k-t} haben:

$$\frac{\|\delta_x\|}{\|x\|} \leq c \cdot 10^{k-t}$$

Faustformel:

Bei $\kappa(A) \approx 10^k$ ist die Lösung auf $k - t - 1$ stellen genau.

7. Direkte Verfahren für lineare Gleichungssysteme

7.1. Einführung

- Geodätische Untersuchungen (vgl. Kapitel 5)
- Stromnetze (Graphen) (vgl. Kapitel 3)

7.2. Das Gaußsche Eliminationsverfahren

Sei $A \in \mathbb{K}^{n\times n}$ eine invertierbare Matrix und $b \in \mathbb{K}^n$ ein gegebener Vektor. Gesucht ist die Lösung $x \in \mathbb{K}^n$ des linearen Gleichungssystems

$$Ax = b \tag{7.1}$$

Das Gaußsche Eliminationsverfahren ist aus Linearer Algebra bekannt. Ziel ist es in diesem Kapitel, sich mit den numerischen Eigenschaften zu beschäftigen.

$$A = (a_{i,j})_{i,j}, \quad b = (b_i)_i, \quad x = (x_i)_i$$

$$(7.1) \Leftrightarrow \begin{array}{ccccccccc} a_{1,1}x_1 & + & a_{1,2}x_2 & + & \dots & + & a_{1,n}x_n & = & b_1 \\ a_{2,1}x_1 & + & a_{2,2}x_2 & + & \dots & + & a_{2,n}x_n & = & b_2 \\ \vdots & & \vdots & & \vdots & & \vdots & & \vdots \\ a_{n,1}x_1 & + & a_{n,2}x_2 & + & \dots & + & a_{n,n}x_n & = & b_n \end{array}$$

A invertierbar, also $\det(A) \neq 0$, also existiert ein $a_{i,1} \neq 0$.
Ohne Einschränkung: $a_{1,1} \neq 0$.
Somit können wir die Variable x_1 aus den Gleichungen $2, \dots, n$ eliminieren, indem wir von dem Zeilen $2, \dots, n$ die erste Zeile multipliziert mit $l_{i,1} = \frac{a_{i,1}}{a_{1,1}}$ subtrahieren. Somit erhalten wir

$$\begin{array}{c|ccccccccc} a_{1,1}x_1 & + & a_{1,2}x_2 & + & \dots & + & a_{1,n}x_n & = & b_1 \\ \hline & + & (a_{2,2} - l_{2,1}a_{1,2})x_2 & + & \dots & + & (a_{2,n} - l_{2,1}a_{1,n})x_n & = & (b_2 - l_{2,1}b_1) \\ & & \vdots & & \vdots & & \vdots & & \vdots \\ & + & (a_{n,2} - l_{n,1}a_{1,2})x_2 & + & \dots & + & (a_{n,n} - l_{n,1}a_{1,n})x_n & = & (b_n - l_{n,1}b_1) \end{array}$$

Somit haben wir noch $n-1$ unbekannte $x_2, \dots, x_n$ und das reduzierte Gleichungssystem

$$\begin{array}{ccccccc} a_{2,2}^{(2)}x_2 & + & \dots & + & a_{2,n}^{(2)}x_n & = & b_2^{(2)} \\ \vdots & & \vdots & & \vdots & & \vdots \\ a_{n,2}^{(2)}x_2 & + & \dots & + & a_{n,n}^{(2)}x_n & = & b_n^{(2)} \end{array}$$

mit $\begin{array}{ll} a_{i,k}^{(2)} = a_{i,k} - l_{i,1}a_{1,k} & \\ b_i^{(2)} = b_i - l_{i,1}b_1 & i,k = 2, \dots, n \\ l_{i,1} = \frac{a_{i,1}}{a_{1,1}} & \end{array}$

Eliminiert man nun x_2 aus den Gleichungen $i = 3, \ldots, n$ und wendet diese Prozedur rekursiv auf das jeweils entstandene kleiner Gleichungssystem an, so erhalten wir ein äquivalentes Gleichungssystem der Form:

$$\begin{array}{ccccccccccccc}
a_{1,1}x_1 & + & a_{1,2}x_2 & + & a_{1,3}x_3 & + & \ldots & + & a_{1,n-1}x_{n-1} & + & a_{1,n}x_n & = & b_1 \\
 & & a^{(2)}_{2,2}x_2 & + & a^{(2)}_{2,3}x_3 & + & \ldots & + & a^{(2)}_{2,n-1}x_{n-1} & + & a^{(2)}_{2,n}x_n & = & b^{(2)}_2 \\
 & & & & a^{(3)}_{3,3}x_3 & + & \ldots & + & a^{(3)}_{3,n-1}x_{n-1} & + & a^{(3)}_{3,n}x_n & = & b^{(3)}_3 \\
 & & & & & & \ddots & & \vdots & & \vdots & & \vdots \\
 & & & & & & & & a^{(n-1)}_{n-1,n-1}x_{n-1} & + & a^{(n-1)}_{n-1,n}x_n & = & b^{(n-1)}_{n-1} \\
 & & & & & & & & & & a^{(n)}_{n,n}x_n & = & b^{(n)}_n
\end{array}$$

mit den Koeffizienten $a^{(k)}_{i,j}$ und b^j_i definiert durch

$$\begin{aligned}
a^{(k+1)}_{i,j} &= a^{(k)}_{i,j} - l_{i,k}a^{(k)}_{k,j} \\
b^{(k+1)}_{i} &= b^{(k)}_{i} - l_{i,k}b^{(k)}_{k} \quad \text{für } k = 1, \ldots, n \text{ und } i, j = k+1, \ldots, n \\
l_{i,k} &= \tfrac{a^{(k)}_{i,k}}{a^{(k)}_{k,k}}
\end{aligned}$$

Das Gaußsche Eliminationsverfahren bringt das Gleichungssystem auf Dreiecksgestalt und $x_n, \ldots, x_1$ lassen sich durch Rückwärtssubstitution berechnen. In Rechnerarithmetik kann folgendes Problem auftreten: $l_{i,k}$ kann sehr groß werden und in $a^{(k+1)}_{i,j} = a^{(k)}_{i,j} - l_{i,k}a^{(k)}_{k,j}$ können signifikante Ziffern verloren gehen.

Beispiel:

$$\begin{pmatrix} \varepsilon & 1 \\ 1 & 1 \end{pmatrix}$$

In exakter Arithmetik: $a^{(2)}_{2,2} = 1 - \frac{1}{\varepsilon}$
In Rechnerarithmetik: $\mathrm{rd}(a^{(2)}_{2,2}) = -\frac{1}{\varepsilon}$, wenn ε sehr klein bzw. $\varepsilon \ll 1$.
Dies ist aber die Lösung des Gleichungssystems mit $a_{2,2} = 0$.

Zur Vermeidung dieses Problems betrachtet man die Strategie der **Spaltenpivotsuche**. Dabei nennt man die **Diagonalelemente**, unterhalb derer die Spalte eliminiert werden soll, **Pivotelemente**. Im k-ten Eliminationsschritt werden also die k-te und m-te Zeile vertauscht, wobei m definiert ist durch

$$|a^{(k)}_{m,k}| := \max_{k \le i \le n} |a^{(k)}_{i,k}|$$

Aus der Invertierbarkeit von A folgt $a^{(k)}_{m,k} \neq 0$ und aus der Maximalität folgt die obere Schranke

$$l_{i,k} < 1, \quad i = k+1, \ldots, n$$

Vollständige Pivotsuche

11.06.2012
16. Vorlesung

Im k-ten Eliminationsschritt werden die Spalten k und s und die Zeilen k und r vertauscht, wenn

$$|a^{(k)}_{r,s}| := \max_{k \le i,j \le n} |a^{(k)}_{i,j}|$$

Die Spaltenpivotsuche benötigt im *worst case* $\mathcal{O}(n^2)$ Vergleiche.
Die vollständige Pivotsuche benötigt im *worst case* $\mathcal{O}(n^3)$ Vergleiche.

Die Spaltenpivotsuche ist in der Regel ausreichend, die vollständige Pivotsuche wird nur in Spezialfällen angewendet.

Algorithmus 7.2.1 Gaußsches Eliminationsverfahren mit Pivotsuche

Voraussetzung:
Es sei bereits ein Programm SWAP(A) vorhanden, welches die Pivotsuche durchführt (Spaltenpivotsuche oder vollständige Pivotsuche).

1. Elimination:

for $k \leftarrow 1$ **to** n **do**
  $A = \text{SWAP}(A)$
  for $i \leftarrow k+1$ **to** n **do**
    $l = \frac{a_{i,k}}{a_{k,k}}$
    for $j \leftarrow k+1$ **to** n **do**
      $a_{i,j} = a_{i,j} - l a_{k,j}$
    end for
  end for
end for

2. Rückwärtssubstitution:

for $k \leftarrow n$ **to** 1 **do**
  $x_k = b_k$
  for $i \leftarrow k+1$ **to** n **do**
    $x_k = x_k - a_{k,i} x_i$
  end for
  $x_k = \frac{x_k}{a_{k,k}}$
end for

Äquilibrierung

Exkurs: Rademacher

Meist führt man vor dem Gauß-Algorithmus eine sogenannte **Äquilibrierung** durch, d. h. man multipliziert mit einer Diagonalmatrix D die alle Zeilensummen der Matrix auf eins transformiert.

$$Ax = b \Leftrightarrow DAx = Db \text{ mit } d_i = \left(\sum_{j=1}^{n} |a_{i,j}| \right)^{-1}$$

Ziel dieser Skalierung ist es, die Konditionszahl des Gleichungssystems zu verringern, was den Einfluss von Störungen der Eingabedaten (z. B. durch Rundungsfehler) auf die Lösung verringert. Äquilibrierung ist eine Möglichkeit der **Vorkonditionierung** linearer Gleichungssysteme, allerdings im Regelfall nicht besonders effektiv, da die durch die Diagonalmatrix gegebene Approximation der Inversen nicht gut ist.
Bei der vollständigen Pivotierung wird vor der Durchführung eine Äquilibrierung, sowohl zeilenweise als auch spaltenweise, vorgenommen.

7.3. Anzahl der Rechenoperationen

Wir fassen eine Multiplikation bzw. Division und eine Addition bzw. Subtraktion zu einer sogenannten **floating point operation (kurz: *flop*)** zusammen.

$$1\ flop = 1 \text{ Multiplikation/Division} + 1 \text{ Addition/Subtraktion}$$

Die Komplexität eines Algorithmus wird häufig durch $flops$ gekennzeichnet. Man bedient sich dabei meist der Landauschen $\mathcal{O}$-Notation. Bezeichnet N_k die Anzahl $flops$, des k-ten Eliminationsschrittes, so gilt:

$$N_k = (n-k)^2 + \mathcal{O}(n-k)$$

In Komplexitätsabschätzungen werden meist nur Terme höchster Ordnung betrachtet. Als Aufwand für die Gauß-Elimination erhalten wir somit:

$$\begin{aligned}\sum_{k=1}^{n-1} N_k &= \sum_{k=1}^{n-1}(n-k)^2 + \mathcal{O}(n-k)\\ &= \frac{n^3}{3} + \mathcal{O}(n^2)\\ &= \mathcal{O}(n^3)\end{aligned}$$

7.4. Die *LR*-Zerlegung

Definition 7.4.1 (*LR*-Zerlegung):
Seien $A, L, R \in \mathbb{K}^{n\times n}$, wobei L eine untere und R eine obere Dreiecksmatrix sei, so dass

$$A = LR$$

*Diese Produktdarstellung heißt **LR-Zerlegung** der Matrix A.*

Aus einer gegebenen LR-Zerlegung $A = LR$ erhält man die Lösung von

$$Ax = b \Leftrightarrow L\underbrace{Rx}_{=y} = b$$

wie folgt:

1. Löse $Ly = b$
2. Löse $Rx = y$

Im ersten Schritt wird durch vorwärtseinsetzen (MATLAB-Implementierung) die selbe rechte Seite aus b erzeugt, wie bei der Gauß-Elimination. Im zweiten Schritt berechnet man hieraus die Lösung durch rückwärtseinsetzten (MATLAB-Implementierung). Man sieht leicht, dass mit $R = (r_{i,k})_{i,k}$ und $L = (l_{i,k})_{i,k}$ gilt:

$$\begin{aligned} y_1 &= \frac{b_1}{l_{1,1}} \\ y_2 &= \frac{b_2 - l_{2,1}y_1}{l_{2,2}} \\ &\vdots \\ y_n &= \frac{b_n - l_{n,1}y_1 - \cdots - l_{n,n-1}y_{n-1}}{l_{n,n}} \end{aligned} \quad \text{und} \quad \begin{aligned} x_n &= \frac{y_n}{r_{n,n}} \\ x_{n-1} &= \frac{y_{n-1} - r_{n-1,n}x_n}{r_{n-1,n-1}} \\ &\vdots \\ x_1 &= \frac{y_1 - r_{1,2}x_2 - \cdots - r_{1,n}x_n}{r_{1,1}} \end{aligned}$$

Jeder dieser Prozesse benötigt $\frac{n^2}{2} + \mathcal{O}(n)$ *flops*.

Existenz und Konstruktion einer LR-Zerlegung

Elementarmatrix $L_j \in \mathbb{K}^{n\times n},\ j \in \{1,\dots,n\}$ ist definiert als

$$L_j = \begin{pmatrix} e_1 & e_2 & \cdots & e_{j-1} & l_j & e_{j+1} & \cdots & e_{n-1} & e_n \end{pmatrix}$$

wobei die $e_i,\ i = 1,\dots,n$ die Einheitsspaltenvektoren sind und $l_j \in \mathbb{K}^n$ ist ein Spaltenvektor der Form

$$l_j = \begin{pmatrix} 0 & 0 & \cdots & \underset{\substack{\uparrow \\ j\text{-ter Eintrag}}}{1} & -l_{j+1,j} & \cdots & -l_{n,j} \end{pmatrix}^{\mathsf{H}}$$

Die Matrix L_j sieht wie folgt aus:

$$L_j = \begin{pmatrix} 1 & 0 & \cdots & \cdots & \cdots & \cdots & \cdots & 0 \\ 0 & 1 & \ddots & & & & & \vdots \\ \vdots & \ddots & \ddots & \ddots & & & & \vdots \\ \vdots & & 0 & 1 & \ddots & & & \vdots \\ \vdots & & \vdots & -l_{j+1,j} & 1 & \ddots & & \vdots \\ \vdots & & \vdots & -l_{j+2,j} & 0 & \ddots & \ddots & \vdots \\ \vdots & & \vdots & \vdots & \vdots & \ddots & 1 & 0 \\ 0 & \cdots & 0 & \underset{\substack{\uparrow \\ j\text{-te Spalte}}}{-l_{n,j}} & 0 & \cdots & 0 & 1 \end{pmatrix}$$

Eigenschaften von Elementarmatrizen

1. Anwendung auf einen Vektor $a = \begin{pmatrix} a_1 & \dots & a_n \end{pmatrix}^{\mathsf{H}}$:

$$L_j \begin{pmatrix} a_1 \\ \vdots \\ a_j \\ a_{j+1} \\ \vdots \\ a_n \end{pmatrix} = \begin{pmatrix} a_1 \\ \vdots \\ a_j \\ a_{j+1} - l_{j+1,j}a_j \\ \vdots \\ a_n - l_{n,j}a_j \end{pmatrix}$$

2. Anwendung auf eine Matrix $A = \begin{pmatrix} a_1 \\ \vdots \\ a_n \end{pmatrix}$, wobei $a_i,\ i = 1, \ldots, n$ die Zeilenvektoren von A sind:

$$L_j \cdot A = L_j \begin{pmatrix} a_1 \\ \vdots \\ a_n \end{pmatrix} = \begin{pmatrix} a_1 \\ \vdots \\ a_j \\ a_{j+1} - l_{j+1,j} a_j \\ \vdots \\ a_n - l_{n,j} a_j \end{pmatrix}$$

Sind die $l_{i,k}$ wie im Gaußschen Eliminationsverfahren gewählt, so entspricht $L_j A$ dem j-ten Eliminationsschritt im Gaußschen Eliminationsverfahren

3. L_j ist invertierbar, da aus 1. sofort folgt:

$$La = 0 \Leftrightarrow a = 0$$

4. Als Inverse der Elementarmatrix L_j haben wir:

$$L_j^{-1} = \begin{pmatrix} 1 & 0 & \cdots & & \cdots & \cdots & \cdots & 0 \\ 0 & 1 & \ddots & & & & & \vdots \\ \vdots & \ddots & \ddots & \ddots & & & & \vdots \\ \vdots & & 0 & 1 & \ddots & & & \vdots \\ \vdots & & \vdots & l_{j+1,j} & 1 & \ddots & & \vdots \\ \vdots & & \vdots & l_{j+2,j} & 0 & \ddots & \ddots & \vdots \\ \vdots & & \vdots & \vdots & \vdots & \ddots & 1 & 0 \\ 0 & \cdots & 0 & l_{n,j} & 0 & \cdots & 0 & 1 \end{pmatrix}$$

($l_{n,j}$: j-te Spalte)

5. Für $j < k$ gilt:

$$L_j L_k = \begin{pmatrix} 1 & 0 & \cdots & & \cdots & \cdots & \cdots & & \cdots & \cdots & \cdots & 0 \\ 0 & 1 & \ddots & & & & & & & & & \vdots \\ \vdots & \ddots & \ddots & \ddots & & & & & & & & \vdots \\ \vdots & & 0 & 1 & \ddots & & & & & & & \vdots \\ \vdots & & \vdots & -l_{j+1,j} & \ddots & \ddots & & & & & & \vdots \\ \vdots & & \vdots & -l_{j+2,j} & 0 & \ddots & \ddots & & & & & \vdots \\ \vdots & & \vdots & \vdots & \vdots & \ddots & \ddots & \ddots & & & & \vdots \\ \vdots & & \vdots & \vdots & \vdots & & 0 & 1 & \ddots & & & \vdots \\ \vdots & & \vdots & \vdots & \vdots & & \vdots & -l_{k+1,k} & \ddots & \ddots & & \vdots \\ \vdots & & \vdots & \vdots & \vdots & & \vdots & -l_{k+2,k} & 0 & \ddots & \ddots & \vdots \\ \vdots & & \vdots & \vdots & \vdots & & \vdots & \vdots & \vdots & \ddots & 1 & 0 \\ 0 & \cdots & 0 & -l_{n,j} & 0 & \cdots & 0 & -l_{n,k} & 0 & \cdots & 0 & 1 \end{pmatrix}$$

($-l_{n,j}$: j-te Spalte; $-l_{n,k}$: k-te Spalte)

Wir benötigen noch Permutationsmatrizen, die uns erlauben, die Zeilen- und Spaltenvertauschungen beim Gaußschen Eliminationsverfahren zu beschreiben.
Sei $\{i_1, \dots, i_n\}$ eine Permutation von $\{1, \dots, n\}$ und es sei e_i der i-te Einheitsvektor. Dann heißt

$$P = \begin{pmatrix} e_{i_1} & \dots & e_{i_n} \end{pmatrix}^{\mathsf{H}}$$

die zur Permutation $\{i_1, \dots, i_n\}$ gehörige **Permutationsmatrix**.

Eigenschaften von Permutationsmatrizen

1. Anwendung auf einen Vektor $a = \begin{pmatrix} a_1 & \dots & a_n \end{pmatrix}^{\mathsf{H}} \in \mathbb{K}^n$

$$Pa = \begin{pmatrix} a_{i_1} \\ \vdots \\ a_{i_n} \end{pmatrix}$$

2. Anwendung auf eine Matrix $A = \begin{pmatrix} a_1 \\ \vdots \\ a_n \end{pmatrix} \in \mathbb{K}^{n \times n}$ mit den Zeilenvektoren a_i, $i = 1, \dots, n$

$$PA = \begin{pmatrix} a_{i_1} & \dots & a_{i_n} \end{pmatrix}^{\mathsf{H}}$$

   Dies führt auf die Vertauschung von Zeilen entsprechend der Permutation

3. Anwendung einer Matrix $A \in \mathbb{K}^{n \times n}$ mit den Spaltenvektoren $a_i,\ i = 1, \dots, n$ auf P^{H}:

$$\begin{aligned} AP^{\mathsf{H}} &= \begin{pmatrix} a_1 & \dots & a_n \end{pmatrix} P^{\mathsf{H}} \\ &\overset{2.}{=} \begin{pmatrix} a_{i_1} & \dots & a_{i_n} \end{pmatrix} \end{aligned}$$

   Dies entspricht der Vertauschung der Spalten entsprechend der Permutation

4. Permutationsmatrizen sind invertierbar: $P^{-1} = P^{\mathsf{H}}$, d. h. Permutationsmatrizen sind unitär (orthogonal für $\mathbb{K} = \mathbb{R}$)

Sei nun P eine Permutationsmatrix, welche nur Zeilen unterhalb der j-ten Zeile vertauscht. In der Permutation $\{i_1, \dots, i_n\}$ von $\{1, \dots, n\}$ gilt für die ersten j Indizes: $i_1 = 1, \dots, i_j = j$. Dann gilt $PL_jP^{\mathsf{H}} = \hat{L}_j$, wobei $\hat{L}_j$ wie L_j aufgebaut ist, nur für $k > j$ gilt $\hat{L}_{k,j} = l_{i_k,j}$. Dies folgt aus der Darstellung:

$$PL_jP^{\mathsf{H}} = P \left(I + \begin{pmatrix} 1 & 0 & \cdots & \cdots & \cdots & \cdots & \cdots & 0 \\ 0 & 1 & \ddots & & & & & \vdots \\ \vdots & \ddots & \ddots & \ddots & & & & \vdots \\ \vdots & & 0 & 1 & \ddots & & & \vdots \\ \vdots & & \vdots & -l_{j+1,j} & 1 & \ddots & & \vdots \\ \vdots & & \vdots & -l_{j+2,j} & 0 & \ddots & \ddots & \vdots \\ \vdots & & \vdots & \vdots & \vdots & \ddots & 1 & 0 \\ 0 & \cdots & 0 & -l_{n,j} & 0 & \cdots & 0 & 1 \\ & & & \uparrow & & & & \\ & & & j\text{-te Spalte} & & & & \end{pmatrix} \right) P^{\mathsf{H}}$$

Hieraus folgt mit 4. sofort die Behauptung. Wir werden nun das Gaußsche Eliminationsverfahren mit Hilfe von Permutationsmatrizen und Elementarmatrizen beschreiben. Dazu seien

- $P_j :=$ Permutationsmatrix, welche die Zeilenvertauschung vor dem j-ten Eliminationsschritt ausführt, z. B. bei der Spaltenpivotsuche. Bei Anwendung von P_j werden also nur die Zeilen $j, \ldots, n$ vertauscht.
- $L_j :=$ Elementarmatrix mit $l_{i,j} := \frac{a_{i,j}^{(j)}}{a_{j,j}^{(j)}}$, $i = j+1, \ldots, n$, wobei $a_{i,j}^{(j)}$ das i,j-te Element der durch vorangegangene Elimination und Permutation transformierten Matrix A vor Beginn der j-ten Elimination und **nach** der Permutation ist.

Das Gaußsche Eliminationsverfahren lautet dann:

$$L_{n-1}P_{n-1} \cdot \ldots \cdot L_2P_2L_1P_1(A \mid b) = (R \mid y)$$

Diese Produktdarstellung lässt sich noch vereinfachen indem wir:

$$P_kL_j = \hat{L}_jP_k \text{ für } k > j \text{ ausnutzen}$$

Bezeichnen wir die in den Zeilen $k > j$ „permutierte“ Elementarmatrix $\hat{L}_j$ mit L_j^k, so gilt:

$$\begin{aligned} L_{n-1}P_{n-1} \cdot \ldots \cdot L_2P_2L_1P_1 &= L_{n-1}P_{n-1} \cdot \ldots \cdot L_3P_3L_2L_1^2P_2P_1 \\ &= L_{n-1}P_{n-1} \cdot \ldots \cdot L_3L_2^3P_3L_1^2P_2P_1 \\ &= L_{n-1}P_{n-1} \cdot \ldots \cdot L_3L_2^3(L_1^2)^3P_3P_2P_1 \\ &\vdots \\ &= L_{n-1}\tilde{L}_{n-2}\tilde{L}_{n-3} \cdot \ldots \cdot \tilde{L}_2\tilde{L}_1P_{n-1} \cdot \ldots \cdot P_2P_1 \end{aligned}$$

wobei die $\tilde{L}_j$ die entsprechend permutierten Elementarmatrizen L_j sind. Die Elementarmatrizen sind invertierbar, daher erhalten wir die linke Dreiecksmatrix

$$L := (\tilde{L}_1)^{-1}(\tilde{L}_2)^{-1} \cdot \ldots \cdot (\tilde{L}_{n-2})^{-1}(\tilde{L}_{n-1})^{-1}$$

und die Permutationsmatrix

$$P := P_{n-1} \cdot \ldots \cdot P_2P_1$$

$\Rightarrow PA = LR$

Damit haben wir folgenden Satz bewiesen.

Satz 7.4.1 (Existenz der *LR*-Zerlegung):
Zu jeder Matrix $A \in \mathbb{K}^{n\times n}$ gibt es eine Permutationsmatrix P, eine linke Dreiecksmatrix $L = (l_{i,j})_{i,j}$ mit $l_{i,i} = 1$, $i = 1, \ldots, n$ und eine rechte Dreiecksmatrix R, so dass

$$PA = LR$$

14.06.2012
17. Vorlesung

Der Satz 7.4.1 ist ohne Permutation im Allgemeinen falsch.
Ein Programm, welches die LR-Zerlegung einer (eventuell) permutierten Matrix A berechnet, erhält man aus Algorithmus 7.2.1 zur Gauß-Elimination.

Man speichert die Werte $l = \frac{a_{i,k}}{a_{k,k}}$ als $l_{i,k}$ ab und erhält so die untere Dreiecksmatrix L. Die obere Dreiecksmatrix R entsteht aus A durch überschreiben. Die Permutationsmatrix P erhält man schließlich aus dem Produkt aller durchgeführten Zeilenvertauschungen.

Löst man ein lineares Gleichungssystem mit der LR-Zerlegung, so hat man bezüglich des Rechenaufwandes keinen Vorteil, sofern man das lineare Gleichungssystem nur für eine rechte Seite löst. Aufwand: $\mathcal{O}(n^3)$.
Hat man mehrere rechte Seiten, muss die LR-Zerlegung nur einmal durchgeführt werden. Es muss dann für jede rechte Seite nur eine Vorwärts- und eine Rückwärtssubstitution durchgeführt werden.
Der Aufwand beträgt dann $\mathcal{O}(n^2)$.

Um aus einer gegebenen LR-Zerlegung $PA = LR$ die Lösung von $Ax = b$ zu erhalten muss man folgendermaßen vorgehen:

1. Löse $Ly = Pb$
2. Löse $Rx = y$

Algorithmus 7.4.1 LR-Zerlegung mit Spaltenpivotsuche

```
function [L,R,P] = LR(A)
n = size(A,1);
J = 1:n; % row index vector (to be permuted)
for k = 1:n
 % determine the maximal value a_max
 % and its index in the _shortened_ vector abs(A(k:n,k))
 [a_max i_max_rel] = max(abs(A(k:n,k)));
 if(a_max > A(k,k)) % need to swap?
  i_max = (k-1) + i_max_rel; % add offset to get row index
  tmp_row = A(i_max,:); % exchange k-th row with pivot row
  A(i_max,:) = A(k,:);
  A(k,:) = tmp_row;
  tmp = J(i_max); % permute corresponding row indices
  J(i_max) = J(k);
  J(k) = tmp;
 end
 % eliminate k-th entry in i-th row k+1 <= i <= n
 for i = k+1:n
  l_ik = A(i,k) / A(k,k);
  A(i,k:n) = A(i,k:n) - l_ik * A(k,k:n);
  A(i,k) = l_ik; % store l_ik in eliminated entry
 end
end
L = tril(A, -1) + eye(n);
R = triu(A);
P = eye(n); % create permutation matrix
P = P(J,:);
end
```

Exkurs: Rademacher

Korollar 7.4.1.a:
Es sei $PA = LR$ die LR-Zerlegung der Matrix A. Dann gilt

$$\det(A) = \pm \prod_{i=1}^{n} r_{i,i}$$

wobei das Vorzeichen in $\det(A)$ positiv ist, wenn eine gerade Anzahl an Zeilenvertauschungen vorgenommen wurde. Dementsprechend ist es negativ wenn eine ungerade Anzahl an Zeilenvertauschungen vorgenommen wurde.

7.5. Rückwärtsstabilität

Wir führen den komponentenweisen Betrag einer Matrix $M = (m_{i,j})_{i,j}$ ein:

$$|M| = \left(|m_{i,j}|\right)_{i,j}$$

Satz 7.5.1:
Sei $A \in \mathbb{K}^{n\times n}$ eine Matrix, die eine LR-Zerlegung besitzt. Dann erfüllen die berechneten Faktoren $\tilde{L}$ und $\tilde{R}$ der LR-Zerlegung die Gleichung

$$A + \delta_A = \tilde{L}\tilde{R}$$

mit einer Störung $\delta_A \in \mathbb{K}^{n\times n}$, für die gilt:

$$|\delta_A| \leq 3 \cdot (n-1) \cdot eps \cdot (A + \tilde{L}\tilde{R}) + \mathcal{O}(eps^2)$$

Satz 7.5.2:
Seien $\tilde{L}$ und $\tilde{R}$ die berechneten Faktoren der LR-Zerlegung der Matrix $A \in \mathbb{K}^{n\times n}$. Beim lösen von $\tilde{L}\tilde{y} = b$ und $\tilde{R}x = \tilde{y}$ ergibt sich

$$(A + E)\tilde{x} = b$$

mit $|E| \leq n \cdot eps \cdot (3 \cdot |A| + 5 \cdot |\tilde{L}||\tilde{R}|) + \mathcal{O}(eps^2)$.

Bemerkung 7.5.1:
Ohne Pivotsuche können die Faktoren $|\tilde{L}|$ und $|\tilde{R}|$ groß werden. Da Permutationen ohne Rundungsfehler durchgeführt werden können, gilt Satz 7.5.2 auch für PA, wobei P eine entsprechende Permutationsmatrix ist, die durch Spaltenpivotsuche entsteht. Bei Spaltenpivotsuche wissen wir, dass für die Elemente $\tilde{l}_{i,j}$ der Matrix $\tilde{L}$ die Abschätzung $|\tilde{l}_{i,j}| \leq 1$ gilt.

$$\|\tilde{L}\|_\infty \leq n$$

Aus Satz 7.5.2 ergibt sich

$$\|E\|_\infty \leq n \cdot eps \cdot (3 \cdot \|A\|_\infty + 5n \cdot \|\tilde{R}\|_\infty) + \mathcal{O}(eps^2)$$

Es bleibt $\|\tilde{R}\|_\infty$ gegen $\|A\|_\infty$ abzuschätzen. Dazu definieren wir den **Wachstumsfaktor** $\rho_n(A)$ als

$$\rho_n(A) := \frac{\alpha_{\max}}{\max_{i,j=1,\ldots,n}|a_{i,j}|}$$

wobei $\alpha_{\max}$ der Betrag des betragsmäßig größten Elements ist, welches im Laufe der Gauß-Elimination in allen Matrizen $A = A^{(1)}, \ldots, A^{(n)} = \tilde{R}$ auftritt.

$$\|\tilde{R}\|_\infty \leq n \cdot \max_{i,j=1,\ldots,n}|\tilde{r}_{i,j}| \leq n \cdot \alpha_{\max} \text{ und } \max_{i,j=1,\ldots,n}|a_{i,j}| \leq \|A\|_\infty$$

$$
\begin{aligned}
\Rightarrow \|\tilde{R}\|_\infty &\leq n \cdot \frac{\alpha_{\max}}{\|A\|_\infty} \cdot \|A\|_\infty \\
&\leq \frac{n \cdot \alpha_{\max}}{\max_{i,j=1,\ldots,n}|a_{i,j}|} \cdot \|A\|_\infty \\
&= n \cdot \rho_n(A) \cdot \|A\|_\infty \\
\Rightarrow \|E\|_\infty &\leq n \cdot eps \cdot (3\|A\|_\infty + 5n^2 \cdot \rho_n(A) \cdot \|A\|_\infty) + \mathcal{O}(eps^2) \\
\Rightarrow \frac{\|E\|_\infty}{\|A\|_\infty} &\leq 8n^3 \cdot \rho_n(A) \cdot eps
\end{aligned}
$$

n^3 ist hier mathematisches Artefakt. Wichtig ist der Wachstumsfaktor $\rho_n(A)$. Die Rückwärtsstabilität der Gauß-Elimination hängt von der Größe des Wachstumsfaktors $\rho_n(A)$ ab.

Satz 7.5.3:
Für eine beliebige Matrix $A \in \mathbb{K}^{n\times n}$ und die Gauß-Elimination mit Spaltenpivotsuche erfüllt der Wachstumsfaktor die Bedingung

$$\rho_n(A) \leq 2^{n-1}$$

Ist die Abschätzung in Satz 7.5.3 scharf?

Beispiel 7.5.1 (Wilkinson):

$$A = \begin{pmatrix} 1 & 0 & \ldots & 0 & 1 \\ -1 & \ddots & \ddots & \vdots & \vdots \\ \vdots & \ddots & \ddots & 0 & \vdots \\ \vdots & & \ddots & \ddots & 1 \\ -1 & \ldots & \ldots & -1 & 1 \end{pmatrix} \in \mathbb{R}^{n\times n}$$

Man rechnet leicht nach, dass $\alpha_{\max} = 2^{n-1}$ und somit ist $\rho_n(A) = 2^{n-1}$.
Überraschendes Ergebnis: Für allgemeine Matrizen A ist die Gauß-Elimination nicht rückwärtsstabil. Für die meisten in der Praxis vorkommenden Matrizen ist der Wachstumsfaktor nicht so groß.

Für symmetrische (hermitesche) positiv definite Matrizen gilt:

$$\rho_n(A) = \mathcal{O}(1)$$

Mit statistischen Hilfsmitteln kann gezeigt werden, dass die Gauß-Elimination mit Spaltenpivotsuche für eine große Klasse von **praxisrelevanten Matrizen rückwärtsstabil** ist, siehe z. B. [6].

18.06.2012
18. Vorlesung

Wenn wir nun $Ax = b$ mit Hilfe der Gauß-Elimination mit Spaltenpivotsuche gelöst haben, so stellt sich die Frage, ob für die berechnete Lösung $\tilde{x}$ gilt:

$$(A + E)\tilde{x} = b \text{ mit } \frac{\|E\|}{\|A\|} = \mathcal{O}(eps)? \tag{7.2}$$

Sei nun $(A + E)\tilde{x} = b$, dann betrachten wir das Residuum

$$
\begin{aligned}
r &= b - A\tilde{x} \\
\Rightarrow 0 &= b - (A + E)\tilde{x} = r - E\tilde{x}
\end{aligned} \tag{7.3}
$$

Mit Hilfe des Residuums werden wir ein Kriterium zur Überprüfung von (7.2) entwickeln

$$\begin{aligned}(7.3) \Rightarrow \|r\|_2 &= \|E\tilde{x}\|_2 \leq \|E\|_2 \cdot \|\tilde{x}\|_2 \\ \Rightarrow \frac{\|R\|_2}{\|A\|_2} &\geq \frac{\|r\|_2}{\|A\|_2 \cdot \|\tilde{x}\|_2} \text{ (untere Abschätzung)}\end{aligned}$$

Hat das relative Residuum $\frac{r}{\|A\|_2 \cdot \|\tilde{x}\|_2}$ eine große Norm, so kann (7.2) nicht erfüllt sein. Dies gilt auch für die Matrixnorm.
Es gilt auch die Umkehrung: Ist die euklidische Norm des relativen Residuums „klein“, so gilt (7.2). zu zeigen: Es existiert eine Matrix E mit $(A+E)\tilde{x} = b$ und kleiner Norm $\|E\|_2$. Dazu definiere

$$E := \frac{r\tilde{x}^\mathsf{T}}{\|\tilde{x}\|_2^2}$$

Man sieht leicht, dass für das dyadische Produkt yz^T mit $y, z \in \mathbb{K}^n$ gilt:

$$\|yz^\mathsf{T}\|_2 = \|y\|_2\|z\|_2$$

Hieraus folgt

$$\|E\|_2 = \frac{\|r\tilde{x}\|_2}{\|\tilde{x}\|_2^2} = \frac{\|r\|_2\|\tilde{x}\|_2}{\|\tilde{x}\|_2^2} = \frac{\|r\|_2}{\|\tilde{x}\|_2} = \|A\|_2\frac{\|r\|_2}{\|A\|_2\|\tilde{x}\|_2}$$

Des weiteren gilt

$$b - (A+E)\tilde{x} = (b - A\tilde{x}) - E\tilde{x} = r - \frac{r\tilde{x}^\mathsf{T}\tilde{x}}{\|\tilde{x}\|} = 0$$

und somit auch

$$(A+E)\tilde{x} = b$$

Eine analoge Aussage kann mit etwas mehr Aufwand für die Maximumsnorm bewiesen werden [vgl. 7].
Zusammenfassend kann man sagen, dass die **Norm des relativen Residuums**, also

$$\frac{\|r\|}{\|A\| \cdot \|\tilde{x}\|}$$

ein **verlässlicher Stabilitätsindikator** ist, wobei $\|\cdot\|$ die euklidische Norm oder die Maximumsnorm sein kann.

7.6. Die Cholesky-Zerlegung

In diesem Abschnitt werden wir uns mit hermitesch bzw. symmetrisch positiv definiten Matrizen beschäftigen. Die LR-Zerlegung ist in diesem Fall wesentlich einfacher.

Zur Erinnerung: Grundlagen aus der Linearen Algebra

Sei $A \in \mathbb{K}^{n\times n}$, dann gilt

1. Ist $\mathbb{K} = \mathbb{C}$, dann heißt A **hermitesch**, falls $A^\mathsf{H} = A$
   Ist $\mathbb{K} = \mathbb{R}$, so heißt A **symmetrisch**, falls $A^\mathsf{T} = A$

2. Ist A hermitesch, so ist $\langle Ax, y\rangle_2 = \langle y, Ax\rangle_2$ reell und A besitzt nur reelle Eigenwerte sowie ein zugehöriges Orthonormalsystem aus Eigenvektoren
3. Ist $\langle Ax, x\rangle \geq 0 \; \forall 0 \neq x \in \mathbb{K}^n$, so heißt A **positiv semidefinit**
   Gilt die strenge Ungleichung $\forall 0 \neq x \in \mathbb{K}^n$, so heißt A **positiv definit**
4. Die Untermatrizen $A_k := (a_{i,j})_{1 \leq i,j \leq k}$, $k = 1, \ldots, n$, der hermiteschen Matrix sind genau dann positiv definit, wenn A positiv definit ist. Somit sind alle Diagonalelemente einer hermitesch positiv definiten Matrix positiv
5. A hermitesch positiv definit $\iff$ alle Eigenwerte von A sind positiv

Satz 7.6.1:
*Sei $A \in \mathbb{K}^{n \times n}$ eine hermitesch bzw. symmetrisch positiv definite Matrix. Dann existiert **genau eine obere Dreiecksmatrix** $R \in \mathbb{K}^{n \times n}$ mit positiven Diagonalelementen, so dass*

$$A = R^H R$$

*Diese Zerlegung heißt **Cholesky-Zerlegung**.*

Beweis (per vollständiger Induktion):
$n = 1$: trivial ✓

$n - 1 \mapsto n$: Der Satz ist richtig für $n - 1$. Da A positiv definit ist, ist auch die Untermatrix A_{n-1} positiv definit und besitzt genau eine Cholesky-Zerlegung $A_{n-1} = R_{n-1}^{\mathsf{H}} R_{n-1}$.

Dann gilt die Zerlegung

$$A = \begin{pmatrix} A_{n-1} & c \\ c & \alpha \end{pmatrix} = \begin{pmatrix} R_{n-1}^{\mathsf{H}} & 0 \\ r^{\mathsf{H}} & \beta \end{pmatrix} \begin{pmatrix} R_{n-1} & r \\ 0 & \beta \end{pmatrix} =: R^{\mathsf{H}} R$$

wenn folgende Gleichungen erfüllt sind

$$\begin{aligned} R_{n-1}^{\mathsf{H}} r &= c \\ r^{\mathsf{H}} r + \beta^2 &= \alpha \end{aligned}$$

Offenbar ist die erste Gleichung eindeutig lösbar, da R_{n-1} invertierbar. Somit existiert ein eindeutig bestimmter Vektor r und aus der zweiten Gleichung erhalten wir $\beta^2 = \alpha - r^{\mathsf{H}} r$. Betrachte

$$\begin{aligned} 0 < \det(A) &= \det(R^{\mathsf{H}}) \det(R) \\ &= |\det(R_{n-1})|^2 \beta^2 \end{aligned}$$

Also ist $\beta^2 > 0$ und somit existiert ein eindeutig bestimmtes $\beta > 0$.

□

Der Beweis liefert uns direkt einen Algorithmus. Sei $A = R^{\mathsf{H}} R$ die Cholesky-Zerlegung von A, dann gilt folgender Zusammenhang zwischen dem $a_{j,k}$ und dem $r_{i,k}$.

$$\begin{aligned} a_{j,k} &= \sum_{i=1}^{j} \overline{r}_{i,j} r_{i,k} \quad k > j \\ a_{k,k} &= \sum_{i=1}^{k} |r_{i,k}|^2 \end{aligned}$$

Hieraus ergibt sich folgender Algorithmus:

Algorithmus 7.6.1 Cholesky-Zerlegung

1: **for** $k \leftarrow 1$ **to** n **do**
2: **for** $j \leftarrow 1$ **to** $k-1$ **do**
3: $r_{j,k} = \dfrac{a_{j,k} - \sum_{i=1}^{j-1} \overline{r}_{i,j} r_{i,j}}{\overline{r}_{j,j}}$
4: **end for**
5: $r_{k,k} = \sqrt{a_{k,k} - \sum_{i=1}^{k-1} |r_{i,k}|^2}$
6: **end for**

Im Anhang befindet sich eine MATLAB-Implementierung

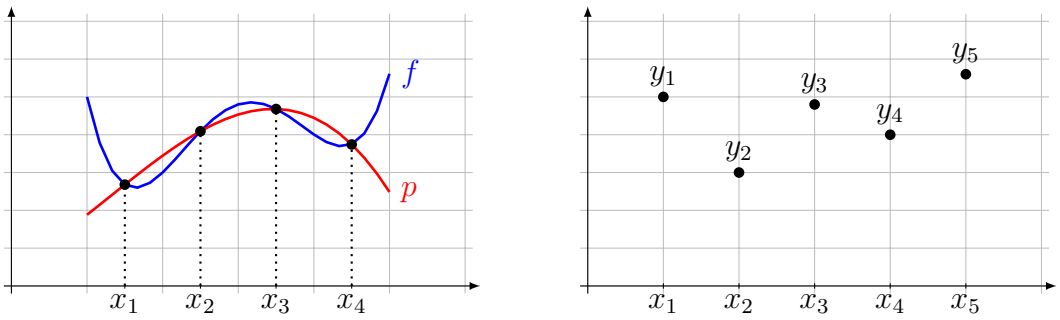
Der Aufwand der Cholesky-Zerlegung entspricht $\frac{n^3}{6} + \mathcal{O}(n^2)$ *flops* und beträgt somit etwa die Hälfte des Aufwands einer LR-Zerlegung. Hat man die Cholesky-Zerlegung $A = R^{\mathsf{H}} R$ einer hermitesch positiv definiten Matrix berechnet, so löst man das lineare Gleichungssystem $Ax = b$ wie folgt:

1. Löse $R^{\mathsf{H}} y = b$
2. Löse $Rx = y$

Wachstumsfaktor für die Cholesky-Zerlegung: $\rho_n(A) \in \mathcal{O}(1)$.
Die Cholesky-Zerlegung ist **rückwärtsstabil**.

8. Polynominterpolation

8.1. Einführung



Problem:
Sei

$$P_n = \left\{ \sum_{k=0}^{n} a_k x_k \colon x, a_k \in \mathbb{C},\ k = 0, \dots, n \right\}$$

die Menge aller Polynome mit komplexen oder reellen Koeffizienten vom Grad $\leq n$.
Gegeben seien $n+1$ Stützpunkte $(x_i, y_i) \in \mathbb{C}^2$, $i = 0, \dots, n$.
Gesucht ist ein Polynom $p \in P_n$ mit $p(x_i) = y_i \ \forall i = 0, \dots, n$.

8.2. Lagrange-Interpolation

Satz 8.2.1 (Existenz und Eindeutigkeit):
Zu beliebigen $n+1$ Stützpunkten (x_i, y_i), $i = 0, \dots, n$ mit $x_i \neq x_k$ für $i \neq k$ existiert genau ein Polynom $p \in P_n$ mit $p(x_i) = y_i \ \forall i = 0, \dots, n$.

Beweis:

1. Eindeutigkeit:
   Annahme: $\exists p_1, p_2$, so dass $p_1(x_i) = p_2(x_i) = y_i \ \forall i$
   $\Rightarrow p = p_1 - p_2 \in P_n \ \wedge \ p(x_i) = 0$
   $\Rightarrow p$ ist Polynom vom Grad $\leq n$ mit mindestens $n+1$ Nullstellen
   $\Rightarrow p \equiv 0 \Rightarrow p_1 = p_2$

2. Existenz (konstruktiv):
   Nach Lagrange konstruieren wir spezielle Interpolationspolynome $L_i \in P_n$, $i = 0, \dots, n$ mit $L_i(x_k) = \delta_{i,k} = \begin{cases} 1, & \text{falls } i = k \\ 0, & \text{sonst} \end{cases}$. Folgende Polynome besitzen diese Eigenschaft: $L_i(x) = \prod_{\substack{k=0 \\ k \neq i}}^{n} \left(\frac{x - x_k}{x_i - x_k} \right)$. Das Interpolationspolynom p ergibt sich durch die **Lagrangesche Interpolationsformel**:

$$\boxed{\sum_{i=0}^{n} y_i L_i(x) = p(x) = \sum_{i=0}^{n} y_i \prod_{\substack{k=0 \\ k \neq i}}^{n} \left(\frac{x - x_k}{x_i - x_k} \right)}$$

□

21.06.2012
19. Vorlesung

Bemerkung 8.2.1:

1. Das Interpolationspolynom $p \in P_n$ hängt linear von den Stützwerten y_i, $i = 0, \ldots, n$ ab
2. Für praktische Rechnungen mit großer Anzahl Stützstellen $n \in \mathbb{N}$ ist dieser Ansatz weniger geeignet
3. Das Lagrangeinterpolationspolynom ist nützlich, wenn viele Interpolationsprobleme mit gleichen Stützstellen $x_i,\ i = 0, \ldots, n$ aber jeweils verschiedenen Stützwerten $y_i,\ i = 0, \ldots, n$ gelöst werden müssen

Beispiel:

i	0	1	2
x_i	0	1	3
y_i	1	3	2

$$L_i(x) = \sum_{i=0}^{n} y_i \prod_{\substack{k=0 \\ k \neq i}}^{n} \left(\frac{x - x_k}{x_i - x_k} \right)$$

Gesucht ist $p \in P_2$ mit $p(x_i) = y_i,\ i = 0, 1, 2$.
Dazu berechnen wir zunächst die $L_i,\ i = 0, 1, 2$:

$$L_0(x) = \frac{(x - x_1)(x - x_2)}{(x_0 - x_1)(x_0 - x_2)} = \frac{(x-1)(x-3)}{(0-1)(0-3)} = \frac{(x-1)(x-3)}{3}$$
$$L_1(x) = \frac{(x - x_0)(x - x_2)}{(x_1 - x_0)(x_1 - x_2)} = \frac{(x-0)(x-3)}{(1-0)(1-3)} = \frac{x(3-x)}{2}$$
$$L_2(x) = \frac{(x - x_0)(x - x_1)}{(x_2 - x_0)(x_2 - x_1)} = \frac{(x-0)(x-1)}{(3-0)(3-1)} = \frac{x(x-1)}{6}$$

Aus der Lagrangeschen Interpolationsformel folgt:

$$\begin{aligned} p(x) &= y_0 L_0(x) + y_1 L_1(x) + y_2 L_2(x) \\ &= 1 \cdot \frac{(x-1)(x-3)}{3} + 3 \cdot \frac{x(3-x)}{2} + 2 \cdot \frac{x(x-1)}{6} \\ &= \frac{-5x^2 + 17x + 6}{6} \end{aligned}$$

Probe: $p(0) = 1$
$p(1) = 3$
$p(3) = 2$

Der Nachteil der Lagrangeschen Darstellung des Interpolationspolynoms ist, dass bei Hinzunahme einer weiteren Stützstelle x_{n+1} alle Berechnungen erneut durchgeführt werden müssen.
Wünschenswert wäre eine Darstellung, bei der die vorherigen wiederverwendet werden können. Eine solche Darstellung liefert die **Newtonsche Interpolationsformel**.

8.3. Newtonsche Interpolationsformel und dividierte Differenzen

Um ein Interpolationspolynom $p \in P_n$ mit $p(x_i) = y_i,\ i = 0, \dots, n$ zu finden, machen wir folgenden Ansatz:

$$\begin{aligned} p(x) &= p_{0,1,\dots,n}(x) \\ &:= a_0 + a_1(x - x_0) + a_2(x - x_0)(x - x_1) + \cdots + a_n(x - x_0)(x - x_1)\dots(x - x_{n-1}) \in P_n \end{aligned}$$

Aus $p(x_i) = y_i,\ i = 0, \dots, n$ erhalten wir folgende Bestimmungsgleichungen:

$$\begin{aligned} p(x_0) &= a_0 &&= y_0 \\ p(x_1) &= a_0 + a_1(x_1 - x_0) &&= y_1 \\ \vdots \quad &= \qquad\qquad \vdots &&= \vdots \\ p(x_n) &= a_0 + a_1(x_n - x_0) + a_2(x_n - x_0)(x_n - x_1) + \cdots + a_n(x_n - x_0)(x_n - x_1)\dots(x_n - x_{n-1}) &&= y_n \end{aligned}$$

Die Koeffizienten $a_i,\ i = 0, \dots, n$ des Interpolationspolynoms $p_{0,1,\dots,n}(x)$ können hieraus mit n Divisionen und $n(n-1)$ Multiplikationen berechnet werden. Es gibt einen weiteren Ansatz, der nur $\frac{n(n+1)}{2}$ Divisionen benötigt, die sogenannten **dividierten Differenzen**.

Definition 8.3.1 (dividierte Differenzen):
Gegeben seien die Stützpunkte $(x_i, y_i), \quad i = 0, \dots, n$, *dann definieren wir die* ***dividierten Differenzen*** *als*

1. $[y_i] := y_i, \quad i = 0, \dots, n$
2. $[y_i, \dots, y_k] := \dfrac{[y_{i+1}, \dots, y_k] - [y_i, \dots, y_{k-1}]}{x_k - x_i}, \quad 0 \le i < k \le n$

Beispiel:

$$\begin{aligned} [y_i] &= y_i && && i = 0, 1, 2 \\ [y_0, y_1] &= \frac{[y_1] - [y_0]}{x_1 - x_0} && = \frac{y_1 - y_0}{x_1 - x_0} \\ [y_1, y_2] &= \frac{[y_2] - [y_1]}{x_2 - x_1} && = \frac{y_2 - y_1}{x_2 - x_1} \\ [y_0, y_1, y_2] &= \frac{[y_1, y_2] - [y_0, y_1]}{x_2 - x_0} && = \frac{1}{x_2 - x_0}\left(\frac{y_2 - y_1}{x_2 - x_1} - \frac{y_1 - y_0}{x_1 - x_0}\right) \end{aligned}$$

Berechnet man die dividierten Differenzen per Hand, so ist es nützlich, diese in folgendem Differenzenschema anzuordnen:

$$\begin{array}{l|lll} x_0 & [y_0] & & \\ & & [y_0, y_1] & \\ x_1 & [y_1] & & [y_0, y_1, y_2] \\ & & [y_1, y_2] & & [y_0, y_1, y_2, y_3] \\ x_2 & [y_2] & & [y_1, y_2, y_3] \\ & & [y_2, y_3] & \\ x_3 & [y_3] & & \end{array}$$

Satz 8.3.1 (Rekursionsformel nach Neville):
Es sei $p_{i,\dots,k+1} \in P_{k+1}$ also nach Satz 8.2.1 eindeutig bestimmte Interpolationspolynom mit $p_{i,\dots,k+1}(x_j) = y_j,\ j = 0,\dots,k+1$. Dann gilt:

$$p_{i,\dots,k+1}(x) = \frac{(x - x_i)p_{i+1,\dots,k+1}(x) + (x_{k+1} - x)p_{i,\dots,k}(x)}{x_{k+1} - x_i} \tag{8.1}$$

Beweis:
Einsetzen in die rechte Seite von (8.1) ergibt:

$$\begin{aligned} p_{i,\dots,k}(x_i) &= y_i \\ &= p_{i,\dots,k+1}(x_i) \\ p_{i+1,\dots,k+1}(x_{k+1}) &= y_{k+1} \\ &= p_{i,\dots,k+1}(x_{k+1}) \end{aligned}$$

$$\frac{(x_j - x_i)\overbrace{p_{i+1,\dots,k+1}(x_j)}^{=y_j} + (x_{k+1} - x_j)\overbrace{p_{i,\dots,k}(x_j)}^{=y_j}}{x_{k+1} - x_i} = \frac{(x_{k+1} - x_i)y_j}{x_{k+1} - x_i} = y_j$$

$p_{i,\dots,k+1}(x_j),\ j = i+1,\dots,k$ $\quad x_i \quad x_{i+1} \quad \cdots \quad \cdots \quad \cdots \quad x_k \quad x_{k+1}$

Die Behauptung folgt direkt mit Satz 8.2.1, da $p_{i,\dots,k+1}(x)$ hiernach eindeutig bestimmt ist. □

Satz 8.3.2:
Für die Koeffizienten des Newtonschen Interpolationspolynoms $p_{0,1,\dots,n}(x) \in P_n$ gilt:

$$a_i = [y_0, y_1, \dots, y_i], \quad i = 0,\dots,n$$

Beweis (per vollständiger Induktion):
Induktion über $m = k - i$:

$$P_{i,\dots,k}(x) = [y_i] + [y_i, y_{i+1}](x - x_i) + \dots + [y_i,\dots,y_k](x - x_i)\cdot\dots\cdot(x - x_{k-1})$$

Da $p_i \in P_0$ konstanten: $p_i(x_i) = y_i$ ergibt sich für den Induktionsanfang $m = 0$.
$p_i(x_i) = y_i = [y_i]$. Im Induktionsschluss sei die Behauptung für $m \geq 0$ richtig, es gilt also $p_{i,\dots,k}(x) = [y_i,\dots,y_k]x^{k-i} + q_1 \in P_{k-i}$ mit $q_1 \in P_{k-i-1}$ und
$p_{i+1,\dots,k+1}(x) = [y_{i+1},\dots,y_{k+1}]x^{k-i} + q_2 \in P_{k-i}$ mit $q_2 \in P_{k-i-1}$. Aus der Rekursionsformel nach Neville folgt:

$$\begin{aligned} p_{i,\dots,k+1}(x) &= \frac{(x - x_i)p_{i+1,\dots,k+1}(x) + (x_{k+1} - x)p_{i,\dots,k}(x)}{x_{k+1} - x_i} \\ &\overset{\text{Induktions-voraussetzung}}{=} \frac{(x - x_i)[y_{i+1},\dots,y_{k+1}] + (x_{k+1} - x)[y_i,\dots,y_k]}{x_{k+1} - x_i}x^{k-i} + r_1 \\ &= \frac{[y_{i+1},\dots,y_{k+1}] - [y_i,\dots,y_k]}{x_{k+1} - x_i}x^{k-i+1} + r_2 \\ &\overset{\text{Definition 8.3.1 dividierte Differenzen}}{=} [y_i,\dots,y_{k+1}]x^{k-i+1} + r_2 \\ &= [y_i,\dots,y_{k+1}](x - x_i)\cdot\dots\cdot(x - x_k) + r_3 \end{aligned}$$

Hierbei sind r_1, r_2, r_3 jeweils passende Restpolynome aus P_{k-i}. Weiterhin gilt für $j = i,\dots,k$: $r_3(x_j) = p_{i,\dots,k+1}(x_j) = y_j$. Da das Interpolationspolynom $p_{i,\dots,k}(x) \in P_{k-i}$ mit

$p_{i,\dots,k}(x_j) = y_j,\ j = i, \dots, k$ eindeutig bestimmt ist, folgt

$$r_3(x) = p_{i,\dots,k}(x)$$

Nach Induktionsvoraussetzung ist $r_3(x)$ aber schon in Newtonscher Interpolationsform und somit folgt die Behauptung.

□

25.06.2012
20. Vorlesung

Beispiel:

i	x_i	y_i
0	0	1
1	1	3
2	3	2

Das zugehörige Differenzenschema der dividierten Differenzen:

x_i	y_i	[ ]	[ ]
0	$\boxed{1}$		
		$\boxed{2}$	
1	3		$\boxed{-\frac{5}{6}}$
		$-\frac{1}{2}$	
3	2		

Wir erhalten:

$$p(x) = P_{0,1,2}(x) = 1 + 2x - \frac{5}{6}x(x-1) = 1 + \frac{17}{6}x - \frac{5}{6}x^2$$

Fügen wir noch eine Stützstelle hinzu, z. B. $x_3 = 2,\ y_3 = 4$, so lautet das Differenzenschema:

x_i	y_i	[ ]	[ ]	[ ]
0	$\boxed{1}$			
		$\boxed{2}$		
1	3		$\boxed{-\frac{5}{6}}$	
		$-\frac{1}{2}$		$\boxed{-\frac{1}{3}}$
3	2		$-\frac{3}{2}$	
		-2		
2	4			

Somit ergibt sich für das Interpolationspolynom:

$$p(x) = P_{0,1,2,3}(x) = 1 + 2x - \frac{5}{6}x(x-1) - \frac{1}{3}x(x-1)(x-3)$$

8.4. Interpolationsfehler

In diesem Abschnitt wollen wir untersuchen, welchen Fehler man macht, wenn man eine gegebene Funktion f durch ein Polynom p approximiert, so dass f und p in bestimmten Stützstellen x_i übereinstimmen.
Es sei $f \in \mathcal{C}^{(n+1)}([a,b])$, $[a,b] \subset \mathbb{R}$ und $x_i \in [a,b]$, $i = 0, \dots, n$. Die Stützstellen seien

paarweise verschieden, d. h. $x_i \neq x_j$ für $i \neq j$. Die Stützwerte seien durch

$$y_i = f(x_i), \quad i = 0, \ldots, n$$

gegeben. Mit $p \in P_n$ bezeichnen wir das eindeutig bestimmte Interpolationspolynom mit $p(x_i) = y_i$, $i = 0, \ldots, n$. Für ein beliebiges $x \in [a, b]$ sei I_x das kleinste Intervall, welches die Punkte $x_0, \ldots, x_n$ und x enthält.

Satz 8.4.1:
Zu jedem $x \in [a, b]$ existiert ein $\xi \in I_x$, so dass

$$f(x) - p(x) = \omega(x) \frac{f^{(n+1)}(\xi)}{(n+1)!}$$

mit $\omega(x) = \prod_{j=0}^{n}(x - x_j)$ und $f^{(n+1)}(\xi) = \frac{\mathrm{d}^{(n+1)} f}{\mathrm{d}x^{(n+1)}}(\xi)$.

Beweis:
Sei $\bar{x} \in I_x$ beliebig mit $\bar{x} \neq x_j$, $j = 0, \ldots, n$. Dann definieren wir $F(x) := f(x) - p(x) - K\omega(x)$ mit $K = \textit{konstant}$ und wählen K so, dass $F(\bar{x}) = 0$ ist, d. h.

$$K := \left(\frac{f - p}{\omega} \right) (\bar{x})$$

Die Funktion F hat in I_x genau $n + 2$ Nullstellen, nämlich $x_0, \ldots, x_n, \bar{x}$. Aus dem Satz von Rolle erhalten wir, dass $F^{(n+1)}$ in I_x eine Nullstelle ξ hat. Da $p^{(n+1)} \equiv 0$ gilt

$$F^{(n+1)}(x) = f^{(n+1)}(x) - K(n+1)!$$

und wir erhalten

$$K = \left(\frac{f - p}{\omega} \right) (\bar{x}) = \frac{f^{(n+1)}(\xi)}{(n+1)!}$$

Somit gilt:

$$f(\bar{x}) - p(\bar{x}) = \omega(\bar{x}) \frac{f^{(n+1)}(\xi)}{(n+1)!}$$

Diese Gleichung gilt offensichtlich auch für $\bar{x} = x_i$, $i \in \{0, \ldots, n\}$.

□

Korollar 8.4.1.a:
Für den Interpolationsfehler gilt:

$$|f(x) - p(x)| \leq |\omega(x)| \cdot \max_{\xi \in I_x} \frac{|f^{(n+1)}(\xi)|}{(n+1)!}$$

Wir betrachten nun den Fall äquidistanter Stützstellen in $[a, b]$. Sei $n \in \mathbb{N}$ und $x_j := a + j \cdot h$, $j = 0, \ldots, n$ mit der Schrittweite

$$h := \frac{b - a}{n}$$

Sei nun $x \in [a, b]$, dann existiert ein $\theta \in \mathbb{R}$ mit $0 \leq \theta \leq n$, so dass

$$x = a + \theta \cdot h$$

Weiterhin gilt:

$$\omega(x) = \prod_{j=0}^{n}(x - x_j) = \prod_{j=0}^{n}(\theta h - jh) = h^{(n+1)} \prod_{j=0}^{n}(\theta - j)$$

Ist $p \in P_n$ das Interpolationspolynom in den Stützstellen $\big(x_j, f(x_j)\big)$, $j = 0, \ldots, n$ so gilt für den Interpolationsfehler bei äquidistanten Stützstellen:

$$|f(x) - p(x)| \leq \frac{h^{(n+1)}}{(n+1)!} \left(\prod_{j=0}^{n} |\theta - j| \right) \max_{\xi \in I_x} \left| f^{(n+1)}(\xi) \right|$$

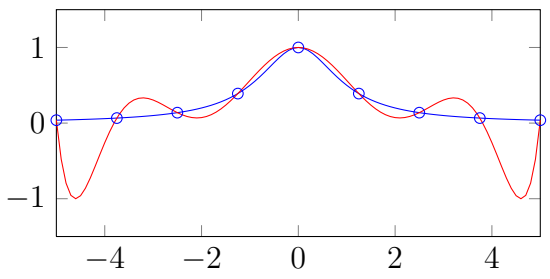
1. Halten wir die Anzahl n der Stützstellen fest und betrachten $b - a \to 0$, so haben wir $|f(x) - p(x)| = \mathcal{O}(h^{(n+1)})$
2. Betrachten wir andererseits ein festes Intervall und eine wachsende Anzahl von Stützstellen, also $n \to \infty$, so verbessert sich die Konvergenz im Allgemeinen nicht, wie das folgende Beispiel 8.4.1 zeigt.

Beispiel 8.4.1 (von Runge):
Wir betrachten die Funktion

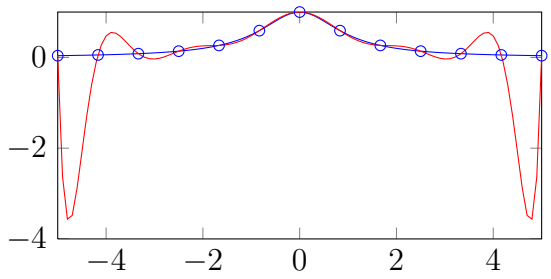
$$f(x) := \frac{1}{1 + x^2} \quad x \in [-5, 5]$$

Approximieren wir $f(x)$ nun durch ein Polynom $p \in P_n$ mit wachsendem $n \in \mathbb{N}$ und den äquidistanten Stützstellen $x_j = -5 + \frac{10j}{n}$, $j = 0, \ldots, n$ sehen wir, dass der Interpolationsfehler in der Maximumsnorm mit wachsendem n steigt. An den Intervallrändern zeigt das Interpolationspolynom mit wachsendem n ein stark oszillierendes Verhalten.

Approximation der Runge-Funktion mit n Stützstellen:



$n = 9$



$n = 13$

Es stellt sich die Frage, unter welchen Voraussetzungen man Funktionen durch Polynome gut approximieren kann.

Satz 8.4.2 (Weierstraßscher Approximationssatz):
Sei $f\colon [a, b] \to \mathbb{R}$ eine stetige Funktion und sei $\varepsilon > 0$ ein beliebig vorgegebener Wert. Dann existiert ein Polynom p, so dass

$$\forall x \in [a, b] \text{ die Abschätzung } |f(x) - p(x)| < \varepsilon \text{ gilt}$$

Beweis:
[Siehe 8, Teil II, Seite 449 ff]

□

Der Weierstraßsche Approximationssatz sagt also aus, dass es zu einer gegebenen stetigen Funktion f eine Folge von Polynomen gibt, die gleichmäßig gegen f konvergiert. Im Beispiel 8.4.1 wurden äquidistante Stützstellen verwendet. Die schlechte Approximation kann durch eine geschickte Wahl der Stützstellen verbessert werden.

Idee: Wir versuchen $\max_{x\in[a,b]}|\omega(x)|$ minimal zu bekommen.

Zunächst halten wir fest, dass es genügt die Aufgabe auf dem Intervall $[-1,1]$ zu lösen.

$$\begin{aligned} x\colon [a,b] &\to [-1,1] \\ x(t) &\mapsto \frac{2t-a-b}{b-1} \end{aligned}$$

Sowie die Umkehrabbildung

$$\begin{aligned} t\colon [-1,1] &\to [a,b] \\ t(x) &\mapsto \frac{1-x}{2}a + \frac{1+x}{2}b \end{aligned}$$

Ist nun $p \in P_n$ mit führendem Koeffizienten 1 die Lösung des Minimax-Problems

$$\min_{\substack{q\in P_n \\ a_n=1}} \max_{x\in[a,b]} |q(x)|$$

so ist $\hat{p}(t) := p\Big(x(t)\Big)$ Lösung des Minimax-Problems

$$\min_{\substack{\hat{q}\in P_n \\ a_n=\frac{2^n}{(b-a)^n}}} \max_{t\in[-1,1]} |\hat{q}(t)|$$

Als nächstes definieren wir die Tschebyscheff-Polynome auf $[-1,1]$.

Definition 8.4.1 (Tschebyscheff-Polynome):
Für $n = 0, 1, \ldots$ definieren wir die ***Tschebyscheff-Polynome*** *durch*

$$T_n(x) := \cos(n \cdot \arccos(x)), \quad x \in [-1,1]$$

Man rechnet leicht nach, dass folgende Rekursion erfüllt ist:

$$T_0(x) = 1, \quad T_1(x) = x, \quad T_n(x) = 2x \cdot T_{n-1}(x) - T_{n-2}(x)$$

für $x \in [-1,1]$ und $n \geq 2$.

28.06.2012
21. Vorlesung

Aus der Definition 8.4.1 folgt: Der führende Koeffizient a_n von T_n ist 2^{n-1} (für $n \geq 1$). Die Tschebyscheff-Polynome erfüllen eine bestimmte Minimax-Aufgabe auf $[-1,1]$.

Satz 8.4.3:

1. *Für jedes Polynom $p_n \in P_n$ mit führendem Koeffizienten $a_n \neq 0$ existiert ein $x \in [-1,1]$, so dass*
$$|p_n(x)| \geq \frac{|a_n|}{2^{(n-1)}} \text{ gilt}$$

2. *Die Tschebyscheff-Polynome $T_n(x)$ sind minimal bezüglich der Maximumsnorm $\|\cdot\|_\infty$ unter den Polynomen vom Grad n mit führendem Koeffizienten $2^{(n-1)}$, d. h. sie lösen folgende Minimax-Aufgabe*
$$\min_{\substack{p \in P_n \\ a_n = 2^{(n-1)}}} \max_{x \in [-1,1]} |p(x)|$$

Beweis:

1. Angenommen $p_n \in P_n$ mit führendem Koeffizienten $a_n = 2^{n-1}$ und $|p_n(x)| < 1 \; \forall x \in [-1,1]$. Hieraus folgt:
$$p_n(x) - T_n(x) \in P_{n-1}$$
da $T_n(x)$ als führenden Koeffizienten 2^{n-1} hat. Nach Voraussetzung gilt: $-1 < p_n(x) < 1 \; \forall x \in [-1,1]$.

   1.1. Für $\bar{x}_{2k} := \cos\left(\frac{2k\pi}{n}\right) \in [-1,1]$ gilt:
$$T_n(\bar{x}_{2k}) = 1, \quad p_n(\bar{x}_{2k}) < 1 \quad \Rightarrow \quad p_n(\bar{x}_{2k}) - T_n(\bar{x}_{2k}) < 0$$

   1.2. Für $\bar{x}_{2k+1} := \cos\left(\frac{(2k+1)\pi}{n}\right) \in [-1,1]$ gilt:
$$T_n(\bar{x}_{2k+1}) = -1, \quad p_n(\bar{x}_{2k+1}) > -1 \quad \Rightarrow \quad p_n(\bar{x}_{2k+1}) - T_n(\bar{x}_{2k+1}) > 0$$

   Hieraus folgt, dass $p_n - T_n$ in den Punkten $\bar{x}_k, \; k = 0, \ldots, n$ alternierend das Vorzeichen wechselt und somit in $[-1,1]$ mindestens n Nullstellen hat. Da $p_n - T_n \in P_{n-1} \Rightarrow p_n = T_n$. Dies ist ein Widerspruch zur Annahme, dass $|p_n(x)| < 1 \; \forall x \in [-1,1]$.
   $\Rightarrow$ Also muss es für jedes $p_n \in P_n$ mit $a_n = 2^{n-1}$ ein $x \in [-1,1]$ geben, so dass $|p_n(x)| \geq 1$. Für beliebige Polynome $p_n \in P_n$ mit $a_n \neq 0$ folgt die Behauptung mit $\tilde{p}_n := \frac{2^{n-1}}{a_n} p_n$, da $\tilde{a}_n = 2^{n-1}$.
$$|\tilde{p}_n(x)| \geq 1 \quad \Leftrightarrow \quad \left|\frac{2^{n-1}}{a_n}\right| \cdot |p_n(x)| \geq 1$$

2. Es genügt zu zeigen, dass für alle Polynome $p_n \in P_n$ mit führendem Koeffizienten $a_n = 2^{n-1}$ gilt:
$$\max_{x \in [-1,1]} |T_n(x)| \leq \max_{x \in [-1,1]} |p_n(x)|$$
   $\max_{x \in [-1,1]} |T_n(x)| = 1 \leq \max_{x \in [-1,1]} |p_n(x)|$.

□

Gesucht ist nun
$$\omega(x) := \prod_{j=0}^{n} (x - x_j) \in P_n$$

mit Nullstellen $x_j \in [-1,1]$, so dass $\max_{x\in[-1,1]}|\omega(x)|$ minimal wird. Das gesuchte Polynom ist nach Satz 8.4.3 das Polynom $\frac{1}{2^{n-1}}T_n(x)$, dessen Nullstellen

$$x_i = \cos\left(\frac{2i+1}{2n+2}\cdot\pi\right), \quad i = 0,\dots,n$$

die **Tschebyscheff-Knoten** sind.

Sind die Stützstellen die Tschebyscheff-Knoten, so gilt für den Interpolationsfehler

$$|f(x) - p(x)| \leq \frac{1}{2^n(n+1)!}\max_{\xi\in I_x}|f^{(n+1)}(\xi)|$$

8.5. Hermite-Interpolation

Das Problem der Polynominterpolation (siehe Abschnitt 8.1) kann verallgemeinert werden, indem man in den Stützstellen nicht nur Funktionswerte des Polynoms vorschreibt, sondern auch dessen Ableitungen.

Problem:
Sei

$$P_n = \left\{\sum_{k=0}^{n} a_k x_k \colon x, a_k \in \mathbb{C},\ k = 0,\dots,n\right\}$$

die Menge aller Polynome mit komplexen oder reellen Koeffizienten vom Grad $\leq n$.
Gegeben seien $n+1$ Stützpunkte $(x_i, y_i^{(k)}) \in \mathbb{C}^2,\ i = 0,\dots,m,\ k = 0,\dots,(n_i)-1$ mit $n+1 = \sum_{i=1}^{m} n_i$.
Gesucht ist ein Polynom $p \in P_n$ mit

$$p^{(k)}(x_i) = y_i^{(k)}, \quad i = 0,\dots,m, \quad k = 0,\dots,(n_i)-1$$

Die Lagrangesche-Interpolation ist also ein Spezialfall der Hermite-Interpolation.

Satz 8.5.1 (Existenz und Eindeutigkeit):
Zu beliebigen paarweise verschiedenen Stützstellen $x_i,\ i = 0,\dots,m$ und zugehörigen Stützwerten $y_i^{(k)},\ k = 0,\dots,(n_i)-1,\ i = 0,\dots,m$ gibt es genau ein Polynom $p \in P_n$ mit $n+1 = \sum_{i=1}^{m} n_i$, so dass

$$p^{(k)}(x_i) = y_i^{(k)}, \quad i = 0,\dots,m, \quad k = 0,\dots,(n_i)-1$$

erfüllt ist.

Beweis:
Es sei angenommen p_1 und p_2 seien zwei Polynome, welche die Hermite Interpolationsbedingung erfüllen. Somit gilt für $q(x) := p_1(x) - p_2(x)$:

$$q^{(k)}(x_i) = 0, \quad i = 0,\dots,m, \quad k = 0,\dots,(n_i)-1$$

Somit ist x_i mindestens n_i-fache Nullstelle von q. Also hat q mindestens $n+1 = \sum_{i=0}^{m} n_i$ Nullstellen. Da jedoch $q \in P_n \Rightarrow q \equiv 0$.
Die Existenz der Hermiten Interpolation lässt sich direkt aus der Eindeutigkeit schließen.

Dazu betrachten wir die $n+1$ Koeffizienten c_i von

$$p(x) = c_0 + c_1 x + \cdots + c_n x^n$$

als Lösungen des linearen Gleichungssystems, welches durch die Hermiten Interpolationsbedingungen gegeben ist. Aus der Eindeutigkeit des Interpolationspolynoms folgt, dass die Matrix des linearen Gleichungssystems nicht singulär ist.

□

Für den Interpolationsfehler der Hermiten Interpolation gilt folgender Satz, der analog zu Satz 8.4.1 zu beweisen ist.

Satz 8.5.2:
Sei $f \in \mathcal{C}^{n+1}([a,b])$, $[a,b] \subset \mathbb{R}$ eine gegebene Funktion und $x_i \in [a,b]$, $i = 0,\ldots,m$ seien paarweise verschiedene Stützstellen. Das Polynom p besitzt höchstens den Grad n und erfülle erfülle in den Stützstellen die Interpolationsbedingung

$$p^{(k)}(x_i) = f^{(k)}(x_i), \quad i = 0,\ldots,m, \quad k = 0,\ldots,(n_i)-1$$

wobei $p^{(k)}, f^{(k)}$ die k-ten Ableitungen von p, f sind und $n+1 = \sum_{i=1}^{m} n_i$. Dann existiert zu jedem $x \in [a,b]$ ein $\xi \in I_x$, so dass

$$f(x) - p(x) = \omega(x)\frac{f^{(n+1)}(\xi)}{(n+1)!}$$

wobei I_x das kleinste Intervall ist, welches $x_0,\ldots,x_m$ und x enthält und

$$\omega(y) := (y-x_0)^{n_0} \cdot (y-x_1)^{n_1} \cdot \ldots \cdot (y-x_m)^{n_m}$$

8.6. Spline-Interpolation

Der Begriff des Spline kommt aus dem Schiffbau und bezeichnet eine dünne, biegsame Latte, eine sogenannte Straklatte, die zwischen Gewichte („Knoten“) eingespannt wird und so Kurven darstellt. Ein Kurvenlineal, etwa im Entwurf beim Straßenbau, ist auch ein Spline. Ein Spline kann zwischen den Knoten als Polynom dargestellt werden, die an den Knoten durch gewisse Differenzierbarkeitsanforderungen – so glatt wie benötigt – zusammengefügt werden. In der Numerik sind Splines daher Funktionen, welche auf den durch die Knoten gebildeten Intervalle durch stückweise Polynome dargestellt werden und die in den Knoten gewissen Differenzierbarkeitsbedingungen genügen.

Definition 8.6.1 (Spline):
Es seien $x_0,\ldots,x_n \in [a,b] \subset \mathbb{R}$, $a = x_0 < x_1 < \ldots < x_n = b$, $n \in \mathbb{N}$. Die Funktion $s_k(x)$ heißt ***Spline*** *der Ordnung $k \in \mathbb{N}$ relativ zu $x_0,\ldots,x_n$ genau dann, wenn*

1. $s_k|_{[x_j,x_{j+1}]} \in P_k$, $j = 0,\ldots,n-1$
2. $s_k \in \mathcal{C}^{k-1}([a,b])$ *(für $k=0$ ist dies die leere Menge $\emptyset$)*

Beispiel 8.6.1:

1. Jedes Polynom $p \in P_k$ ist ein Spline der Ordnung k

2. $n = 0,\ x_0 = t$

$$s(x) := (x-t)_t^{k-1} := \begin{cases} (x-t)^{k-1}, & x > t \\ 0, & \text{sonst} \end{cases}$$

Dann ist s ein Spline der Ordnung k relativ zu x_0

02.07.2012
22. Vorlesung

Im Allgemeinen besteht ein Spline auf jedem Teilintervall $[x_j, x_{j+1}]$ aus einem Polynom vom Grad $\leq k$. Der Grad kann von Teilintervall zu Teilintervall verschieden sein. Es kann Unstetigkeiten in der k-ten Ableitung geben.
Nach den Bedingungen aus Definition 8.6.1 gilt für $j = 0, \ldots, n-1$

$$s_{k,j} := s_k|_{[x_j, x_{j+1}]} \in P_k \Rightarrow s_{j,k}(x) = \sum_{i=0}^{k} s_{i,j}(x - x_j)^i$$

mit passenden Koeffizienten $s_{i,j} \in \mathbb{R}$. Es müssen $(k+1) \cdot n$ Koeffizienten $s_{i,j}$ bestimmt werden. Nach der zweiten Bedingung aus Definition 8.6.1 gilt

$$s_k \in \mathcal{C}^{k-1}([a,b]) \Rightarrow \text{für } j = 1, \ldots, n, \quad m = 0, \ldots, k-1 \text{ gelten } s_{k,j-1}^{(m)}(x_j) = s_{k,j}^{(m)}(x_j)$$

Dies ergibt $k \cdot (n-1)$ Bedingungen zur Bestimmung der Koeffizienten $s_{i,j}$. Somit bleiben $(k+1)n - k(n-1) = n + k$ Freiheitsgrade bzw. unbestimmte Koeffizienten.
Angenommen die Splines sind interpolierend, d. h. für gegebene Werte $y_i,\ i = 0, \ldots, n$ gilt

$$s_k(x_j) = y_j$$

Jetzt bleiben immer noch $k-1$ Freiheitsgrade. Die Bestimmung dieser verbleibenden Koeffizienten führt auf verschiedene Klassen von Splines.

Definition 8.6.2 (periodische, natürliche und vollständige Splines):

1. ***Periodische Splines:***

$$s_k^{(m)}(a) = s_k^{(m)}(b), \quad m = 0, \ldots, k-1$$

2. ***Natürliche Splines:***
Falls für $k = 2l-1$ mit $l \geq 2$ gilt, dann gilt für $j = 0, \ldots, l-2$:

$$s_k^{(l+j)}(a) = s_k^{(l+j)}(b) = 0$$

3. ***Vollständige Splines ($k = 3$):***

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

Interpolierende, kubische Splines

Dazu seien $x_i \in [a,b] \subset \mathbb{R},\ i = 0, \ldots, n$ gegebene Stützstellen mit
$a = x_0 < x_1 < \ldots < x_n = b$ und $y_i,\ i = 0, \ldots, n$ gegebene Stützwerte.
Gesucht ist ein Spline $s_3(x)$ vom Grad 3 mit

$$y_i = s_3(x_i), \quad m_i := s_3'(x_i), \quad M_i := s_3''(x_i), \quad i = 0, \ldots, n$$

Da $s_{3,i-1} \in P_3$, ist $s''_{3,i-1} \in P_1$ also linear. Somit gilt:

$$s''_{3,i-1}(x) = M_{i-1}\frac{x_i - x}{h_i} + M_i\frac{x - x_{i-1}}{h_i}, \quad x \in [x_{i-1}, x_i]$$

mit $h_i := x_i - x_{i-1}$.
Zweimaliges Integrieren von $s''_{3,i-1}$:

$$s_{3,i-1}(x) = M_{i-1}\frac{(x_i - x)^3}{6h_i} + M_i\frac{(x - x_{i-1})^3}{6h_i} + c_{i-1}(x - x_{i-1}) + \tilde{c}_{i-1}$$

mit Integrationskonstanten $c_{i-1}, \tilde{c}_{i-1}$. Diese bestimmt man mit Hilfe der Interpolationsbedingungen $s_3(x_{i-1}) = y_{i-1},\ s_3(x_i) = y_i$ woraus folgt

$$\begin{aligned}\tilde{c}_{i-1} &= y_{i-1} - M_{i-1}\frac{h_i^2}{6}\\ c_{i-1} &= \frac{y_i - y_{i-1}}{h_i} - \frac{h_i}{6}(M_i - M_{i-1})\end{aligned}$$

mit $i = 0, \ldots, n-1$.
Aus der Stetigkeit der ersten Ableitung ergibt sich mit $s'_3(x_i^{\pm}) := \lim_{t\to 0} s'_3(x_i \pm t)$ die Beziehung:

$$\begin{aligned}s'_3(x_i^-) &= \frac{h_i}{6}M_{i-1} + \frac{h_i}{3}M_i + \frac{y_i - y_{i-1}}{h_i}\\ &= -\frac{h_{i+1}}{3}M_i - \frac{h_{i+1}}{6}M_{i+1} + \frac{y_{i+1} - y_i}{h_{i+1}}\\ &= s'_3(x_i^+)\end{aligned}$$

für $i = 1, \ldots, n-1$.

Notation:

$$\begin{aligned}\mu_i &:= \frac{h_i}{h_i + h_{i+1}}\\ \lambda_i &:= \frac{h_{i-1}}{h_i + h_{i+1}}\\ \alpha_i &:= \frac{6}{h_i + h_{i+1}}\left(\frac{y_{i+1} - y_i}{h_{i+1}} - \frac{y_i - y_{i-1}}{h_i}\right)\end{aligned}$$

Damit ergibt sich folgendes lineares Gleichungssystem

$$\mu_i M_{i-1} + 2M_i + \lambda_i M_{i+1} = \alpha_i$$

mit $i = 1, \ldots, n-1$.
Dieses System hat $n+1$ Unbekannte $(M_0, \ldots, M_n)$ aber nur $n-1$ Gleichungen. Es fehlen also noch zwei Bedingungen. Im Allgemeinen sehen diese wie folgt aus:

$$\begin{aligned}2M_0 + \lambda_0 M_1 &= \alpha_0\\ \mu_n M_{n-1} + 2M_n &= \alpha_n\end{aligned}$$

Bei natürlichen Splines haben wir

$$s''_3(a) = s''_3(b) = 0 \quad \Longrightarrow \quad \lambda_0 = \mu_n = \alpha_0 = \alpha_n = 0$$

Im Allgemeinen ist zur Bestimmung der Koeffizienten M_i ein tridiagonales Gleichungssystem zu lösen.

$$\begin{pmatrix} 2 & \lambda_0 & 0 & \dots & 0 \\ \mu_1 & 2 & \ddots & \ddots & \vdots \\ 0 & \ddots & \ddots & \ddots & 0 \\ \vdots & \ddots & \ddots & 2 & \lambda_{n-1} \\ 0 & \dots & 0 & \mu_n & 2 \end{pmatrix} \begin{pmatrix} M_0 \\ \vdots \\ M_n \end{pmatrix} = \begin{pmatrix} \alpha_0 \\ \vdots \\ \alpha_n \end{pmatrix}$$

Satz 8.6.1:
Sei $f \in \mathcal{C}^2([a,b])$ und sei s_3 der natürliche, kubische Spline, der f interpoliert. Dann gilt

$$\int_a^b \left(s_3''(x)\right)^2 \mathrm{d}x \leq \int_a^b \left(f''(x)\right)^2 \mathrm{d}x$$

wobei Gleichheit genau dann gilt, wenn $f = s_3$.

Beweis (nach [9, Seite 245]):
Aus der Identität $f'' = s_3'' + (f'' - s_3'')$ ergibt sich:

$$\int_a^b (f'')^2 \,\mathrm{d}x = \int_a^b (s_3'')^2 \,\mathrm{d}x + 2\int_a^b s_3''(f'' - s_3'') \,\mathrm{d}x + \underbrace{\int_a^b (f'' - s_3'')^2 \,\mathrm{d}x}_{\geq 0}$$

Es genügt also zu zeigen, dass der mittlere Summand verschwindet. Aus den Randbedingungen für natürliche Splines folgt:

$$\int_a^b s_3''(f'' - s_3'') \,\mathrm{d}x = \left[s_3''(f' - s_3')\right]_a^b - \int_a^b s_3'''(f' - s_3') \,\mathrm{d}x$$

wobei s_3''' auf den Teilintervallen (x_i, x_{i+1}) konstant ist, da s_3 dort kubisch ist. Im Allgemeinen ist s_3''' in x_i unstetig.
Es sei $d_i := s_3'''(x) = s_{3,i}'''(x),\ x \in (x_i, x_{i+1})$. Dann gilt unter der Voraussetzung $s_3''(a) = s_3''(b) = 0$

$$\begin{aligned} \int_a^b s_3''(f'' - s_3'') \,\mathrm{d}x &= -\sum_{i=1}^n \int_{x_{i-1}}^{x_i} d_i(f' - s_3') \,\mathrm{d}x \\ &= -\sum_{i=1}^n d_i \int_{x_{i-1}}^{x_i} (f' - s_3') \,\mathrm{d}x \\ &= \underbrace{-\sum_{i=1}^n d_i \left(\left(\underbrace{f(x_i) - s_3(x_i)}_{=0} \right) - \left(\underbrace{f(x_{i-1}) - s_3(x_{i-1})}_{=0} \right) \right)}_{=0} \end{aligned}$$

$\Rightarrow$ Behauptung.

□

Bemerkung 8.6.1:
Der Satz gilt allgemeiner für kubische interpolierende Splines mit der Eigenschaft

$$s_3''(b)\Big(f'(b) - s_3'(b)\Big) = s_3''(a)\Big(f'(a) - s_3'(a)\Big)$$

und somit für natürliche, vollständige und periodische Splines.

Satz 8.6.2:
Es sei s_3 der vollständige, kubische Spline, der $f \in \mathcal{C}^4([a,b])$ in den Knoten $x_0, \dots, x_n$ interpoliert. Sei $h := \max_{i\in\{0,\dots,n-1\}}|x_{i+1} - x_i|$. Dann gilt:

$$\|f - s_3\|_\infty \leq \frac{5}{384} h^4 \|f^{(4)}\|_\infty$$

Kubische interpolierende Splines haben folgende interessante Eigenschaft:
$f\colon [a,b] \to \mathbb{R}$ sei gegeben, $f \in \mathcal{C}^2([a,b])$, f beschreibe eine parametrisierte Kurve in der Ebene.
Bereits aus der Analysis II bekannt:

$$\kappa(t) := \frac{f''(t)}{\Big(1 + f'(t)^2\Big)^{\frac{3}{2}}}$$

Nimmt man nun an, dass lokale Änderungen klein sind, d. h. $|f'(t)| \ll 1$, so erhalten wir als approximierte Krümmung

$$\kappa(t) \approx f''(t)$$

Messen wir die Krümmung der gesamten Kurve in der L_2-Norm, so sagt Satz 8.6.1 aus, dass kubische, interpolierende Splines dieses Funktional minimieren. Beschreibt nun $f(t)$ die Lage einer dünnen Holzlatte, so ist durch

$$E(f) = \int_a^b \left(\frac{f''(t)}{\Big(1 + f'(t)^2\Big)^{\frac{3}{2}}} \right)^2 \mathrm{d}t$$

die Biegeenergie der Latte gegeben. Für kleine Auslenkungen gilt:

$$E(f) \approx \int_a^b \Big(f''(t)\Big)^2 \,\mathrm{d}t \overset{\text{Satz 8.6.1}}{\geq} \int_a^b \Big(s_3''(t)\Big)^2 \,\mathrm{d}t$$

9. Numerische Integration

05.07.2012
23. Vorlesung

9.1. Einführung

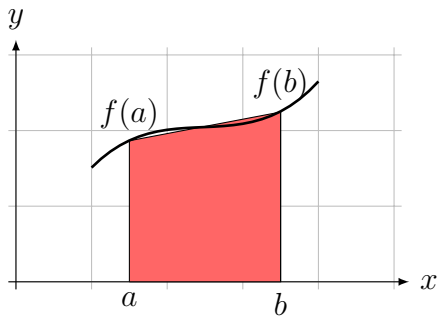
$f(x) = e^{-x^2}$ einfach zu differenzieren, aber $\int_a^b e^{-x^2}\,\mathrm{d}x$ lässt sich nicht einfach durch algebraische oder transzendente Funktionen berechnen.

Andere Beispiele:

- Elliptische Integrale (Bogenlänge von Ellipsen berechnen)

Prinzipiell kann ein Integral $\int_a^b f(x)\,\mathrm{d}x$, welches eine Zahl darstellt, numerisch berechnet werden. Ein zugehöriges Verfahren, welches das Integral unter Benutzung der Funktionswerte $f(x)$ berechnet, nennt man **numerisches Integrationsverfahren** oder **numerisches Quadraturverfahren**.

9.2. Die Trapezregel



$$\begin{aligned} F &= m \cdot h \\ m &= \frac{f(a)+f(b)}{2} \\ h &= b-a \end{aligned}$$

Aus der nebenstehenden Abbildung erhält man als erste Idee zur Approximation von $\int_a^b f(x)\,\mathrm{d}x$ folgende Formel:

$$\int_a^b f(x)\,\mathrm{d}x \approx \frac{b-a}{2} \cdot \Big(f(a)+f(b)\Big)$$

Die soeben hergeleitete Formel heißt **Trapezregel**. Man kann diese auch analytisch herleiten. Dazu konstruieren wir eine lineare Funktion $p \in P_1$ mit $p(a) = f(a)$, $p(b) = f(b)$ und wählen $\int_a^b p(x)\,\mathrm{d}x \approx \int_a^b f(x)\,\mathrm{d}x$. Für $p(x)$ haben wir:

$$p(x) = f(a) + \frac{f(b)-f(a)}{b-a}(x-a)$$

Integrieren von $p(x)$ ergibt:

$$\int_a^b p(x)\,\mathrm{d}x = \frac{b-a}{2}\Big(f(a)+f(b)\Big) \approx \int_a^b f(x)\,\mathrm{d}x$$

Als nächstes leiten wir mit Hilfe der Darstellung des Interpolationsfehlers eine Abschätzung des Integrationsfehlers her (Satz 8.4.1)

$$\forall x \in [a,b] \quad \exists \xi_x \in [a,b]\colon f(x) - p(x) = (x-a)(x-b)\frac{f''(\xi_x)}{2}$$

Sei nun $I_h(f) := \frac{b-a}{2}\Big(f(a)+f(b)\Big) = \int_a^b p(x)\,\mathrm{d}x = I(p)$. Dann gilt:

$$\begin{aligned} I(f) - I_h(f) &= \int_a^b f(x)\,\mathrm{d}x - \int_a^b p(x)\,\mathrm{d}x \\ &= \int_a^b f(x) - p(x)\,\mathrm{d}x \\ &= \int_a^b (x-a)(x-b)\frac{f''(\xi_x)}{2}\,\mathrm{d}x \\ &\overset{\text{MWS}^1}{=} \frac{f''(\eta)}{2}\int_a^b (x-a)(x-b)\,\mathrm{d}x \\ &= -\frac{f''(\eta)}{12}(b-a)^3 = -\frac{f''(\eta)}{12}h^3 \end{aligned}$$

mit $h := b-a$.
Für den Fehler der Trapezregel gilt somit:
Unter der Voraussetzung $f \in \mathcal{C}^2([a,b])$ $\exists \eta \in (a,b)$, so dass

$$\int_a^b f(x)\,\mathrm{d}x - I_h(f) = -\frac{f''(\eta)}{12}h^3$$

mit $h = b-a$ fest und vorgegebenem f.
Eine genauere Approximation ergibt die **zusammengesetzte Trapezregel**. Dazu betrachten wir folgende Zerlegung von $[a,b]$:

$$\begin{gathered} a = x_0 < x_1 < \ldots < x_{n-1} < x_n = b \\ h = \frac{b-a}{n}, \quad x_i = a + ih, \quad i = 0,\ldots,n \end{gathered}$$

Wir approximieren das Integral auf jedem Teilintervall durch die Trapezregel.

$$\int_{x_{i-1}}^{x_i} f(x)\,\mathrm{d}x \approx \frac{h}{2}\Big(f(x_{i-1}) + f(x_i)\Big)$$

und erhalten so die zusammengesetzte Trapezregel.

$$\begin{aligned} \int_a^b f(x)\,\mathrm{d}x &= \sum_{i=1}^{n}\int_{x_{i-1}}^{x_i} f(x)\,\mathrm{d}x \\ &\approx \sum_{i=1}^{n}\frac{h}{2}\Big(f(x_{i-1}) + f(x_i)\Big) \\ &= h\left(\frac{f(x_0)}{2} + f(x_1) + f(x_2) + \cdots + f(x_{n-1}) + \frac{f(x_n)}{2}\right) \end{aligned}$$

Wir leiten nun eine **Fehlerabschätzung** für die zusammengesetzte Trapezregel her:

$$I_h(f) := h\left(\frac{f(x_0)}{2} + f(x_1) + f(x_2) + \cdots + f(x_{n-1}) + \frac{f(x_n)}{2}\right)$$

[1] Mittelwertsatz

$$\begin{aligned}
\Rightarrow \quad \int_a^b f(x)\,\mathrm{d}x - I_h(f) &= \sum_{i=1}^{n}\left\{\int_{x_{i-1}}^{x_i} f(x)\,\mathrm{d}x - \frac{h}{2}\Big(f(x_{i-1}) + f(x_i)\Big)\right\} \\
&\overset{\substack{\text{Fehlerabschätzung}\\ =\\ \text{Trapezregel}}}{=} -\frac{h^3}{12}\sum_{i=1}^{n} f''(\eta_i) \\
&= -\frac{h^3}{12}\frac{b-a}{n}\sum_{i=1}^{n} f''(\eta_i)
\end{aligned}$$

mit $\eta_i \in (x_{i-1}, x_i),\ i = 1, \ldots, n$.
Da durch $\frac{1}{n}\sum_{i=1}^{n} f''(\eta_i)$ der arithmetische Mittelwert von $f''(\eta_i),\ i = 1, \ldots, n$ gegeben ist, gilt:

$$\frac{1}{n}\sum_{i=1}^{n} f''(\eta_i) \in \left[\min_{i=1,\ldots,n} f''(\eta_i), \max_{i=1,\ldots,n} f''(\eta_i)\right]$$

Da nach Voraussetzung $f \in \mathcal{C}^2([a,b])$, ist f'' stetig und aus dem Mittelwertsatz der Differenzialrechnung folgt:

$$\exists \eta \in (a,b),\ \text{so dass}\ \frac{1}{n}\sum_{i=1}^{n} f''(\eta_i) = f''(\eta)$$

Für die Fehlerdarstellung der zusammengesetzten Trapezregel gilt somit:

$$\int_a^b f(x)\,\mathrm{d}x - I_h(f) = -\frac{b-a}{12} f''(\eta) h^2$$

Zwei direkte Folgerungen:

1. Die Approximation des Integrals kann immer genauer bestimmt werden indem man Punkte x_i hinzunimmt.
2. Der Fehler bei der zusammengesetzten Trapezregel ist $\mathcal{O}(h^2)$.
   Verdoppeln wir die Anzahl der Punkte, so vervierfachen wir die Genauigkeit.

9.3. Newton-Cotes-Formeln

Eine naheliegende Idee, um numerische Integrationsverfahren mit einer hohen Genauigkeit zu erhalten, ist die Erhöhung des Polynomgrades des integrierenden Polynoms. Dazu betrachten wir die Zerlegung:

$$\begin{gathered}
a = x_0 < x_1 < \ldots < x_{n-1} < x_n = b \\
h = \frac{b-a}{n}, \quad x_i = a + ih, \quad i = 0, \ldots, n
\end{gathered}$$

Sei $p_n \in P_n$ das integrierende Polynom mit $p_n(x_i) = f(x_i),\ i = 0, \ldots, n$.
Nach der Interpolationsformel von Lagrange gilt:

$$p_n(x) = \sum_{i=0}^{n} f_i \cdot L_i(x) \text{ mit } L_i = \prod_{\substack{k=0\\k\neq i}}^{n} \frac{x - x_k}{x_i - x_k} \text{ und } f_i = f(x_i)$$

Wir führen nun eine Variable $t \in \mathbb{R}$ ein und eine neue Funktion $\varphi_i(t)$, so dass

$$x = a + ht$$
$$L_i(x) = \varphi_i(t) = \prod_{\substack{k=0 \\ k \neq i}}^{n} \frac{t-h}{i-h}$$

gilt.

$$\begin{aligned} \Rightarrow \int_a^b p_n(x)\,\mathrm{d}x &= \sum_{i=1}^{n} f_i \int_a^b L_i(x)\,\mathrm{d}x \\ &\overset{\substack{\text{Substituiere} \\ x=a+ht}}{=} h \sum_{i=0}^{n} f_i \underbrace{\int_0^n \varphi_i(t)\,\mathrm{d}t}_{=:\alpha_i} \\ &= h \sum_{i=0}^{n} \alpha_i f_i \end{aligned}$$

mit den **Gewichten** $\alpha_i := \int_0^n \varphi_i(t)\,\mathrm{d}t$.

Bemerkung 9.3.1:
Die Gewichte α_i hängen nicht von f oder den Intervallgrenzen a, b ab. Wir erhalten somit folgende Gewichte und somit auch Integrationsformeln:

1. $n = 1$: **Trapezregel**
$$\alpha_0 = \frac{1}{2} = \alpha_1$$

2. $n = 2$: **Simpsonregel**
$$\begin{aligned} \alpha_0 &= \int_0^2 \varphi_0(t)\,\mathrm{d}t \\ &= \int_0^2 \frac{t-1}{0-1}\frac{t-2}{0-2}\,\mathrm{d}t \\ &= \frac{1}{2}\int_0^2 (t^2 - 3t + 2)\,\mathrm{d}t \\ &= \frac{1}{2}\left(\frac{8}{3} - \frac{12}{2} + 4\right) = \frac{1}{3} \\ \alpha_1 &= \int_0^2 \frac{t-0}{1-0}\frac{t-2}{1-2}\,\mathrm{d}t \\ &= -\int_0^2 (t^2 - 2t)\,\mathrm{d}t = \frac{4}{3} \\ \alpha_2 &= \int_0^2 \frac{t-0}{2-0}\frac{t-1}{2-1}\,\mathrm{d}t \\ &= \frac{1}{2}\int_0^2 (t^2 - t)\,\mathrm{d}t = \frac{1}{3} \end{aligned}$$

$\Rightarrow$ Simpsonregel: $\alpha_0 = \frac{1}{3},\ \alpha_1 = \frac{4}{3},\ \alpha_2 = \frac{1}{3}$

$$I_h(f) = \frac{h}{3}\Big(f(x_0) + 4f(x_1) + f(x_2)\Big)$$

Im Allgemeinen haben die **Newton-Cotes-Formeln** folgende Gestalt:

$$\int_a^b f(x)\,\mathrm{d}x \approx \int_a^b p_n(x)\,\mathrm{d}x = h\sum_{i=0}^{n}\alpha_i f_i$$

mit $f_i = f(x_i),\ h := \frac{b-a}{n}$.

09.07.2012
24. Vorlesung

Satz 9.3.1:
Die Gewichte $\alpha_i,\ i = 0,\dots,n$ der Newton-Cotes-Formel sind rationale Zahlen mit der Eigenschaft

$$\sum_{i=0}^{n}\alpha_i = n$$

Beweis:
Sei $f(x) = 1$, dann folgt aus der Eindeutigkeit des Interpolationspolynoms, dass $p_n(x) = 1$.

$$\overset{\text{Newton-Cotes}}{\Rightarrow} \quad (b-a) = h\cdot\sum_{i=0}^{n}\alpha_i$$

$$\Leftrightarrow \quad n = \sum_{i=0}^{n}\alpha_i$$

□

Nach Konstruktion sind die Newton-Cotes-Formeln exakt für Polynome $p_n \in P_n$, es gilt also

$$I_h(p) = h\sum_{i=0}^{n} f_i\alpha_i = \int_a^b p(x)\,\mathrm{d}x =: I(p)$$

Allgemeiner gilt für eine gegebene Menge von Knoten, dass die Integrationsformel dadurch eindeutig bestimmt ist. Dabei müssen die Knoten nicht äquidistant sein.

Satz 9.3.2:
Es seien $x_0,\dots,x_n$ paarweise verschiedene Knoten aus $[a,b]$. Dann existiert genau eine Integrationsformel

$$I_h(f) = (b-a)\sum_{i=0}^{n}\tilde{\alpha}_i f_i$$

die für alle Polynome $p \in P_n$ exakt ist, d. h. $I_h(p) = \int_a^b p(x)\,\mathrm{d}x\ \forall p \in P_n$.

Beweis:

„⇒“: Man wähle $\tilde{\alpha}_i := \frac{1}{b-a}\int_a^b L_i(x)\,\mathrm{d}x,\ L_i = \prod_{\substack{k=0\\k\neq i}}^{n}\frac{x-x_k}{x_i-x_k}$. Dann ist $I_h(f)$ exakt für alle $f = p \in P_n$.

„⇐“: Angenommen $I_h(p) = I(p)\ \forall p \in P_n$, dann gilt für $L_i \in P_n$

$$I_h(L_i) = I(L_i) = \int_a^b L_i(x)\,\mathrm{d}x = (b-a)\tilde{\alpha}_i$$

Man erhält somit die $\tilde{\alpha}_i$ auf eindeutige Art und Weise. Aus

$$I_h(f) = \sum_{i=0}^{n}\beta_i f_i$$

folgt $(b-a)\tilde{\alpha}_i = \sum_{j=0}^{n} \beta_i L_i(x_j) = \beta_i$. Somit gilt

$$I_h(f) = (b-a)\sum_{i=0}^{n} \tilde{\alpha}_i f_i$$

□

Satz 9.3.3:
Sei $h = \frac{b-a}{2}$, dann ist die ***Simpsonregel***

$$I_h(f) = \frac{h}{3}\left(f(a) + 4f\left(\frac{b+a}{2}\right) + f(b)\right)$$

exakt für Polynome $p \in P_3$. Ist $f \in \mathcal{C}^4([a,b])$, so gilt für den Approximationsfehler

$$\int_a^b f(x)\,\mathrm{d}x - I_h(f) = -\frac{f^{(4)}(\xi)}{90}h^5$$

für ein $\xi \in (a,b)$.

Beweis:
Sei $p \in P_2$ das integrierende Polynom mit $p(a) = f(a)$, $p(b) = f(b)$ und $p(\frac{a+b}{2}) = f(\frac{a+b}{2})$, so dass

$$I_h(p) = \int_a^b p(x)\,\mathrm{d}x$$

Aus Satz 8.4.1 folgt

$$f(x) - p(x) = \frac{f'''(\xi)}{6} \cdot \omega(x)$$

mit $\omega(x) = (x-a)(x-\frac{a+b}{2})(x-b)$. Im Unterschied zur Trapezregel nimmt $\omega(x)$ positive und negative Werte auf $[a,b]$ an. Deshalb können wir nicht analog vorgehen. Stattdessen konstruieren wir ein Hilfspolynom $q \in P_3$, so dass

$$\int_a^b f(x) - q(x)\,\mathrm{d}x = \int_a^b f(x) - p(x)\,\mathrm{d}x$$

Durch direktes nachrechnen folgt, dass für alle $c \in \mathbb{R}$ gilt

$$\int_a^b c\omega(x)\,\mathrm{d}x = 0$$

Für alle $c \in \mathbb{R}$ hat $q(x) := p(x) + c\omega(x)$ die Eigenschaft $\int_a^b q(x)\,\mathrm{d}x = \int_a^b p(x)\,\mathrm{d}x$. Wir wählen nun die Konstante $c \in \mathbb{R}$, so dass $q'(\frac{a+b}{2}) = f'(\frac{a+b}{2})$. Dazu betrachten wir $q'(\frac{a+b}{2}) = p'(\frac{a+b}{2}) + c(\frac{b-a}{2})(\frac{a-b}{2})$ woraus folgt

$$c = \frac{(p'-f')(\frac{a+b}{2})}{\left(\frac{b-a}{2}\right)^2} = \frac{(p'-f')(\frac{a+b}{2})}{h^2}$$

Nun betrachten wir die Hilfsfunktion

$$r(t) := f(t) - q(t) - \frac{f(x)-q(x)}{(x-a)\left(x-\frac{a+b}{2}\right)^2(x-b)}(t-a)\left(t-\frac{a+b}{2}\right)^2(t-b)$$

Dann gilt: $r(t) = 0 \Leftrightarrow t \in \left\{a, b, \frac{a+b}{2}, x\right\}$. Zwischen zwei aufeinanderfolgenden Nullstellen von r liegt eine Nullstelle von r'. Des weiteren ist nach Konstruktion von q bzw. der Wahl von c auch $\frac{a+b}{2}$ eine Nullstelle von r'. Somit hat r' mindestens vier Nullstellen in (a, b)

$$\Rightarrow r^{(4)} \text{ hat mindestens eine Nullstelle } \xi_x$$

Ausrechnen: $r^{(4)}(t) = f^{(4)}(t) - \frac{f(x)-q(x)}{(x-a)\left(x-\frac{a+b}{2}\right)^2(x-b)} \cdot 4!$
Somit gilt für den Integrationsfehler:

$$\forall x \in [a, b]\ \exists \xi_x \in (a, b) \colon f(x) - q(x) = \frac{1}{4!} f^{(4)}(\xi_x)(x-a)\left(x - \frac{a+b}{2}\right)^2 (x-b)$$

Offensichtlich gilt für alle $x \in [a, b]$:

$$(x-a)\left(x - \frac{a+b}{2}\right)^2 (x-b) \leq 0$$

Nun können wir wie bei der Trapezregel vorgehen. Da $f \in \mathcal{C}^{(4)}([a, b])$, ist $f^{(4)} \in \mathcal{C}([a, b])$. $\Rightarrow$ Es existieren ein Minimum m und ein Maximum M von $f^{(4)}$ auf $[a, b]$. Es gilt also folgende Abschätzung:

$$-\frac{m}{4!}(x-a)\left(x - \frac{a+b}{2}\right)^2 (x-b) \leq f(x) - q(x) \leq -\frac{M}{4!}(x-a)\left(x - \frac{a+b}{2}\right)^2 (x-b)$$

Aus dem Mittelwertsatz folgt: $\exists \mu \in (a, b)$, so dass

$$\begin{aligned}
\int_a^b f(x) - q(x)\,\mathrm{d}x &= -\frac{1}{4!} f^{(4)}(\mu) \int_a^b (x-a)\left(x - \frac{a+b}{2}\right)^2 (x-b)\,\mathrm{d}x \\
&= -\frac{f^{(4)}(\mu)}{24} \frac{4h^5}{15} \\
&= -\frac{f^{(4)}(\mu)}{90} h^5
\end{aligned}$$

Da die vierte Ableitung eines Polynoms vom Grad ≤ 3 verschwindet, folgt aus der Fehlerdarstellung sofort die Exaktheit der Simpsonregel für $p \in P_3$.

□

Man unterscheidet zwischen geschlossenen und offenen Newton-Cotes-Formeln

1. **Geschlossene Newton-Cotes-Formeln:**
   $x_0 = a, \quad x_n = b, \quad x_i = a + ih, \quad h = \frac{b-a}{n}, \quad i = 1, \ldots, n-1$

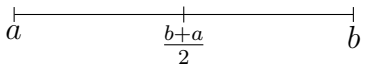
   Beispiele: 1.1. Trapezregel $(n = 1)$

   1.2. Simpsonregel $(n = 2)$

   1.3. $\frac{3}{8}$-Regel $(n = 3)$

2. **Offene Newton-Cotes-Formeln:**
   $x_0 = a + h, \quad x_n = b - h, \quad x_i = a + ih, \quad h = \frac{b-a}{n+2}, \quad n \geq 0, \quad i = 1, \ldots, n+1$

   Beispiele: 2.1. Mittelpunktsregel $(n = 1)$



Im folgenden Satz sind die Fehlerdarstellungen für offene und geschlossene Newton-Cotes-Formeln angegeben.

Satz 9.3.4:
Für jede Newton-Cotes-Formel $I_h(f)$, die einer geraden Zahl n entspricht, gilt unter der Voraussetzung $f \in \mathcal{C}^{n+2}([a,b])$ die Fehlerdarstellung

$$\int_a^b f(x)\,\mathrm{d}x - I_h(f) = \frac{M_n}{(n+2)!} h^{n+3} f^{(n+2)}(\xi) \tag{9.1}$$

mit $\xi \in (a,b)$ und $M_n = \begin{cases} \int_0^n t \cdot \Pi_{n+1}(t)\,\mathrm{d}t < 0 & \text{(geschlossene Formel)} \\ \int_{-1}^{n+1} t \cdot \Pi_{n+1}(t)\,\mathrm{d}t > 0 & \text{(offene Formel)} \end{cases}$
Dabei wurde zur Abkürzung $\Pi_{n+1}(t) := \prod_{i=0}^{n}(t-i)$ benutzt.
Aus (9.1) ergibt sich, dass der Exaktheitsgrad $n+1$ ist. Unter der Voraussetzung $f \in \mathcal{C}^{n+1}([a,b])$ gilt für ungerade Zahlen n die analoge Fehlerdarstellung

$$\int_a^b f(x)\,\mathrm{d}x - I_h(f) = \frac{K_n}{(n+1)!} h^{n+2} f^{(n+1)}(\xi)$$

mit $K_n = \begin{cases} \int_0^n \Pi_{n+1}(t)\,\mathrm{d}t < 0 & \text{(geschlossene Formel)} \\ \int_{-1}^{n+1} \Pi_{n+1}(t)\,\mathrm{d}t > 0 & \text{(offene Formel)} \end{cases}$
Der Exaktheitsgrad ist hier n.

Beweis:
[Siehe 10, Theorem 9.2]

□

12.07.2012
25. Vorlesung

Bemerkung 9.3.2:
Newton-Cotes-Formeln höherer Ordnung werden kaum benutzt. Besser ist es, zusammengesetzte Formeln zu verwenden, die auf jedem Teilintervall Formeln niedriger Ordnung benutzen, wie die Trapez- oder die Simpsonregel.

Als Beispiel betrachten wir die **zusammengesetzte Simpsonregel:**
Dazu sei $n = 2m,\ m \in \mathbb{N},\ h = \frac{b-a}{n},\ x_i = a + ih,\ i = 0, \ldots, n$. Für $f \in \mathcal{C}^4([a,b])$ gilt

$$\begin{aligned}
\int_a^b f(x)\,\mathrm{d}x &= \sum_{i=1}^{m} \int_{x_{2i-2}}^{x_{2i}} f(x)\,\mathrm{d}x \\
&\overset{2}{=} \sum_{i=1}^{m} \left\{ \frac{h}{3}\Big(f(x_{2i-2}) + 4f(x_{2i-1}) + f(x_{2i})\Big) - \frac{h^5}{90} f^{(4)}(\xi_i) \right\} \text{ mit } \xi_i \in (x_{2i-2}, x_{2i}) \\
&= \frac{h}{3}\left\{ f(x_0) + 4\sum_{i=1}^{m} f(x_{2i-1}) + 2\sum_{i=1}^{m-i} f(x_{2i}) + f(x_n) \right\} - \frac{h^5}{90}\sum_{i=1}^{m} f^{(4)}(\xi_i) \\
&= I_h(f) - \frac{h^4(b-a)}{180}\left\{ \frac{1}{m}\sum_{i=1}^{m} f^{(4)}(\xi_i) \right\} \\
&= I_h(f) - \frac{h^4(b-a)}{180} f^{(4)}(\xi)
\end{aligned}$$

Für ein $\xi \in (a,b)$. Im letzten Schritt sind wir vollkommen analog zur zusammengesetzten Trapezregel vorgegangen. Wir haben folgenden Satz 9.3.5 bewiesen.

[2] Simpsonregel auf (x_{2i-2}, x_{2i}) angewendet

Satz 9.3.5:
Sei $m \in \mathbb{N}$ und $h = \frac{b-a}{2m}$. Dann gilt für die zusammengesetzte Simpsonregel

$$I_h(f) := \frac{h}{3}\left(f(x_0) + 4\sum_{i=1}^{m} f(x_{2i-1}) + 2\sum_{i=1}^{m-1} f(x_{2i}) + f(x_n)\right)$$

die Fehlerdarstellung

$$\int_a^b f(x)\,\mathrm{d}x - I_h(f) = -\frac{h^4(b-a)}{180} f^{(4)}(\xi)$$

für ein $\xi \in (a,b)$.

9.4. Gauß-Integration

Bisher sind wir bei der Konstruktion von Quadraturformeln von einer vorgegebenen Wahl von Stützstellen ausgegangen und haben geeignete Gewichte gesucht. Wir werden nun versuchen, auch die Stützstellen optimal zu wählen, um so Quadraturformeln maximaler Genauigkeit zu erhalten.
Statt des Integrals einer Funktion $f(x)$ über ein Intervall $[a,b]$ betrachten wir

$$I(f) := \int_a^b \omega(x) f(x)\,\mathrm{d}x \text{ mit } a,b \in [-\infty,\infty]$$

Dabei heißt $\omega(x)$ **Gewichtsfunktion** und soll folgende Eigenschaften haben:

1. $\omega(x) \geq 0\ \forall x \in [a,b]$
2. $\omega(x)$ ist messbar auf $[a,b]$
3. Alle Momente $\mu_k := \int_a^b x^k \omega(x)\,\mathrm{d}x,\ k = 0,1,2,\ldots$ existieren und sind endlich
4. Für alle Polynome $s(x)$ mit $\int_a^b \omega(x)s(x)\,\mathrm{d}x = 0$ und $s(x) \geq 0\ \forall x \in [a,b]$ gilt $s(x) \equiv 0$

Diese Bedingungen sind erfüllt, falls $\omega(x)$ stetig ist und $\omega(x) > 0\ \forall x \in [a,b]$. Wir suchen jetzt eine Integrationsformel der Form

$$G_n(f) := \sum_{j=1}^{n} A_j f(x_j)$$

mit Gewichten A_j und Integrationspunkten $x_j,\ j = 1,\ldots,n$ mit größtmöglicher Genauigkeit. Es soll

$$I(p) = G_n(p) \quad \forall p \in P_k$$

mit größtmöglichem $k \geq n+1$ gelten. Eine solche Integrationsformel hat $2n$ freie Parameter, die Gewichte A_j und die Integrationspunkte x_j. Für die Newton-Cotes-Formeln und $\omega(x) \equiv 1$ hatten wir

$$I_h(p) = I(p) \text{ mit } k = \begin{cases} n+1, & n \text{ gerade} \\ n, & n \text{ ungerade} \end{cases}$$

Wir werden nun Integrationsformeln konstruieren, so dass

$$G_n(p) = I(p) \quad \forall p \in P_{2n-1}$$

Dies ergibt dann $2n$ Bedingungen für die $2n$ Parameter in G_n.

Satz 9.4.1:
Es gibt keine Integrationsformel G_n, die in P_{2n} exakt ist.

Beweis:
Angenommen es gelte $G_n(p) = I(p)\ \forall p \in P_{2n}$, dann würde dies insbesondere für

$$p(x) := \prod_{j=1}^{n} (x - x_j)^2$$

gelten. Offenbar ist für diese Wahl $G_n(p) = 0$, aber $I(p) \neq 0$, da $p(x) \geq 0$ und nicht identisch verschwindet (vergleiche Eigenschaft 4. der Gewichtsfunktion $\omega(x)$).

□

Wir werden eine Integrationsformel G_n konstruieren, die in P_{2n-1} exakt ist. Dazu benötigen wir Orthogonalpolynome, die mit Hilfe des Schmidtschen Orthogonalisierungsverfahren konstruiert werden.

Satz 9.4.2:
Die Polynome $p_0, \ldots, p_n$ seien für $x \in [a, b]$ rekursiv definiert durch

$$p_0(x) := 1$$
$$p_i(x) := x^i - \sum_{j=0}^{i-1} \frac{\langle x^i, p_j \rangle}{\langle p_j, p_j \rangle} p_j(x), \quad i = 1, \ldots, n$$

mit dem inneren Produkt $\langle f, g \rangle := \int_a^b \omega(x) \cdot f(x) \cdot g(x)\,\mathrm{d}x$. Dann gilt:

1. $p_i \in P_i, \quad i = 0, \ldots, n$
2. $\langle p_i, p_j \rangle = 0, \quad j < i, \quad i, j = 0, \ldots, n$
3. *p_n hat n einfache, reelle Nullstellen, die alle in (a, b) liegen*

Beweis:
1. und 2. per vollständiger Induktion ($\rightsquigarrow$ Literatur)
3. $\rightsquigarrow$ Literatur

□

Lemma 9.4.1:
Es gilt

$$\langle p_n, q \rangle = 0 \quad \forall q \in P_{n-1}$$

Beweis:
Jedes Polynom $q \in P_{n-1}$ kann als Linearkombination von Orthogonalpolynomen $p_0, \ldots, p_{n-1}$ dargestellt werden.

□

Für einige klassische Integrationsprobleme tragen die Orthogonalpolynome bestimmte Namen:

$[a,b]$	$\omega(x)$	Name
$[-1,1]$	1	Legendere-Polynome
$[-1,1]$	$(1-x^2)^{-\frac{1}{2}}$	Tschebyscheff-Polynome
$[0,\infty]$	e^{-x}	Laguerre-Polynome
$[-\infty,\infty]$	e^{-x^2}	Hermite-Polynome

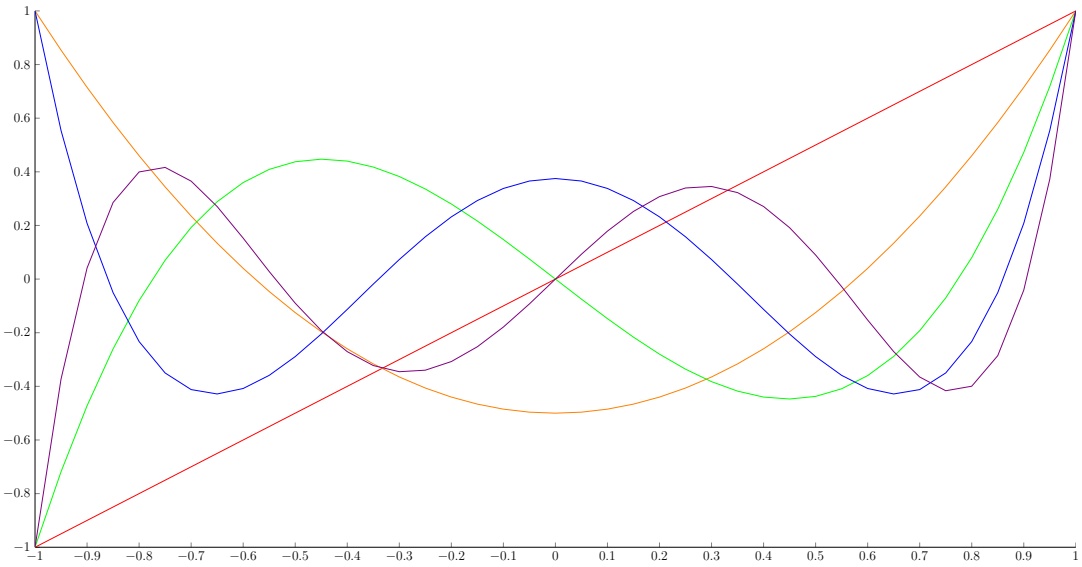


Bild der Legendere-Polynome der Ordnungen 1 bis 5

Man bestimmt A_j nun so, dass G_n exakt ist für P_{2n-1}.

Satz 9.4.3:
Es seien x_j, $j = 1, \ldots, n$ die Nullstellen des Orthogonalpolynoms p_n und

$$A_j = \int_a^b \omega(x) \prod_{\substack{i=1 \\ i \neq j}}^{n} \left(\frac{x - x_i}{x_j - x_i} \right)^2 \mathrm{d}x$$

Dann sind die zugehörigen Integrationsformeln G_n exakt für $p \in P_{2n-1}$. Sind die Nullstellen des Orthogonalpolynoms vorgegeben und die Integrationsformel G_n ist exakt für P_{2n-1}, dann müssen die Gewichte die oben angegebene Gestalt haben.

Beweis:
$\rightsquigarrow$ Literatur

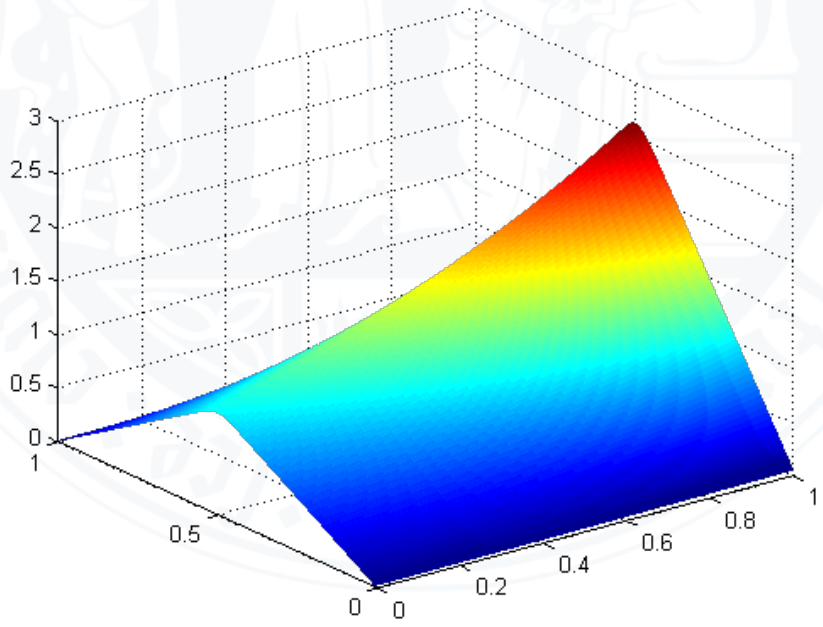
□

Notizen zur Vorlesung

Numerische Mathematik II

im Wintersemester 2012/2013

– Prof. Dr. Axel Klawonn –



Mathematisches Institut
Universität zu Köln

1. Gewöhnliche Differentialgleichungen

09.10.2012
1. Vorlesung

1.1. Einführung

Wachstum einer Bakterienpopulation, die kontinuierlich wächst. Zu einem Zeitpunkt t sei die Anzahl der Bakterien $P(t)$. Nach einem Zeitschritt Δt sei der Zuwachs

$$\Delta P = P(t + \Delta t) - P(t)$$

Wählt man einen Zeitschritt nicht „zu groß“, so erscheint es sinnvoll, dass die neue Größe ΔP proportional zu $P(t)$ und Δt ist, d. h. es gibt eine Konstante $c_w > 0$, so dass $\Delta P \approx c_w P(t)\Delta t$. Dies ist nur sinnvoll, wenn Δt klein genug, da sonst Bakterien nicht berücksichtigt werden, die ständig hinzukommen und selbst zum Wachstum beitragen. Betrachte daher den Grenzübergang $\Delta t \to 0$. Es gilt:

$$\frac{\Delta P}{\Delta t} \approx c_w P(t) \Rightarrow \lim_{\Delta t \to 0} \frac{\Delta P}{\Delta t} = \frac{\mathrm{d}P(t)}{\mathrm{d}t} = P'(t)$$

Annahme zur Beschreibung des Bakterienwachstums $P'(t) = c_w P(t)$. Es handelt sich um eine **gewöhnliche Differentialgleichung**. Zur Lösung dieser gewöhnlichen Differentialgleichung machen wir folgenden heuristischen Ansatz. Betrachte $P'(t) = \frac{\mathrm{d}P(t)}{\mathrm{d}t}$ als Bruch.

$$\frac{\mathrm{d}P}{\mathrm{d}t} = c_w P \text{„}\Leftrightarrow\text{“} \frac{1}{P}\,\mathrm{d}P = c_w\,\mathrm{d}t$$

Unbestimmte Integration ergibt:

$$\int \frac{1}{P}\,\mathrm{d}P = \int c_w\,\mathrm{d}t + c \Rightarrow \ln|P| = c_w\,\mathrm{d}t + c$$

Betrachte P wieder als Funktion von t, so folgt:

$$|P(t)| = e^{c_w t + c} = \tilde{c} \cdot e^{c_w t} \text{ mit } \tilde{c} = e^c$$

Da keine negative Anzahl Bakterien vorkommen wird, gilt:

$$P(t) = \tilde{c} \cdot e^{c_w t}$$

In der Lösung taucht immer noch die Unbekannte Integrationskonstante $\tilde{c}$ auf. Zur Bestimmung benutzt man die Anfangspopulation $P(0) = P_0$.

$$P_0 = P(0) = \tilde{c}$$

Als Lösung erhalten wir: $P(t) = P_0 \cdot e^{c_w t}$.
Dieses Verfahren nennt man **Trennung der Veränderlichen**. Hier heuristisch, lässt sich aber auch mathematisch exakt begründen. Da die Lösung der Differentialgleichung vom Anfangswert der Bakterienpopulation abhängt, nennt man ein solches Problem auch **Anfangswertproblem**. Diese Differentialgleichung ist analytisch lösbar, im Allgemeinen benötigt man jedoch numerische Verfahren.

$u'(t) = f\big(u(t)\big) \wedge u(0) = u_0$ mit einer gegebenen Funktion $f = f(u)$. Ist f nicht linear, lässt sich die Trennung der Veränderlichen nicht mehr direkt anwenden. Wir nehmen an, dass $u \in \mathcal{C}^2(\mathbb{R})$ und $\Delta t > 0$. Taylorentwicklung von u und t ergibt:

$$\begin{aligned} u(t+\Delta t) &= u(t) + \Delta t \cdot u'(t) + \frac{(\Delta t)^2}{2} \cdot u''(t+\xi) \text{ mit } \xi \in (0, \Delta t) \\ \Rightarrow u'(t) &= \frac{u(t+\Delta t) - u(t)}{\Delta t} + \mathcal{O}(\Delta t) \end{aligned} \tag{1.1}$$

Die Idee ist es diese Darstellung zur Konstruktion eines numerischen Verfahrens zu verwenden. Dazu nehmen wir an, dass $\Delta t > 0$ eine feste Zeitschrittweite sei. Wir führen folgende Bezeichnungen ein:

$$t_m := m \cdot \Delta t, \quad v_m \approx u(t_m), \quad m = 0, 1, 2, \ldots$$

Die Werte v_m stellen also eine Approximation der exakten Lösung u im Punkt t_m dar. Dabei soll der Anfangswert $v_0 = u(t_0) = u_0$ exakt dargestellt werden.
Vernachlässigen wir in (1.1) die Terme erster Ordnung in Δt, so gilt:

$$\frac{u(t_{m+1}) - u(t_m)}{\Delta t} \approx u'(t_m) = f\big(u(t_m)\big)$$

Sei nun v_m schon bekannt, dann lässt sich v_{m+1} durch folgende Gleichung definieren:

$$\begin{aligned} \frac{v_{m+1} - v_m}{\Delta t} &= f(v_m) \\ \boxed{v_{m+1} = v_m + \Delta t \cdot f(v_m)} &\text{ mit } m = 0, 1, 2, \ldots \end{aligned}$$

Dieses Verfahren nennt man das **explizite Eulerverfahren** oder auch **Eulersche Polygonzugverfahren**.

11.10.2012
2. Vorlesung

Frage: Konvergiert das explizite Eulerverfahren gegen die analytische Lösung, wenn die Zeitschrittweite $\Delta t \to 0$?

Wir beschränken uns auf endliche Intervalle $t \in [0, t^*]$. Ohne Einschränkung sei $\frac{t^*}{\Delta t} \in \mathbb{N}$. Der Fehler e_m in jedem Zeitschritt t_m ist gegeben durch

$$\begin{aligned} e_m &:= v_m - u_m \\ u_m &:= u(t_m) \end{aligned}$$

Unter Konvergenz des Verfahrens verstehen wir:

$$\lim_{\Delta t \to 0} \max_{m=0,\ldots,\frac{t^*}{\Delta t}} |e_m| = 0$$

Aus den Gleichungen:

$$\begin{array}{rll} & v_{m+1} & = v_m + \Delta t \cdot f(v_m) \\ - & u_{m+1} & = u_m + \Delta t \cdot f(u_m) + \mathcal{O}\big((\Delta t)^2\big) \\ \hline & e_{m+1} & = e_m + \Delta t\big(f(v_m) - f(u_m)\big) + \mathcal{O}\big((\Delta t)^2\big) \end{array}$$

Annahme: f sei lipschitzstetig bezüglich $|\cdot|$ mit einer Lipschitzkonstanten $L > 0$.

$$\begin{aligned}\Rightarrow |e_{m+1}| &\leq |e_m| + \Delta t \underbrace{|f(v_m) - f(u_m)|}_{\leq L \cdot |v_m - u_m|} + \mathcal{O}\left((\Delta t)^2\right) \\ &\leq (1 + \Delta t \cdot L) \cdot |e_m| + \mathcal{O}\left((\Delta t)^2\right)\end{aligned}$$

Behauptung:
$e_m \leq \frac{c}{L}\Delta t\Big((1 + \Delta t \cdot L)^m - 1\Big)$ mit $m = 0, 1, \ldots, \frac{t^*}{\Delta t}$

Beweis (per vollständiger Induktion):
$m = 0$: $|e_0| = |v_0 - u_0| = 0$ ✓
$m \mapsto m + 1$:

$$\begin{aligned}|e_{m+1}| &\leq (1 + \Delta t)|e_m| + c \cdot (\Delta t)^2 \\ &\overset{\text{I.V.}}{\leq} (1 + \Delta t \cdot L)\left(\frac{c}{L}\Delta t\Big((1 + \Delta t \cdot L)^m - 1\Big)\right) + c \cdot (\Delta t)^2 \\ &= \frac{c}{L}\Delta t\left((1 + \Delta t \cdot L)^{m+1} - 1 \underbrace{+\Delta t L - \Delta t L}_{=0}\right) \\ &= \frac{c}{L}\Delta t\Big((1 + \Delta t \cdot L)^{m+1} - 1\Big) \quad \checkmark\end{aligned}$$

Da $\Delta t \cdot L > 0$, folgt: $1 + \Delta t \cdot L < e^{\Delta t L} \Rightarrow (1 + \Delta t L)^m < e^{m\Delta t \cdot L}$ mit $m = 0, 1, \ldots$.
Betrachten wir nun $t \in [0, t^*]$, so gilt für $m = 0, \ldots, \frac{t^*}{\Delta t}$:

$$\begin{aligned}|e_m| &\leq \frac{c}{L}\Delta t\left(e^{m\Delta t \cdot L} - 1\right) \\ &\leq \underbrace{\left(\frac{c}{L} \cdot (e^{t^* \cdot L} - 1)\right)\Delta t}_{\lim\limits_{\Delta t \to 0} \to 0}\end{aligned}$$

□

Satz 1.1.1:
Das explizite Eulerverfahren ist konvergent mit der Konvergenzrate $\mathcal{O}(\Delta t)$, wobei Δt die Zeitschrittweite ist.

1.2. Theoretische Grundlagen

$$\begin{cases} y' = f(x, y), \quad x \in I, \text{ mit } I \subset \mathbb{R} \text{ Intervall} \\ y(x_0) = y_0 \end{cases}$$

Allgemein betrachten wir im Folgenden ein System gewöhnlicher Differentialgleichungen erster Ordnung.

$$\begin{aligned} y_1' &= f_1\Big(x, y_1(x), y_2(x), \ldots, y_n(x)\Big) \\ y_2' &= f_2\Big(x, y_1(x), y_2(x), \ldots, y_n(x)\Big) \\ &\vdots \qquad\qquad\qquad \vdots \\ y_n' &= f_n\Big(x, y_1(x), y_2(x), \ldots, y_n(x)\Big) \end{aligned}$$

Gesucht sind dabei n reelle Funktionen $y_i(x),\ i = 1,\dots,n$ einer reellen Variablen $x \in I \subset \mathbb{R}$.

Kompakte Form: $\boxed{y' = f(x,y)}$ mit $y' := \begin{pmatrix} y_1' \\ \vdots \\ y_n' \end{pmatrix},\ f(x,y) = \begin{pmatrix} f_1(x,y_1,y_2,\dots,y_n) \\ \vdots \\ f_n(x,y_1,y_2,\dots,y_n) \end{pmatrix}$

Anfangswertproblem:
$y' = f(x,y)$ mit den Anfangswerten $y(x_0) = y_0 = \begin{pmatrix} y_{1,0} & y_{2,0} & \cdots & y_{n,0} \end{pmatrix}^\mathsf{T}$

Es ist ausreichend Existenz- und Eindeutigkeitsaussagen für Systeme erster Ordnung zu zeigen (und auch numerische Verfahren hierfür zu entwickeln), da man Differentialgleichungen höherer Ordnung hierauf zurückführen kann.

Bertachte: $g^{(m)} = f\left(x, y(x), y^{(1)}(x), y^{(2)}(x), \dots, y^{(m-1)}(x)\right),\ m \geq 1$

Dazu führen wir folgende Hilfsfunktionen ein:
$z_1(x) := y(x), \quad z_2(x) := y^{(1)}(x), \quad z_3(x) := y^{(2)}(x), \quad \dots, \quad z_m(x) := y^{(m-1)}(x)$

$$z' = \begin{pmatrix} z_1' \\ z_2' \\ \vdots \\ z_{m-1}' \\ z_m' \end{pmatrix} = \begin{pmatrix} z_2 \\ z_3 \\ \vdots \\ z_m \\ f(x,z_1,z_2,\dots,z_m) \end{pmatrix}$$

Mit der Transformation können wir gewöhnliche Differentialgleichungen m-ter Ordnung der Form $\begin{cases} y^{(m)} = f\left(x, y(x), y^{(1)}(x), \dots, y^{(m-1)}(x)\right) \\ y^{(i)}(x_0) = y_{i,0}, \qquad\qquad\qquad\qquad i = 0,\dots,m-1 \end{cases}$ immer als System erster Ordnung behandeln.

Satz 1.2.1 (Existenz und Eindeutigkeit):
Sei $S := \{(x,y)\colon a \leq x \leq b,\ y \in \mathbb{R}^n\}$ mit $-\infty < a \leq b < \infty$,
$f\colon S \to \mathbb{R}^n,\ f \in \mathcal{C}(S)$ weiterhin sei f lipschitzstetig bezüglich y, d. h. es existiert eine positive Konstante L, so dass

$$\|f(x,y_1) - f(x,y_2)\| \leq L \cdot \|y_1 - y_2\| \quad \forall (x,y_i) \in S, \quad i = 1,2$$

Dann existiert zu jedem $x_0 \in [a,b]$ und jedem $y_0 \in \mathbb{R}^n$ genau eine für $x \in [a,b]$ definierte Funktion $y(x)$ mit:

1. $y(x) \in \mathcal{C}^1([a,b])$
2. $y'(x) = f\Big(x, y(x)\Big)\ \forall x \in [a,b]$
3. $y(x_0) = y_0$

Satz 1.2.2 (stetige Abhängigkeit von den Anfangswerten):
Sei $S := \{(x,y)\colon a \leq x \leq b,\ y \in \mathbb{R}^n\}$ mit $-\infty < a \leq b < \infty$,
$f\colon S \to \mathbb{R}^n,\ f \in \mathcal{C}(S)$ weiterhin sei f lipschitzstetig bezüglich y mit Lipschitzkonstante $L > 0$. Weiterhin sei $a \leq x_0 \leq b$. Dann gilt für die Lösung $y(x,S)$ des Anfangswertproblem

$$\begin{cases} y' = f(x,y), & x \in [a,b] \\ y(x_0,s) = S \end{cases}$$

die Abschätzung $\|y(x,s_1) - y(x,s_2)\| \leq e^{L\cdot|x-x_0|} \cdot \|s_1 - s_2\|$.

Beweis:
Nach Definition von $y(x,s)$ gilt nach Hauptsatz der Differential- und Integralrechnung

$$\begin{aligned} y(x,s) &= y(x_0,s) + \int_{x_0}^{x} y'(t,s)\,\mathrm{d}t \\ &= y(x_0,s) + \int_{x_0}^{x} f\Big(t, y(t,s)\Big)\,\mathrm{d}t \end{aligned}$$

Sei $x \in [a,b]$, dann $y(x,s_1) - y(x,s_2) = (s_1 - s_2) + \int_{x_0}^{x} f\Big(t, y(t,s_1)\Big) - f\Big(t, y(t,s_2)\Big)\,\mathrm{d}t$

$$\begin{aligned} \overset{\Delta\text{-Ungl.}}{\Rightarrow} \|y(x,s_1) - y(x,s_2)\| &\leq \|s_1 - s_2\| + \left\| \int_{x_0}^{x} f\Big(t, y(t,s_1)\Big) - f\Big(t, y(t,s_2)\Big)\,\mathrm{d}t \right\| \\ &\overset{\substack{\Delta\text{-Ungl.} \\ \Rightarrow \\ \text{Lipschitz-} \\ \text{stetigkeit} \\ \text{von } f}}{\leq} \|s_1 - s_2\| + \int_{x_0}^{x} \Big\| \underbrace{f\Big(t, y(t,s_1)\Big) - f\Big(t, y(t,s_2)\Big)}_{\leq L \cdot \|y(t,s_1) - y(t,s_2)\|} \Big\|\,\mathrm{d}t \\ &\leq \|s_1 - s_2\| + L \cdot \int_{x_0}^{x} \Big\| y(t,s_1) - y(t,s_2) \Big\|\,\mathrm{d}t \end{aligned}$$

Definiere $\Phi(x) := \int_{x_0}^{x} \Big\| y(t,s_1) - y(t,s_2) \Big\|\,\mathrm{d}t$.
Es gibt: $\Phi'(x) = \Big\| y(x,s_1) - y(x,s_2) \Big\|$. Für $x \geq x_0$ folgt sofort

$$\underbrace{\Phi'(x) - L \cdot \Phi(x)}_{=:\alpha(x)} \leq \|s_1 - s_2\|$$

Betrachte folgendes Anfangswertproblem:

$$\begin{cases} \Phi'(x) = \alpha(x) + L \cdot \Phi(x) \\ \Phi(x_0) = 0 \end{cases}$$

Für $x \geq x_0$ hat dieses Anfangswertproblem die Lösung: $\Phi(x) = e^{L\cdot(x-x_0)} \cdot \int_{x_0}^{x} \alpha(t) e^{-L(t-x_0)}\,\mathrm{d}t$.
Die Lösung erhält man durch Variation der Konstanten. Man kann aber auch einfach nachrechnen, dass $\Phi(x)$ eine Lösung ist. Es gilt:

$$\begin{aligned} 0 &\leq \Phi(x) \\ &\overset{\alpha(x) \leq \|s_1 - s_2\|}{\leq} e^{L(x-x_0)} \|s_1 - s_2\| \underbrace{\int_{x_0}^{x} e^{-L(t-x_0)}\,\mathrm{d}t}_{= \left[-\frac{1}{L} e^{-L(t-x_0)}\right]_{x_0}^{x}} \\ &= \frac{1}{L} \|s_1 - s_2\| \left(e^{L(x-x_0)} - 1 \right) \end{aligned}$$

Mit diesen Abschätzungen erhalten wir:

$$\begin{aligned} \Big\| y(x,s_1) - y(x,s_2) \Big\| &= \Phi'(x) \\ &= \alpha(x) + L \cdot \Phi(x) \\ &\leq \|s_1 - s_2\| + \|s_1 - s_2\| \cdot \left(e^{L \overbrace{|x - x_0|}^{x \geq x_0}} - 1 \right) \\ &= e^{L|x-x_0|} \|s_1 - s_2\| \end{aligned}$$

Für $x < x_0$ kann man analog vorgehen.

□

16.10.2012
3. Vorlesung

Satz 1.2.3 (Grönwall-Lemma):
Die reelle Funktion $\Phi(t)$ sei stetig in dem Intervall $J = [0, a]$ und es sei $\Phi(t) \leq \alpha + \beta \cdot \int_0^t \Phi(\tau)\,\mathrm{d}\tau$ in J mit $\beta > 0$. Dann gilt:

$$\Phi(t) \leq \alpha e^{\beta t} \in J$$

Satz 1.2.4 (verallgemeinertes Grönwall-Lemma):
Die reelle Funktion $\Phi(t)$ sei stetig in dem Intervall $J = [0, a]$ und es gelte $\Phi(t) \leq \alpha + \int_0^t h(s) \cdot \Phi(s)\,\mathrm{d}s$ in J wobei $\alpha \in \mathbb{R}$ und $h(t) \geq 0$ und stetig in J sei. Dann gilt:

$$\Phi(t) \leq \alpha \cdot e^{H(t)} \text{ mit } H(t) = \int_0^t h(s)\,\mathrm{d}s$$

1.3. Numerische Behandlung von Anfangswertaufgaben

1.3.1. Allgemeine Einschrittverfahren

Wir folgen hier der Darstellung in [11, Kapitel 3]

Betrachte folgendes Anfangswertproblem:

$$\begin{cases} y' = f\big(t, y(t)\big), & t \in I := [t_0, t_0 + T] \\ y(t_0) = y_0 \end{cases} \tag{1.2}$$

Wiederholung: Hier für das explizite Eulerverfahren
Für y' gilt näherungsweise:

$$\frac{y(t+h) - y(t)}{h} \approx y'(t) = f\big(t, y(t)\big)$$

d. h. $y(t + h) \approx y(t) + h \cdot f\big(t, y(t)\big)$. Wir unterteilen das Intervall $I = [t_0, t_0 + T]$ in Teilintervalle $[t_i, t_{i+1}]$, $i = 0, \ldots, n-1$, $n \in \mathbb{N}$, indem wir ein Gitter I^h einführen:

$$I^h = \{t_0, t_1, t_2, \ldots, t_n\}$$

Der Einfachheit halber nehmen wir zunächst eine äquidistante Unterteilung:

$$t_i = t_0 + i \cdot h, \quad i = 0, \ldots, n, \quad h = \frac{T}{n}$$

Gesucht sind die approximativen Werte von y in den Gitterpunkten t_i. Wir bezeichnen diese Werte mit $u_i := u_h(t_i)$, $i = 1, \ldots, n$ wobei $u_h : I^h \to \mathbb{R}^n$ eine **Gitterfunktion** ist.

Explizites Eulerverfahren

1. $u_0 = y_0 = y(t_0)$ exakt
2. $\frac{u_{i+1} - u_i}{h} = f(t_i, u_i) \quad i = 0, \ldots, n-1$
   $\Leftrightarrow u_{i+1} = u_i + h \cdot f(t_i, u_i) \quad i = 0, \ldots, n-1$

Das explizite Eulerverfahren ist ein Spezialfall eines **allgemeinen Einschrittverfahren** der Form:

$$\begin{cases} u_0 = y_0 \\ u_{i+1} = u_i + h_i \cdot \Phi(t_i, u_i, h_i), \quad i = 0, \dots, n-1 \end{cases} \tag{1.3}$$

Die Funktion Φ heißt **Inkrementfunktion** und beschreibt das jeweilige Einschrittverfahren.
Explizites Eulerverfahren: $\Phi(t_i, u_i, h_i) = f(t_i, u_i)$.
Da sich u_{i+1} explizit, d. h. ohne lösen einer weiteren Gleichung, aus u_i berechnen lässt, nennt man solche Verfahren **explizit** (im Gegenteil zu **impliziten Verfahren**).

Bemerkung 1.3.1:
Explizite Einschrittverfahren lassen sich durch komponentenweise Betrachtung direkt auf den vektoriellen Fall übertragen.

$$\begin{aligned} \vec{u}_{i+1} &= \begin{pmatrix} u_1 \\ \vdots \\ u_n \end{pmatrix}_{i+1} = \begin{pmatrix} u_1^{(i+1)} \\ \vdots \\ u_n^{(i+1)} \end{pmatrix} \\ \Rightarrow \vec{u}_{i+1} &= \begin{pmatrix} u_1 \\ \vdots \\ u_n \end{pmatrix}_{i+1} + h_i \cdot \begin{pmatrix} \Phi_1(t_i, \vec{u}_i, h_i) \\ \vdots \\ \Phi_n(t_i, \vec{u}_i, h_i) \end{pmatrix} \\ \vec{u}_{i+1} &= \vec{u}_i + h_i \cdot \vec{\Phi}_1(t_i, \vec{u}_i, h_i) \end{aligned}$$

Fragen:

1. Konvergiert ein solches Einschrittverfahren und wenn ja unter welchen Bedingungen?
2. Wie gut ist die Approximation?

Gesucht ist der **globale Fehler**:

$$\begin{aligned} e_h(t_i) &= y(t_i) - u(t_i) \\ &= y_i - u_i =: e_i \end{aligned}$$

Sei zunächst wieder $h = konstant$, dann gilt für u_h beim Einschrittverfahren nach Konstruktion $\frac{u_h(t+h)-u_h(t)}{h} - \Phi\Big(t, u_h(t), h\Big) = 0$. Ersetze u_h durch y, dann:

$$\tau_h(t, h) := \frac{y(t+h) - y(t)}{h} - \Phi(t, y, h)$$

τ_h heißt **lokaler Diskretisierungsfehler** in $\Big(t, y(t)\Big)$. Der Wert $\tau(t, h)$ ist der Fehler, der bei genau einem Schritt des Einschrittverfahrens mit exaktem Startwert entsteht. Da wir in der Praxis nicht in jedem Schritt den exakten Startwert haben, ist dies ein lokaler Fehler. Für das explizite Eulerverfahren gilt

$$\tau(t, h) = \frac{y(t+h) - y(t)}{h} - \underbrace{f\Big(t, y(t)\Big)}_{=y'} = \mathcal{O}(h)$$

Also ist der lokale Diskretisierungsfehler hier $\mathcal{O}(h)$. Eine vernünftige Folgerung für Konvergenz ist:

$$\lim_{h \to 0} \tau(t, h) = 0$$

Definition 1.3.1 (Konsistenz):
Das explizite Einschrittverfahren (1.3) *zur Lösung des Anfangswertproblem* (1.2) *heißt* ***konsistent****, wenn für alle* $t \in [t_0, t_0 + T]$ *bei genügend oft differenzierbaren Funktionen* f *gilt, dass*

$$\lim_{h\to 0} \tau(t,h) = \lim_{h\to 0} \left(\frac{y(t+h) - y(t)}{h} - \Phi\Big(t, y(t), h\Big) \right) = 0$$

Das Verfahren heißt ***konsistent der Ordnung*** $p \in \mathbb{N}$*, falls* $\tau(t,h) = \mathcal{O}(h^p)$.

Bemerkung 1.3.2:
Konsistenz bedeutet also insbesondere $y'(t) = f\Big(t, y(t)\Big) = \lim\limits_{\substack{h\to o \\ h\neq 0}} \Phi\Big(t, y(t), h\Big)$.

Satz 1.3.1:
Das explizite Eulerverfahren ist konsistent von der Ordnung 1.

Definition 1.3.2 (Konvergenz):
Das explizite Einschrittverfahren (1.3) *zur Lösung des Anfangswertproblem* (1.2) *heißt* ***konvergent****, falls für genügend glattes* f *gilt:*

$$\lim_{h\to 0} \max_{i=0,\dots,n} |e_h(t_i)| = \lim_{h\to 0} \max_{i=0,\dots,n} |y(t_i) - u_h(t_i)| = 0$$

Das Verfahren heißt ***konvergent von der Ordnung*** $p \in \mathbb{N}$*, falls* $\max\limits_{i=0,\dots,n} |e_h(t_i)| = \mathcal{O}(h^p)$ *gilt.*

Lemma 1.3.1 (diskretes Grönwall-Lemma):
Für $j = 0, 1, \dots, m-1$ *seien* $\eta_j, \rho_j, z_j \geq 0$ *und* z_j *genüge der Ungleichung*

$$z_{j+1} \leq (1 + \rho_j) z_j + \eta_j$$

Dann gilt:

$$z_j \leq \left(z_0 + \sum_{i=0}^{j-1} \eta_i \right) \cdot e^{\sum_{i=0}^{j-1} \rho_i} \quad \forall j = 0, 1, \dots, m$$

Beweis (per vollständiger Induktion):
$j = 0$:

$$\begin{aligned} z_0 &= (z_0 + 0) \cdot e^0 \\ &= \left(z_0 + \sum_{i=0}^{-1} \eta_i \right) \cdot e^{\sum_{i=0}^{-1} \rho_i} \quad \checkmark \end{aligned}$$

$j \mapsto j + 1$:
Nach Voraussetzung gilt:

$$\begin{aligned} z_{j+1} &\leq (1 + \rho_j) z_j + \eta_j \\ &\overset{\text{I.V.}}{\leq} (1 + \rho_j) \left(z_0 + \sum_{i=0}^{j-1} \eta_i \right) \cdot e^{\sum_{i=0}^{j-1} \rho_i} + \eta_j \end{aligned}$$

Aus $1 + x \leq e^x$ folgt $1 + \rho_j \leq e^{\rho_j}$

$$\begin{aligned} \Rightarrow z_{j+1} &\leq \left(z_0 + \sum_{i=0}^{j-1} \eta_i \right) \cdot e^{\sum_{i=0}^{j} \rho_i} + \eta_j \\ &\overset{\eta_j \geq 0}{\leq} \left(z_0 + \sum_{i=0}^{j} \eta_i \right) \cdot e^{\sum_{i=0}^{j} \rho_i} \end{aligned}$$

□

Satz 1.3.2 (Konvergenz expliziter Einschrittverfahren):
Wir betrachten ein explizites Einschrittverfahren der Form (1.3) zur Lösung des Anfangswertproblem (1.2). Gegeben seien die Schrittweiten h_i, $i = 0, \ldots, n-1$ und wir setzen

$$h := \max_{i=0,\ldots,n-1} h_i$$

Die Inkrementfunktion $\Phi(t, y, h)$ erfülle bezüglich der zweiten Komponente die Lipschitzbedingung $\left|\Phi(t, y_1, h) - \Phi(t, y_2, h)\right| \leq L \cdot |y_1 - y_2|$ mit der Lipschitzkonstanten $L > 0$, hierbei sei $|\cdot|$ eine passende Norm. Dann ist ein explizites Einschrittverfahren konvergent von der Ordnung p, wenn es konsistent von der Ordnung p ist. Ist

$$\tau_h := \max_{t \in [t_0, t_0+T]} \left|\tau_h(t, h)\right|$$

der maximale Diskretisierungsfehler, dann gilt für den globalen Fehler $e_h(t_i)$ im Punkt t_i die Abschätzung:

$$\begin{aligned} e_h(t_i) &= \left|y(t_i) - y_h(t_i)\right| \\ &\leq \left(\left|y(t_0) - u_h(t_0)\right| + (t_i - t_0)\tau_h \right) \cdot e^{L \cdot (t_i - t_0)} \end{aligned}$$

18.10.2012
4. Vorlesung

Beweis:
Es gilt:

$$\begin{array}{lll} & y(t_{j+1}) & = y(t_0) + h_j \Phi\Big(t_j, y(t_j), h_j\Big) + h_j \tau(t_j, h_j) \\ - & u_{j+1} & = u_j + h_j \Phi(t_j, u_j, h_j) \\ \hline & e_{j+1} & = \underbrace{(y(t_j) - u_j)}_{e_j} + h_j \Big(\Phi(t_j, y(t_j), h_j) - \Phi(t_j, u_j, h_j) \Big) + h_j \tau(t_j, h_j) \end{array}$$

Mit $\tau_h := \max\limits_{t \in [t_0, t_0+T]} \left|\tau(t, h)\right|$ ergibt sich

$$|e_{j+1}| \overset{\text{Lipschitzbedingung}}{\underset{\text{für } \Phi}{\leq}} |e_j| + h_j L |e_j| + h_j \tau_j = (1 + \underbrace{h_j L}_{=:\rho_j}) \underbrace{|e_j|}_{=:z_j} + h_j \underbrace{\tau_h}_{=:\eta_j}$$

$$\begin{aligned} \overset{\text{Lemma 1.3.1}}{\Rightarrow} \quad |e_j| &\leq \left(|e_0| + \sum_{i=0}^{j-1} h_i \tau_h \right) e^{\sum_{i=0}^{j-1} h_i L} \\ &\overset{h_i = t_{i+1} - t_i}{=} \Big(|y(t_0) - u_h(t_0)| + |t_j - t_0| \tau_h \Big) e^{L(t_j - t_0)} \end{aligned}$$

□

Für ein explizites Einschrittverfahren mit Φ lipschitzstetig bezüglich der zweiten Komponente gilt also:

$$\text{Konsistenz} \Rightarrow \text{Konvergenz}$$

Bestimmen der Konsistenzordnung eines Einschrittverfahren

Wir betrachten (1.2), dabei sei $f\colon G \subset \mathbb{R}^2 \to \mathbb{R}$ hinreichend oft differenzierbar. Mit $y^{(i+1)}$ bezeichnen wir die $(i+1)$-te Ableitung der Funktion $y(x)$. Offensichtlich gilt:

$$\begin{aligned} y^{(i+1)}(x) &= \frac{\partial^{(i+1)} y(x)}{\partial x^{i+1}} \\ &= \frac{\partial^i}{\partial x^i} y'(x) \\ &= \frac{\partial^i}{\partial x^i} f\Big(x, y(x)\Big) \end{aligned} \tag{1.4}$$

Somit haben wir

$i = 1$: $y''(x) = \frac{\partial}{\partial x} f\Big(x, y(x)\Big) = f_x + f_y f$ mit $f_x = f_x\Big(x, y(x)\Big) = \frac{\partial f}{\partial x}\Big(x, y(x)\Big)$
$i = 2$: $y'''(x) = f_{xx} + f_{xy}y' + (f_x + f_y y')f_y + f(f_{xy} + f_{yy}y') = f_{xx} + 2f_{xy}f + f^2 f_{yy} + (f_x + f f_y)f_y$

Sei $f \in \mathcal{C}^p(G), \quad p \in \mathbb{N}$, so ist $y \in \mathcal{C}^{p+1}$ und besitzt die Taylorentwicklung:

$$y(x+h) = y(x) + hy'(x) + \cdots + \frac{h^p}{p!} y^{(p)}(x) + \mathcal{O}\left(h^{p+1}\right)$$

Zur Berechnung der Konsistenzordnung eines gegebenen Einschrittverfahren betrachten wir die Taylorentwicklung der zugehörigen Inkrementfunktion $\Phi(t, y, h, f)$ und berechnen den lokalen Diskretisierungsfehler $\tau_h(t, f)$ unter Anwendung von (1.4).
Taylorentwicklung in zwei Veränderlichen:

$$\begin{aligned} f(x+h, y+k) = f(x,y) &+ \left[h\frac{\partial f}{\partial x}(x,y) + k\frac{\partial f}{\partial y}(x,y)\right] \\ &+ \left[\frac{h^2}{2}\frac{\partial^2 f}{\partial x^2}(x,y) + hk\frac{\partial^2 f}{\partial x \partial y}(x,y) + \frac{k^2}{2}\frac{\partial^2 f}{\partial y^2}(x,y)\right] \\ &+ \dots \\ &+ \frac{1}{n!}\sum_{j=0}^{n}\binom{n}{j} h^{n-j}k^j \frac{\partial^n f(x,y)}{\partial x^{n-j}\partial y^j} \\ &+ \mathcal{O}\left(h^{n+1} + k^{n+1}\right) \end{aligned}$$

Methode von Heun

$y_{k+1} = y_k + h \cdot \Phi(x_k, y_k, h, f)$ mit $\Phi(x, y, h, f) := \frac{1}{2}\Big\{f(x,y) + f\Big(x + h, y + hf(x,y)\Big)\Big\}$
Bestimmung der Konsistenzordnung der **Methode von Heun**:
Aus der Taylorentwicklung von $f(x, y)$ erhalten wir:

$$f\Big(x + h, y + \underbrace{hf(x,y)}_{=k}\Big) = f + [hf_x + hff_y] + \left[\frac{h^2}{2}f_{xx} + h^2 f f_{xy} + \frac{h^2}{2}f^2 f_{yy}\right] + \mathcal{O}\left(h^3\right)$$

$$\Rightarrow \Phi(x,y,h,f) \overset{\text{Definition von } \Phi}{=} f + \frac{h}{2}(f_x + f f_y) + \frac{h^2}{4}(f_{xx} + 2f f_{xy} + f^2 f_{yy}) + \mathcal{O}\left(h^3\right)$$

Die Taylorentwicklung von $y = y(x)$ ergibt:

$$\begin{aligned}
y(x+h) &= y(x) + hy'(x) + \frac{h^2}{2}y''(x) + \frac{h^3}{6}y^{(3)}(x) + \mathcal{O}\left(h^4\right) \\
&\overset{(1.4)}{=} y(x) + hf(x,y) + \frac{h^2}{2}(f_x + f_y f) + \frac{h^3}{6}\Big((f_{xx} + 2ff_{xy} + f^2 f_{yy}) + (f_x + ff_y)f_y\Big) + \mathcal{O}\left(h^4\right) \\
\Rightarrow \tau_h(x) &= \frac{y(x+h) - y(x)}{h} - \Phi(x,y,h,f) \\
&= \left[\cancel{f} + \cancel{\frac{h}{2}(f_x + f_y f)} + \frac{h^2}{6}((f_{xx} + 2ff_{xy} + f^2 f_{yy}) + (f_x + ff_y)f_y)\right] \\
&\quad - \left[\cancel{f} + \cancel{\frac{h}{2}(f_x + ff_y)} + \frac{h^2}{4}(f_{xx} + 2ff_{xy} + f^2 f_{yy})\right] + \mathcal{O}\left(h^3\right) \\
&= \mathcal{O}\left(h^2\right)
\end{aligned}$$

Die Methode von Heun hat die Konsistenzordnung zwei und damit auch die Konvergenzordnung zwei.

Konsistenzordnung expliziter Einschrittverfahren

Stufe	Name	Ordnung
$m = 1$	explizites Eulerverfahren	$\mathcal{O}(h)$
$m = 2$	Methode von Heun	$\mathcal{O}(h^2)$
$m = 3$	einfache Kutta-Regel	$\mathcal{O}(h^3)$
$m = 3$	(Standard) Runge-Kutta-Verfahren	$\mathcal{O}(h^4)$

1.3.2. Verfahren höherer Ordnung: Runge-Kutta-Verfahren

23.10.2012
5. Vorlesung

Ziel: Verfahren höherer Ordnung mittels Quadraturformel herleiten.
Betrachte das Anfangswertproblem (1.2). Sei $f\colon D \subset \mathbb{R}^{n+1} \to \mathbb{R}^n$ stetig und $I^h := \{t_0, t_1, \ldots, t_n = t_0 + T\}$ ein Gitter mit $t_i < t_j,\ i \neq j$.
Integration von (1.2):

$$y(t) = y(t_0) + \int_{t_0}^{t} \underbrace{f\Big(s, y(s)\Big)}_{=y'} \,\mathrm{d}s$$

Hieraus ergibt sich für $[t_j, t_{j+1}] \subset I$

$$y(t_{j+1}) = y(t_j) + \int_{t_j}^{t_{j+1}} f\Big(s, y(s)\Big)\,\mathrm{d}s$$

Im Folgenden sei $y(t_{j+1}) := y_{j+1}$ und $y(t_j) := y_j$.

Idee: Approximiere $\int_{t_j}^{t_{j+1}} f\Big(s, y(s)\Big)\,\mathrm{d}s$ durch Quadraturverfahren der Form

$$h_j \cdot \sum_{l=0}^{m} \gamma_l f\Big(s_l, y(s_l)\Big), \quad s_l \in [t_j, t_{j+1}]$$

Da $y(s_l)$ im Allgemeinen nicht bekannt ist, benötigen wir eine Näherung von $f\Big(s_l, y(s_l)\Big)$.

Je nach Wahl der Näherung erhalten wir verschiedene Verfahren:

1. **Trapezregel:** $j = 0,\ t_1 = t_0 + h$

$$\int_{t_0}^{t_0+h} f\Big(s, y(s)\Big)\,\mathrm{d}s = \frac{h}{2}\Big\{f(t_0, y_0) + f\Big(t_0 + h, y(t_0 + h)\Big)\Big\} + \mathcal{O}\left(h^3\right)$$

$\Rightarrow$ Einschrittverfahren: $\boxed{u_{j+1}} = u_j + \frac{h}{2}f(t_j, u_j) + \frac{h}{2}f(t_{j+1}, \boxed{u_{j+1}})$ also im Allgemeinen ein **implizites** Verfahren. Für den lokalen Diskretisierungsfehler gilt:

$$\begin{aligned}\tau(t,h) &= \frac{y(t,h) - y(t)}{h} - \frac{f(t, y(t)) + f(t+h, y(t+h))}{2}\\ &= \cancel{\frac{1}{h}\left(\int_t^{t+h} f(s, y(s))\,\mathrm{d}s\right)} \cancel{-\frac{1}{h}\left(\int_t^{t+h} f(s, y(s))\,\mathrm{d}s\right)} + \mathcal{O}(h^2)\\ &= \mathcal{O}(h^2)\\ &\Rightarrow \text{Konsistenzordnung } p = 2\end{aligned}$$

2. **Mittelpunktsregel:** $j = 0,\ t_1 = t_0 + h$

$$\int_{t_0}^{t_0+h} f\Big(s, y(s)\Big)\,\mathrm{d}s = h \cdot f\left(t_0 + \frac{h}{2}, y\left(t_0 + \frac{h}{2}\right)\right) + \mathcal{O}\left(h^3\right)$$

Hieraus ergibt sich das Verfahren:

$$u_{j+2} = u_j + 2hf(t_{j+1}, u_{j+1})$$

Beispiel für ein explizites **Mehrschrittverfahren** (**Hier:** Zweischrittverfahren)

Man kann den fehlenden Wert für y auch wie folgt approximieren

$$y(t_0 + h) \approx y(t_0) + hf\Big(t_0, y(t_0)\Big) \text{ (explizites Eulerverfahren)}$$

Einsetzen in die Trapezregel ergibt:

$$\int_{t_0}^{t_0+h} f\Big(s, y(s)\Big)\,\mathrm{d}s \approx \frac{h}{2}\Big\{f(t_0, y_0) + f\Big(t_0 + h, y_0 + hf(t_0, y_0)\Big)\Big\}$$

$\Rightarrow$ Verfahren von Heun

$$\begin{aligned}u_0 &= y_0\\ u_{k+1} &= u_k + h_k \underbrace{\frac{1}{2}\Big(f(t_k, u_k) + f\Big(t_k + h_k, u_k + h_k f(t_k, u_k)\Big)\Big)}_{=:\Phi(t_k, u_k, h_k)}\end{aligned}$$

Mittelpunktsregel (analog):

$$\Phi(t, y, h) = f\left(t + \frac{h}{2}, y + \frac{h}{2}f(t, y)\right)$$

Das so erhaltene Verfahren bezeichnet man auch als **modifiziertes Eulerverfahren** (von Collatz). Beide Verfahren benötigen zwei Funktionsauswertungen von f und haben die Konvergenzordnung $p = 2$.

Runge-Kutta-Verfahren

Im Ansatz $y_{j+1} = y_j + \int_{t_j}^{t_{j+1}} f\Big(s, y(s)\Big)\,\mathrm{d}s$ sollen die Integrale systematisch durch Quadraturformeln approximiert werden, d. h.

$$\int_{t_j}^{t_{j+1}} f\Big(s, y(s)\Big)\,\mathrm{d}s \approx h_j \sum_{l=1}^{m} \gamma_l f\Big(s_l, y(s_l)\Big), \quad s_l \in [t_j, t_{j+1}]$$

Der systematische Ansatz zur Approximation der Werte $f\Big(s_l, y(s_l)\Big)$ führt auf die Klasse der Runge-Kutta-Verfahren.

Bezeichnung: $k_l \approx f\Big(s_l, y(s_l)\Big),\ l = 1, \ldots, m$ zunächst betrachten wir nur explizite Runge-Kutta-Verfahren.

Explizite Runge-Kutta-Verfahren

$y(s_l)$ wird mit Hilfe von $k_i,\ i = 1, \ldots, l-1$ berechnet. Sei

$$\begin{aligned} y(s_l) &\approx y_j + h_j(\beta_{l,1}k_1 + \cdots + \beta_{l,l-1}k_{l-1}) \\ &= y_j + h_j \sum_{i=1}^{l-1} \beta_{l,i}k_i \end{aligned}$$

Weiterhin sei $s_1 := t_j$ und $s_l := t_j + \alpha_l h_j$, wobei $\alpha_l := \sum_{i=1}^{l-1} \beta_{l,i}$
Hieraus konstruieren wir ein Gleichungssystem

$$\begin{aligned} k_1 &= f(t_j, y_j) \\ k_2 &= f(t_j + \alpha_2 h_j, y_j + h_j\beta_{2,1}k_1) \\ k_3 &= f\Big(t_j + \alpha_3 h_j, y_j + h_j(\beta_{3,1}k_1 + \beta_{3,2}k_2)\Big) \\ \vdots &= \qquad\qquad \vdots \\ k_m &= f\Big(t_j + \alpha_m h_j, y_j + h_j(\beta_{m,1}k_1 + \beta_{m,2}k_2) + \cdots + \beta_{m,m-1}k_{m-1}\Big) \end{aligned} \tag{1.5}$$

Hieraus erhalten wir das Verfahren:

$$\boxed{u_{j+1} = u_j + h_j(\gamma_1 k_1 + \gamma_2 k_2 + \cdots + \gamma_m k_m)} \tag{1.6}$$

wobei die γ_i noch zu bestimmen sind.

Definition 1.3.3 (explizites Runge-Kutta-Verfahren):
Ein Verfahren der Form (1.6) *mit Stufenwerten wie in* (1.5) *heißt m-stufiges* ***explizites Runge-Kutta-Verfahren****. Die übliche Darstellungsart ist die* ***Butcher-Tabelle****.*

$$\begin{array}{c|ccccc} 0 & & & & & \\ \alpha_2 & \beta_{2,1} & & & & \\ \alpha_3 & \beta_{3,1} & \beta_{3,2} & & & \\ \vdots & \vdots & \vdots & \ddots & & \\ \alpha_{m-1} & \beta_{m-1,1} & \beta_{m-1,2} & \cdots & \ddots & \\ \alpha_m & \beta_{m,1} & \beta_{m,2} & \cdots & \beta_{m,m-1} & \\ \hline & \gamma_1 & \gamma_2 & \cdots & \gamma_{m-1} & \gamma_m \end{array}$$

Beispiele für explizite Runge-Kutta-Verfahren:

1. Explizites Eulerverfahren (1. Ordnung)
   $u_{j+1} = u_j + h_j \cdot f(t_j, u_j),\ m = 1$

$$\begin{array}{c|c} 0 & \\ \hline & 1 \end{array} \qquad \gamma_1 = 1$$

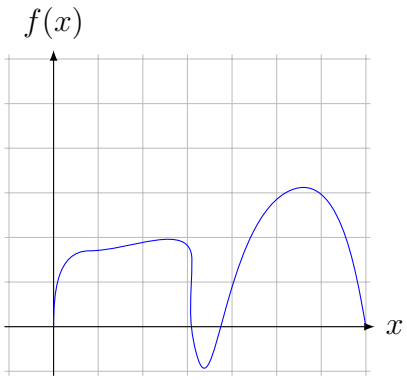
2. Modifiziertes Eulerverfahren (2. Ordnung, Mittelpunktsregel für Quadratur)
3. Verfahren von Heun (2. Ordnung, Trapezregel)
4. Runge-Verfahren (3. Ordnung)
   $m = 3$

$$\begin{array}{c|ccc} 0 & & & \\ \frac{1}{2} & \frac{1}{2} & & \\ 1 & 0 & 1 & \\ \hline & 0 & 0 & 1 \end{array} \qquad \begin{array}{llllll} \alpha_2 & = \frac{1}{2} & \alpha_3 & = 1 & & \\ \beta_{2,1} & = \frac{1}{2} & \beta_{3,1} & = 0 & \beta_{3,2} & = 1 \\ \gamma_1 & = 0 & \gamma_2 & = 0 & \gamma_3 & = 1 \end{array}$$

$$\begin{aligned} & & k_3 &\overset{(1.5)}{=} f\Big(t_j + \underbrace{1}_{=\alpha_3} k_j, y_j + h_j(\underbrace{1}_{=\beta_{3,2}} k_2)\Big) \\ u_{j+1} = u_j + h_j\gamma_3 k_3 \quad & k_2 &= f\left(t_j + \underbrace{\frac{1}{2}}_{=\alpha_2} h_j, y_j + h_j \underbrace{\frac{1}{2}}_{=\beta_{2,1}} k_1\right) \\ & k_1 &= f(t_j, y_j) \end{aligned}$$

5. Klassisches Runge-Verfahren (4. Ordnung, Simpsonregel für Quadratur)
   $m = 4$

$$\begin{array}{c|cccc} 0 & & & & \\ \frac{1}{2} & \frac{1}{2} & & & \\ \frac{1}{2} & 0 & \frac{1}{2} & & \\ 1 & 0 & 0 & 1 & \\ \hline & \frac{1}{6} & \frac{1}{3} & \frac{1}{3} & \frac{1}{6} \end{array}$$



1.3.3. Schrittweitensteuerung und eingebettete Verfahren

Darstellung folgt hier [12, Abschnitt 7.2.5]

Schrittweitensteuerung

In Bereichen, in denen sich die Lösung stark ändert, sollte eine kleine Schrittweite gewählt werden, in Bereichen mit schwacher Änderung kann mit großen Schrittweiten gerechnet werden.

Gegeben: t_0, y_0 und Fehlerschranke $\varepsilon > 0$

Gesucht: Möglichst große Schrittweite h, so dass der Diskretisierungsfehler $e_h(t_i) := y(t_i) - u_h(t_i)$ die Abschätzung $e_h(t_i) \leq \varepsilon$ erfüllt.

25.10.2012
6. Vorlesung

Dann fordern wir $\varepsilon \approx \kappa \cdot eps$, wobei eps die Maschinengenauigkeit ist und

$$\kappa \approx \max\Big\{|y(t)| \colon t \in [t_i, t_i + h]\Big\}$$

Eine $\kappa \cdot eps \approx \varepsilon$ entsprechende Wahl von h garantiert dann, dass

$$|e_h(t_i + h)| \leq \varepsilon \approx \kappa \cdot eps \tag{1.7}$$

d. h. der relative Fehler ist dann in der Größenordnung der Maschinengenauigkeit. Ist (1.7) erfüllt, so stimmt die berechnete Lösung in $t_i + h = t_{i+1}$ im Rahmen der Maschinengenauigkeit mit der exakten Lösung überein.

Frage: Wie wählt man eine Schrittweite $h = h(\varepsilon)$, so dass $\kappa \cdot eps \approx \varepsilon$?

Idee: Verwende zwei Einschrittverfahren verschiedener Ordnung und benutze die Differenz der Lösung zur Schätzung der Schrittweite. Zum **rechnen** ein Verfahren p-ter Ordnung und zum **schätzen** ein Verfahren $p+1$-ter Ordnung.

Die Verfahren seien durch Inkrementfunktionen Φ_I, Φ_{II} gegeben, d. h.

$$\begin{aligned}
\hat{u}_{i+1} &= \hat{u}_i + h_i \Phi_I(t_i, \hat{u}_i, h) \quad (p\text{-ter Ordnung}) \\
\overline{u}_{i+1} &= \overline{u}_i + h_i \Phi_{II}(t_i, \overline{u}_i, h) \quad (p+1\text{-ter Ordnung})
\end{aligned}$$

Mit jeweils einer unbekannten Funktion: φ_I unf φ_{II} ausgewertet in $y(t_i)$ gilt:

$$\begin{aligned}
& y(t_{i+1}) - \overline{u}_{i+1} && = h^{p+1}\varphi_{II}\Big(y(t_i)\Big) + \mathcal{O}\Big(h^{p+2}\Big) \\
- \quad & y(t_{i+1}) - \hat{u}_{i+1} && = h^{p}\varphi_{I}\Big(y(t_i)\Big) + \mathcal{O}\Big(h^{p+1}\Big) \\
\hline
& \hat{u}_{i+1} - \overline{u}_{i+1} && = -h^p\varphi_I\Big(y(t_i)\Big) + \underbrace{h^{p+1}\Phi_{II}\Big(y(t_i)\Big)}_{\in \mathcal{O}(h^{p+1})} + \mathcal{O}\Big(h^{p+1}\Big) \\
& && = -h^p\varphi_I\Big(y(t_i)\Big) + \mathcal{O}\Big(h^{p+1}\Big)
\end{aligned}$$

$$\overline{u}_{i+1} - \hat{u}_{i+1} = \underbrace{y(t_{i+1}) - \hat{u}_{i+1}}_{= e_h(t_{i+1})} + \mathcal{O}\left(h^{p+1}\right)$$

Für den Fehler erhalten wir die Abschätzung:

$$e_h(t_{i+1}) = y(t_{i+1}) - \hat{u}_{i+1} \approx \overline{u}_{i+1} - \hat{u}_{i+1}$$

Es genügt, die Differenz $\overline{u}_{i+1} - \hat{u}_{i+1}$ abzuschätzen.
Der Aufwand verdoppelt sich, da zwei Einschrittverfahren durchgeführt werden. Um diesen Aufwand zu verringern, hat Fehlberg (1964 – 1970) den Vorschlag gemacht, das Verfahren Φ_I in Φ_{II} einzubetten. Dieser Ansatz heißt auch **Runge-Kutta-Fehlberg-Verfahren** (RKF-Verfahren).

Gegeben:

$$\begin{aligned}
\hat{u}_0 &= \overline{u}_0 \\
\hat{u}_{i+1} &= \overline{u}_i + h_i \Phi_I(t_i, \overline{u}_i, h) \quad (p\text{-ter Ordnung}) \\
\overline{u}_{i+1} &= \overline{u}_i + h_i \Phi_{II}(t_i, \overline{u}_i, h) \quad (p+1\text{-ter Ordnung})
\end{aligned}$$

mit $\Phi_I(x,y,h) := \sum_{k=0}^{p} \hat{\gamma}_k f_k(x,y,h)$ und $\Phi_{II}(x,y,h) := \sum_{k=0}^{p+1} \overline{\gamma}_k f_k(x,y,h)$, wobei

$$\begin{aligned} k_h = f_k &:= f_k(x,y,h) \\ &= f\left(x + \alpha_k h, y + h \sum_{l=0}^{k-1} \beta_{kl} f_l\right) \end{aligned}$$

Beide Verfahren benutzen dieselben Funktionswerte $k_l,\ l = 1,\ldots,p$ und für Φ_{II} muss nur k_{p+1} zusätzlich berechnet werden. Daher spricht man von **eingebetteten Verfahren**. Da die beiden Verfahren Φ_I und Φ_{II} die Ordnung p bzw. $p+1$ haben sollen, müssen die Koeffizienten $\alpha_k, \beta_{kl}, \overline{\gamma_k}$ und $\hat{\gamma}_k$ so gewählt werden, dass für die lokalen Diskretisierungsfehler gilt:

$$\begin{aligned} \tau_{h,I}(t_{i+1}) &= \frac{y(t_{i+1}) - y(t_i)}{h} - \Phi_I(t_i, y(t_i), h) \ = \mathcal{O}\left(h^p\right) \\ \tau_{h,II}(t_{i+1}) &= \frac{y(t_{i+1}) - y(t_i)}{h} - \Phi_{II}(t_i, y(t_i), h) = \mathcal{O}\left(h^{p+1}\right) \end{aligned}$$

Die gesuchten Koeffizienten kann man im Prinzip wie gewöhnliche Runge-Kutta-Verfahren bestimmen. Der Ansatz führt allerdings im Allgemeinen auf ein nicht lineares Gleichungssystem *(Für Details, [siehe 12, Abschnitt 7.2.5])*.
Stattdessen betrachten wir ein konkretes eingebettetes Runge-Kutta-Verfahren mit $p = 2$ und $p = 3$ *[11, Beispiel 11.11]*.

$p = 2$

0		
1	1	
	$\frac{1}{2}$	$\frac{1}{2}$

$$u_{i+1} = u_i + h_i\left(\tfrac{1}{2}k_1 + \tfrac{1}{2}k_2\right)$$

$p = 3$

0			
1	1		
$\frac{1}{2}$	$\frac{1}{4}$	$\frac{1}{4}$	
$\hat{u}$	$\frac{1}{2}$	$\frac{1}{2}$	0
$\overline{u}$	$\frac{1}{6}$	$\frac{1}{6}$	$\frac{2}{3}$

Erweiterte Butcher-Tabelle

Somit erhalten wir für $\overline{u}_0 = \hat{u}_0$ die Verfahren:

$$\begin{aligned} \hat{u}_{i+1} &= \overline{u}_i + \frac{h_i}{2}(k_1 + k_2) && p = 2 \\ \overline{u}_{i+1} &= \overline{u}_i + \frac{h_i}{6}(k_1 + k_2 + 4k_3) && p = 3 \\ k_1 &= f(t_i, \overline{u}_i) \\ k_2 &= f(t_i + h_i, \overline{u}_i + h_i k_1) \\ k_3 &= f\left(t_i + \frac{h_i}{2}, \overline{u}_i + \frac{h_i}{4}(k_1 + k_2)\right) \end{aligned}$$

Schrittweitensteuerung mit eingebetteten Verfahren

Es gilt:

$$\hat{u}_{k+1} - \overline{u}_{k+1} = h \cdot \left(\Phi_I(t_k, \overline{u}_k, h) - \Phi_{II}(t_k, \overline{u}_k, h)\right) \tag{1.8}$$

Da die Verfahren die Ordnung p und $p+1$ haben gilt:

$$\begin{aligned} & \Phi_I(t, y(t), h) - \frac{y(t+h) - y(t)}{h} \approx h^p c_I(t) \\ & \Phi_{II}(t, y(t), h) - \frac{y(t+h) - y(t)}{h} \approx h^{p+1} c_{II}(t) \\ \overset{(1.8)}{\Rightarrow} \quad & \hat{u}_{k+1} - \overline{u}_{k+1} \approx h^{p+1} c_I(t_k) \end{aligned} \tag{1.9}$$

Angenommen, der $(k+1)$-te Iterationsschritt ist erfolgreich abgeschlossen, dann gilt für eine vorgegebene Fehlerschranke $\varepsilon > 0$: $|\hat{u}_{k+1} - \overline{u}_{k+1}| \leq \varepsilon$. Näherungsweise gilt dann auch:

$$\underbrace{h_k^{p+1}}_{=h} |c_I(t_k)| \leq \varepsilon$$

Damit die neue Schrittweite h_{k+1} die Fehlerabschätzung $|\hat{u}_{k+2} - \overline{u}_{k+2}| \leq \varepsilon$ liefert, muss

$$h_{k+1}^{p+1} |c_I(t_{k+1})| \leq \varepsilon$$

gelten.

$$\overset{(1.9)}{\Rightarrow} |c_I(t_k)| \approx \frac{\hat{u}_{k+1} - \overline{u}_{k+1}}{h_k^{p+1}}$$

Aus $c_I(t_k) \approx c_i(t_{k+1})$ ergibt sich die Folgerung

$$\begin{aligned} & |\hat{u}_{k+1} - \overline{u}_{k+1}| \left| \frac{h_{k+1}}{h_k} \right|^{p+1} \leq \varepsilon \\ \Rightarrow \quad & h_{k+1} := h_k \left(\frac{\varepsilon}{|\hat{u}_{k+1} - \overline{u}_{k+1}|} \right)^{\frac{1}{p+1}} \end{aligned}$$

Aufgrund numerischer Experimente schlagen Stoer/Bulirsch [12, Seite 136] folgende Modifikation vor:

$$h_{k+1} := \alpha h_k \left(\frac{\varepsilon h_k}{|\hat{u}_{k+1} - \overline{u}_{k+1}|} \right)^{\frac{1}{p+1}}$$

wobei α geeignet zu wählen ist, z. B. $\alpha = 0,9$.

1.3.4. Implizite Einschrittverfahren

30.10.2012
7. Vorlesung

Das implizite Eulerverfahren

Wiederholung:
Sei $y \in \mathcal{C}^2([a,b]),\ x, \tilde{x} \in [a,b]$

$$\overset{\text{Taylor-Entwicklung}}{\Rightarrow} y(x) = y(\tilde{x}) + y'(x)(x - \tilde{x}) + \frac{y''(\xi)}{2}(x - \tilde{x})^2$$

wobei ξ passend zwischen x und $\tilde{x}$ gewählt wird.
Setze $x = x_{k+1}, \quad \tilde{x} = x_k, \quad x_{k+1} = x_k + h$.

$$\begin{aligned} y(x_{k+1}) &= y(x_k) + h \cdot y'(x_k) + \mathcal{O}\left(h^2\right) \\ &= y(x_k) + h \cdot f\Big(x_k, y(x_k)\Big) + \mathcal{O}\left(h^2\right) \end{aligned}$$

$$\Rightarrow u_{k+1} = u_k + h \cdot f(x_k, u_k) \text{ (expliziter Euler (vorwärts))}$$

Wählen wir stattdessen in der Taylorentwicklung

$$x = x_k, \quad \tilde{x} = x_{k+1}, \quad x_{k+1} = x_k + h$$

(d. h. rückwärtsentwickeln), dann gilt

$$\begin{aligned} y(x_k) &= y(x_{k+1}) - h \cdot y'(x_{k+1}) + \mathcal{O}\left(h^2\right) \\ &= y(x_{k+1}) - h \cdot f\Big(x_{k+1}, y(x_{k+1})\Big) + \mathcal{O}\left(h^2\right) \end{aligned}$$

Hieraus ergibt sich das implizite Eulerverfahren (rückwärts):

$$u_{k+1} = u_k + h \cdot f(x_{k+1}, u_{k+1})$$

Im Allgemeinen muss hier pro Iterationsschritt ein nichtlineares Gleichungssystem gelöst werden. Ist dieses System lösbar?

Satz 1.3.3:
Sei $f\colon [0,T] \times \mathbb{R}^n \to \mathbb{R}^n$ eine stetig differenzierbare Funktion.
Weiterhin sei $f(x,y)$ bezüglich y lipschitzstetig mit einer Lipschitzkonstanten $L > 0$ und es sei $h > 0$ so klein, dass $hL < 1$. Dann existiert für alle $(x,y) \in [0,T] \times \mathbb{R}^n$ eine eindeutig bestimmte Lösung v des nichtlinearen Gleichungssystems

$$v = y + h \cdot f(x,v)$$

Beweis:
Banachscher-Fixpunktsatz: Gesucht wird Fixpunkt v von

$$T\colon \mathbb{R}^n \to \mathbb{R}^n \quad \wedge \quad Tv = y + hf(x,v)$$

Aus der Lipschitzstetigkeit von f bezüglich y folgt

$$\|Tv - Tw\| = h\|f(x,v) - f(x,w)\| \leq \underbrace{hL}_{:=q} \|v - w\|$$

$\Rightarrow$ Behauptung

□

Satz 1.3.4:
Es gelten die Voraussetzungen des Satzes (1.3.3)*. Dann konvergiert das implizite Eulerverfahren und es hat die Konvergenzordnung $p = 1$.*

Beweis:
Für den lokalen Diskretisierungsfehler $\tau_h(x)$ des impliziten Eulerverfahren gilt:

$$\tau_h(x) = \frac{y(x+h) - y(x)}{h} - \Phi\Big(x + h, y(x+h)\Big) = \mathcal{O}(h)$$

Für den lokalen Diskretisierungsfehler

$$\begin{array}{rll} & y_{k+1} & = y_k + hf(x_{k+1}, y_{k+1}) + h\tau_h(x_k) \\ - & u_{k+1} & = u_k + hf(x_{k+1}, u_{k+1}) \\ \hline & e_{k+1} & = y_k - u_k + h\Big(f(x_{k+1}, y_{k+1}) - f(x_{k+1}, u_{k+1})\Big) + h\tau_h(x_k) \end{array}$$

$e_k = \|y_k - u_k\|, \quad \tau_h = \max_k \|\tau_h(x_k)\|$

$$
\begin{aligned}
\Rightarrow \qquad e_{k+1} &\le e_k + h\Big\|f(x_{k+1}, y_{k+1}) - f(x_{k+1}, u_{k+1})\Big\| + h\tau_h \\
&\le e_k + hLe_{k+1} + h\tau_h \\
\Rightarrow \qquad e_{k+1} &\le \frac{e_k}{1-hL} + \frac{h\tau_h}{1-hL} \\
&= \left(1 + \underbrace{\frac{hL}{1-hL}}_{\rho_j}\right)\underbrace{e_k}_{z_j} + \underbrace{\frac{h\tau_h}{1-hL}}_{\eta_j} \\
\overset{\text{Lemma 1.3.1}}{\Rightarrow} \qquad e_k &\le \left(\underbrace{e_0}_{=0} + \sum_{i=0}^{k-1} \frac{h}{1-hL}\tau_h\right) e^{\sum_{i=0}^{k-1} \frac{hL}{1-hL}} \\
\Rightarrow \qquad e_k &\le \left(\frac{h}{1-hL}\tau_h\right) \cdot k \cdot e^{k\frac{hL}{1-hL}} \\
&\le \frac{T}{1-TL}\tau_h e^{\frac{TL}{1-TL}} \\
&\le \tilde{c}\tau_h \\
&\le \tilde{c}h \\
&= \mathcal{O}(h)
\end{aligned}
$$

□

Bemerkung 1.3.3:
Wenn wir in den Sätzen (1.3.3) und (1.3.4) nur die einseitige Lipschitzbedingung

$$\Big(f(x,y) - f(x,z)\Big) \cdot (y-z) \le L\|y-z\|^2 \quad \forall y, z \in \mathbb{R}^n$$

mit einer Konstanten $L \in \mathbb{R}$ verlangen, so gelten alle Aussagen in diesen beiden Sätzen analog.
Da in diesem Fall L auch negativ sein darf, ergibt sich manchmal aus $hL < 1$ keine Einschränkung an die Schrittweite.

Implizite Runge-Kutta-Verfahren

Analog zu den expliziten Runge-Kutta-Verfahren setzen wir:

$$
\begin{aligned}
k_1 &= f\Big(x_j + \alpha_1 h_j, u_j + h_j(\beta_{1,1}k_1 + \cdots + \beta_{1,m}k_m)\Big) \\
k_2 &= f\Big(x_j + \alpha_2 h_j, u_j + h_j(\beta_{2,1}k_1 + \cdots + \beta_{2,m}k_m)\Big) \\
\vdots \; &= \qquad\qquad \vdots \\
k_m &= f\Big(x_j + \alpha_1 h_j, u_j + h_j(\beta_{m,1}k_1 + \cdots + \beta_{m,m}k_m)\Big)
\end{aligned}
$$

wobei $\alpha_l = \sum_{i=1}^{m} \beta_{l,i}$.
Das implizite Runge-Kutta-Verfahren lautet dann

$$u_{j+1} = u_j + h_j(\gamma_1 k_1 + \cdots + \gamma_m k_m)$$

Notation Butcher-Tabelle

$$\begin{array}{c|ccc} \alpha_1 & \beta_{1,1} & \cdots & \beta_{1,m} \\ \alpha_2 & \beta_{2,1} & \cdots & \beta_{2,m} \\ \vdots & \vdots & & \vdots \\ \alpha_{m-1} & \beta_{m-1,1} & \cdots & \beta_{m-1,m} \\ \alpha_n & \beta_{m,1} & \cdots & \beta_{m,m} \\ \hline & \gamma_1 & \cdots & \gamma_m \end{array}$$

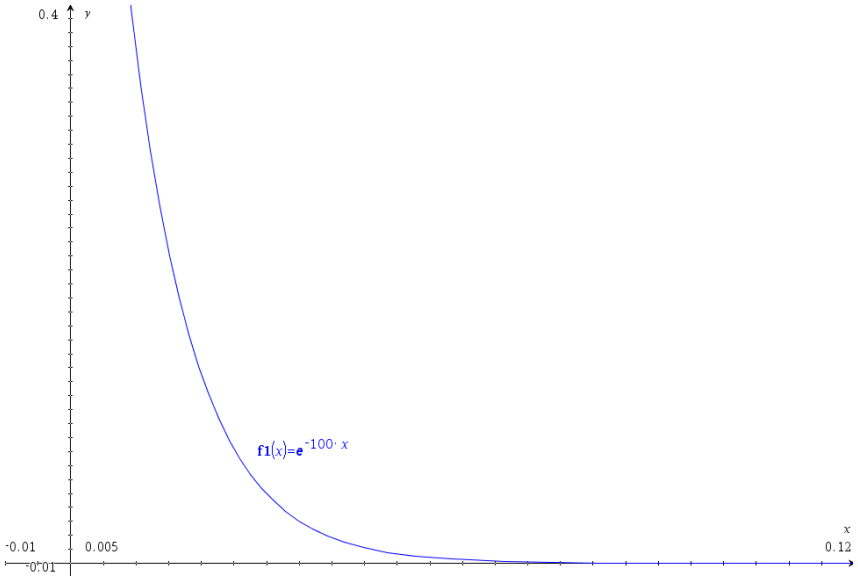
1.3.5. Absolute Stabilität

06.11.2012
8. Vorlesung

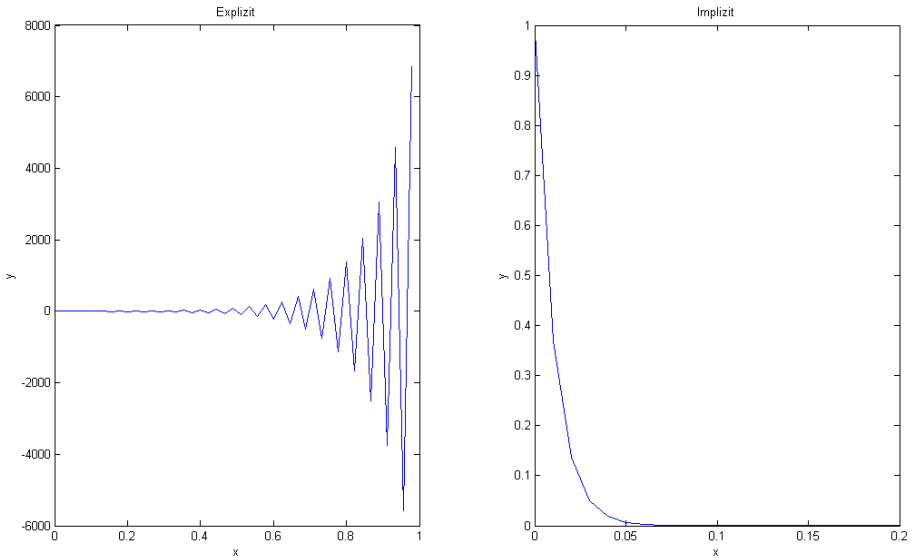
Wir folgen hier der Darstellung in [11, Kapitel 12]

Testproblem: $\begin{cases} y' = -100y \text{ in } [0,T] \\ y(0) = 1 \end{cases}$

Exakte Lösung: $y(t) = e^{-100t} \xrightarrow{t\to\infty} 0$



Ein explizites Eulerverfahren führt erst ab $h < \frac{1}{50}$ zu vernünftigen Ergebnissen, implizite Eulerverfahren haben diese Einschränkung nicht.



Betrachte ein explizites Eulerverfahren für das Testproblem:

$$\begin{aligned} u_{j+1} &= u_j + h \cdot f(t_j, u_j) \\ &= u_j - 100hu_j \\ &= (1 - 100h)u_j \\ &= (1 - 100h)^{j+1} \cdot \underbrace{y_0}_{=1} \end{aligned}$$

Damit auch die numerische Lösung gegen 0 geht, d. h. $u_j \xrightarrow{j\to\infty} 0$ muss

$$\begin{aligned} & |1 - 100h| < 1 \\ \Leftrightarrow \quad & -1 < 1 - 100h < 1 \\ \Leftrightarrow \quad & 0 < h < \frac{1}{50} \end{aligned}$$

Definition 1.3.4 (absolute Stabilität):
Gegeben sei die (skalare) Differentialgleichung $y' = \lambda y,\ y(0) = 1$ für festes $\lambda \in \mathbb{C}$. Dieses Problem bezeichnet man auch als Testproblem oder Testaufgabe. Hierzu betrachten wir ein allgemeines (explizites oder implizites) Einschrittverfahren mit fester Schrittweite h gegeben durch

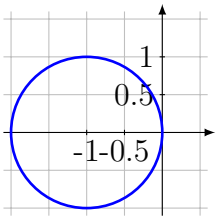
$$u_{j+1} = u_j + h \cdot \Phi(ih, u_j, u_{j+1}, h), \quad j = 0, 1, 2, \ldots$$

*mit $u_0 = 1$. Der Bereich **absoluter Stabilität** dieses Verfahrens ist definiert als*

$$\mathcal{A} := \left\{ z \in \mathbb{C} \colon z = \lambda h \ \wedge\ |u_{j+1}| < |u_j| \quad \forall j = 0, 1, 2, \ldots \right\}$$

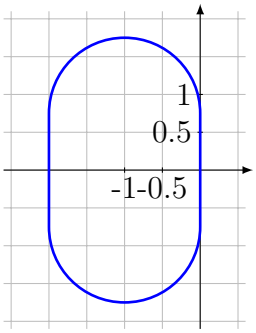
Beispiele:

1. Explizites Eulerverfahren:



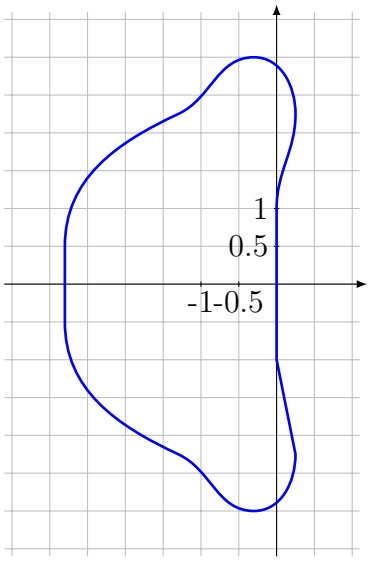
$$\begin{aligned} & y' = \lambda y,\ y(0) = 1 \\ u_{i+1} &= u_i + h\lambda u_i = (1 + h\lambda)u_i \\ \Rightarrow \quad \mathcal{A} &= \{ z \in \mathbb{C} \colon z = \lambda h \ \wedge\ |1 + z| < 1 \} \end{aligned}$$

2. Modifiziertes Eulerverfahren:



$$\begin{aligned} & y' = \lambda y,\ y(0) = 1 \\ \Rightarrow \quad u_{i+1} &= u_i + hf\left(t_i + \tfrac{h}{2}, u_i + \tfrac{h}{2} f(t_i, u_i)\right) \\ &= u_i + h\lambda\left(u_i + \tfrac{h}{2}\lambda u_i\right) \\ &= \left(1 + h\lambda + \tfrac{(h\lambda)^2}{2}\right) u_i \\ \Rightarrow \quad \mathcal{A} &= \left\{ z \in \mathbb{C} \colon z = \lambda h \ \wedge\ \left|1 + z + \tfrac{z^2}{2}\right| < 1 \right\} \end{aligned}$$

3. Klassisches Runge-Kutta-Verfahren 4. Ordnung:



Hierfür ergibt sich das Stabilitätsgebiet:

$$\mathcal{A} = \left\{ z \in \mathbb{C} \colon z = \lambda h \ \wedge \ \left| 1 + z + \frac{z^2}{2} + \frac{z^3}{6} + \frac{z^4}{24} \right| < 1 \right\}$$

Die exakte Lösung konvergiert für negative reelle λ streng monoton gegen 0, wenn $t \to \infty$. Damit dies auch für die numerische Lösung gilt, muss $\lambda h \in \mathcal{A}$ gelten.

1. Explizites Eulerverfahren:

$$|1 + \lambda h| < 1 \Leftrightarrow 0 < h < -\frac{2}{\lambda}$$

2. Modifiziertes Eulerverfahren:

$$\begin{aligned} h\lambda \in \mathcal{A} \Leftrightarrow & \quad \left| 1 + h\lambda + \frac{(h\lambda)^2}{2} \right| < 1 \\ \Leftrightarrow & \quad -1 < 1 + h\lambda + \frac{(h\lambda)^2}{2} < 1 \\ \Leftrightarrow & \quad -1 < -1 - h\lambda - \frac{(h\lambda)^2}{2} < 1 \\ \Rightarrow & \quad 0 \underset{(*)}{<} -h\lambda - \frac{(h\lambda)^2}{2} \underset{(**)}{<} 2 \end{aligned}$$

$$\begin{aligned} \Rightarrow & \quad h\lambda < -\frac{(h\lambda)^2}{2} \\ \Leftrightarrow & \quad 2 < -h\lambda \qquad (*) \\ \Leftrightarrow & \quad h < -\frac{2}{\lambda} \end{aligned}$$

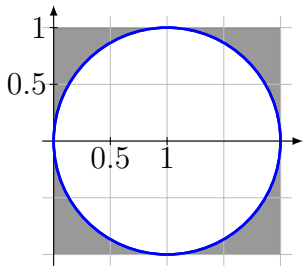
$$\begin{aligned} \Rightarrow & \quad -h\lambda - \frac{(h\lambda)^2}{2} < 2 \\ \Leftrightarrow & \quad -2h\lambda - (h\lambda)^2 - 4 < 0 \qquad (**) \\ \Leftrightarrow & \quad (h\lambda + 1)^2 + 3 > 0 \end{aligned}$$

Die Ungleichung $(**)$ ist immer erfüllt und somit muss die Schrittweite beim modifizierten Eulerverfahren auch die Bedingung $\boxed{h < -\frac{2}{\lambda}}$ erfüllen.

Implizites Eulerverfahren:

$$\text{Für } y' = \lambda y,\ y(0) = 1 \Rightarrow u_{i+1} = u_i + h\lambda u_{i+1}$$
$$\Leftrightarrow u_{i+1} = \frac{1}{1 - h\lambda} u_i$$

Als absolutes Stabilitätsgebiet erhalten wir:



$$\mathcal{A} = \left\{ z \in \mathbb{C} \colon z = \lambda h \ \wedge\ 1 < |1 - z| \right\}$$

Für reelles, negatives λ ist offensichtlich $z = \lambda h \in \mathcal{A}$ **ohne** Bedingung an h immer erfüllt. Wir können zeigen:

$$\lambda h \in \mathcal{A} \Leftrightarrow 1 < |1 - \lambda h| = \Big(1 - Re(\lambda h)\Big)^2 + \Big(1 - Im(\lambda h)\Big)^2$$

Dies ist für alle $\lambda \in \mathbb{C}$ mit $Re(\lambda) < 0$ **ohne** Bedingung an h erfüllt. Dies erklärt das bessere numerische Verhalten des impliziten Eulerverfahren zu beginn der Vorlesung für $y' = -100y,\ y(0) = 1$.

1.3.6. Steife Differentialgleichungen

Zunächst überlegen wir uns, wie wir den Begriff der Stabilität auf Systeme gewöhnlicher Differentialgleichungen übertragen können. Wir betrachten dazu

$$y'(x) = Ay(x), \quad A \in \mathbb{R}^{n \times n}, \quad y = \begin{pmatrix} y_1 \\ \vdots \\ y_n \end{pmatrix} \tag{1.10}$$

Hat A n verschiedene Eigenwerte, so hat die Lösung von (1.10) die Form:

$$y(x) = \sum_{j=1}^{n} c_j e^{\lambda_j x} \cdot v_j$$

mit Konstanten $c_j,\ j = 1, \ldots, n$ und einer Basis $v_j,\ j = 1, \ldots, n$ aus Eigenvektoren von A zu den Eigenwerten λ_j.

Es gilt:

$$e^{\lambda_j x} = e^{Re(\lambda_j)x} \underbrace{e^{iIm(\lambda_j)x}}_{\in \partial B(0,1)}$$

Also bestimmt der Realteil von λ_j das Wachstumsverhalten von $y(x)$. Nehmen wir an, dass $Re(\lambda_j) < 0,\ j = 1, \ldots, n$, so gilt:

$$\lim_{x \to \infty} \|y(x)\| = 0$$

Unter dieser Voraussetzung haben wir eine exponentiell abklingende Lösung vorliegen. Analog zum skalaren Fall soll für die numerische Lösung gelten:

$$\|u_{j+1}\| < \|u_j\|$$

A hat n verschiedene Eigenwerte $\Rightarrow$ A diagonalisierbar $(A = Q^{-1}DQ)$ $D = \underset{i=1,\dots,n}{\text{diag}}\,(\lambda_i)$ und n Spaltenvektoren von Q sind die Eigenvektoren v_j.
Führen wir $z = Qy$ als neue Variable ein.

$$y' = Ay = Q^{-1}D\underbrace{Qy}_{=z}$$

$$z' = (Qy)' = Qy' \overset{y'=Ay}{=} \underbrace{Q(Q^{-1}}_{=I} D \underbrace{Qy}_{=z}) = Dz$$

Somit haben wir ein entkoppeltes System gewöhnlicher Differentialgleichungen erhalten:

$$z = \begin{pmatrix} z_1 \\ \vdots \\ z_n \end{pmatrix}, \qquad \boxed{\begin{array}{c} z_1' = \lambda_1 z_1 \\ \vdots \\ z_n' = \lambda_n z_n \end{array}}$$

08.11.2012
9. Vorlesung

Der Stabilitätsbegriff lässt sich komponentenweise übertragen. Da die Eigenwerte einer Matrix auch komplex sein können, ist eine natürliche Verallgemeinerung des stabilen Falls die Forderung:

$$Re(\lambda_j) < 0, \quad j = 1, \dots, n$$

Da wir im Allgemeinen für jede gewöhnliche Differentialgleichung $z_i' = \lambda_i z_i,\ i = 1, \dots, n$ ein anderes Stabilitätsgebiet erhalten, wird man im Allgemeinen auch sehr unterschiedliche Anforderungen an die Schrittweite bekommen (abhängig von der Größe von $Re(\lambda_j)$).
$\Rightarrow$ implizite Verfahren sind dann von Vorteil

Allgemein betrachten wir Systeme der Form

$$y' = Ay + g(x) \tag{1.11}$$

mit $y = \begin{pmatrix} y_1 \\ \vdots \\ y_n \end{pmatrix}$, $\quad A \in \mathbb{R}^{n\times n}$, $\quad g = \begin{pmatrix} g_1 \\ \vdots \\ g_n \end{pmatrix}$.

Annahme: A hat n verschiedene Eigenwerte $(\lambda_j)_{j=1,\dots,n}$.

Dann hat die Lösung von (1.11) die Form:

$$y(x) = \sum_{j=1}^{n} c_j e^{\lambda_j x} v_j + \psi(x)$$

$(v_j)_{j=1,\dots,n}$ Eigenvektoren zu den Eigenwerten $(\lambda_j)_{j=1,\dots,n}$, c_j, $j = 1, \dots, n$ positive Konstanten und einer Funktion $\psi(x)$ Haben die Eigenwerte negativen Realteil, d. h. $Re(\lambda_j) < 0,\ j = 1, \dots, n$ so verhält sich $y(x)$ für $x \to \infty$ wie $\psi(x)$. Interpretiert man x als Zeit, so nennt man ψ auch den **stationären Anteil** der Lösung und $\sum_{j=1}^{n} c_j e^{\lambda_j x} v_j$ den **instationären** oder **transienten Anteil**.

Definition 1.3.5 (steife Differentialgleichungssysteme):
Ein lineares System gewöhnlicher Differentialgleichung der Form

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

heißt ***steif*** *(engl. stiff), wenn für die Eigenwerte $\lambda_1, \dots, \lambda_n \in \mathbb{C}$ von A gilt:*

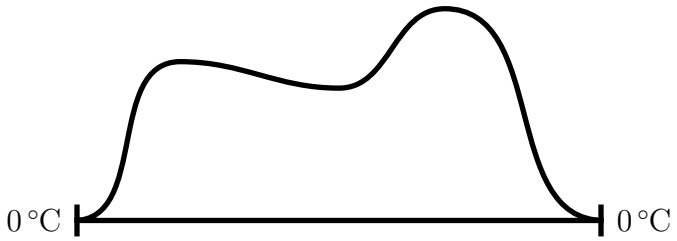
1. *$Re(\lambda_j) < 0,\ j = 1, \dots, n$*
2. *Es gibt sowohl Eigenwerte mit großem als auch mit kleinem Betrag, d. h.*

$$\max_{j=1,\dots,n} |\lambda_j| \gg \min_{j=1,\dots,n} |\lambda_j|$$

Ist $n = 1$, so besitzt der Eigenwert λ_1 einen (relativ) großen Betrag.

Wir betrachten jetzt die **Wärmeleitungsgleichung** in einer Raumdimension. Es handelt sich um eine partielle Differentialgleichung.

$y(t, x)$ = Wärmeverteilung zu einem Zeitpunkt t im Punkt x eines Stabes oder Drahtes (Herleitung später in Abschnitt 2.2.1)



$$\frac{\partial y}{\partial t}(t, x) = \frac{\partial^2 y}{\partial x^2}(t, x), \quad t \geq 0, \quad x \in [0, 1]$$

Wir geben folgende **Anfangswerte** vor:

$$y(0, x) = \underbrace{\Phi(x)}_{\text{Anfangswärmeverteilung}}, \quad x \in [0, 1]$$

Des weiteren geben wir **Randwerte** vor:

$$y(t, 0) = y(t, 1) = 0, \quad \forall t \geq 0$$

Diese Randwerte lassen sich so interpretieren, dass die Temperatur an beiden Stabenden konstant auf 0 °C gehalten wird. Dementsprechend benötigen wir für die Anfangswärmeverteilung $\Phi(x)$ eine Kompartibilitätsbedingung $\Phi(0) = \Phi(1) = 0$.

Idee: Um ein numerisches Verfahren zur Lösung der Wärmeleitgleichung zu entwickeln, approximieren wir $\frac{\partial^2 y}{\partial x^2}$ durch einen Differentialquotienten zweiter Ordnung.

Dazu betrachten wir die Taylorentwicklungen um x:

$$y(t, x+h) = y(t, x) + h\frac{\partial y}{\partial x}y(t, x) + \frac{h^2}{2}\frac{\partial^2 y}{\partial x^2}y(t, x) + \frac{h^3}{3!}\frac{\partial^3}{\partial x^3}y(t, x) + \frac{h^4}{4!}\frac{\partial^4}{\partial x^4}y(t, x) + \mathcal{O}\left(h^5\right)$$
$$y(t, x-h) = y(t, x) - h\frac{\partial y}{\partial x}y(t, x) + \frac{h^2}{2}\frac{\partial^2 y}{\partial x^2}y(t, x) - \frac{h^3}{3!}\frac{\partial^3}{\partial x^3}y(t, x) + \frac{h^4}{4!}\frac{\partial^4}{\partial x^4}y(t, x) + \mathcal{O}\left(h^5\right)$$

Addieren der beiden Entwicklungen und auflösen nach $\frac{\partial^2 y}{\partial x^2}$ ergibt:

$$\frac{\partial^2 y}{\partial x^2}(t,x) = \frac{y(t,x+h) - 2y(t,x) + y(t,x-h)}{h^2} + \mathcal{O}\left(h^2\right)$$

Wähle das äquidistante Gitter $I_h := \{x_0, x_1, \ldots, x_{m+1}\}, \quad x_j = jh, \quad h = \frac{1}{m+1}$, $j = 0, \ldots, m+1$. Ersetze in der Wärmeleitungsgleichung $\frac{\partial y}{\partial t} = \frac{\partial^2 y}{\partial x^2}$ die rechte Seite durch den Differenzenquotienten:

$$\frac{\partial y}{\partial t}(t,x) = \frac{y(t,x-h) - 2y(t,x) + y(t,x+h)}{h^2} + \mathcal{O}\left(h^2\right)$$

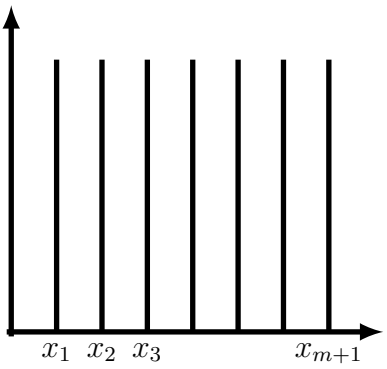
Lassen wir den Term $\mathcal{O}(h^2)$ for, so erhalten wir in jedem Gitterpunkt x_i eine Näherung an $y(t,x_i)$, die wir mit $\tilde{y}_i(t)$ bezeichnen.

$$\frac{\partial y}{\partial t}(t,x_i) = \frac{\tilde{y}_{i-1}(t) - 2\tilde{y}_i(t) + \tilde{y}_{i+1}(t)}{h^2}, \quad i = 1, \ldots, m$$

Die Wärmeleitungsgleichung (partielle Differentialgleichung) haben wir somit auf ein System gewöhnlicher Differentialgleichungen erster Ordnung transformiert:

$$\tilde{y}' = A\tilde{y}, \quad \tilde{y} = (\tilde{y}_1, \ldots, \tilde{y}_m)^\mathsf{T}$$

mit den Anfangswerten $\tilde{y}(0) = \left(\Phi(x_1), \ldots, \Phi(x_{m+1})\right)^\mathsf{T}$ und $A = \frac{1}{h^2}\begin{pmatrix} -2 & 1 & 0 & \cdots & 0 \\ 1 & -2 & 1 & \ddots & \vdots \\ 0 & \ddots & \ddots & \ddots & 0 \\ \vdots & \ddots & 1 & -2 & 1 \\ 0 & \cdots & 0 & 1 & -2 \end{pmatrix}$



Diese Vorgehensweise zur Lösung der Wärmeleitgleichung nennt man die **(vertikale) Linienmethode** (*engl. method of lines*). Die Matrix A ist diagonalisierbar und hat die Eigenwerte

$$\lambda_j = (m+1)^2 \cdot \left(2\cos\left(\frac{j\pi}{m+1}\right) - 2\right), \quad j = 1, \ldots, m$$

A ist symmetrisch, also sind λ_j reell. Alle Eigenwerte sind negativ und es gibt betragsmäßig große und kleine Eigenwerte. Das vorliegende System gewöhnlicher Differentialgleichungen ist also steif und wird sogar umso steifer, je feiner die Ortsdiskretisierung h ist.

2. Partielle Differentialgleichungen

13.11.2012
10. Vorlesung

2.1. Die Advektionsgleichung $u_t + cu_x = 0$

$$u_t + cu_x = 0$$
$$u = u(x,t), \quad u_t = \frac{\partial u}{\partial t}(x,t), \quad u_x = \frac{\partial u}{\partial x}(x,t)$$

Ohne Einschränkung: $c \geq 0$, $c = konstant$
sonst betrachte $v(x,t) := u(-x,t) \Rightarrow$ Transformationsgleichung mit $\tilde{c} := -c > 0$
$v_t + \tilde{c}v_x = 0$

2.1.1. Physikalische Herleitung

Eine Flüssigkeit bewege sich mit **konstanter Geschwindigkeit** in Richtung der positiven x-Achse. Sie transportiere dabei einen Schadstoff.

$u(x,t) :=$ **Dichte** (Masse pro Längeneinheit) des Schadstoffes im Punkt (x,t)

$q(x,t) :=$ **Fluss** des Schadstoffes (Masse, die in einer Zeiteinheit an dem Ortspunkt x vorbeifließt)

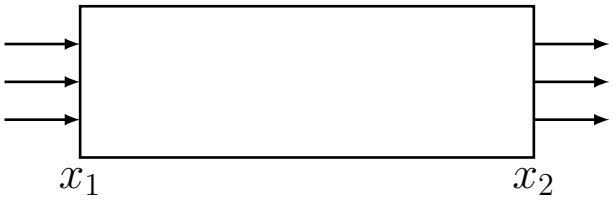
$c = \frac{dx}{dt} = konstant =$ Geschwindigkeit = Ableitung Ort nach Zeit
Es gilt folgender Zusammenhang zwischen Fluss, Dichte und Geschwindigkeit:

$$q(x,t) = c \cdot u(x,t)$$

Sei $V = (x_1, x_2) \subset \mathbb{R}$ ein offenes Intervall. Dann ist die in V enthaltene Masse gegeben durch

$$M = M(t) = \int_V u(x,t)\,\mathrm{d}x$$

Masse kann nicht erzeugt oder vernichtet werden, daher ist eine Massenänderung in V nur möglich, wenn Masse mit der Flüssigkeit ein- und ausströmt.



Keine zeitliche Änderung der Masse

$$\Rightarrow 0 = \frac{\partial M}{\partial t} = \frac{\partial}{\partial t}\left(\int_V u(x,t)\,\mathrm{d}x\right) = \int_V u_t(x,t)\,\mathrm{d}x$$

Differenz zwischen einfließender und ausfließender Masse gleich 0, d. h.

$$0 = q(x_1,t) - q(x_2,t) \quad \forall t \geq 0$$

Hieraus ergibt sich mit $x_1 \neq x_2$:

$$\frac{\int_{x_1}^{x_2} u_t(x,t)\,\mathrm{d}x}{x_2 - x_1} + \frac{q(x_2,t) - q(x_1,t)}{x_2 - x_1} = 0$$

Bezeichnen wir mit $F(x)$ die Stammfunktion von $u_t(x,t)$ bezüglich x, so folgt:

$$\frac{F(x_2) - F(x_1)}{x_2 - x_1} + \frac{q(x_2,t) - q(x_1,t)}{x_2 - x_1} = 0$$

Grenzübergang: $x_1 \to x_2 =: x$ ergibt mit $F'(x) = u_t(x,t)$

$$u_t(x,t) + q_x(x,t) = 0$$

Da $q_x = c \cdot u_x (q = cu, c = konstant)$, folgt $\boxed{u_t(x,t) + cu_x(x,t) = 0}$ (kurz: $u_t + cu_x = 0$)

2.1.2. Allgemeine Lösung

Variablentransformation:

$$\xi := x - ct \Leftrightarrow x = \xi + ct \tag{2.1}$$

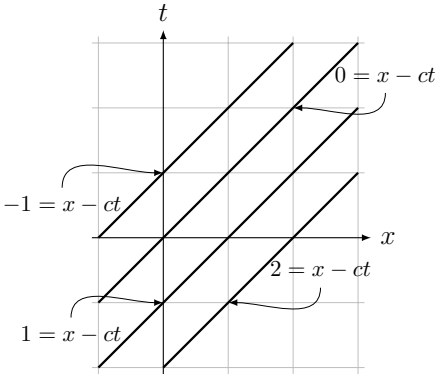
Betrachte die Hilfsfunktion:

$$\begin{aligned} v(\xi,t) &:= u(\xi + ct, t) \\ &\overset{\text{Kettenregel}}{\Rightarrow} \frac{\partial v}{\partial t}(\xi,t) = \frac{\partial u}{\partial x}(\xi + ct, t)\frac{\partial(\xi + ct)}{\partial t} + \frac{\partial u}{\partial t} \\ &= cu_x + u_t = 0 \end{aligned}$$

Also ist die Hilfsfunktion $v(\xi,t)$ unabhängig von t und es existiert eine Funktion $\varphi(\xi)$ die nur von ξ abhängt, so dass $v(\xi,t) = \varphi(\xi)$.
$\Rightarrow \varphi(\xi) = v(\xi,t) = u(\xi + ct, t) \overset{(2.1)}{\Rightarrow} \boxed{u(x,t) = \varphi(x - ct)}$
Sei umgekehrt $\varphi\colon \mathbb{R} \to \mathbb{R}, \quad \varphi \in \mathcal{C}^1(\mathbb{R})$ mit $u(x,t) := \varphi(x - ct)$.
$\Rightarrow u_t + cu_x = -c\varphi'(x - ct) + c\varphi'(x - ct) = 0.$

Lemma 2.1.1:
Die Funktion $u \in \mathcal{C}^1(\mathbb{R} \times \mathbb{R})$ ist eine Lösung der Gleichung $u_t + cu_x = 0$ genau dann, wenn es eine Funktion $\varphi \in \mathcal{C}^1(\mathbb{R})$ gibt mit $u(x,t) = \varphi(x - ct), \quad (x,t) \in \mathbb{R} \times \mathbb{R}$.

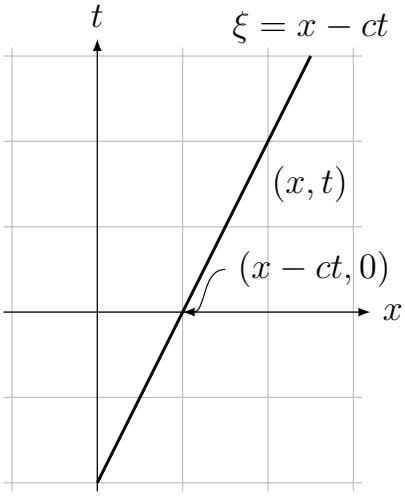
2.1.3. Charakteristiken



Sei $\xi \in \mathbb{R}$, dann folgt aus der allgemeinen Lösung, vergleiche Lemma 2.1.1, dass jede Lösung $u(x,t)$ auf der Geraden C_ξ mit
$C_\xi := \left\{(x,t) \in \mathbb{R}^2 : x - ct = \xi\right\}$ $\xi = konstant$
und gegeben, konstant. $(x,t) \in C_\xi$
$\Rightarrow u(x,t) = \varphi(x - ct) = \varphi(\xi) = konstant$
Die Geraden C_ξ heißen **Charakteristiken**.

Als nächstes betrachten wir **Anfangswert** und **Anfangs-Randwertproblem**:

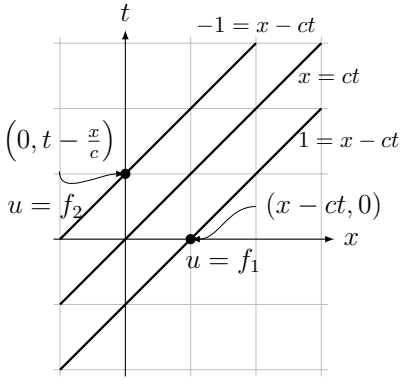
1. Anfangswertproblem:



Gegeben sei $u_t + cu_x = 0$
$(x,t) \in \mathbb{R} \times \mathbb{R}_+$ mit den Anfangswerten
$u(x,0) = f_1(x), \quad x \in \mathbb{R}$ $u(x,t) = konstant$
$\forall (x,t) \in C_\xi \Rightarrow (x-ct,0) \in C_\xi$,
da $(x-ct) - c \cdot 0 = x - ct = \xi$
$\Rightarrow u(x,t) = u(x-ct,0) = f_1(x-ct)$

Fazit: Anfangswerte pflanzen sich entlang der Charakteristiken fort.

2. Erstes Anfangs-Randwertproblem:

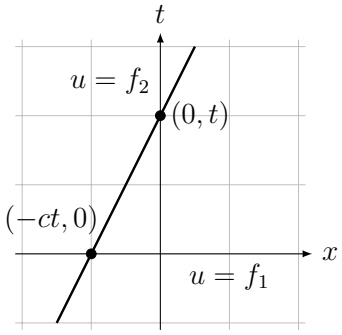


Gegeben sei $u_t + cu_x = 0$, $(x,t) \in \mathbb{R}_+ \times \mathbb{R}_+$
mit dem Anfangswert: $u(x,0) = f_1(x)$, $x \in \mathbb{R}_+$
und mit dem Randwert: $u(0,t) = f_2(t)$, $t \in \mathbb{R}_+$

Hierfür ergibt sich die Lösung:

$$u(x,t) = u(x-ct,0) \quad = f_1(x-ct) \qquad x \geq ct$$
$$u(x,t) = u\left(0, t - \frac{x}{c}\right) = f_2\left(x - \frac{t}{c}\right) \qquad x < ct$$

3. Zweites Anfangs-Randwertproblem:



$u_t + cu_x = 0, \quad (x,t) \in \mathbb{R} \times \mathbb{R}_+$
Anfangswert: $u(x,0) = f_1(x), \quad x \in \mathbb{R}$
Randwert: $u(0,t) = f_2(t), \quad t \in \mathbb{R}_+$

Da $(0,t)$ und $(-ct,0)$ auf der selben Charakteristik liegen, gilt:

$$f_2(t) = u(0,t) = u(-ct,0) = f_1(-ct)$$

Bei gegebenem f_1 kann f_2 nicht frei gewählt werden und umgekehrt. Das zweite Anfangs-Randwertproblem ist also möglicherweise wiedersprüchlich.
$\Rightarrow$ nicht jedes Anfangs-Randwertproblem ist sinnvoll.

Definition 2.1.1 (sachgemäß gestelltes Problem):
Ein Anfangswertproblem oder ein Anfangs-Randwertproblem heißt ***sachgemäß gestellt*** *(gut gestellt) (*engl. properly posed, well posed*), wenn gilt:*

1. ***Existenz:*** *Es gibt mindestens eine Lösung*
2. ***Eindeutigkeit:*** *Es gibt höchstens eine Lösung*
3. ***stetige Abhängigkeit von den Anfangs- und Randwerten:*** *Die Lösung u ist von den Anfangs- und Randwerten stetig abhängig, z. B. von f_1, f_2*

Bemerkung 2.1.1:
Die Punkte eins bis drei hängen natürlich davon ab,

- in welchem Raum die Lösungen liegen sollen
- in welcher Topologie die stetige Abhängigkeit gelten soll

Für die Gleichung $u_t + cu_x = 0$ gilt:

1. das betrachtete Anfangswertproblem ist gut gestellt
2. das erste Anfangs-Randwertproblem ist gut gestellt
3. das zweite Anfangs-Randwertproblem ist im Allgemeinen nicht gut gestellt (schlecht gestellt)

15.11.2012
11. Vorlesung
Einführung in C-Prog.

2.1.4. Charakteristiken für die Anfangswertaufgabe der Advektionsgleichung

20.11.2012
12. Vorlesung

Die homogene Advektionsgleichung

$$\begin{cases} u_t(x,t) + c(x,t)u_x(x,t) = 0, & x \in \mathbb{R}, \quad t > 0 \\ u(x,0) = \varphi(x), & x \in \mathbb{R} \end{cases} \tag{2.2}$$

Voraussetzung: $c(x,t)$ und $\varphi(x)$ genügend glatt.
Anfangswertprobleme heißen auch **Cauchyprobleme**.

Charakteristikenverfahren

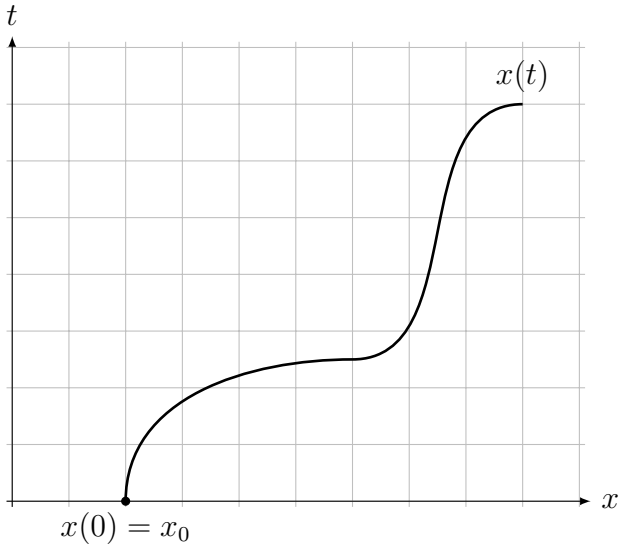
Die Charakteristiken des Anfangswertproblem (2.2) sind Kurven in der x-t-Ebene, die wie folgt definiert werden:

$$\begin{cases} \dfrac{\mathrm{d}x(t)}{\mathrm{d}t} = c(x(t),t), & t > 0 \\ x(0) = x_0 \end{cases} \tag{2.3}$$

Die Lösung $x = x(t)$ dieser Gleichung definiert eine Kurve

$$\Gamma := \Big\{(x(t),t), \quad t > 0\Big\}$$

mit dem Anfangspunkt $(x_0, 0)$ zum Zeitpunkt $t = 0$.



Diese Kurven nennt man **Charakteristiken**. Entlang dieser Charakteristiken für u bzw. $u(x(t),t)$ gilt:

$$\begin{aligned}\frac{\mathrm{d}u(x(t),t)}{\mathrm{d}t} &= u_t + u_x\frac{\mathrm{d}x(t)}{\mathrm{d}t} = u_t + u_x c(x,t) = 0\\ &\Rightarrow u(x(t),t) = \textit{konstant} \text{ auf } \Gamma\\ &\Rightarrow u(x,t) = u(x_0,0) = \varphi(x_0)\end{aligned}$$

Fazit: Wir können das Cauchyproblem (2.2) lösen, indem wir die gewöhnliche Differentialgleichung (2.3) lösen.

Beispiel 2.1.1:

$$\begin{cases} u_t + xu_x = 0, & x \in \mathbb{R},\ t > 0\\ u(x,0) = \varphi(x), & x \in \mathbb{R}\end{cases}$$

Für die Charakteristik ergibt sich $\begin{cases}\dfrac{\mathrm{d}x(t)}{\mathrm{d}t} = x(t), & t > 0\\ x(0) = x_0\end{cases}$

Als Lösung erhalten wir:

$$x(t) = x_0 \cdot e^t \Leftrightarrow x_0 = x \cdot e^{-t}$$

Somit ergibt sich für die Lösung von (2.2): $\boxed{u(x,t) = \varphi(x_0) = \varphi\left(xe^{-t}\right)}$

Die inhomogene Advektionsgleichung

$$\begin{cases} u_t + a(x,t)u_x = b(x,t), & x \in \mathbb{R},\ t > 0\\ u(x,0) = \varphi(x), & x \in \mathbb{R}\end{cases}$$

Die Lösungen werden wieder als Lösung folgender gewöhnlichen Differentialgleichung definiert:

$$\frac{\mathrm{d}x(t)}{\mathrm{d}t} = a(x,t), \quad t > 0, \quad x(0) = x_0$$

Wie zuvor betrachten wir $u(x(t),t)$ auf der Charakteristik.

$$\begin{aligned}\frac{\mathrm{d}}{\mathrm{d}t}u\Big(x(t),t\Big) &= u_t + \frac{\mathrm{d}x}{\mathrm{d}t}u_x \\ &= u_t + a(x,t)u_x \\ &= b\Big(x(t),t\Big)\end{aligned}$$

Die Lösung entlang der Charakteristik $(x(t),t)$ wird gegeben durch:

$$\boxed{u\Big(x(t),t\Big) = \varphi(x_0) + \int_0^t b\Big(x(s),s\Big)\,\mathrm{d}s}$$

Beispiel 2.1.2:

$$\begin{cases} u_t + u_x = x, & x \in \mathbb{R},\ t > 0 \\ u(x,0) = \varphi(x), & x \in \mathbb{R} \end{cases}$$

Die Charakteristiken sind definiert durch $\begin{cases} \dfrac{\mathrm{d}x(t)}{\mathrm{d}t} = 1, & t > 0 \\ x(0) = x_0 \end{cases}$

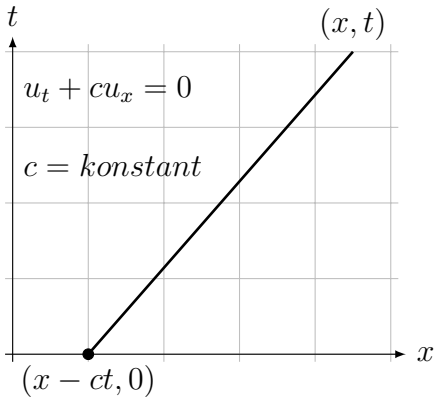
$$\begin{aligned}\Rightarrow x(t) = x_0 + t \Rightarrow u\Big(x(t),t\Big) &= \varphi(x_0) + \int_0^t x(s)\,\mathrm{d}s \\ &= \varphi(x_0) + x_0 t + \frac{1}{2}t^2\end{aligned}$$

Da $x_0 = x - t$, folgt

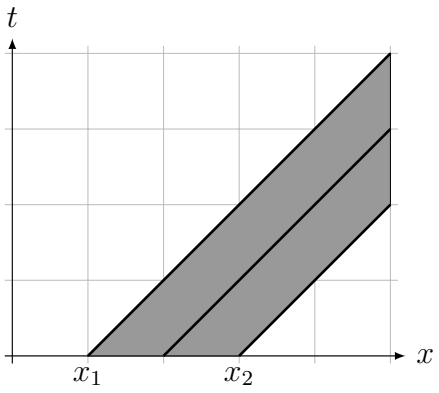
$$\begin{aligned}u(x,t) &= \varphi(x-t) + (x-t)t + \frac{1}{2}t^2 \\ &= \boxed{\varphi(x-t) + xt - \frac{1}{2}t^2}\end{aligned}$$

Abhängigkeits- und Bestimmtheitsbereich

Abbildung 2.1.: Abhängigkeits- und Bestimmtheitsbereich



(a) Abhängigkeitsbereich



(b) Bestimmtheitsbereich

(a) Die Lösung von u hängt im Punkt (x,t) nur von $(x-ct,0)$ ab, d. h. der **Abhängigkeitsbereich** (*engl. domain of dependence*) des Punktes (x,t) ist der Punkt $(x-ct,0)$.

Allgemeiner gilt:
Der **Abhängigkeitsbereich** eines Punktes (x,t) ist die Punktmenge zum Zeitpunkt $t=0$, von der allein der Wert der Lösung in (x,t) abhängt, d. h. eine Änderung der Anfangswerte außerhalb des Abhängigkeitsbereiches beeinflusst nicht den Wert $u(x,t)$ *[siehe 13, Seite 307]*.

(b) Die Werte von $u(x,0)$ für $x \in [x_1,x_2]$ bestimmen die Lösung im Streifen

$$\Omega := \Big\{(x,t) : x_1 - ct \le x - ct \le x_2 - ct\Big\}$$

d. h. Ω ist der **Bestimmtheitsbereich** (*engl. domain of influence/determinance*).

Allgemeiner gilt:
Der **Bestimmtheitsbereich** von I ist das Gebiet $\Omega \subset \mathbb{R}^2$, in dem dann die Lösung durch die Anfangswerte $u(x,0), \quad x \in I$ bestimmt ist.

2.1.5. Differenzenverfahren für die Advektionsgleichung

Approximation von Ableitungen durch Differenzenquotienten

Sei $u \in \mathcal{C}^2(G), \quad G \subset \mathbb{R}^2$
Taylorentwicklung:

$$\begin{aligned} u(x+h,t) &= u(x,t) + u\frac{\partial u}{\partial x}(x,t) + \frac{u^2}{2}\frac{\partial^2 u}{\partial x^2}(x,t) + \dots \\ \Rightarrow \frac{\partial u}{\partial x}(x,t) &= \underbrace{\frac{u(x+h,t)-u(x,t)}{h}}_{=:D_{+x,h}u(x,t)} + \mathcal{O}(h) \end{aligned}$$

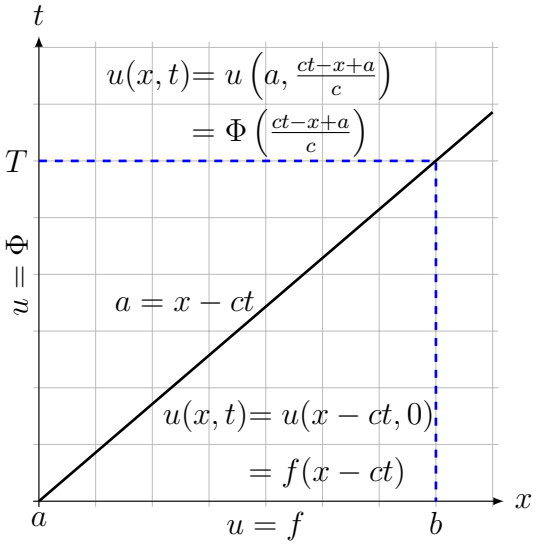
Analog:

$$\begin{aligned} \frac{\partial u}{\partial x}(x,t) &= \underbrace{\frac{u(x,t)-u(x-h,t)}{h}}_{=:D_{-x,h}u(x,t)} + \mathcal{O}(h) \\ \frac{\partial u}{\partial x}(x,t) &= \underbrace{\frac{u(x+h,t)-u(x-h,t)}{2h}}_{=:D_{0x,h}u(x,t)} + \mathcal{O}\left(h^2\right) \\ \frac{\partial u}{\partial x}(x,t) &= \frac{u(x,t+h)-u(x,t)}{h} + \mathcal{O}(h) \end{aligned}$$

Manchmal schreiben wir auch D_{0x} statt $D_{0x,h}$ etc.

Das Modellproblem (Anfangs-Randwertproblem)

$$\begin{aligned} u_t + cu_x &= 0, && (x,t) \in [a,b] \times [0,T] \\ u(x,0) &= f(x), && x \in [a,b] \\ u(a,t) &= \Phi(t), && t \in [0,T] \end{aligned}$$



Es seien nun $M, N \in \mathbb{N}$ und

$$
\begin{aligned}
h := \Delta x &:= \frac{b-a}{N} \\
k := \Delta t &:= \frac{T}{M} \\
x_i &:= a + ih, \quad i = 0, 1, \ldots, N \\
t_j &:= jk, \quad j = 0, 1, \ldots, M \\
G &:= (a,b) \times (0,T) \subset \mathbb{R}^2 \\
G_{h,k} &:= \left\{(x_i, t_j) \colon i = 0, 1, \ldots, N, \quad j = 0, 1, \ldots, M\right\} = \text{Die Menge der Gitterpunkte in } \overline{G}
\end{aligned}
$$

Die exakte Lösung $u(x,t)$ des Modellproblems wird in den Gitterpunkten durch eine Näherung $u^{h,k}$ approximiert:

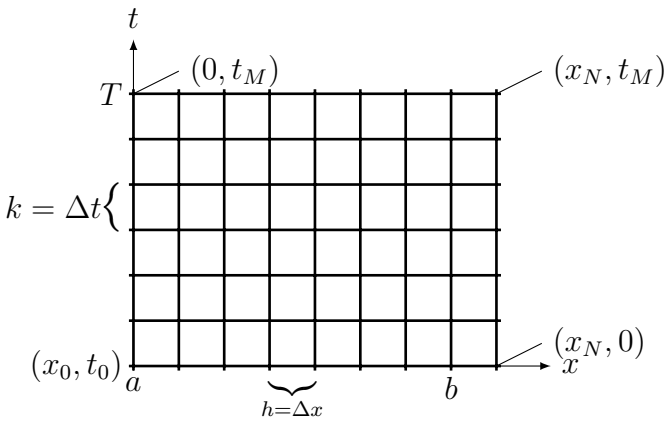
$$u(x_i, t_j) \approx u^{h,k}(x_i, t_j) =: u_{i,j}^{h,k}$$

Die Werte

$$
\begin{aligned}
u_{i0}^{h,k} &:= u(x_i, 0) = f(x_i), \quad i = 0, 1, \ldots, N \text{ und} \\
u_{0j}^{h,k} &:= u(0, t_j) = \Phi(t_j), \quad j = 0, 1, \ldots, M
\end{aligned}
$$

sind bekannt.

Das Gitter $G_{h,k}$:



Ein erstes Differenzenverfahren (Raumvorwärtsverfahren)

27.11.2012
14. Vorlesung

Sei $u(x,t)$ eine Lösung der Advektionsgleichung: $u_t + cu_x = 0$
$\Rightarrow D_{+t,k}u^{h,k}(x_i,t_j) + cD_{+x,h}u^{h,k}(x_i,t_j) = \underbrace{\cancel{\mathcal{O}(h)} + \cancel{\mathcal{O}(k)}}_{=0}, \quad i = 0,1,\ldots,N, \quad j = 0,1,\ldots,M$

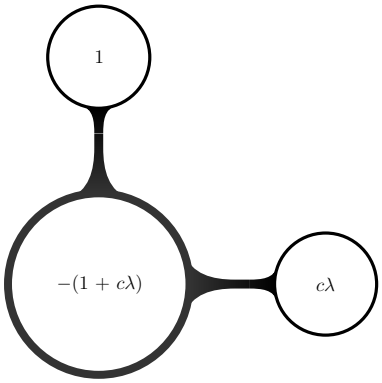
$$\begin{aligned} & D_{+t}u_{i,j}^{h,k} + cD_{+x,h}u_{i,j}^{h,k} = 0 \\ \Leftrightarrow \quad & \frac{u_{i,j+1}^{h,k} - u_{i,j}^{h,k}}{k} + c\frac{u_{i+1,j}^{h,k} - u_{i,j}^{h,k}}{h} = 0 \end{aligned} \tag{2.4}$$

Annahme: $\frac{k}{h} = konstant =: \lambda \in \mathbb{R}$

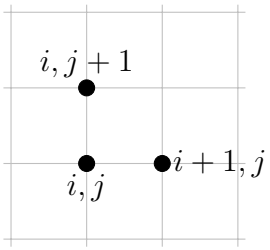
Dann genügt es u^h statt $u^{h,k}$ zu schreiben.

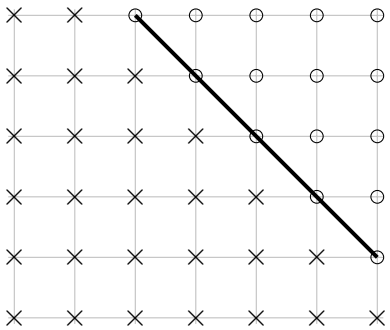
$$\begin{aligned} \Leftrightarrow \quad u_{i,j+1}^{h,k} &= u_{i,j}^{h,k} - c\underbrace{\frac{k}{h}}_{=\lambda}\left(u_{i+1,j}^{h,k} - u_{i,j}^{h,k}\right) \\ &= (1+c\lambda)u_{i,j}^{h,k} - c\lambda u_{i+1,j}^{h,k} \\ \overset{(2.4)}{\Rightarrow} \quad u_{i,j+1}^{h} &- (1+c\lambda)u_{i,j}^{h} + c\lambda u_{i+1,j}^{h} = 0 \end{aligned}$$

„Molekül“:



$\widehat{=}$





$\times$ = bekannt oder berechenbar
$\circ$ = unbekannt oder nicht berechenbar

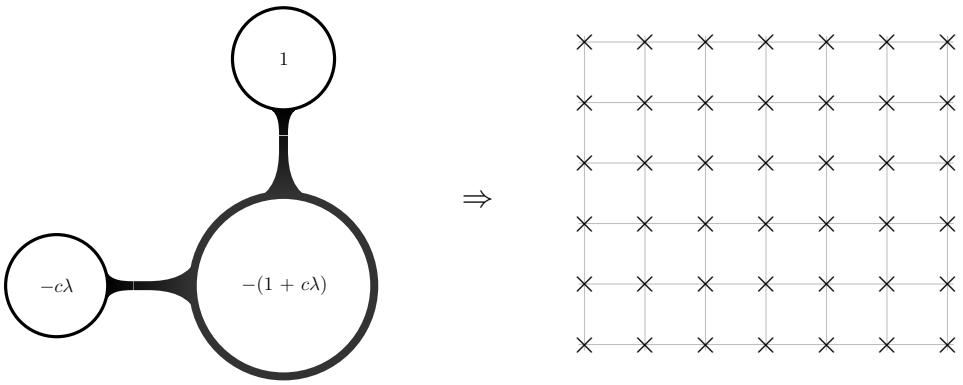
Dieser Ansatz ist offensichtlich nicht geeignet zur Lösung des Anfangswertproblem der Advektionsgleichung.

Ein zweites Differenzenverfahren (Raumrückwärts)

Hier approximieren wir die Raumableitung u_x durch $D_{-x}u$

$$\begin{aligned} \Rightarrow \quad & D_{+t}u^h + cD_{-x}u^h = 0 \\ \Leftrightarrow \quad & \frac{u^h_{i,j+1} - u^h_{i,j}}{k} - c\frac{u^h_{i,j} - u^h_{i-1,j}}{h} = 0 \\ \overset{\lambda=\frac{k}{h}}{\Leftrightarrow} \quad & u^h_{i,j+1} - (1 - c\lambda)u^h_{i,j} - c\lambda u^h_{i-1,j} = 0 \end{aligned}$$

„Molekül“:



Fazit: Nicht jeder Differenzenansatz funktioniert für das betrachtete Modellproblem.

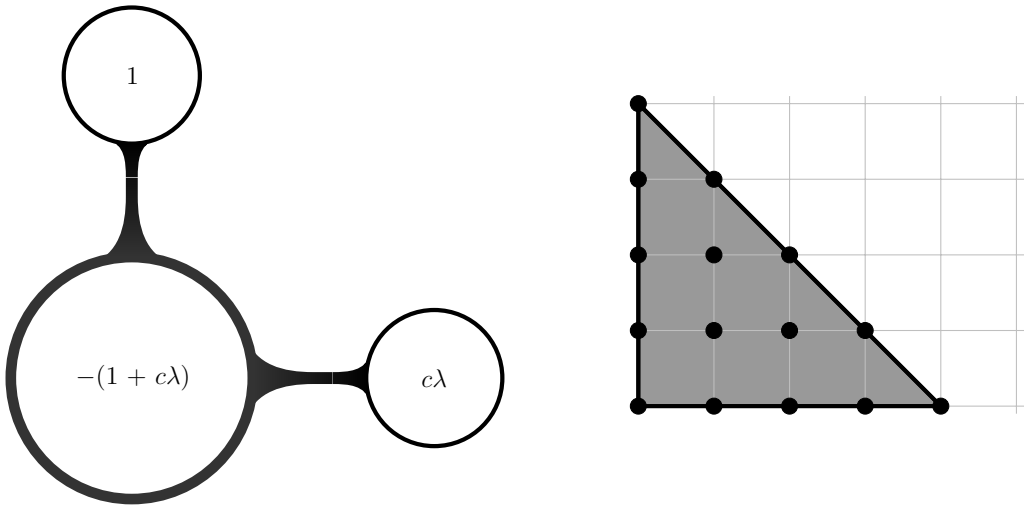
2.1.6. Die Courant-Friedrichs-Lewy-Bedingung

Der Abhängigkeitsbereich der Advektionsgleichung $u_t + cu_x = 0$ im Punkt (x, t) ist:

$$AB\Big(u, (x,t)\Big) = (x - ct, 0)$$

Auch für die Differenzengleichung lässt sich ein Abhängigkeitsbereich angeben. Betrachten wir zunächst das Raumvorwärtsverfahren:

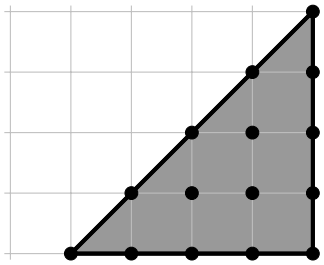
$$u^h_{i,j+1} - (1 + c\lambda)u^h_{i,j} + c\lambda u^h_{i+1,j} = 0$$



Der Wert $u^h_{i,j+1}$ im Punkt $(x_i, t_j + 1)$ hängt zunächst von den beiden Gitterpunkten (x_i, t_j) und (x_{i+1}, t_j) ab. Durch rekursives vorgehen sehen wir, dass der Abhängigkeitsbereich von u^h im Punkt (x_i, t_{j+1}) das Intervall $[x_i, x_{i+j+1}]$ auf der x-Achse ist. Aus $x_i = a + ih$ und $t_j = jk$ mit $\lambda = \frac{k}{h}$ folgt für $(x,t) \in G_h$

$$AB\Big(u^h, (x,t)\Big) : \left\{(\overline{x}, 0) : x \leq \overline{x} \leq x + \frac{t}{\lambda}\right\}$$

Für das zweite Differenzenverfahren (Raumrückwärts) ergibt sich analog:



$u^h_{i,j+1} - (1 - c\lambda)u^h_{i,j} - c\lambda u^h_{i-1,j} = 0$
$\Rightarrow$ für $(x,t) \in G_h$

$$AB(u^h, (x,t)) = \left\{(\overline{x}, 0) : x - \frac{t}{\lambda} \leq \overline{x} \leq x\right\}$$

Satz 2.1.1 (CFL-Bedingung):
Für die Konvergenz eines Differenzenverfahrens ist es notwendig, dass der Abhängigkeitsbereich der Differentialgleichung im Abschluss des Abhängigkeitsbereich der Differenzengleichung beim Grenzübergang von h gegen 0 enthalten ist. Für alle $(x,t) \in G$ gilt dann:
Ist $(x_h, t_h) \in G_h$ eine Folge von Gitterpunkten mit $\lim_{h\to 0}(x_h, t_h) = (x,t)$ und $\xi \in AB(u,(x,t))$, dann gibt es für alle $\varepsilon > 0$ ein $\delta > 0$, so dass

$$B_\varepsilon(\xi) \cap AB\Big(u^h, (x_h, t_h)\Big) \neq \emptyset$$

für $0 < h < \delta$, wobei $B_\varepsilon(\xi)$ der Kreis um ξ mit Radius ε ist.

Beweis:
Angenommen, die CFL-Bedingung ist nicht erfüllt.
$\Rightarrow \exists (x,t) \in G$ und $\xi \in AB(u,(x,t))$, sowie eine Folge $(x_h, t_h) \in G_h$ mit $(x_h, t_h) \xrightarrow{h\to 0} (x,t)$ und ein $\varepsilon > 0$, so dass

$$B_\varepsilon(\xi) \cap AB\Big(u^h, (x_h, t_h)\Big) = \emptyset \quad \forall h > 0$$

Hieraus folgt, dass Änderungen der Anfangs- und Randwerte in $B_\varepsilon(\xi)$ die Lösung u in (x,t) ändert, aber nicht die Approximation u^h in (x_h, t_h). Es ist möglich, die Anfangs-Randwerte so zu wählen, dass der Limes $\lim_{h\to 0} u^h(x_h, t_h)$ entweder nicht existent oder nicht gegen $u(x,t)$ konvergiert.

□

Bemerkung 2.1.2:
Diese Bedingung geht zurück auf eine Arbeit von R. Courant, K.O. Friedrichs und H. Lewy *[siehe 14]*.

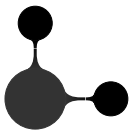
Beispiel 2.1.3:

29.11.2012
15. Vorlesung

$$u_t + cu_x = 0$$
$$AB\Big(u,(x,t)\Big) = (x - ct, 0)$$

1. Zeitvorwärts/Raumvorwärts:

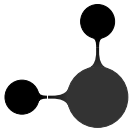
$$u^h_{i,j+1} - (1 + c\lambda)u^h_{i,j} + c\lambda u^h_{i+1,j} = 0$$



$AB\Big(u^h,(x,t)\Big) = \Big\{(\overline{x},0)\colon x \le \overline{x} \le x + \frac{t}{\lambda}\Big\} = \Big[x, x + \frac{t}{\lambda}\Big], \quad \lambda = \frac{k}{h}$
Überprüfen der CFL-Bedingung:
(x,t) gegeben. $\xi = (x - ct, 0), \quad \varepsilon = \frac{ct}{2}, \quad (x_h, t_h) \xrightarrow{h\to 0} (x,t)$
$B_\varepsilon(\xi) \cap \Big[x, x + \frac{t}{\lambda}\Big] = \emptyset \quad \forall h > 0$

2. Das zweite Differenzenverfahren (Raumrückwärts):

$$u^h_{i,j+1} - (1 - c\lambda)u^h_{i,j} - c\lambda u^h_{i-1,j}$$



Hierfür ergibt sich:
$AB\Big(u^h,(x,t)\Big) = \Big\{(\overline{x},0)\colon x - \frac{t}{\lambda} \le \overline{x} \le x\Big\} = \Big[x - \frac{t}{\lambda}, x\Big]$
Es gilt:

$$\begin{aligned} x - ct \in \Big[x - \frac{t}{\lambda}, x\Big] \\ \Leftrightarrow \quad x - \frac{t}{\lambda} \le x - ct \le x \\ \Leftrightarrow \quad 0 \le ct \le \frac{t}{\lambda} \\ \Leftrightarrow \quad 0 \le c\lambda \le 1 \end{aligned}$$

CFL-Bedingung für das Zeitvorwärts/Raumrückwärts-Verfahren lautet:

$$\boxed{c\lambda \le 1}, \quad \lambda = \frac{k}{h} \Rightarrow c\frac{k}{h} \le 1$$

Die CFL-Bedingung ist nur eine notwendige Bedingung, daher
Konvergenzbeweis bezüglich $\|\cdot\|_\infty$.
Sei $u^h(x,t)$ die numerische Lösung, die mit dem Verfahren

$$D_{+t}u^h + cD_{-x}u^h = 0$$

berechnet wird. Wie gezeigt ist die CFL-Bedingung $c\lambda \le 1$.

Notation:

$u_{i,j}$	:=	$u(x_i,t_j)$	=	exakte Lösung in (x_i,t_j)
$e_{i,j}$	:=	$e^h_{i,j} = u_{i,j} - u^h_{i,j}$	=	Fehler in (x_i,t_j)

Satz 2.1.2:
Sei $u \in \mathcal{C}^2(\overline{G})$ eine Lösung von

$$\begin{aligned} u_t + cu_x &= 0, && (x,t) \in G = (a,b) \times (0,T) \\ u(x,0) &= f(x), && x \in [a,b] \\ u(0,t) &= \Phi(t) = 0, && t \in [0,T] \end{aligned}$$

Weiterhin sei u^h die numerische Approximation, die durch das Verfahren

$$D_{+t}u^h + cD_{-x}u^h = 0$$

berechnet wird und es gelte $c\lambda \leq 1$.
Dann gilt:

$$|u^h_{i,j} - u_{i,j}| \leq (j \cdot \Delta t) M_2 (c+\lambda)\frac{\Delta x}{2} \text{ mit } M_2 := \sup_{\overline{G}} \max_{p+q=2} \left| \frac{\partial^2 u}{\partial x^p \partial t^q} \right|$$

Beweis:
Taylorentwicklung ergibt:

$$\frac{u_{i,j+1} - u_{i,j}}{\Delta t} + c\frac{u_{i,j} - u_{i-1,j}}{\Delta x} = f_{i,j} \tag{2.5}$$

mit $1 \leq i \leq N, \quad 1 \leq j \leq M-1$ und

$$f_{i,j} = \frac{\partial^2 u}{\partial t^2}(x_i, \tau)\frac{\Delta t}{2} - c\frac{\partial^2 u}{\partial x^2}(\xi, t_i)\frac{\Delta x}{2}$$

wobei $\tau \in (t_j, t_{j+1})$ und $\xi \in (x_{i-1}, x_i)$.
Dann gilt:

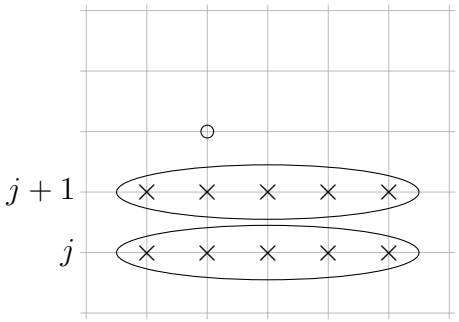
$$|f_{i,j}| \leq F := M_2(\lambda + c)\frac{\Delta x}{2}$$

Die Approximation u^h erfüllt:

$$\frac{u^h_{i,j+1} - u^h_{i,j}}{\Delta t} + c\frac{u^h_{i,j} - u^h_{i-1,j}}{\Delta x} = 0 \tag{2.6}$$

$$\begin{aligned} &\overset{(2.5),\,(2.6)}{\Rightarrow} && \frac{e_{i,j+1} - e_{i,j}}{\Delta t} + c\frac{e_{i,j} - e_{i-1,j}}{\Delta x} = f_{i,j} \\ &\Rightarrow && e_{i,j+1} = (1 - c\lambda)e_{i,j} + c\lambda e_{i-1,j} + \Delta t f_{i,j} \end{aligned}$$

Zusätzlich gilt:
$$\begin{aligned} e_{i,0} &:= u_{i,0} - u^h_{i,0} = 0 \quad 0 \leq i \leq N \\ e_{0,j} &:= u_{0,j} - u^h_{0,j} = 0 \quad 0 \leq j \leq M \end{aligned}$$



$$E_j := \max_{0\le i\le N} |e_{i,j}| \text{ (maximaler Fehler im Zeitschritt } j)$$

$$\begin{aligned} \Rightarrow \quad |e_{i,j+1}| &\le |1-c\lambda|E_j + |c\lambda|E_j + \Delta t\cdot F \\ \Rightarrow \quad E_{j+1} &\overset{0<c\lambda\le 1}{\le} E_j + \Delta t\cdot F \end{aligned}$$

Es gilt: $E_0 = \max_{0\le i\le N}|e_{i,0}| = 0$

$$\begin{aligned} \Rightarrow E_j &\le j\Delta t F \\ &= j\Delta t M_2(\lambda + c)\frac{\Delta x}{2} \end{aligned}$$

□

2.1.7. Das Lax-Wendroff-Verfahren

Betrachte das Modellproblem (Anfangswertproblem)

$$\begin{cases} u_t + cu_x = 0 & (x,t)\in(a,b)\times(0,T) \\ u(x,0) = f(x) & x\in[a,b] \end{cases}$$

Annahme: Lösung genügend glatt.

Taylorentwicklung um t:

$$u(x,t+\Delta t) = u(x,t) + \Delta t\frac{\partial u}{\partial t}(x,t) + \frac{\Delta t^2}{2}\frac{\partial u^2}{\partial t^2}(x,t) + \mathcal{O}\left((\Delta t)^3\right) \tag{2.7}$$

Aus der Differentialgleichung $u_t + cu_x = 0$ folgt: $u_t = -cu_x$

$$\begin{aligned} \Rightarrow u_{tt} = \frac{\partial^2 u}{\partial t^2} &= \frac{\partial}{\partial t}\left(\frac{\partial u}{\partial t}\right) = -c\frac{\partial}{\partial t}\frac{\partial u}{\partial x} \\ &= -c\frac{\partial}{\partial x}\frac{\partial u}{\partial t} = c^2 u_{xx} \end{aligned}$$

Ersetzt man in der Taylorentwicklung (2.7) u_t durch $-cu_x$ und u_{tt} durch c^2u_{xx} so erhält man:

$$u(x,t+\Delta t) = u(x,t) - c\Delta t u_x(x,t) + c^2\frac{\Delta t^2}{2}u_{xx}(x,t) + \mathcal{O}\left((\Delta t)^3\right)$$

$$\overset{\substack{u(x_i,t_j)=u_{i,j}\\ \Leftrightarrow \\ k=\Delta t}}{} u_{i,j+1} = u_{i,j} - cku_x(x_i,t_j) + \frac{c^2k^2}{2}u_{xx}(x_i,t_j) + \mathcal{O}\left(k^3\right) \tag{2.8}$$

Nächster Schritt: Approximieren der Ableitung nach x

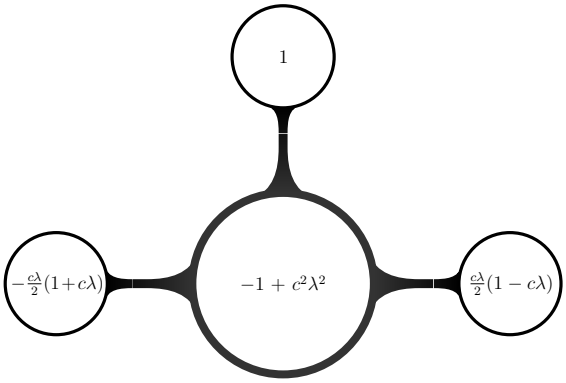
$\overset{h=\Delta x}{\Rightarrow}$

1. $\frac{\partial u}{\partial x}(x_i,t_j) = D_{0x,h}u(x_i,t_j) + \mathcal{O}(h^2) = \frac{u_{i+1,j}-u_{i-1,j}}{2h} + \mathcal{O}(h^2)$
2. $\frac{\partial^2 u}{\partial x^2}(x_i,t_j) = \frac{u_{i+1,j}-2u_{i,j}+u_{i-1,j}}{h^2} + \mathcal{O}(h^2) = D_{+x}D_{-x}u^h + \mathcal{O}(h^2)$
   $\overset{(2.8)}{\Rightarrow} u_{i,j+1} = u_{i,j} - ckD_{0x}u_{i,j} + \frac{c^2k^2}{2}D_{+x}D_{-x}u_{i,j} + \underbrace{\mathcal{O}\left(k^3\right) + \mathcal{O}\left(kh^2\right) + \mathcal{O}\left(k^2h^2\right)}_{\text{vernachlässigen}}$

Fortlassen der Fehlerterme höherer Ordnung ergibt das **Lax-Wendroff-Verfahren**:

$$u_{i,j+1}^h = u_{i,j}^h - ck\frac{u_{i+1,j}^h - u_{i-1,j}^h}{2h} + \frac{(ck)^2}{2}\frac{u_{i+1,j}^h - 2u_{i,j}^h + u_{i-1,j}^h}{h^2}$$
$$\overset{\lambda=\frac{k}{h}}{\Leftrightarrow} u_{i,j+1}^h = \frac{c\lambda}{2}(1+c\lambda)u_{i-1,j}^h + (1-c^2\lambda^2)u_{i,j}^h - \frac{c\lambda}{2}(1-c\lambda)u_{i+1,j}^h$$

„Molekül“:



Bemerkung 2.1.3:

1. Für das Lax-Wendroff-Verfahren gilt die CFL-Bedingung $|c\lambda| \leq 1$ und es konvergiert unter dieser Bedingung. Die CFL-Bedingung ist hier also auch hinreichend.
2. Das Lax-Wendroff-Verfahren konvergiert mit der Ordnung 2 ($e^h \in \mathcal{O}(h^2 + k^2)$).
3. Wenn das Modellproblem (Anfangs-Randwertproblem) betrachtet wird, muss auch $u(b,t)$ vorgeschrieben werden, obwohl dies für die analytische Lösung nicht notwendig ist, vgl. auch Raumvorwärts.

Literatur: [15].

2.1.8. Zusammenfassung einfacher Differenzenverfahren

04.12.2012
16. Vorlesung

1. Zeit-Vorwärts-Raum-Vorwärts-Verfahren

$$\underbrace{\frac{u_{i,j+1}^h - u_{i,j}^h}{k}}_{D_{+t}u_{i,j}^h} + c\underbrace{\frac{u_{i+1,j}^h - u_{i,j}^h}{h}}_{cD_{+x}u_{i,j}^h} = 0$$
$$\Leftrightarrow u_{i,j+1}^h = (1+c\lambda)u_{i,j}^h - c\lambda u_{i+1,j}^h$$

   Nachteil: Erfüllt die CFL-Bedingung **nicht**! Dieses Verfahren sollte nicht verwendet werden.

2. Zeit-Vorwärts-Raum-Rückwärts-Verfahren

$$\underbrace{\frac{u_{i,j+1}^h - u_{i,j}^h}{k}}_{D_{+t}u_{i,j}^h} + c\underbrace{\frac{u_{i,j}^h - u_{i-1,j}^h}{h}}_{cD_{-x}u_{i,j}^h} = 0$$
$$\Leftrightarrow u_{i,j+1}^h = (1-c\lambda)u_{i,j}^h + c\lambda u_{i-1,j}^h$$

3. Zeit-Vorwärts-Raumzentriert-Verfahren

$$\underbrace{\frac{u_{i,j+1}^h - u_{i,j}^h}{k}}_{D_{+t}u_{i,j}^h} + \underbrace{c\frac{u_{i+1,j}^h - u_{i-1,j}^h}{h}}_{cD_{0x}u_{i,j}^h} = 0$$

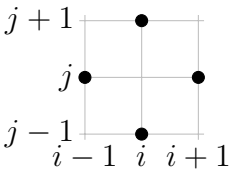
$$\Leftrightarrow u_{i,j+1}^h = u_{i,j}^h - \frac{c\lambda}{2}(u_{i+1,j}^h - u_{i-1,j}^h)$$

4. Lax-Friedrichs-Verfahren

$$\frac{u_{i,j+1}^h - \frac{1}{2}(u_{i-1,j}^h + u_{i+1,j}^h)}{k} + c\frac{u_{i+1,j}^h - u_{i-1,j}^h}{2h} = 0$$

5. Leap-Frog-Verfahren

$$\frac{u_{i,j+1}^h - u_{i,j-1}^h}{2k} + c\frac{u_{i+1,j}^h - u_{i-1,j}^h}{2h} = 0$$



6. Lax-Wendroff-Verfahren

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

2.2. Die Wärmeleitgleichung

2.2.1. Modellgleichung

$$u_t = u_{xx}, \quad (x,t) \in (a,b) \times (c,d)$$

Verwandte Gleichungen:
Viele Gleichungen lassen sich auf die Modellgleichung reduzieren:

1. $u_t = au_{xx},\ a = konstant$
   Transformation:

$$\begin{aligned} \tau := at \Leftrightarrow t &= \frac{\tau}{a} \\ \Rightarrow u_\tau = \frac{\partial}{\partial\tau}u(x,\tau) &= u_t(x,t)\frac{\partial t}{\partial\tau} = \frac{1}{a}\underbrace{u_t(x,t)}_{=a\cdot u_{xx}} = u_{xx} \\ \Rightarrow u_\tau &= u_{xx} \end{aligned}$$

2. $u_t = a(t)u_{xx},\ a(t) > 0$
   Transformation:

$$\begin{aligned} A(t) &= \int_0^t a(\eta)\,\mathrm{d}\eta, \quad t = \Phi(\tau) \\ U(x,\tau) &:= u\Big(x, \Phi(\tau)\Big) \\ U_\tau(x,\tau) &= \frac{\partial u}{\partial\tau}\Big(x,\Phi(\tau)\Big) = u_t(x,t)\frac{\partial\Phi(\tau)}{\partial\tau} \\ &= u_t\left(\frac{\partial A(t)}{\partial\tau}\right)^{-1} = u_t\frac{1}{a(t)} = u_{xx}(x,t) = u_{xx}\Big(x,\Phi(\tau)\Big) = U_{xx}(x,\tau) \end{aligned}$$

2.2.2. Analytische Lösung der Wärmeleitungsgleichung - Trennung der Veränderlichen

Wir betrachten $u_t = u_{xx}$ und machen folgenden **Ansatz**:

$$u(x,t) = X(x)T(t)$$

Notation:

$$\frac{d^2X}{dx} = X'', \quad \frac{dT}{dt} = \dot{T}$$

$$\Rightarrow u_t = u_{xx} \Leftrightarrow \underbrace{\frac{\dot{T}}{T}}_{\text{hängt von } t \text{ ab}} = \underbrace{\frac{X''}{X}}_{\text{hängt von } x \text{ ab}} = \underbrace{\lambda}_{\text{unabhängig von } x,t} = konstant$$

Hieraus ergeben sich die beiden gewöhnlichen Differentialgleichungen

$$\begin{aligned} X'' - \lambda X &= 0 \\ \dot{T} - \lambda T &= 0 \end{aligned} \tag{2.9}$$

mit den allgemeinen Lösungen

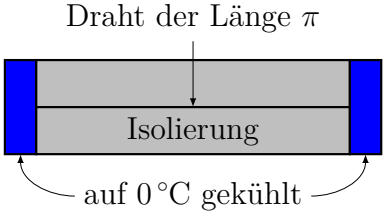
$$\begin{aligned} X(x) &= ae^{\sqrt{\lambda}x} + be^{-\sqrt{\lambda}x} \\ T(t) &= c \cdot e^{\lambda t} \end{aligned}$$

wobei $a, b, c \in \mathbb{C}$. Da wir $X(x) \cdot T(t)$ betrachten, können wir ohne Einschränkung $c = 1$ annehmen. Da die Wärmeleitgleichung linear ist, erhalten wir aus der Summe spezieller Lösungen selbst wieder eine Lösung (**Superpositionsprinzip**).

Beispiel 2.2.1 (Anfangs-Randwertproblem):

$$\begin{aligned} u_t = u_{xx}, &\qquad (x,t) \in (0,\pi) \times (0,\infty) \\ u(x,0) = \varphi(x), &\qquad x \in [0,\pi] \\ u(0,t) = u(\pi,t) = 0, &\qquad t \in [0,\infty) \end{aligned}$$

Physikalischer Hintergrund:



Ein Stab der Länge π mit konstantem Querschnitt hat zur Zeit $t_0 = 0$ die Temperaturverteilung φ. Der Stab ist isoliert, so dass der Wärmestrom nur in x-Richtung statt findet. Die Endpunkte des Stabes werden für die Zeit $t \geq 0$ auf 0 °C gehalten.

Zur analytischen Lösung dieses Problems machen wir folgenden Rechenansatz:

$$u(x,t) = \sum_{n=0}^{\infty} C_n X_n(x) T_n(t)$$

wobei C_n Konstanten und $X_n(x), T_n(t)$ Lösungen der gewöhnlichen Differentialgleichung (2.9) zu gegebenen Konstanten $\lambda = \lambda_n$ sind. Die Konstanten λ_n werden so gewählt, dass die Randwerte $u(0,t) = u(\pi,t) = 0$ erfüllt sind, d. h.

$$X_n(0) = X_n(\pi) = 0$$

Setzen wir $\lambda := -n^2, \quad n = 0, 1, 2, \ldots$ so folgt:

$$X_n(x) = \sin(nx), \quad \left(a = \frac{1}{2i}, \quad b = -\frac{1}{2i}\right)$$
$$T_n(t) = e^{-n^2 t}$$

Also:

$$u(x,t) = \sum_{n=1}^{\infty} C_n \cdot \sin(nx) \cdot e^{-n^2 t}$$

Die Koeffizienten C_n bestimmen wir mit Hilfe der Anfangsbedingung

$$u(x,0) = \varphi(x)$$
$$\varphi(x) = u(x,0) = \sum_{n=1}^{\infty} C_n \cdot \sin(nx)$$

Hieraus folgt direkt, dass die Konstanten C_n die Koeffizienten der Fourier-Sinusreihenentwicklung der Funktion $\varphi(x)$ sind:

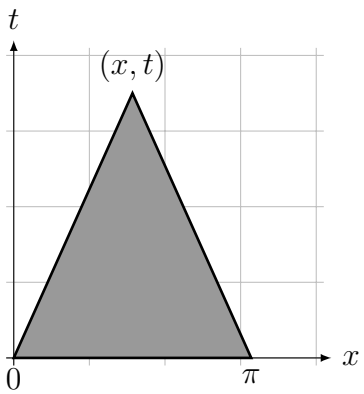
$$C_n = \frac{2}{\pi} \int_0^{\pi} \varphi(x) \cdot \sin(nx)\,\mathrm{d}x, \quad n = 1, 2, \ldots$$

Die Lösung des Anfangs-Randwertproblem (2.2.1) ist also gegeben durch:

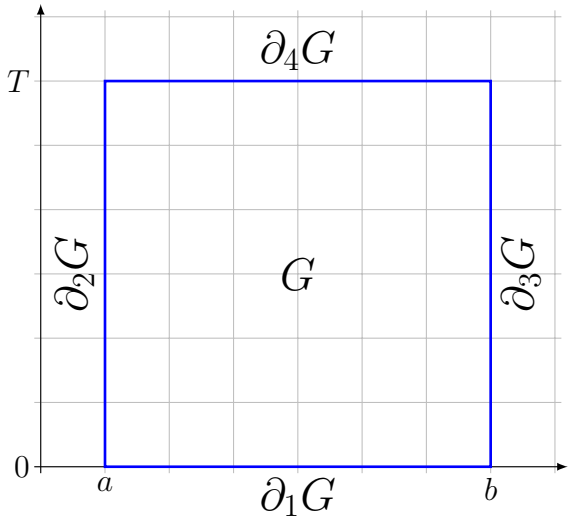
$$\boxed{u(x,t) = \sum_{n=1}^{\infty} \left(\frac{2}{\pi} \int_0^{\pi} \varphi(x) \cdot \sin(nx)\,\mathrm{d}x\right) \cdot \sin(nx) \cdot e^{-n^2 t}}$$

Offenbar hängt die Lösung $u(x,t)$ im Punkt (x,t) von allen Werten $u(x,0)$ mit $x \in [0,\pi]$ ab. Der Abhängigkeitsbereich ist also

$$AB\Big(u,(x,t)\Big) = \Big\{(x,0) : 0 \le x \le \pi\Big\} = [0,\pi]$$



2.2.3. Das Maximumprinzip



Notation:

$$\begin{aligned} G &:= (a,b)\times(0,T) \\ \partial_1 G &:= (a,b)\times\{0\} \\ \partial_2 G &:= \{0\}\times[0,T] \\ \partial_3 G &:= \{b\}\times[0,T] \\ \partial_4 G &:= (a,b)\times\{T\} \end{aligned}$$

Satz 2.2.1 (Maximumprinzip für die Wärmeleitungsgleichung):
Es gelte $u(x,t)\in\mathcal{C}(\overline{G})$, $u\in\mathcal{C}^2(G\cup\partial_4 G)$ *und* $u_t = u_{xx}$ *für* $(x,t)\in G\cup\partial_4 G$. *Dann nimmt* u *das Maximum auf dem* ***parabolischen Rand*** R *an, mit*

$$R := \partial_1 G\cup\partial_2 G\cup\partial_3 G$$

06.12.2012
17. Vorlesung

Beweis:
Maximum existiert, da $u\in\mathcal{C}(\overline{G})$, $\overline{G}$ kompakt.

$$M := \max_{\overline{G}} u(x,t)$$

Angenommen der Satz gilt nicht, dann gibt es ein $\varepsilon > 0$, so dass

$$\max_{R} u(x,t) \le M-\varepsilon$$

und es gibt $(x_0,t_0)\in G\cup\partial_4 G$, so dass $M = u(x_0,t_0)$

$$\begin{aligned} \Rightarrow\quad u_t(x_0,t_0) &\ge 0 \\ u_x(x_0,t_0) &= 0 \\ u_{xx}(x_0,t_0) &\le 0 \end{aligned}$$

Wir definieren folgende Hilfsfunktion:

$$v_k(x,t) := u(x,t) + k\cdot(t_0-t),\quad k = konstant > 0$$

Daraus folgt:

1. $v_k(x_0,t_0) = u(x_0,t_0) = M$
2. $k(t_0-t)\le kT$

Wähle $k>0$, so dass $kT < \frac{\varepsilon}{2}$, dann gilt:

$$\max_{R} v_k \le M - \frac{\varepsilon}{2} \tag{2.10}$$

$v_k \in \mathcal{C}(\overline{G})$, $\overline{G}$ kompakt

$\Rightarrow$ v_k nimmt Maximum in $\overline{G}$ an, etwa in $(x_1, t_1) \in \overline{G}$

$\Rightarrow$ $v_k(x_1, t_1) \geq v_k(x_0, t_0) \overset{1.}{=} M$

$\overset{(2.10)}{\Rightarrow}$ $(x_1, t_1) \in G \cup \partial_4 G$

Weiterhin gilt:

$$\begin{array}{rlll} & 0 & \leq (v_k)_t(x_1,t_1) & = u_t(x_1,t_1) - k \\ - & 0 & \geq (v_k)_{xx}(x_1,t_1) & = u_{xx}(x_1,t_1) \\ \hline & 0 & \leq (v_k)_t(x_1,t_1) - (v_k)_{xx}(x_1,t_1) & = \underbrace{u_t(x_1,t_1) - u_{xx}(x_1,t_1)}_{=0, \quad (x_1,t_1) \in G \cup \partial_4 G} - k \end{array}$$

$0 < k \leq 0$ $\quad \lightning$

□

Bemerkung 2.2.1:
Für Verallgemeinerung des Maximumprinzips, [siehe 16].

2.2.4. Eindeutigkeit und stetige Abhängigkeit von den Rand- und Anfangswerten

Satz 2.2.2:
Seien u, v Lösungen von $u_t = u_{xx}$ in $G \cup \partial_4 G$ und $u|_{\partial_k G} = f_k, \quad v|_{\partial_k G} = g_k, \quad k = 1, 2, 3.$ Dann gilt:

$$\max_{\overline{G}} |u - v| \leq \max_{1 \leq k \leq 3} \max_{\partial_k G} |f_k - g_k|$$

Beweis:
Wir definieren die Hilfsfunktion

$$w := u - v$$

Dann folgt aus dem Maximumprinzip:

1. $\max_{\overline{G}} w \leq \max_R w$
2. $\max_{\overline{G}} (-w) \leq \max_R (-w)$

$\Rightarrow \max_{\overline{G}} |w| \leq \max_R |w| = \max_R |u - v| = \max\limits_{1 \leq k \leq 3} \max\limits_{\partial_k G} |f_k - g_k|$

□

Folgerungen:

1. Die Lösungen des Anfangs-Randwertproblem der Wärmeleitungsgleichung hängen stetig von den Rand- und Anfangswerten ab.
2. Die Lösung des Anfangs-Randwertproblem ist eindeutig.

$\Rightarrow$ Das Anfangs-Randwertproblem der Wärmeleitungsgleichung ist gut gestellt.

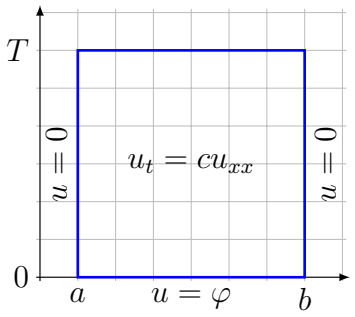
2.2.5. Explizite Differenzenverfahren für die Wärmeleitungsgleichung

Bei der Wärmeleitungsgleichung müssen wir im Gegensatz zur Advektionsgleichung eine Ableitung (in Raumrichtung) zweiter Ordnung approximieren. Wir können dabei vorgehen wie bei der Linienmethode (siehe Ende von Sektion 1.3.6).
Sei $u \in \mathcal{C}^4(G)$, dann gilt:

$$\frac{\partial^2 u}{\partial x^2}(x,t) = D_{+x}D_{-x}u(x,t) + \mathcal{O}\left(h^2\right)$$
$$D_{+x}D_{-x}u(x,t) = \frac{u(x+h,t) - 2u(x,t) + u(x-h,t)}{h^2}$$

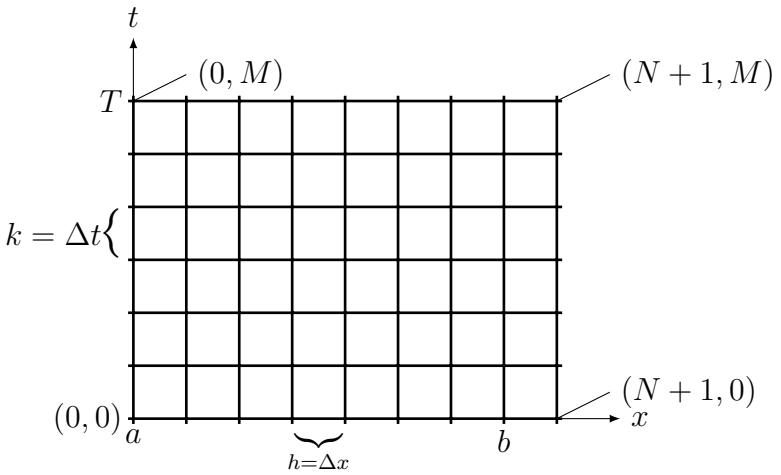
Beispiel 2.2.2 (Modellproblem):

$$\begin{aligned} u_t &= cu_{xx}, && (x,t) \in (a,b) \times (0,T) \\ u(x,0) &= \varphi(x), && x \in [a,b] \\ u(a,t) &= u(b,t) = 0, && t \in [0,T] \end{aligned}$$



Gitter $G^{h,k}$: $h := \Delta x := \frac{b-a}{N+1}$, $k := \Delta t := \frac{T}{M}$

$$G_{h,k} := \left\{ (x_i, t_j) : \begin{cases} x_i := a + ih, & i = 0,1,\ldots,N+1 \\ t_j := jk, & j = 0,1,\ldots,M \end{cases} \right\}$$



Die exakte Lösung des Modellproblems (2.2.2) wird in den Gitterpunkten durch eine Approximation $u^{h,k}$ angenähert.

$$u_{i,j} := u(x_i, t_j) \approx u^{h,k}(x_i, t_j) =: u_{i,j}^{h,k}$$

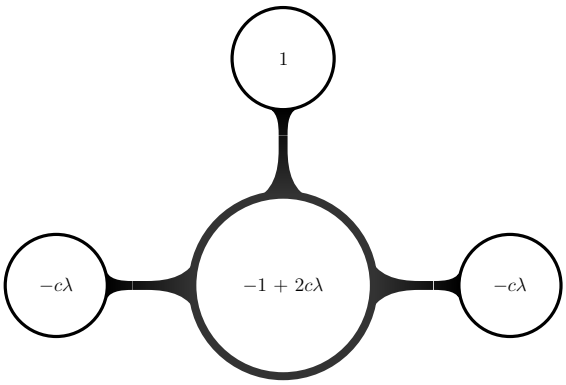
Die Anfangs- und Randwerte sind bekannt:

$$\begin{aligned} u_{i,0}^{h,k} &:= u(x_i, 0) = \varphi(x_i), && i = 0,1,\ldots,N+1 \\ u_{0,j}^{h,k} &:= u(a, t_j) = 0, && j = 0,1,\ldots,M \\ u_{N+1,j}^{h,k} &:= u(b, t_j) = 0, && j = 0,1,\ldots,M \end{aligned}$$

Für die inneren Punkte $(x_i, t_j), \quad 1 \leq i \leq N, \quad 1 \leq j \leq M$ benutzen wir die Differenzengleichung

$$\begin{aligned} & D_{+t} u_{i,j}^{h,k} = c D_{+x} D_{-x} u_{i,j}^{h,k} \\ \Leftrightarrow \quad & \frac{u_{i,j+1}^{h,k} - u_{i,j}^{h,k}}{k} = c \frac{u_{i+1,j}^{h,k} - 2u_{i,j}^{h,k} + u_{i-1,j}^{h,k}}{h^2} \\ \Leftrightarrow \quad & u_{i,j+1}^{h,k} = u_{i,j}^{h,k} + c\frac{k}{h^2}\left(u_{i+1,j}^{h,k} - 2u_{i,j}^{h,k} + u_{i-1,j}^{h,k}\right) \end{aligned}$$

Unter der Annahme $\lambda := \frac{k}{h^2} = konstant$ lautet das Molekül für dieses Verfahren (klassisches explizites Verfahren oder Zeit-Vorwärts-Raum-Vorwärts-Rückwärts):



Begründung: $\lambda = \frac{k}{h^2} = konstant.$
Aus der Lösung des Anfangs-Randwertproblem mit Hilfe der Trennung der Veränderlichen, wissen wir:

$$AB\Big(u, (x,t)\Big) = [a,b]$$

Analog zur Vorgehensweise bei der Advektionsgleichung ergibt sich für das Differenzenverfahren für $(x,t) \in G^{h,k}$

$$AB\Big(u^{h,k}, (x,t)\Big) = [a,b] \cap \left[x - \frac{th}{k}, x + \frac{th}{k}\right]$$

Damit die CFL-Bediengung erfüllt wird, ist es erforderlich, dass

$$\lim_{h,k \to 0} \frac{h}{k} = \infty$$

Dies ist nicht für $\frac{k}{h} = konstant$ erfüllt, aber für $\frac{k}{h^2} = \lambda = konstant.$

2.2.6. Konvergenz des klassischen expliziten Differenzenverfahrens

Satz 2.2.3:
Folgende Voraussetzungen seien erfüllt:

1. $u = u(x,t)$ *sei die Lösung des Modellproblems* (2.2.2) *mit* $a = -1, \quad b = 1$ *sowie* $u \in \mathcal{C}^4((-1,1) \times (0,T))$
2. u^h *sei die Lösung des klassischen expliziten Differenzenverfahrens mit* $\lambda := \frac{\Delta t}{(\Delta x)^2}$ *und* $c\lambda \leq \frac{1}{2}$

Dann gilt:

$$|u^h_{i,j} - u_{i,j}| \leq T\left[\frac{\Delta t}{2}M_2 + \frac{(\Delta x)^2}{12}M_4\right], \quad 0 \leq i \leq N+1, \quad 0 \leq j \leq M$$

wobei $M_p := \max_{r+s=p} \max_{(x,t)\in\overline{G}} \left|\frac{\partial^p u(x,t)}{\partial x^r \partial t^s}\right|, \quad p = 2, 4.$

Beweis:
Wir definieren den globalen Fehler in (x_i, t_j) als $e_{i,j} := u_{i,j} - u^h_{i,j}, \quad 0 \leq i \leq N+1$, $0 \leq j \leq M$, bzw. $1 \leq i \leq N, \quad 1 \leq j \leq M$. Der Fehler erfüllt die Differenzengleichung

$$\frac{e_{i,j+1} - e_{i,j}}{k} - c \cdot \left(\frac{e_{i+1,j} - 2e_{i,j} + e_{i-1,j}}{h^2}\right) = \tau^h_{i,j} \tag{2.11}$$

mit $\tau^h_{i,j} \leq K := \frac{\Delta t}{2}M_2 + \frac{(\Delta x)^2}{12}M_4$ (Restglied abschneiden, Taylorentwicklung). Anfangs- und Randwerte ergeben:

$$\begin{aligned} e_{0,j} = 0 = e_{N+1,j}, &\quad 0 \leq j \leq M \\ e_{i,0} = 0, &\quad 0 \leq i \leq N+1 \end{aligned}$$

Sei $E_j := \max_{0\leq i\leq N+1} |e_{i,j}|$

$$\begin{aligned} \Rightarrow E_0 &= 0 \\ E_j &= \max_{1\leq i\leq N} |e_{i,j}| \end{aligned} \tag{2.12}$$

$$\begin{aligned} \overset{(2.11)}{\Rightarrow} |e_{i,j+1}| &= |e_{i,j} + c\lambda e_{i+1,j} - 2c\lambda e_{i,j} + c\lambda e_{i-1,j} + k\tau^h_{i,j}| \\ &\leq |1 - 2c\lambda| \cdot |e_{i,j}| + |c\lambda| \cdot |e_{i+1,j}| + |c\lambda| \cdot |e_{i-1,j}| + k|\tau^h_{i,j}| \\ &\leq \Big\{|1 - 2c\lambda| + 2|c\lambda|\Big\}E_j + \Delta t \cdot K \end{aligned}$$

Da $0 < c\lambda < \frac{1}{2}$, folgt $|1 - 2c\lambda| = 1 - 2c\lambda$

$$\begin{aligned} \Rightarrow E_{j+1} &\leq E_j + \Delta t \cdot K \\ &\underset{\text{induktiv}}{\overset{(2.12)}{\leq}} (j+1)\underbrace{\Delta t K}_{\leq T}, \quad 0 \leq j \leq M \end{aligned}$$

□

2.2.7. Von-Neumannsche Stabilitätsanalyse

11.12.2012
18. Vorlesung

Spezielle Lösungen des Differenzenverfahrens

[17, Kapitel 4.2]

Betrachte das Modellproblem (2.2.2) mit $a = 0, \quad b = 1$ und das klassische explizite Differenzenverfahren

$$\frac{u^h_{i,j+1} - u^h_{i,j}}{\Delta t} = \frac{u^h_{i+1,j} - 2u^h_{i,j} + u^h_{i-1,j}}{(\Delta x)^2} \quad 0 \le i \le N+1, \quad 0 \le j \le M$$

mit den diskreten Randwerten $u^h_{0,j} = u^h_{N+1,j} = 0, \quad 0 \le j \le M$. Wir suchen nun spezielle Lösungen der Differentialgleichung und verwenden dazu, wie bei der Differentialgleichung, den Ansatz der **Trennung der Veränderlichen**. Gesucht sind diskrete Lösungen der Form:

$$w^h_{i,j} = X^h_i T^h_j, \quad i = 0, \dots, N+1, \quad j = 0, \dots, M$$

wobei $\underbrace{X^h := (X^h_i)}_{\text{unabhängig von } j}$, $\underbrace{T^h := (T^h_j)}_{\text{unabhängig von } i}$. Einsetzen in das Differenzenverfahren ergibt:

$$\frac{X^h_i T^h_{j+1} - X^h_i T^h_j}{\Delta t} = \frac{X^h_{i+1} T^h_j - 2X^h_i T^h_j + X^h_{i-1} T^h_j}{(\Delta x)^2}$$

Ohne Einschränkung gilt: $X^h_i \neq 0, \quad T^h_j \neq 0.$

$$\frac{T^h_{j+1} - T^h_j}{\Delta t T^h_j} = \frac{X^h_{i+1} - 2X^h_i + X^h_{i-1}}{(\Delta x)^2 X^h_i}$$

$\Rightarrow \exists -\mu = konstant$, so dass

$$\frac{X^h_{i+1} - 2X^h_i + X^h_{i-1}}{(\Delta x)^2 X^h_i} = -\mu = \frac{T^h_{j+1} - T^h_j}{\Delta t T^h_j}$$

Daraus folgt:

1. $X^h_{i+1} - 2X^h_i + X^h_{i-1} = -\mu(\Delta x)^2 X^h_i$
2. $T^h_{j+1} - T^h_j = -\mu \Delta t T^h_j$

$w^h_{i,0} = X^h_i T^h_0 = \varphi(x_i)$
$\overset{\text{Anfangswerte hängen nur von } \varphi(x) \text{ ab}}{\Rightarrow}$ Ohne Einschränkung: $T^h_0 := 1$

$$2. \Leftrightarrow T^h_j = (1 - \mu\Delta t) T^h_j$$
$$\overset{T^h_0 = 1}{\underset{\text{induktiv}}{\Leftrightarrow}} T^h_j = (1 - \Delta t \mu)^j, \quad 0 \le j \le M$$

Randbedingungen: $w^h_{0,j} = w^h_{N+1,j} = 0$
$\Rightarrow X^h_0 = 0 = X^h_{N+1}$

$\overset{1.}{\Rightarrow} A_h x^h = -\mu x^h$ (Eigenwertproblem)

$$h = \Delta x, \quad A_h = \frac{1}{h^2}\begin{pmatrix} -2 & 1 & 0 & \cdots & 0 \\ 1 & -2 & 1 & \ddots & \vdots \\ 0 & \ddots & \ddots & \ddots & 0 \\ \vdots & \ddots & 1 & -2 & 1 \\ 0 & \cdots & 0 & 1 & -2 \end{pmatrix}$$

Für die Eigenwerte ergibt sich

$$\begin{aligned} -\mu_p &= -\frac{2}{h^2}\Big(\cos(p\pi h) - 1\Big) \\ &\overset{2\sin^2(\alpha)=1-\cos(2\alpha)}{=} \frac{4}{h^2}\sin^2\left(\frac{p\pi h}{2}\right) \end{aligned}$$

und für die zugehörigen Eigenvektoren:

$$X_p^h = \left(x_{p,1}^h, x_{p,2}^h, \ldots, x_{p,N}^h\right)^\mathsf{T}, \quad p = 1, \ldots, N$$

mit $X_{p,i}^h = \sin(p\pi \underbrace{i \cdot h}_{\substack{a=0 \\ = x_i}})$, $\quad i = 1, \ldots, N$.

Alle speziellen Lösungen der Differentialgleichungen erhalten wir somit

$$w^p := w^{h,p} := \left(w_{i,j}^{h,p}\right)_{i,j}, \quad p = 1, \ldots, N$$

d. h. $w^{h,p}$ hat in (x_i, t_j) den Wert $w_{i,j}^{h,p}$. Das Superpositionsprinzip gilt auch hier:

$$v = \sum_{p=1}^{N} \gamma_p w^p$$

mit $\gamma_p \in \mathbb{C}, \quad p = 1, \ldots, N$ ist Lösung des klassischen expliziten Differenzenverfahrens. Wir bestimmen die Koeffizienten mit Hilfe der Anfangswerte

$$v_{i,0} := v(x_i, 0) = \varphi(x_i), \quad i = 0, \ldots, N+1$$

Da $T_0^h = 1$, also $w_{i,0}^{h,p} = X_{p,i}^h, \quad i = 0, \ldots, N+1$ gilt:

$$\begin{aligned} \varphi(x_i) &\overset{\substack{\text{Definition} \\ \text{von } v}}{=} \sum_{p=1}^{N} \gamma_p w_{i,0}^{h,p} \\ &= \sum_{p=1}^{N} \gamma_p X_{p,i}^h \\ &\overset{\substack{\text{Randwerte} \\ \text{sind } 0}}{=} \sum_{p=0}^{N+1} \gamma_p \sin(p\pi x_i) \end{aligned}$$

Die Eigenvektoren bilden ein Orthogonalsystem

$$\begin{aligned}\gamma_p &= 2h \sum_{p=0}^{N+1} \varphi(x_i) X_{p,i}^h \\ &= 2h \sum_{p=0}^{N+1} \varphi(x_i) \sin(p\pi x_i)\end{aligned}$$

Vergleich der analytischen und der numerischen Lösung

Die **analytische Lösung** lautet:

$$u(x,t) = \sum_{k=1}^{\infty} C_k e^{-k^2\pi^2 t} \sin(k\pi x)$$

mit $C_k = 2\int_0^1 \varphi(x)\sin(k\pi x)\,\mathrm{d}x, \quad k = 1, 2, \ldots$.
Als diskrete Lösung haben wir berechnet:

$$u_{i,j}^h = \sum_{p=1}^{N+1} \gamma_p (1 - \Delta t \mu_p)^j \sin(p\pi x_i)$$

mit $\mu_p = \frac{4}{h^2}\sin^2(\frac{p\pi h}{2})$ und $\gamma_p = 2h\sum_{l=0}^{N+1}\varphi(x_l)\sin(p\pi x_l)$.

Es gilt also:

$$|u_{i,j} - u_{i,j}^h| = \left| \sum_{k=1}^{N+1} \left\{ C_k e^{-k^2\pi^2 t_j} - \gamma_k (1 - \Delta t \mu_k)^j \right\} \sin(k\pi x_i) + \sum_{k=N+2}^{\infty} \underbrace{C_k}_{\substack{\text{beschränkt,} \\ \text{wenn } \varphi(x) \\ \text{stetig}}} e^{-k^2\pi^2 t_j} \underbrace{\sin(k\pi x_i)}_{\leq 1} \right|$$

Betrachte $t_j \geq \underline{t} > 0$

$$\begin{aligned}\left| \sum_{k=N+2}^{\infty} C_k e^{-k^2\pi^2 t_j} \sin(k\pi x_i) \right| &\leq C \sum_{k=N+2}^{\infty} e^{-k^2\pi^2 \underline{t}} \\ &\leq C \sum_{k=N+2}^{\infty} (e^{-\pi^2 \underline{t}})^k \\ &= C (e^{-\pi^2 \underline{t}})^{N+2} \cdot \frac{1}{1 - e^{-\pi^2 \underline{t}}} \\ &\xrightarrow[\approx]{N\to\infty} 0\end{aligned}$$

Wir betrachten in der endlichen Summe die zeitabhängigen Terme:

1. $k^2\pi^2 t_j \geq 0 \Rightarrow e^{-k^2\pi^2 t_j} \leq 1 \quad \forall k$
2. Gilt für irgendeinen Index k
$$|1 - \Delta t \mu_k| > 1$$
so wächst $|1 - \Delta t \mu_k|^j$ unbeschränkt bzw. wird sehr groß für $j = 1, \ldots, M$.

Ein Wachstumsverhalten wie unter 2. beschrieben, tritt bei der analytischen Lösung nicht auf, vgl. 1..
Daher ist es notwendig, dass $|1 - \Delta t \mu_k| \leq 1 \quad \forall k = 1, \ldots, N+1$.

Hieraus folgt:

$$\begin{aligned} & 1 \geq |1 - \Delta t \mu_k| = \left| 1 - \Delta t \frac{4}{(\Delta x)^2} \sin^2\left(k\pi \frac{\Delta x}{2}\right) \right| \\ \Leftrightarrow \quad & 0 \leq \frac{4\Delta t}{(\Delta x)^2} \sin^2\left(k\pi \frac{\Delta x}{2}\right) \leq 2 \\ \Leftrightarrow \quad & \frac{\Delta t}{(\Delta x)^2} \leq \frac{1}{2} \end{aligned}$$

Die Von-Neumannsche Stabilitätsanalyse

Modellproblem:

$$\begin{aligned} u_t &= u_{xx} && (x,t) \in (0,1) \times (0,\infty) \\ u(0,t) &= u(1,t) = 0 && t \in [0,\infty) \\ u(x,0) &= \varphi(x) && x \in [0,1] \end{aligned}$$

Gegeben sei eine **Familie spezieller Lösungen** von $u_t = u_{xx}$

$$\mathcal{F} := \Big(T_k(t)\sin(k\pi x)\Big)_{k\in\mathbb{N}}$$

13.12.2012
19. Vorlesung

wobei $T_k(t) := e^{-(k\pi)^2 t}$

Aus $\sin(x) = \frac{1}{2i}(e^{ix} - e^{-ix})$ ergibt sich $\mathcal{F} = \Big(T_k(t)e^{ik\pi x}\Big)_{k\in\mathbb{Z}}$.
Weiterhin sei eine **Familie diskreter spezieller Lösungen** gegeben:

$$\mathcal{F}_\Delta := \Big((a_k)^j e^{ik\pi x_l}\Big)_{k\in\mathbb{Z}}$$

wobei (a_k) die Zeitabhängigkeit der Lösung beschreibt.

Beispiel (klassisches explizites Differenzenverfahren):

$$a_k = (1 - \Delta t \mu_k)$$

Der Faktor a_k wird **Verstärkungsfaktor** genannt.

Definition 2.2.1 (von-Neumann-stabil):
*Ein Differenzenverfahren heißt **von-Neumann-stabil**, wenn gilt:*

$$\max_k |(a_k)^j| \leq \max_k |T_k(t_j)| \quad \forall t_j \in [0,T] \ (T = \infty \text{ zugelassen})$$

Bemerkung 2.2.2:
Für ein von-Neumann-stabiles Differenzenverfahren gilt: Das Wachstum der numerischen speziellen Lösung ist durch das Wachstum der entsprechenden analytischen speziellen Lösung beschränkt.

Beispiel 2.2.3 (klassisches explizites Differenzenverfahren):

$$u_t = u_{xx}$$

Einsetzten spezieller Lösung der Form $u_k(x,t) = T_k(t)e^{ik\pi x}$ ergibt

$$\begin{aligned} T_k'(t) &= -(k\pi)^2 T_k(t) \\ \overset{T_k(0)=1}{\Leftrightarrow} \quad T_k(t) &= e^{-(k\pi)^2 t} \end{aligned}$$

Das klassische explizite Differenzenverfahren lautet:

$$\frac{u^h_{l,j+1} - u^h_{l,j}}{\Delta t} = \frac{u^h_{l+1,j} - 2u^h_{l,j} + u^h_{l-1,j}}{h^2}$$

Einsetzten der speziellen Lösung der Form $u^h_{l,j} = (a_k)^j e^{ik\pi x_l}$ in das Differenzenverfahren ergibt:

$$\frac{(a_k)^{j+1} - (a_k)^j}{\Delta t} e^{ik\pi x_l} = \frac{e^{ik\pi x_{l+1}} - 2e^{ik\pi x_l} + e^{ik\pi x_{l-1}}}{(\Delta x)^2} \cdot (a_k)^j$$

mit $x_l = l\Delta x \quad (a_k \neq 0)$ folgt hieraus:

$$\begin{aligned} \frac{a_k - 1}{\Delta t} &= \frac{e^{ik\pi\Delta x} - 2 + e^{-ik\pi\Delta x}}{(\Delta x)^2} \\ &= \frac{2}{(\Delta x)^2}(\cos(k\pi\Delta x) - 1) \\ &= -\frac{4}{(\Delta x)^2}\sin^2\left(\frac{k\pi\Delta x}{2}\right) \\ \Leftrightarrow a_k &= 1 - \frac{4(\Delta t)}{(\Delta x)^2}\sin^2\left(\frac{k\pi\Delta x}{2}\right) \end{aligned}$$

Es gilt weiterhin:

$$|T_k(t_j)| \leq 1 \quad \forall k$$

Also muss gelten:

$$\begin{aligned} & a_k|^j \leq 1 \quad \forall j \geq 0 \\ \Leftrightarrow \quad & |a_k| \leq 1 \\ \Leftrightarrow \quad & \Delta t \geq \frac{(\Delta x)^2}{2} \end{aligned}$$

$\Rightarrow$ Das klassische explizite Differenzenverfahren ist von-Neumann-stabil, wenn $\frac{\Delta t}{(\Delta x)^2} \leq \frac{1}{2}$

2.2.8. Implizite Differenzenverfahren für die Wärmeleitungsgleichung

Bisher gesehen:
$\frac{\Delta t}{(\Delta x)^2} \leq \frac{1}{2}$ ist notwendig und hinreichend für die Konvergenz des klassischen expliziten Differenzenverfahren für $u_t = u_{xx}$. Angenommen, wir benötigen $\Delta x = \frac{1}{100}$ um genügend Genauigkeit bezüglich der räumlichen Dimension zu erhalten, dann gilt:

$$\Delta t = \frac{1}{20000} \text{ (sehr kleine Zeitschrittweite)}$$

Ziel: Konstruktion von Verfahren, die nicht einer so starken Restriktion unterworfen sind.

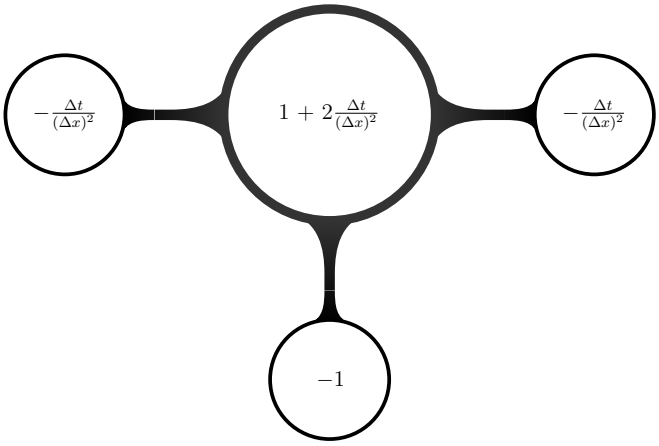
Betrachte das Modellproblem (2.2.2) mit $c = 1,\ a = 0,\ b = 1$.
Betrachte folgende Zeit-Rückwärts-Diskretisierung:

$$D_{-t}u(x,t) = \frac{u(x,t) - u(x,t-\Delta t)}{\Delta t}$$
$$D_{+x}D_{-x}u(x,t) = \frac{u(x+\Delta x,t) - 2u(x,t) + u(x-\Delta x,t)}{(\Delta x)^2}$$

Wir betrachten folgendes Differenzenverfahren:

$$D_{-t}u^h(x_i,t_{j+1}) = D_{+x}D_{-x}u^h(x_i,t_{j+1})$$
$$\Leftrightarrow \frac{u^h_{i,j+1} - u^h_{i,j}}{\Delta t} = \frac{u^h_{i+1,j+1} - 2u^h_{i,j+1} + u^h_{i-1,j+1}}{(\Delta x)^2}, \quad 0 \le i \le N+1, \quad 0 \le j \le M$$

„Molekül“:



Die Randbedingungen ergeben

$$u^h_{0,j} = u^h_{N+1,j} = 0 \quad \forall j = 0,\dots,M$$

und die Anfangswerte

$$u^h_{i,0} = \varphi(x_i) \quad 0 \le i \le N+1$$

Für jeden Zeitschritt j definieren wir den Vektor $u^h_j = \left(u^h_{1,j},\dots,u^h_{N,j}\right)^\mathsf{T}$.
Das Differenzenschema lässt sich dann schreiben als

$$(I + \Delta t A_h)u^h_{j+1} = u^h_j$$

mit I = Einheitsmatrix und $A_h = \frac{1}{(\Delta x)^2}\begin{pmatrix} 2 & -1 & 0 & \cdots & 0 \\ -1 & 2 & -1 & \ddots & \vdots \\ 0 & \ddots & \ddots & \ddots & 0 \\ \vdots & \ddots & -1 & 2 & -1 \\ 0 & \cdots & 0 & -1 & 2 \end{pmatrix}$

Im Unterschied zum expliziten Verfahren muss bei diesem Verfahren in jedem Zeitschritt ein lineares Gleichungssystem der Form

$$(I + \Delta t A_h)v = f$$

gelöst werden $\left(v = u^h_{j+1},\ f = u^h_j\right)$.

Satz 2.2.4:
Die Matrix $(I + \Delta t A_h)$ ist für alle Δt, Δx symmetrisch positiv definit.

$I + \Delta t A_h$ kann also mit Cholesky gelöst werden.

von-Neumannsche Stabilitätsanalyse

Spezielle analytische Lösungen der Wärmeleitungsgleichung

$$u_k(x,t) = T_k(t)e^{ik\pi x}, \quad T_k(t) = e^{-(k\pi)^2 t}$$

Spezielle numerische Lösung

Ansatz: $(a_k)^j e^{ik\pi x_l} = u^h_{l,j}$
Einsetzten in das implizite Schema ergibt:

$$\begin{aligned}
\frac{a_k - 1}{\Delta t} &= \frac{e^{ik\pi\Delta x} - 2 + e^{-ik\pi\Delta x}}{(\Delta x)^2} \cdot a_k \\
&= -\frac{4a_k}{(\Delta x)^2}\sin^2\left(\frac{k\pi\Delta x}{2}\right) \\
\Leftrightarrow a_k &= \frac{1}{1 + \frac{4(\Delta t)}{(\Delta x)^2}\sin^2\left(\frac{k\pi\Delta x}{2}\right)} = \frac{1}{1 + \Delta t \cdot \mu_k} \quad (\mu_k = \text{Eigenwerte von } A_h)
\end{aligned}$$

Es gilt $|T_k(t)| \leq 1$.
Damit das implizite Verfahren von-Neumann-stabil ist, muss gelten:

$$\begin{aligned}
&\max_k \left|(a_k)^j\right| \leq 1 \quad \forall j \geq 0 \\
\Leftrightarrow &\max_k \left|\frac{1}{(1 + \Delta t \mu_k)^j}\right| \leq 1 \quad \forall j
\end{aligned}$$

Da $\Delta t > 0, \quad \mu_k > 0$, folgt $\frac{1}{1+\Delta t\mu_k} \leq 1$
$\Rightarrow$ Die von-Neumannsche Stabilitätsbedingung ist ohne Einschränkung erfüllt.

18.12.2012
20. Vorlesung

Ein (volles) implizites Verfahren ist ohne Einschränkung an Δt und Δx von-Neumann-stabil. Ein solches Verfahren nennt man **unbedingt stabil** (*engl. unconditionally stable*) wohingegen man Verfahren mit Restriktionen an Δt und Δx **bedingt stabil** (*engl. conditionally stable*) nennt.
Von-Neumann-stabil nur notwendig. Es bleibt zu zeigen, dass das implizite Verfahren auch ohne Bedingung an Δt und Δx konvergiert.

Lemma 2.2.1:
Für jede Lösung des impliziten Differenzenschemas gilt:

$$\|u_j\|_\infty \leq \|\varphi\|_\infty$$

wobei $\|u_j\|_\infty = \max_{1\leq l\leq N} |u^h_{l,j}|$ und φ die Anfangswerte vorgibt.
(Betrachte das Modellproblem (2.2.2))

Beweis:
Sei $\lambda := \frac{\Delta t}{(\Delta x)^2}$, dann gilt das implizite Differenzenschema

$$(1+2\lambda)u^h_{l,j+1} = u^h_{l,j} + \lambda(u^h_{l+1,j+1} + u^h_{l-1,j+1})$$

Setzen wir $U_j := \|u_j\|_\infty$, so ergibt sich

$$\begin{aligned}
& (1+2\lambda)u^h_{l,j+1} \leq U_j + 2\lambda U_{j+1} && \forall l = 1,\dots,N \\
\Leftrightarrow\quad & (1+2\lambda)U_{j+1} \leq U_j + 2\lambda U_{j+1} && \\
\Leftrightarrow\quad & U_{j+1} \leq U_j && j = 0,\dots,M \\
& \leq U_0 && \\
& = \|u_0\|_\infty && \\
& = \|\varphi\|_\infty &&
\end{aligned}$$

□

Konvergenz des Verfahrens

Annahme: $u \in \mathcal{C}^4(G \cup \partial_4 G)$

Sei $u = u(x,t)$ die analytische Lösung des Modellproblems. Wie zuvor setzten wir $u_{i,j} = u(x_i, t_j)$ für den Gitterpunkt $(x_i, t_j) \in G^h$. Aus der Taylorentwicklung erhalten wir

$$\frac{u_{i,j+1} - u_{i,j}}{\Delta t} - \frac{u_{i+1,j+1} - 2u_{i,j+1} + u_{i-1,j+1}}{(\Delta x)^2} = \tau_{i,j+1}$$

mit $\tau_{i,j+1} = \tau(x_i, t_{j+1})$ (Restglied) und $|\tau_{i,j+1}| \leq \frac{\Delta t}{2} M_2 + \frac{(\Delta x)^2}{12} M_4 =: K$, mit

$$M_p = \max_{r+s=p} \max_{(x,t)\in\overline{G}} \frac{\partial^p u(x,t)}{\partial x^r \partial t^s}$$

Satz 2.2.5:
Unter der gemachten Voraussetzung gilt:

$$|u_{i,j} - u^h_{i,j}| \leq T\left(\frac{\Delta t}{2} M_2 + \frac{(\Delta x)^2}{12} M_4\right)$$

Beweis:
Sei $e_{i,j} := u_{i,j} - u^h_{i,j}$, dann erfüllt der Fehler $e_{i,j}$ bekanntermaßen auch die Differenzengleichung:

$$\frac{e_{i,j+1} - e_{i,j}}{\Delta t} = \frac{e_{i+1,j+1} - 2e_{i,j+1} + e_{i-1,j+1}}{(\Delta x)^2} + \tau_{i,j+1}$$

und den Anfangsfehler: $e_{i,0} = 0$.
Sei $E_j := \max_{1\leq i\leq N} |e_{i,j}|$, dann gilt:

$$\begin{aligned}
& (1+2\lambda)e_{i,j+1} = e_{i,j} + \lambda(e_{i+1,j+1} + e_{i-1,j+1}) + \Delta t \tau_{i,j+1} && \\
\Rightarrow\quad & (1+2\lambda)|e_{i,j+1}| \leq E_j + 2\lambda(E_{j+1} + \Delta t|\tau_{i,j+1}|) && \forall i \\
\Rightarrow\quad & (1+2\lambda)E_{j+1} \leq E_j + 2\lambda(E_{j+1} + \Delta t K) && \\
\Rightarrow\quad & E_{j+1} \leq E_j + \Delta t K \overset{E_0=0}{\leq} (j+1)\Delta t K \leq TK &&
\end{aligned}$$

□

Wir haben also dieselbe Konvergenzabschätzung wie beim expliziten Verfahren, allerdings ohne die Einschränkung

$$\Delta t \leq (\Delta x)^2 \frac{1}{2}$$

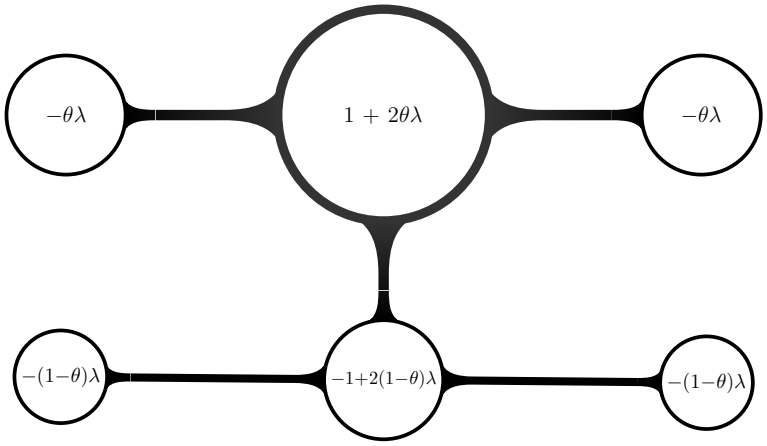
Das Theta-Verfahren

Wir betrachten das Differenzenschema

$$\frac{u_{j+1} - u_j}{\Delta t} + \theta A_h u_{j+1} + (1-\theta) A_h u_j = 0$$

mit $u_j := \left(u^h_{1,j}, \ldots, u^h_{N,j}\right)^\mathsf{T}$ und $\theta \in [0,1]$.
$\theta = 0 \Rightarrow$ klassisches explizites Differenzenverfahren
$\theta = 1 \Rightarrow$ (volles) implizites Differenzenverfahren
$0 < \theta < 1 \Rightarrow$ wie sieht das Molekül aus?

$$\begin{aligned} 0 =& \frac{u^h_{i,j+1} - u^h_{i,j}}{\Delta t} - \theta \frac{u^h_{i+1,j+1} - 2u^h_{i,j+1} + u^h_{i-1,j+1}}{(\Delta x)^2} - (1-\theta)\frac{u^h_{i+1,j} - 2u^h_{i,j} + u^h_{i-1,j}}{(\Delta x)^2} \\ \overset{\lambda := \frac{\Delta t}{(\Delta x)^2}}{\Leftrightarrow}\ 0 =& -(1-\theta)\lambda u^h_{i-1,j} + (-1 + 2(1-\theta)\lambda) u^h_{i,j} - (1-\theta)\lambda u^h_{i+1,j} - \theta\lambda u^h_{i-1,j+1} \\ & + (1 + 2\theta\lambda) u^h_{i,j+1} - \theta\lambda u^h_{i+1,j+1} \end{aligned}$$



Für $\theta = \frac{1}{2}$ ergibt sich

$$|\tau_{i,j}| = \mathcal{O}\left((\Delta t)^2 + (\Delta x)^2\right)$$

Beweisidee: Taylorentwicklung um $\left(x_i, \frac{t_{j+1}+t_j}{2}\right)$
Dieses Verfahren heißt **Crank-Nicolson-Verfahren** (1947). Von-Neumannsche Stabilitätsanalyse ergibt, dass das Crank-Nicolson-Verfahren unbedingt stabil ist.

2.3. Stationäre Diffusionsgleichung

$$u_t - u_{xx} = f$$

Erreichen des zeitunabhängigen Zustands $\Rightarrow u_t = 0$
Stationärer Zustand:

$$-u_{xx} = f$$

2.3.1. Variationsformulierung

Beispiel 2.3.1 (Modellproplem):
Finde eine Funktion $u = u(x) \in \mathcal{C}^2((0,1)) \cap \mathcal{C}([0,1])$, so dass

$$\begin{aligned} -u'' &= f \quad x \in (0,1) \\ u(0) &= 0 \\ u'(1) &= 0 \end{aligned}$$

und $f \in \mathcal{C}((0,1))$ mit $u'' = u_{xx}$

Die Lösung muss offensichtlich zweimal stetig differenzierbar sein. Gibt es einen schwächeren Lösungsbegriff?
Sei $v \in \mathcal{C}^\infty((0,1))$ mit $v(0) = 0$

$$\begin{aligned} \Rightarrow \int_0^1 f(x)v(x)\,\mathrm{d}x \quad &= \quad \int_0^1 -u''(x)v(x)\,\mathrm{d}x \\ &\overset{\text{partielle Integration}}{=} \int_0^1 u'v'\,\mathrm{d}x - u'v\Big|_0^1 \\ &\overset{\substack{u'(1)=0 \\ v(0)=0}}{=} \int_0^1 u'v'\,\mathrm{d}x \end{aligned}$$

Wir definieren den Raum

$$V := \Big\{ v \in \mathcal{C}^\infty((0,1)), \quad v(0) = 0 \Big\}$$

Da $v \in V$ beliebig gewählt war, ergibt sich mit $a(u,v) := \int_0^1 u'v'\,\mathrm{d}x, \quad (f,v) := \int_0^1 fv\,\mathrm{d}x$ das **Variationsproblem**

$$a(u,v) = (f,v) \quad \forall v \in V$$

Ein solches Variationsproblem nennt man auch die **schwache Formulierung** der Differentialgleichung.

1. Ist das Variationsproblem (eindeutig) lösbar?
2. Wenn ja, ist diese Lösung auch Lösung des Modellproblems (2.3.1)?

20.12.2012
21. Vorlesung

Definition 2.3.1 (Hilbertraum):
*Ein **Hilbertraum** ist ein vollständiger, normierter Vektorraum, dessen Norm durch ein Skalarprodukt induziert wird.*

Lemma 2.3.1 (Lax-Milgram):
Sei $(V, \langle\cdot,\cdot\rangle_V)$ ein Hilbertraum, $\|\cdot\|_V := \sqrt{\langle\cdot,\cdot\rangle_V}$ die induzierte Norm, $a(\cdot,\cdot)$ eine stetige, V-elliptische Bilinearform, d. h.

1. *Stetigkeit:*
   Es gebe eine Konstante $c_1 > 0$, so dass
   $$|a(u,v)| \leq c_1 \|u\|_V \|v\|_V \quad \forall u, v \in V$$
2. *V-Elliptizität:*
   Es gebe eine Konstante $\alpha > 0$, so dass
   $$a(u,u) \geq \alpha \|u\|_V^2 \quad \forall u \in V$$

Weiterhin sei F ein stetiges, lineares Funktional $(F\colon V \to \mathbb{R})$ auf V, d. h. es existiere eine Konstante $c_2 > 0$, so dass
$$|F(v)| \leq c_2 \|v\|_V \quad \forall v \in V$$

Dann existiert eine eindeutige Lösung des Variationsproblems $a(u,v) = F(v) \quad \forall v \in V$.

Beweis:
Siehe Numerik partieller Differentialgleichungen.

□

Anwenden des Lemmas von Lax-Milgram auf unser Modellproblem:

$$V := \left\{ v \in \mathcal{C}^\infty((0,1))\colon v(0) = 0 \right\}$$

Inneres Produkt:

$$(u,v)_1 := \int_0^1 u'v' \,\mathrm{d}x + \underbrace{\int_0^1 uv \,\mathrm{d}x}_{(u,v)_0}$$

auf V. Leider ist V bezüglich $(u,v)_1$ nicht vollständig, also kein Hilbertraum. Um das Lemma von Lax-Milgram anwenden zu können, vervollständigen wir V bezüglich $\|\cdot\|_1 := \sqrt{(\cdot,\cdot)_1}$.

Genauer:

$$H^1((0,1)) := \overline{\mathcal{C}^\infty((0,1))}^{\|\cdot\|_1} \text{ (Sobolevraum)}$$

Es gilt:

1. $H^1((0,1)) \subset \mathcal{C}((0,1))$
2. $H^1((0,1))$ ist ein Hilbertraum bezüglich $(\cdot,\cdot)_1$

Sei nun

$$\underbrace{V := \left\{ v \in H^1((0,1)) : v(0) = 0 \right\}}_{\text{Achtung: Geänderte Definition}}$$

Lemma 2.3.2:

1. $a(u,u) \geq \frac{1}{2}\|u\|_1^2 \quad \forall u \in V$
2. $|a(u,v)| \leq \|u\|_1 \|v\|_1 \quad \forall u, v \in V$
3. $|F(v)| \leq \|f\|_0 \|v\|_1 \quad \forall v \in V$
   mit $\|f\|_0 := \sqrt{(f,f)_0} = \sqrt{\int_0^1 f^2 \,\mathrm{d}x}$ *und* $F(v) := (f,v)$

Beweis:

1. $a(u,v) = \int_0^1 u'v' \,\mathrm{d}x$
   zu zeigen:

$$\begin{aligned} a(u,u) &= \int_0^1 (u')^2 \,\mathrm{d}x \\ &\geq \frac{1}{2}\left(\int_0^1 (u')^2 \,\mathrm{d}x + \int_0^1 u^2 \,\mathrm{d}x\right) \end{aligned}$$

   $\Rightarrow$ genügt zu zeigen:

$$\int_0^1 (u')^2 \,\mathrm{d}x \geq \int_0^1 u^2 \,\mathrm{d}x \quad \forall u \in V$$

   zunächst beweisen wir für $u \in V$:

$$\begin{aligned} |u(x)|^2 &= \left|\int_0^x u'(y) \,\mathrm{d}y\right|^2 \\ &\overset{\text{C.S.U.}^1}{\leq} \left(\int_0^x 1^2 \,\mathrm{d}y\right)\left(\int_0^x (u'(y))^2 \,\mathrm{d}y\right) \\ &\overset{H^1((0,1))}{\leq} \underbrace{\left(\int_0^1 1^2 \,\mathrm{d}y\right)}_{=1}\left(\int_0^1 (u'(y))^2 \,\mathrm{d}y\right) \\ &= \int_0^1 (u')^2 \,\mathrm{d}y \end{aligned}$$

$$\begin{aligned} \Rightarrow \int_0^1 \big(u(x)\big)^2 \,\mathrm{d}x &\leq \int_0^1 \left(\int_0^1 (u')^2 \,\mathrm{d}y\right) \mathrm{d}x \\ &\leq \int_0^1 (u')^2 \,\mathrm{d}y \int_0^1 1 \,\mathrm{d}x \\ &= \int_0^1 (u')^2 \,\mathrm{d}x \end{aligned}$$

$\int_0^1 u^2 \,\mathrm{d}x \leq \int_0^1 (u')^2 \,\mathrm{d}x$ ist eine Poincaré-Friedrichs-Ungleichung.

[1] Cauchy-Schwarz-Ungleichung

2. Für beliebige $u, v \in V$ gilt:

$$\begin{aligned} |a(u,v)| &= \left| \int_0^1 u'v' \,\mathrm{d}x \right| \\ &\overset{\text{C.S.U.}^1}{\leq} \left(\int_0^1 (u')^2 \,\mathrm{d}x \right)^{\frac{1}{2}} \left(\int_0^1 (v')^2 \,\mathrm{d}x \right)^{\frac{1}{2}} \\ &\leq \left(\int_0^1 (u')^2 \,\mathrm{d}x + \int_0^1 u^2 \,\mathrm{d}x \right)^{\frac{1}{2}} \left(\int_0^1 (v')^2 \,\mathrm{d}x + \int_0^1 v^2 \,\mathrm{d}x \right)^{\frac{1}{2}} \\ &= \|u\|_1 \|v\|_1 \end{aligned}$$

3. Für beliebiges $v \in V$:

$$\begin{aligned} |F(v)| &= \left| \int_0^1 fv \,\mathrm{d}x \right| \\ &\overset{\text{C.S.U.}^1}{\leq} \underbrace{\left(\int_0^1 f^2 \,\mathrm{d}x \right)^{\frac{1}{2}}}_{=\|f\|_0} \left(\int_0^1 v^2 \,\mathrm{d}x \right)^{\frac{1}{2}} \\ &\leq \|f\|_0 \left(\int_0^1 (v')^2 \,\mathrm{d}x + \int_0^1 v^2 \,\mathrm{d}x \right)^{\frac{1}{2}} \\ &= \|f\|_0 \|v\|_1 \end{aligned}$$

□

Die Variationsformulierung des Modellproblems erfüllt mit $V = \{v \in H^1((0,1)) \colon v(0) = 0\}$ und $(u,v)_V = (u,v)_1$ die Voraussetzungen des Lemmas von Lax-Milgram. Daher existiert eine eindeutig bestimmte Lösung $u \in V$ der schwachen Formulierung, die sogenannte **schwache Lösung**.

Satz 2.3.1:
Sei $f \in \mathcal{C}([0,1])$ und $u \in \mathcal{C}^2((0,1)) \cap \mathcal{C}([0,1])$. Weiterhin sei u eine Lösung des Variationsproblems. Daher löst u auch das ursprüngliche Modellproblem (2.3.1).

Beweis:
Sei $v \in V \cap \mathcal{C}^1((0,1)) \cap \mathcal{C}([0,1])$, dann folgt mit partieller Integration

$$(f,v) = a(u,v) = \int_0^1 -u''v \,\mathrm{d}x + u'(1)v(1)$$

$\Rightarrow \underbrace{(f+u'',v) = 0}_{(*)} \quad \forall v \in V \cap \mathcal{C}^1((0,1)) \cap \mathcal{C}([0,1])$ mit $v(1) = 0$. Sei $w := f + u'' \in \mathcal{C}([0,1])$.
Ist $x \neq 0$, dann existiert ein Intervall $[x_0, x_1] \subset [0,1], \quad x_0 < x_1$, so dass

$$\operatorname{sign}\big(w(x)\big) > 0 \quad \forall x \in (x_0, x_1) \text{ (Stetigkeit)}$$

Definiere $v(x) := \begin{cases} (x-x_0)^2(x-x_1)^2, & x \in [x_0, x_1] \\ 0, & x \in [0,1] \setminus [x_0, x_1] \end{cases}$

[1] Cauchy-Schwarz-Ungleichung

$\Rightarrow (w, v) = 0 \overset{(*)}{\Rightarrow} f + u'' = w = 0$, also $-u'' = f$. Bleibt zu zeigen: $u'(1) = 0$

$$
\begin{aligned}
(f, v) &= \int_0^1 -u''v \,\mathrm{d}x + u'(1)v(1) \\
\Leftrightarrow u'(1)v(1) &= 0 \quad \forall v \in \overset{\Rightarrow v(0)=0}{V} \cap \mathcal{C}^1((0,1)) \cap \mathcal{C}([0,1]) \\
&\overset{v(x)=x}{\Rightarrow} u'(1) = 0
\end{aligned}
$$

Da $u \in V$ schon $u(0) = 0$ impliziert, löst u das Modellproblem.

□

2.3.2. Das Galerkin-Verfahren (Ritz-Galerkin-Verfahren)

08.01.2013
22. Vorlesung

Im Unterschied zum Verfahren der finiten Differenzen, bei denen der Differenzialoperator diskretisiert wird, diskretisieren wir hier den Lösungsraum V.
Es sei $V^h \subset V$ ein endlich dimensionaler Hilbertraum der Dimension $n := \dim(V^h)$ mit der Basis $(\varphi_i)_{i=1,\dots,n}$ um eine approximative Lösung des Variationsproblems

$$a(u, v) = F(v) \quad \forall v \in V$$

zu erhalten, gehen wir wie folgt vor:

1. Ersetze $a(u, v) = F(v) \quad \forall v \in V$ durch
   Finde $u_h \in V^h$, so dass $\boxed{a(u_h, v_h) = F(v_h) \quad \forall v_h \in V^h}$
   $a(\cdot, \cdot) : \underbrace{V^h}_{\subset V} \times \underbrace{V^h}_{\subset V} \to \mathbb{R}, \quad F : V^h \to \mathbb{R}$
   Diskretes Problem wohlgestellt, insbesondere existiert genau eine Lösung $u_h \in V^h$.
2. Entwickle die diskrete Lösung u_h nach der Basis $(\varphi_i)_{i=1,\dots,n}$

$$u_h = \sum_{i=1}^{n} c_i \varphi_i$$

3. Setze diese Entwicklung von u_h in die diskrete Variationsformulierung ein:

$$
\begin{aligned}
a(u_h, v_h) &= a\left(\sum_{i=1}^{n} c_i \varphi_i, v_h\right) \\
&= F(v_h) \quad v_h \in V^h
\end{aligned}
$$

4. Offensichtlich genügt es, die Basisfunktion φ_j als Testfunktion einzusetzen

$$
\begin{aligned}
a\left(\sum_{i=1}^{n} c_i \varphi_i, v_h\right) &= F(\varphi_j) \quad j = 1, \dots, n \\
\sum_{i=1}^{n} c_i a(\varphi_i, \varphi_1) &= F(\varphi_1) \\
\Leftrightarrow \quad \vdots \qquad & \qquad \vdots \qquad (2.13) \\
\sum_{i=1}^{n} c_i a(\varphi_i, \varphi_n) &= F(\varphi_n)
\end{aligned}
$$

5. Definiere die **Steifigkeitsmatrix**

$$\kappa := \Big(a(\varphi_i, \varphi_j)\Big)_{i,j=1,\dots,n}$$

und den **Lastvektor**

$$b := \Big(F(\varphi_j)\Big)_{j=1,\dots,n}$$

6. Das diskrete Variationsproblem ist äquivalent zu (2.13) $\Leftrightarrow \kappa c = b$
wobei $c = (c_1, \dots, c_n)^\mathsf{T}$, der Koeffizientenvektor der Basisentwicklung ist.

Satz 2.3.2:
Sei $a(\cdot,\cdot)$ selbstadjungiert und V-Elliptisch. Dann ist die zugehörige Steifigkeitsmatrix symmetrisch positiv definit.

Beweis:
Seien $u_h, v_h \in V^h$ mit $u_h = \sum_{i=1}^n c_i\varphi_i, \quad v_h = \sum_{j=1}^n d_j\varphi_j$

$$\begin{aligned} a(u_h, v_h) &= a\left(\sum_{i=1}^n c_i\varphi_i, v_h = \sum_{j=1}^n d_j\varphi_j\right) \\ &= \sum_{i,j=1}^n c_i d_j \underbrace{a(\varphi_i, \varphi_j)}_{k_{i,j}} \text{ mit } (k_{i,j})_{i,j} \\ &= d^\mathsf{T}\kappa c \\ &= \langle \kappa c, d\rangle \end{aligned}$$

Hierbei sei $y^\mathsf{T}x = \langle x, y\rangle \quad \forall x, y \in \mathbb{R}^n$.
Dann gilt:

1. $\langle \kappa c, d\rangle = a(u_h, v_h) \overset{\text{Selbst-Adjungiert}}{=} a(v_h, u_h) = \langle \kappa d, c\rangle$
κ symmetrisch folgt direkt aus $\kappa = \Big(a(\varphi_i, \varphi_j)\Big)$

2. $\langle \kappa c, c\rangle = a(u_h, u_h) \overset{\underset{V\text{-Elliptisch}}{a(\cdot,\cdot)}}{\geq} \alpha\|u_h\|_V^2 > 0 \quad \forall u_h \in V^h, \quad u_h \neq 0$
$\Leftrightarrow \langle \kappa c, c\rangle > 0 \quad \forall c \in \mathbb{R}^n, \quad c \neq 0$

□

Satz 2.3.3:
$a(\cdot,\cdot)$ bildet ein Skalarprodukt auf V.

Beweis:
Bilinearformeigenschaften offensichtlich erfüllt. Es gibt $a(u,u) \geq 0 \quad \forall u \in V$ und
$a(u,u) > 0 \quad \forall u \in V, \quad u \neq 0$

□

$\Rightarrow \|u\|_a := \sqrt{a(u,u)}$ ist also eine Norm auf V

Wie gut kann die Approximation, die durch das Galerkin-Verfahren berechnet wird, bestenfalls sein?

Satz 2.3.4 (Bestapproximation des Galerkin-Verfahrens):
Seien $u \in V$ bzw. $u_h \in V^h$, so dass $a(u,v) = F(v) \quad \forall v \in V$ bzw. $a(u_h, v_h) = F(v_h) \quad \forall v_h \in V^h$ und sei $\|u\|_a := \sqrt{a(u,u)}$. Dann gilt:

$$\|u - u_h\|_a = \inf_{v_h \in V^h} \|u - v_h\|_a$$

Das Galerkin-Verfahren liefert also im V^h die Bestapproximation u_h an u in V.

Beweis:
Sei $V^h \subset V$

$$\begin{array}{lll} & a(u, v_h) & = f(v_h) \quad \forall v_h \in V^h \\ - & a(u_h, v_h) & = f(v_h) \quad \forall v_h \in V^h \\ \hline & a(u - u_h, v_h) & = 0 \quad \forall v_h \in V^h \end{array}$$

Das bedeutet: Der Fehler $u - u_h \perp_a V^h$.
Hieraus folgt für beliebige $v_h \in V^h$:

$$\begin{aligned} \|u - u_h\|_a^2 &= a(u - u_h, u - u_h) \\ &= a(u - u_h, u - u_h + v_h) \\ &\leq \|u - u_h\|_a \|u - u_h + v_h\|_a \end{aligned}$$

Für $u \neq u_h$ erhalten wir:

$$\|u - u_h\|_a \leq \|u - \underbrace{u_h + v_h}_{= w_h \in V^h}\|_a \quad \forall v_h \in V^h$$

Also haben wir gezeigt:

$$\|u - u_h\|_a = \inf_{w_h \in V^h} \|u - w_h\|_a$$

□

Ziel: Vorgegeben: Genauigkeit $\varepsilon > 0$
Konstruiere ein $w_h \in V^h$, so dass

$$\|u - w_h\|_a \leq \varepsilon \cdot \|f\|_0$$

$\overset{\text{Satz 2.3.4}}{\Rightarrow} \|u - u_h\|_a \leq \varepsilon \|f\|_0$

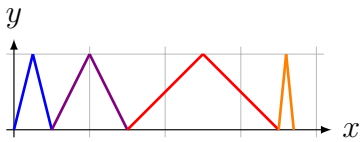
2.3.3. Finite Elemente

In diesem Abschnitt definieren wir den Raum V^h und geben eine Basis $(\varphi_i)_{i=1,\ldots,n}$ an.

Definition 2.3.2 (Konstruktion von V^h):
Sei $0 = x_0 < x_1 < \cdots < x_{n-1} < x_n = 1$ eine Unterteilung von $[0,1]$. Dann definieren wir:

$$V^h := \left\{ v \in \mathcal{C}([0,1]) \colon v|_{[x_{i-1}, x_i]} \text{ linear}, \quad i = 1, \ldots, n, \quad v(0) = 0 \right\}$$

Es gilt: $V^h \subset V \subset H^1((0,1))$.
Hutfunktion-Beispiel:



Basis für V^h

Für jedes $i = 1, \ldots, n$ definieren wir ein $\varphi_i \in V^h$ durch

$$\varphi_i(x_j) = \delta_{i,j} \quad \forall j = 1, \ldots, n$$

mit $\delta_{i,j} = \begin{cases} 1, & \text{falls } i = j \\ 0, & \text{sonst} \end{cases}$ (Kronecker-Delta).

10.01.2013
23. Vorlesung

Die so konstruierte Menge ist eine Basis von V^h.

1. Lineare Unabhängigkeit:
   $\sum_{i=1}^n c_i \underbrace{\varphi_i(x_j)}_{\delta_{i,j}} = 0 \Rightarrow c_j = 0$

2. $V^h = \langle \varphi_1, \ldots, \varphi_n \rangle$
   Dazu definieren wir die Interpolation

   $$I^h : \mathcal{C}([0,1]) \to V^h$$
   $$I^h v(x) := \sum_{i=1}^n v(x_i)\varphi_i(x)$$

   Es genügt dann zu zeigen, dass für $v_h \in V^h$ gilt:

   $$v_h = I^h v_h$$

   Es gilt: $\underbrace{(v_h - I^h v_h)}_{=:w_h}|_{[x_{i-1},x_i]}$ linear und $w_h(x_i) = w_h(x_{i-1}) = 0$.
   $\Rightarrow v_h - I^h v_h$ verschwindet auf $[x_{i-1}, x_i]$. Da $[x_{i-1}, x_i]$ beliebig gewählt war, gilt:

   $$v_h = I^h v_h \text{ auf } [0,1]$$

□

Arbeitsdefinition: Die Basisfunktion $\varphi_i, \quad i = 1, \ldots, n$ nennt man auch (stückweise lineare) Finite-Elemente-Funktionen.

V^h stückweise lineare Funktion.
Für solche Funktionen ist die klassische Ableitung nicht erklärt. Das führt uns auf den Begriff der **schwachen Ableitung**.

Definition 2.3.3 (schwache Ableitung):
*Sei $u \in \mathcal{L}_2((a,b))$, dann heißt v **schwache Ableitung** von u, falls gilt:*

1. $v \in \mathcal{L}_2((a,b))$
2. $\int_a^b \varphi v \,\mathrm{d}x = -\int_a^b \varphi' u \,\mathrm{d}x \quad \forall \varphi \in \mathcal{C}_0^\infty((a,b))$, *wobei* $\varphi' = D\varphi =$ *klassische Ableitung.*

   Für den Fall, dass u klassisch differenzierbar ist, gilt mit partieller Integration

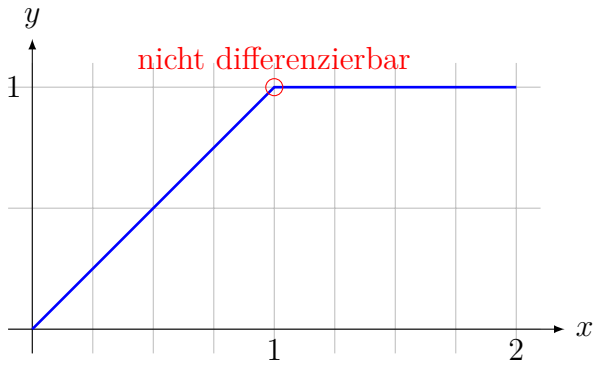
$$\int_a^b \varphi u' \,\mathrm{d}x = -\int_a^b \varphi' u \,\mathrm{d}x$$

   $\varphi(a) = \varphi(b) = 0$, *da* $\varphi \in \mathcal{C}_0^\infty((a,b))$.
   Falls notwendig, bezeichnen wir die schwache Ableitung v mit $D_w u$ ($w \mathrel{\widehat{=}}$ weak derivative).

Beispiel 2.3.2 (schwache Ableitung):

Sei $(a,b) = (0,2)$ und

$$u(x) := \begin{cases} x, & 0 < x \leq 1 \\ 1, & 1 < x < 2 \end{cases}$$



Existiert eine schwache Ableitung von u?
Wenn ja, wie sieht sie aus?
Sei $\varphi \in \mathcal{C}_0^\infty((0,2))$: Finde $v \in \mathcal{L}_2((0,2))$, so dass

$$\int_0^2 \varphi' u \,\mathrm{d}x = -\int_0^2 \varphi v \,\mathrm{d}x$$

$$\begin{aligned}
\int_0^2 \varphi' u \,\mathrm{d}x &= \int_0^1 \varphi' x \,\mathrm{d}x + \int_1^2 \varphi' \cdot 1 \,\mathrm{d}x \\
&\overset{\text{partielle Integration}}{=} -\int_0^1 \varphi \cdot 1 \,\mathrm{d}x + x \cdot \varphi(x)|_0^1 + \varphi(x)|_1^2 \\
&= -\int_0^1 \varphi(x) \,\mathrm{d}x + \cancel{\varphi(1)} + \underbrace{\varphi(2)}_{=0} - \cancel{\varphi(1)} \\
&= -\int_0^1 \varphi(x) \,\mathrm{d}x \\
&= -\int_0^2 \varphi v \,\mathrm{d}x
\end{aligned}$$

mit $v(x) := \begin{cases} 1, & 0 < x \leq 1 \\ 0, & 1 < x < 2 \end{cases} \in \mathcal{L}_2((0,2))$

Es gibt also nicht differenzierbare Funktionen, die schwache Ableitungen besitzen. Es gilt folgender Zusammenhang mit dem Sobolevraum

$$H^1((a,b)) = \overline{\mathcal{C}_0^\infty((a,b))}^{\|\cdot\|_{H^1((a,b))}}$$

$\|u\|^2_{H^1((a,b))} = \|u\|^2_1 := |u|^2_1 + \|u\|^2_0$
$|u|^2_1 := \int_a^b (D_w u)^2 \,\mathrm{d}x, \quad \|u\|^2_0 := \int_a^b u^2 \,\mathrm{d}x$
Es gilt:

$$H^1((a,b)) = \Big\{ u \in \mathcal{L}_2((a,b)), \quad D_w u \text{ existiert} \Big\}$$

(Meyers und Serrin, 1964)

Für Funktionen, die im klassischen Sinne differenzierbar sind, gilt:

$$u' = D_w u$$

Damit folgt auch für $u \in V$ und $u_h \in V^h$:

$$\begin{aligned} \|u - u_h\|^2_a &= \int_0^1 \Big(D_w(u-u_h)\Big)^2 \,\mathrm{d}x \\ &= \sum_{i=1}^n \int_{x_{i-1}}^{x_i} \Big(D_w(u-u_h)\Big)^2 \,\mathrm{d}x \\ &\overset{u\in\mathcal{C}^2((0,1))}{=} \sum_{i=1}^n \int_{x_{i-1}}^{x_i} \Big((u-u_h)'\Big)^2 \,\mathrm{d}x \\ &= \ldots \text{(Beweis zu Satz 2.3.5)} \end{aligned}$$

2.3.4. Fehlerabschätzung

Satz 2.3.5:
Sei $u \in \mathcal{C}^2((0,1))$ *und* $h := \max\limits_{i=1,\ldots,n} \underbrace{(x_i - x_{i-1})}_{=:h_i}$. *Dann gilt:*

$$\|u - I^h u\|_a \le c \cdot h \|u''\|_{\mathcal{L}^2(0,1)}$$

Dabei ist $0 < c =$ *konstant unabhängig von* h *und* $\|u''\|^2_{\mathcal{L}^2(0,1)} = \int_0^1 (u'')^2 \,\mathrm{d}x \quad \forall u \in V$.

Beweis:
Es gilt: $\|u - I^h u\|^2_a = \sum_{i=1}^n \int_{x_{i-1}}^{x_i} \Big((u - I^h u)'\Big)^2 \,\mathrm{d}x$

$$\begin{aligned} \|u''\|^2_{\mathcal{L}^2((0,1))} &= \int_0^1 (u'')^2 \,\mathrm{d}x \\ &= \sum_{i=1}^n \int_{x_{i-1}}^{x_i} (u'')^2 \,\mathrm{d}x \end{aligned}$$

Es genügt zu zeigen:

$$\int_{x_{i-1}}^{x_i} \Big((u - I^h u)'\Big)^2 \,\mathrm{d}x \le c \cdot \underbrace{(x_i - x_{i-1})^2}_{=h_i^2} \int_{x_{i-1}}^{x_i} (u'')^2 \,\mathrm{d}x \tag{2.14}$$

Da $I^h u$ auf $[x_{i-1}, x_i]$ linear ist, gilt:

$$(u - I^h u)'' = u'' \text{ auf } [x_{i-1}, x_i]$$

Also gilt: $\int_{x_{i-1}}^{x_i} \Big((u - I^h u)''\Big)^2 \,\mathrm{d}x = \int_{x_{i-1}}^{x_i} (u'')^2 \,\mathrm{d}x$.

Mit $w := u - I^h u$ genügt es zu zeigen:

$$\int_{x_{i-1}}^{x_i} (w')^2 \,\mathrm{d}x \leq c \cdot (x_i - x_{i-1})^2 \int_{x_{i-1}}^{x_i} (w'')^2 \,\mathrm{d}x \tag{2.15}$$

Substitution: $y := \dfrac{x - x_{i-1}}{x_i - x_{i-1}}$

$$\frac{dy}{dx} = \frac{1}{x_i - x_{i-1}}$$

Dann ergibt sich aus (2.15):

$$\int_0^1 \left(\tilde{w}'(y)\right)^2 \mathrm{d}y \leq c \cdot \int_0^1 \left(\tilde{w}''(y)\right)^2 \mathrm{d}y \tag{2.16}$$

mit $\tilde{w}(y) := w(x_{i-1} + (x_i - x_{i-1})y)$.
Wir haben somit die gesuchte Abschätzung (2.14) auf eine Ungleichung reduziert, die unabhängig von der Schrittweite h_i ist (Skalierungsargument!).
Aus dem Satz von Rolle folgt die Existenz eines $\xi \in (0,1)$, so dass $\tilde{w}'(\xi) = 0$

$$\begin{aligned}
\Rightarrow \qquad \tilde{w}'(y) &= \int_\xi^y \tilde{w}''(x) \,\mathrm{d}x \\
|\tilde{w}'(y)| &= \left| \int_\xi^y \tilde{w}''(x) \,\mathrm{d}x \right| \overset{\text{C.S.U.}^1}{\leq} \underbrace{\left| \int_\xi^y 1^2 \,\mathrm{d}x \right|^{\frac{1}{2}}}_{\leq 1} \left| \int_\xi^y \left(\tilde{w}''(x)\right)^2 \mathrm{d}x \right|^{\frac{1}{2}} \\
\Rightarrow \qquad |\tilde{w}'(y)| &\leq \left| \int_0^1 \left(\tilde{w}''(x)\right)^2 \mathrm{d}x \right|^{\frac{1}{2}} \\
\Rightarrow \qquad \int_0^1 \left(\tilde{w}'(y)\right)^2 \mathrm{d}y &\leq \int_0^1 \left(\int_0^1 \left(\tilde{w}''(x)\right)^2 \mathrm{d}x \right) \mathrm{d}y \\
&= \int_0^1 \left(\tilde{w}''(x)\right)^2 \mathrm{d}x \underbrace{\int_0^1 1 \,\mathrm{d}y}_{=1} \\
&= \int_0^1 \left(\tilde{w}''(y)\right)^2 \mathrm{d}y \\
\Rightarrow \quad \text{(2.16) mit } c = 1
\end{aligned}$$

□

Satz 2.3.6:
Sei $u \in \mathcal{C}^2((0,1))$, $f \in \mathcal{C}((0,1))$ und $u_h \in V^h$ die mit dem Galerkin-Verfahren berechnete Lösung. Dann gilt für den Fehler

$$\|u - u_h\|_1 \leq c \cdot h \|f\|_{\mathcal{L}^2((0,1))}$$

wobei $c > 0$ unabhängig von $h > 0$.

[1] Cauchy-Schwarz-Ungleichung

Beweis:
Da $-u'' = f$ haben wir $\|u''\|_{\mathcal{L}^2((0,1))} = \|f\|_{\mathcal{L}^2((0,1))}$
Aus den Sätzen (2.3.4) und (2.3.5) und der V-Elliptizität folgt:

$$\begin{aligned}
\|u - u_h\|_1 &\leq \frac{1}{\sqrt{\alpha}}\|u - u_h\|_a = \frac{1}{\sqrt{\alpha}} \inf_{v_h \in V^h} \|u - w_h\|_a \\
&\leq \frac{1}{\sqrt{\alpha}}\|u - I^h u\|_a \\
&\leq \frac{d}{\sqrt{\alpha}} h \|u''\|_{\mathcal{L}^2((0,1))} \\
&= \underbrace{\frac{d}{\sqrt{\alpha}}}_{=:c} h \|f\|_{\mathcal{L}^2((0,1))} \\
&= c \cdot h \|f\|_{\mathcal{L}^2((0,1))}
\end{aligned}$$

□

3. Eigenwerte

3.1. Grundlagen

Definition 3.1.1 (Eigenwert):
Sei $A \in \mathbb{K}^{n\times n}$, dann heißt $\lambda \in \mathbb{C}$ ***Eigenwert*** *von $A :\Leftrightarrow \exists x \in \mathbb{C}^n,\ x \neq 0$, so dass*

$$Ax = \lambda x$$

Satz 3.1.1:
λ ist Eigenwert von $A \Leftrightarrow \det(\lambda I - A) = 0$. Das Polynom $\varphi(\lambda) := \det(\lambda I - A)$ heißt charakteristisches Polynom.

Definition 3.1.2 (algebraische und geometrische Vielfachheit):

1. ***algebraische Vielfachheit:***
   $\sigma(\lambda)$ ist die Vielfachheit der Nullstellen λ des charakteristischen Polynoms $\varphi(\lambda)$
2. ***geometrische Vielfachheit:***
   $\rho(\lambda)$ ist die Anzahl der linear unabhängigen Eigenvektoren zum Eigenwert λ

Darstellung des charakteristischen Polynoms:
$\lambda_1, \ldots, \lambda_m$ seien paarweise verschiedene Eigenwerte von A, $\sigma_k = \sigma(\lambda_k), \quad k = 1, \ldots, m$
$\Rightarrow \varphi(\lambda) = \prod_{k=1}^{m} (\lambda - \lambda_k)^{\sigma_k}, \quad \sum_{k=1}^{m} \sigma_k = n$. Es gilt:

$$\sum_{k=1}^{m} \rho_k \leq n \text{ mit } \rho_k = \rho_k(\lambda)$$

Definition 3.1.3 (ähnliche Matrizen):
Zwei Matrizen heißen ***ähnlich*** *$:\Leftrightarrow \exists X \in \mathbb{K}^{n\times n}$, invertierbar so dass $A = XBX^{-1}$.*

Satz 3.1.2:
Seien $A, B \in \mathbb{K}^{n\times n}$ ähnlich. Dann haben sie dieselben Eigenwerte und ihre algebraischen und geometrischen Vielfachheiten stimmen überein.

Beweis:
$A = XBX^{-1}$

$$\begin{aligned}
\Rightarrow \det(\lambda I - A) &= \det(\underbrace{\lambda I - XBX^{-1}}_{=X(\lambda I - B)X^{-1}}) \\
&= \det(X)\det(\lambda I - B)\det(X^{-1}) \\
&= \det(\lambda I - B)
\end{aligned}$$

$\Rightarrow \varphi_A(\lambda) = \varphi_B(\lambda)$ und Eigenwerte und deren algebraische Vielfachheit stimmen überein.
Sei λ Eigenwert von A mit geometrischer Vielfachheit $\rho \Rightarrow$ Es gibt ρ linear unabhängige Eigenvektoren $x_1, \ldots, x_\rho$, d. h.

$$Ax_k = \lambda x_k, \quad k = 1, \ldots, \rho$$

Für $k = 1, \ldots, \rho$ gilt mit $y_k = X^{-1}x_k$

$$\begin{aligned} By_k &= (X^{-1}AX)X^{-1}x_k \\ &= X^{-1}Ax_k \\ &= X^{-1}\lambda x_k \\ &= \lambda y_k \end{aligned}$$

Daraus folgt: $y_1, \ldots, y_\rho$ sind ρ Eigenvektoren von B zum Eigenwert λ.
Lineare Unabhängigkeit von $x_k \Rightarrow y_k$ linear unabhängig.
$\Rightarrow$ geometrische Vielfachheit von λ als Eigenwert von A ist nicht größer als die von B (Symmetrieargument, vertauschen der Rollen von A und B ergibt die Behauptung).

□

15.01.2013
24. Vorlesung

Satz 3.1.3:
Die algebraische Vielfachheit eines Eigenwertes ist mindestens so groß wie seine geometrische Vielfachheit.

Satz 3.1.4 (Jordansche Normalform):
Sei $A \in \mathbb{C}^{n\times n}$. $\exists$ reguläre Matrix $T \in \mathbb{C}^{n\times n}$, so dass

$$J = T^{-1}AT = \begin{pmatrix} c(\lambda_1) & 0 & \cdots & 0 \\ 0 & \ddots & \ddots & \vdots \\ \vdots & \ddots & \ddots & 0 \\ 0 & \cdots & 0 & c(\lambda_n) \end{pmatrix} \text{ mit } c(\lambda_i) = \begin{pmatrix} \lambda_i & 1 & 0 & \cdots & 0 \\ 0 & \lambda_i & \ddots & \ddots & \vdots \\ \vdots & \ddots & \ddots & \ddots & 0 \\ \vdots & & \ddots & \lambda_i & 1 \\ 0 & \cdots & \cdots & 0 & \lambda_i \end{pmatrix}$$

Bemerkung 3.1.1:
Die Jordansche Normalform ist für numerische Berechnungen nicht geeignet, da sie numerisch instabil ist.

Satz 3.1.5:
Sei $A \in \mathbb{K}^{n\times n}$ mit Eigenwerten λ_i, $i = 1, \ldots, n$. Dann gilt für die Determinante $\det(A)$ und die Spur $\operatorname{tr}(A)$ folgender Zusammenhang mit den Eigenwerten:

$$\det(A) = \prod_{i=1}^{n} \lambda_i, \quad \operatorname{tr}(A) = \sum_{i=1}^{n} \lambda_i$$

Beweis:
Sei $\varphi_A(x) := \prod_{i=1}^{n}(x - \lambda_i)$ das charakteristische Polynom von A.

$$\begin{aligned} \det(A) &= (-1)^n \det(-A) \\ &= (-1)^n \varphi_A(0) \\ &= (-1)^n \prod_{i=1}^{n}(-\lambda_i) \\ &= \prod_{i=1}^{n}(\lambda_i) \end{aligned}$$

Beweis für die Spur analog.

□

Definition 3.1.4 (Schursche Normalform):
*Sei $A \in \mathbb{K}^{n\times n}$, dann heißt $A = QTQ^*$ **Schursche Normalform**, wenn Q unitäre Matrix ist und T obere Dreiecksgestalt hat.*

Bemerkung 3.1.2:
Da $Q^* = Q^{-1}$ sind A und T ähnliche Matrizen und haben daher dieselben Eigenwerte. Diese kann man direkt von der Diagonalen von T ablesen.

Satz 3.1.6:
Jede quadratische Matrix $A \in \mathbb{K}^{n\times n}$ besitzt eine Schursche Normalform.

Beweis (per vollständiger Induktion):
$n = 1$: ✓
$n - 1 \mapsto n$:
Sei $x \in \mathbb{K}^n$ ein beliebiger Eigenvektor zum Eigenwert λ von A, zusätzlich sei $x^*x = 1$. Dieser Vektor bilde die erste Spalte einer unitären Matrix $U = \begin{pmatrix} x & \tilde{U} \end{pmatrix}$ mit $\tilde{U} \in \mathbb{K}^{n\times(n-1)}$. Dann gilt:

$$\begin{aligned} U^*AU &= \begin{pmatrix} x^* \\ \tilde{U}^* \end{pmatrix} A \begin{pmatrix} x & \tilde{U} \end{pmatrix} \\ &= \begin{pmatrix} x^*Ax & \overbrace{x^*A\tilde{U}}^{=B} \\ \tilde{U}^* \underbrace{Ax}_{=\lambda x} & \underbrace{\tilde{U}^*A\tilde{U}}_{=C} \end{pmatrix} \\ &= \begin{pmatrix} \lambda & B \\ 0 & \underbrace{C}_{\in\mathbb{K}^{(n-1)\times(n-1)}} \end{pmatrix} \end{aligned}$$

Hierbei haben wir ausgenutzt, dass $Ax = \lambda x$ und x orthogonal zu den Spalten von $\tilde{U}$, da U unitär.
Nach Induktionsvoraussetzung existiert für C eine Schursche Normalform

$$\underbrace{V\tilde{T}V^*}_{=C}$$

Sei nun $Q := U \begin{pmatrix} 1 & 0 \\ 0 & v \end{pmatrix}$
Q unitär:

$$\begin{aligned} QQ^* &= U \begin{pmatrix} 1 & 0 \\ 0 & v \end{pmatrix} \begin{pmatrix} 1 & 0 \\ 0 & v^* \end{pmatrix} U^* \\ &\overset{VV^*=I_{n-1}}{=} U \begin{pmatrix} 1 & 0 \\ 0 & I_{n-1} \end{pmatrix} U^* \\ &= UU^* \\ &= I_n \end{aligned}$$

Weiterhin gilt:

$$Q^*AQ = \begin{pmatrix} \lambda & Bv \\ 0 & \tilde{T} \end{pmatrix} = \text{ Dreiecksmatrix}$$

$\Rightarrow A = QTQ^*$

□

Definition 3.1.5 (unitär diagonalisierbar):
*Sei $A \in \mathbb{K}^{n\times n}$, dann heißt A **unitär diagonalisierbar**, falls es eine unitäre Matrix Q und eine Diagonalmatrix D gibt, so dass $A = QDQ^*$ gilt.*

Die Klasse der unitär diagonalisierbaren Matrizen ist die der normalen Matrizen.

Satz 3.1.7:
Sei $A \in \mathbb{K}^{n\times n}$. Eine unitäre Matrix Q und eine Diagonalmatrix D mit $A = QDQ^*$ existieren genau dann, wenn A normal ist, d. h. $AA^* = A^*A$.

Beweis (per vollständiger Induktion):

„⇒“: Eine Diagonalmatrix ist trivialerweise normal.
Voraussetzung: $A = QDQ^*$

$$\begin{aligned}\Rightarrow AA^* &= (QD\underbrace{Q^*)(Q}_{=I}D^*Q^*)\\ &= QDD^*Q^*\\ &= QD^*DQ^*\\ &= (QD^*Q^*)(QDQ^*)\\ &= A^*A\end{aligned}$$

„⇐“: Voraussetzung: A ist normal.
Sei $A = QTQ^*$ die Schursche Normalform von A. Da A als normal vorausgesetzt wurde, folgt mit den selben Argumenten wie in der Hinrichtung, dass auch T normal ist.

Mit vollständiger Induktion über n zeigen wir, dass dann T Diagonalgestalt hat.
$n = 1$: ✓
$n - 1 \mapsto n$:
$T = \begin{pmatrix} c & v^* \\ 0 & S \end{pmatrix}$, $c \in \mathbb{K}$, $v \in \mathbb{K}^{n-1}$, $S \in \mathbb{K}^{(n-1)\times(n-1)}$ obere Dreiecksmatrix.
Da T normal ist, folgt:

$$\begin{pmatrix} |c|^2 + v^*v & (Sv)^* \\ Sv & SS^* \end{pmatrix} = TT^* = T^*T = \begin{pmatrix} |c|^2 & (cv)^* \\ cv & vv^* + S^*S \end{pmatrix}$$

Aus $|c|^2 + v^*v = |c|^2$ folgt: $\|v\|_{\mathcal{L}_2}^2 = v^*v = 0$
$\Rightarrow v = 0$
$\Rightarrow T = \begin{pmatrix} c & 0 \\ 0 & S \end{pmatrix}$ S ist obere $(n-1)\times(n-1)$-Dreiecksmatrix und $SS^* = S^*S$
$\overset{\text{I.V.}}{\Rightarrow}$ T ist Diagonalmatrix.

□

Satz 3.1.8:
Jede hermitesche Matrix ist unitär diagonalisierbar und ihre Eigenwerte sind alle reell.

Beweis:
A hermitesch $\Leftrightarrow A = A^* \Rightarrow A$ normal
$\overset{\text{Satz 3.1.7}}{\Rightarrow}$ A unitär diagonalisierbar.
Sei D die zugehörige Diagonalmatrix, dann finden sich alle Eigenwerte von A auf der Diagonalen von D und es gilt $D = D^*$. Dies geht nur für reelle Diagonaleinträge.

□

17.01.2013
25. Vorlesung

Korollar 3.1.8.a:
Sei $A \in \mathbb{K}^{n\times n}$ eine hermitesche Matrix mit Eigenwerten λ_i, die in absteigender Weise angeordnet sind:

$$\lambda_1 \geq \lambda_2 \geq \ldots \geq \lambda_n$$

Dann gilt:

$$\lambda_1 = \max_{0\neq x\in\mathbb{C}^n} \frac{x^*Ax}{x^*x}$$
$$\lambda_n = \min_{0\neq x\in\mathbb{C}^n} \frac{x^*Ax}{x^*x}$$

Beweis:
A hermitesch $\Rightarrow$ A unitär diagonalisierbar. $\exists Q$ unitär, so dass $D = Q^*AQ$, wobei D = Diagonalmatrix. Sei $0 \neq x \in \mathbb{C}^n$ beliebig.

$$\begin{aligned}
\frac{x^*Ax}{x^*x} &= \frac{x^*Q\overbrace{(Q^*AQ)}^{=D}Q^*x}{x^*QQ^*x} \\
&= \frac{(Q^*x)^*D(Q^*x)}{\underbrace{(Q^*x)^*}_{y}(Q^*x)} \\
&= \frac{y^*Dy}{y^*y} \\
&= \frac{\sum_{i=1}^n \lambda_i y_i^2}{\sum_{i=1}^n y_i^2} \\
&\leq \lambda_1 \\
\Rightarrow \frac{x^*Ax}{x^*x} &\leq \lambda_1 \quad \forall x \in \mathbb{C}^n, \quad x \neq 0
\end{aligned}$$

$\Rightarrow$ Behauptung (Minimaleigenschaft analog)

□

Es seien $D \in \mathbb{K}^{n\times n}$ Diagonalmatrix, $T \in \mathbb{K}^{n\times n}$ obere Dreiecksmatrix, $X \in \mathbb{K}^{n\times n}$ eine invertierbare Matrix und Q eine unitär Matrix.
Für A quadratisch gilt:

1. A ist diagonalisierbar $\Leftrightarrow$ geometrische und algebraische Vielfachheit stimmen überein
2. A ist unitär diagonalisierbar, d. h. $\exists Q$ unitär so dass $A = QDQ^* \Leftrightarrow A$ ist normal, d. h. $AA^* = A^*A$
3. A besitzt immer eine Schursche Normalform, d. h. $\exists Q$ unitär: $A = QTQ^*$

3.2. Numerische Verfahren zur Eigenwertberechnung

3.2.1. Potenzmethode (Vektoriteration, *engl. power method*)

Ausgehend von einem Vektor $x^{(0)}$ bildet man eine Reihe von Iterierten nach der Iterationsvorschrift:

$$x^{(k+1)} := Ax^{(k)}, \quad k = 0, 1, 2, \ldots$$

Offensichtlich: $x^{(k+1)} = Ax^{(k)} = A^2x^{(k-1)} = \ldots = A^kx^{(0)} \rightsquigarrow$ Potenzmethode

Algorithmus 3.2.1 Potenzmethode

Initialisieren: $k = 0$, Startvektor $x^{(0)}$ mit $\|x^{(0)}\| = 1$
Iterieren: $k \geq 1$

$$
\begin{aligned}
z^{(k)} &:= Ax^{(k-1)} \\
x^{(k)} &:= \frac{z^{(k)}}{\|z^{(k)}\|} \\
\lambda^{(k)} &:= \underbrace{(x^{(k)})^* Ax^{(k)}}_{\text{Rayleigh-Quotienten}}
\end{aligned}
$$

Voraussetzungen an die Matrix:

1. A ist diagonalisierbar
2. Sei X eine quadratische Matrix $X = (x_i)_{i=1,\dots,n}$ deren Spalten x_i die Eigenvektoren von A sind
3. Die Eigenwerte λ_i von A seien wie folgt angeordnet:

$$|\lambda_1| > |\lambda_2| \geq \ldots \geq |\lambda_n|$$

und der betragsmäßig größte Eigenwert habe die algebraische Vielfachheit $\sigma(\lambda_1) = 1$

Wir zeigen, dass $x^{(k)} \to x_1$ (Eigenvektor zum Eigenwert λ_1) konvergiert für $k \to \infty$.

$$
\begin{aligned}
x^{(k)} &= \frac{Ax^{(k-1)}}{\|Ax^{(k-1)}\|} \\
&\overset{\text{induktiv}}{=} \frac{A^k x^{(0)}}{\|A^k x^{(0)}\|}, \quad k \geq 1
\end{aligned}
$$

A diagonalisierbar $\Rightarrow (x_i)_{i=1,\dots,n}$ Eigenvektoren bilden Basis. Entwickle Startvektor $x^{(0)}$ in dieser Basis:

$$x^{(0)} = \sum_{i=1}^{n} \alpha_i x_i$$

$$
\begin{aligned}
\overset{\substack{Ax_i=\lambda_i x_i \\ \Rightarrow \\ i=1,\dots,n}}{} A^k x^{(0)} &= \sum_{i=1}^{n} \alpha_i A^k x_i \\
&= \sum_{i=1}^{n} \alpha_i \lambda_i^k x_i \\
&= \alpha_1 \lambda_1^k \left(x_1 + \underbrace{\sum_{i=2}^{n} \frac{\alpha_i}{\alpha_1} \left(\frac{\lambda_i}{\lambda_1} \right)^k x_i}_{=:y^{(k)}} \right), \quad k = 1, 2, \dots
\end{aligned}
$$

$$x^{(k)} = \frac{\alpha_1 \lambda_1^k (x_1 + y^{(k)})}{\|\alpha_1 \lambda_1^k (x_1 + y^{(k)})\|} = \mu_k \frac{x_1 + y^{(k)}}{\|x_1 + y^{(k)}\|}$$

mit $\mu_k := \operatorname{sign}(\alpha_1 \lambda_1^k)$, nach Voraussetzung $\left|\frac{\lambda_i}{\lambda_1}\right| < 1 \quad \forall i = 2, \dots, n \Rightarrow y^{(k)} \xrightarrow{k \to \infty} 0$.
Also richtet sich die Iterierte $x^{(k)}$ mit wachsendem k in Richtung des Eigenvektors x_1 aus.

Satz 3.2.1:
Sei A diagonalisierbare $n\times n$-Matrix und die Eigenwerte erfüllen die Ordnungsbedingung

$$|\lambda_1| > |\lambda_2| \geq \ldots \geq |\lambda_n|$$

Weiterhin sei $\alpha_1 \neq 0$, dann existiert eine Konstante $c > 0$, so dass

$$\left\|q^{(k)} - x_1\right\|_2 \leq c\left|\frac{\lambda_2}{\lambda_1}\right|^k, \quad k \geq 1$$

wobei $q^{(k)} := \frac{x^{(k)}\|A^k x^{(0)}\|}{\alpha_1 \lambda_1^k} = x_1 + \sum_{i=2}^n \frac{\alpha_i}{\alpha_1}\left(\frac{\lambda_i}{\lambda_1}\right)^k x_i$

Beweis:
A diagonalisierbar, d. h. $\exists X$ invertierbar $A = XDX^{-1}$
Ohne Einschränkung: Spalten x_i von X seien normiert.

$$\begin{aligned}
\left\|q^{(k)} - x_1\right\|_2 &= \left\|x_1 + \sum_{i=2}^n \frac{\alpha_i}{\alpha_1}\left(\frac{\lambda_i}{\lambda_1}\right)^k x_i - x_1\right\|_2 \\
&= \left\|\sum_{i=2}^n \frac{\alpha_i}{\alpha_1}\left(\frac{\lambda_i}{\lambda_1}\right)^k x_i\right\|_2 \\
&\leq \sum_{i=2}^n \left|\frac{\alpha_i}{\alpha_1}\right|\left|\frac{\lambda_i}{\lambda_1}\right|^k \underbrace{\|x_i\|_2}_{=1} \\
&\leq \left|\frac{\lambda_2}{\lambda_1}\right|^k \underbrace{\sum_{i=2}^n \frac{|\alpha_i|}{|\alpha_1|}}_{=:c}
\end{aligned}$$

□

Aus Satz 3.2.1 folgt mit $|\lambda_2| < |\lambda_1|$:

$$\lim_{k\to\infty} q^{(k)} = x_1$$

Es gilt weiterhin:

$$\left\|q^{(k)}\right\|_2^2 = \frac{\|A^k x^{(0)}\|_2^2}{|\alpha_1\lambda_1|^2}\|x^{(k)}\|_2^2$$

Für die Folge der **Rayleigh-Quotienten** gilt:

$$\frac{\left(q^{(k)}\right)^* A\left(q^{(k)}\right)}{\left(q^{(k)}\right)^* q^{(k)}} = \frac{\left(x^{(k)}\right)^* Ax^{(k)}}{\underbrace{\left(x^{(k)}\right)^* x^{(k)}}_{=1}} = \lambda^{(k)}$$

und somit auch die Konvergenz gegen λ_1. Die Konvergenzgeschwindigkeit hängt von der Größe des Quotienten $\left|\frac{\lambda_2}{\lambda_1}\right|$ ab und das Verfahren konvergiert also um so langsamer, je näher $\left|\frac{\lambda_2}{\lambda_1}\right|$ an Eins liegt.

3.2.2. Inverse Iteration

Der Nachteil der Potenzmethode ist, dass immer nur der betragsmäßig größte Eigenwert und der zugehörige Eigenvektor berechnet wird. In diesem Abschnitt betrachten wir ein Verfahren, welches zu einer gegebenen Zahl $\mu \in \mathbb{C}$ den Eigenvektor berechnet, der zu dem Eigenwert gehört, der am dichtesten an μ liegt.

Annahme: μ sei selbst kein Eigenwert von A.

Zur Konstruktion eines solchen Verfahrens wenden wir die Potenzmethode auf

$$M_\mu^{-1} := (A - \mu I)^{-1}$$

an. Die Zahl μ heißt **Shift-Parameter**. Offensichtlich sind die Eigenwerte von M_μ^{-1} gegeben durch $\xi_i := (\lambda_i - \mu)^{-1}, \quad i = 1, \ldots, n$. Angenommen, es gibt ein $m \in \mathbb{N}$, so dass

$$|\lambda_m - \mu| < |\lambda_i - \mu| \quad \forall i = 1, \ldots, n, \quad i \neq m \tag{3.1}$$

Dann liegt λ_m am dichtesten an μ
(3.1) $\Rightarrow \boxed{|\xi_m| > |\xi_i|} \quad i = 1, \ldots, n, \quad i \neq m$

Algorithmus 3.2.2 Inverse Iteration

Initialisieren: $k = 0$, Startvektor $x^{(0)}$ mit $\|x^{(0)}\| = 1$
Iterieren: $k \geq 1$

$$(A - \mu I) z^{(k)} := x^{(k-1)}$$
$$x^{(k)} := \frac{z^{(k)}}{\|z^{(k)}\|}$$
$$\sigma^{(k)} := \underbrace{(x^{(k)})^* A x^{(k)}}_{\text{Rayleigh-Quotienten}}$$

Bemerkung 3.2.1:
Die Eigenvektoren von M_μ oder M_μ^{-1} sind dieselben wie die von A:
Sei $x \in \mathbb{K}^n$ Eigenvektor von A, dann

$$M_\mu x = (A - \mu I)x = Ax - \mu x = (\lambda - \mu)x$$

Sei umgekehrt $x \in \mathbb{K}^n$ Eigenvektor von M_μ:

$$\begin{aligned} Ax &= (M_\mu + \mu I)x \\ &= M_\mu x + \mu x \\ &= (\lambda - \mu)x + \mu x \\ &= \lambda x \end{aligned}$$

Daher kann in der Inversen Iteration der Rayleigh-Quotient mit A statt mit M_μ^{-1} gebildet werden. Zu der inversen Iteration muss in jedem Schritt ein lineares Gleichungssystem gelöst werden. Geschieht dies mit der Gaußelimination, so berechnet man einmal die LR-Zerlegung und führt dann jeweils die Vorwärts-/Rückwärtssubstitution durch.

22.01.2013
26. Vorlesung

Wiederholung:
$x^{(k+1)} = Ax^{(k)} = A^k x^{(0)}$
$x^{(0)} = \sum_{i=1}^{n} \alpha_i x_i$, x_i Eigenvektor von A normiert
Die Vektoriteration oder Potenzmethode hat die Konvergenzgeschwindigkeit $\mathcal{O}\left(\left|\frac{\lambda_2}{\lambda_1}\right|\right)$.
Zusammenhang zwischen Eigenwerten von A und A^{-1}:
Der betragsmäßig kleinste Eigenwert wäre berechenbar durch Anwendung der Potenzmethode auf A^{-1}.

Shift-Strategie:
$\mu \in \mathbb{K}$ und betrachte $(A - \mu I) = M_\mu$. Wende inverse Iteration auf M_μ^{-1} an. Eigenwerte ξ_i von M_μ^{-1} sind gegeben durch $(\lambda_i - \mu)^{-1}$.
$m \in \mathbb{N} \quad |\lambda_m - \mu| < |\lambda_i - \mu| \quad \forall i = 1, \ldots, n, \quad i \neq m \Rightarrow |\xi_m| > |\xi_i|$.
Genau wie bei der Potenzmethode zeigt man, dass

$$\lim_{k\to\infty} q^{(k)} = x_m \text{ und } \lim_{k\to\infty} \sigma^{(k)} = \lambda_m$$

mit $\sigma^{(k)} = \underbrace{(x^{(k)})^* A x^{(k)}}_{\text{Rayleigh-Quotient}}$.
Die Konvergenz ist umso schneller, je dichter μ an λ_m liegt. In jedem Iterationsschritt muss ein lineares Gleichungssystem der Form $(A - \mu I)z = x$ gelöst werden.

A ist diagonalisierbar, d. h. es existiert eine Basis $x_i, \ldots, x_n$ aus Eigenvektoren zu den Eigenwerten $\lambda_1, \ldots, \lambda_n$. Es gilt $x_i^* x_i = 1$ und $x^{(0)} = \sum_{i=1}^{n} \alpha_i x_i$.
Dann gilt für die k-te Iterierte $z^{(k)}$:

$$\begin{aligned} z^{(k)} &= \frac{1}{\|(A-\mu I)^{-(k-1)} x^{(0)}\|} \cdot (A-\mu I)^{-(k-1)} x^{(0)} \\ &= \frac{1}{\|(A-\mu I)^{-(k-1)} x^{(0)}\|} \cdot \sum_{i=1}^{n} \frac{\alpha_i}{(\lambda_i - \mu)^{(k-1)}} \cdot x_i \end{aligned}$$

mit $\lambda_m \approx \mu$.
$\Rightarrow$ Der Anteil des Eigenvektors x_m überwiegt in dieser Summe.
x_m liegt im Bild von $A - \mu I$ und damit ist das System auch lösbar. Als Abbruchkriterium für die Inverse Iteration kann man

$$\|r^{(k)}\|_\infty \leq c \cdot eps \|A\|_\infty$$

benutzen, wobei $r^{(k)} := (A - \mu I)x^{(k)}$ (Residuum).

3.2.3. QR-Verfahren

Idee: Es handelt sich um ein Verfahren, iterativ für eine Matrix die Schursche Normalform zu berechnen.

In diesem Abschnitt beschränken wir uns auf reelle Matrizen $A \in \mathbb{R}^{n\times n}$. Will man auch nur reell rechnen, so kann man A allerdings nicht mehr durch Orthogonaltransformationen auf obere Dreiecksmatrix bringen, sondern nur noch auf obere Block-Dreiecksgestalt.
Zur Erinnerung: Satz 3.1.6 (Schursche Normalform).

Satz 3.2.2:
Sei $A \in \mathbb{R}^{n\times n}$, dann existiert eine orthogonale Matrix $Q \in \mathbb{R}^{n\times n}$, so dass $Q^T AQ$ folgende, obere Dreiecksgestalt hat:

$$Q^T AQ = \begin{pmatrix} R_{1,1} & \cdots & \cdots & R_{1,m} \\ 0 & \ddots & & \vdots \\ \vdots & \ddots & \ddots & \vdots \\ 0 & \cdots & 0 & R_{m,m} \end{pmatrix}$$

wobei jedes $R_{i,i}$ entweder ein Skalar oder eine 2×2-Matrix ist.
Die 1×1-Blöcke enthalten die reellen Eigenwerte und die 2×2-Blöcke die Paare komplex-konjugierter Eigenwerte.

Beweis:
Wir werden im Beweis zwischen reellen und rein komplexen Eigenwerten unterscheiden und in jedem Fall zeigen, dass man die Argumente des Beweises für die Schursche Normalform (Satz 3.1.6) analog verwenden kann.

1. $\lambda \in \mathbb{R}$ ist Eigenwert, dann kann man direkt wie in Satz 3.1.6 vorgehen
2. $\lambda \in \mathbb{C}\setminus\mathbb{R}$ Eigenwert und $z \in \mathbb{C}^m$, $z \neq 0$ zugehöriger Eigenvektor.
   $\Rightarrow \overline{z}$ ist Eigenvektor zum Eigenwert $\overline{\lambda}$, z und $\overline{z}$ sind linear unabhängig
   $z = x + \imath y$

$$\begin{aligned} \alpha z + \beta\overline{z} = 0 &\Leftrightarrow \alpha(x+\imath y) + \beta(x-\imath y) = 0 \\ &\Leftrightarrow (\alpha+\beta)x + (\alpha-\beta)\imath y = 0 \\ &\Leftrightarrow \alpha+\beta = 0 \quad \wedge \quad \alpha-\beta = 0 \\ &\Leftrightarrow \alpha = \beta = 0 \end{aligned}$$

Also sind Real- und Imaginärteil von z linear unabhängig. Es sei $\begin{matrix}\lambda = \alpha + \imath\beta \\ z = x + \imath y\end{matrix}$

Einerseits gilt: $Az = \lambda z = (\alpha+\imath\beta)(x+\imath y) = (\alpha x - \beta y) + (\alpha\imath y + \beta\imath x)$
Andererseits gilt: $Az = Ax + \imath Ay$

Vergleich ergibt:

$$\begin{aligned} Ax &= (\alpha x - \beta y) \\ Ay &= (\alpha y + \beta x) \end{aligned}$$
$$\Rightarrow \quad A\begin{pmatrix} x & y \end{pmatrix} = \begin{pmatrix} x & y \end{pmatrix}\begin{pmatrix} \alpha & \beta \\ -\beta & \alpha \end{pmatrix}$$

x und y linear unabhängig $\Rightarrow \begin{pmatrix} x & y \end{pmatrix}$ hat Rang 2
$\underset{\text{Satz 5.6.1}}{\Rightarrow}$ es existiert eine reduzierte QR-Zerlegung, d. h. es gibt eine Matrix $Z \in \mathbb{R}^{n\times 2}$ mit orthonormalen Spaltenvektoren und eine obere Dreiecksmatrix $U \in \mathbb{R}^{2\times 2}$, so dass $\begin{pmatrix} x & y \end{pmatrix} = \underbrace{Z}_{\widehat{=}Q} U$

$$\begin{aligned}\Rightarrow \quad AZ &= A\begin{pmatrix} x & y \end{pmatrix} U^{-1} \\ &= \begin{pmatrix} x & y \end{pmatrix} \begin{pmatrix} \alpha & \beta \\ -\beta & \alpha \end{pmatrix} U^{-1} \\ &= Z \underbrace{U \begin{pmatrix} \alpha & \beta \\ -\beta & \alpha \end{pmatrix} U^{-1}}_{=:\Lambda} \\ &= Z\Lambda\end{aligned}$$

Statt des komplexen Falls $Az = \lambda z$ haben wir nun den Block-Fall

$$AZ = Z\Lambda$$

Da die Spalten von Z orthogonal sind, lässt sich der Beweis von Satz 3.1.6 analog wiederholen. Dazu definieren wir eine Matrix $\tilde{U} \in \mathbb{R}^{n\times(n-2)}$ mit orthonormalen Spaltenvektoren, so dass

$$U = \begin{pmatrix} Z & \tilde{U} \end{pmatrix}$$

eine orthogonale Matrix ist. Dann gilt:

$$\begin{aligned}U^{-1}AU = U^{\mathsf{T}}AU &= \begin{pmatrix} Z^{\mathsf{T}} \\ \tilde{U}^{\mathsf{T}} \end{pmatrix} A \begin{pmatrix} Z & \tilde{U} \end{pmatrix} \\ &= \begin{pmatrix} \Lambda & Z^{\mathsf{T}}A\tilde{U} \\ 0 & \underbrace{\tilde{U}^{\mathsf{T}}A\tilde{U}}_{=:C} \end{pmatrix}\end{aligned}$$

Die Matrix $\Lambda \in \mathbb{R}^{2\times 2}$ enthält die Eigenwerte λ und $\overline{\lambda}$. Die kleinere Matrix $C \in \mathbb{R}^{(n-2)\times(n-2)}$ kann nun induktiv behandelt werden, wie zuvor.

□ 24.01.2013 27. Vorlesung

Ziel: Transformiere die Matrix A sukzessive (also iterativ) mit Hilfe von Orthogonalmatrizen $Q^{(k)}$ auf die reelle Schursche Normalform (vgl. Satz 3.2.2).

Gegeben: $A^{(0)} := A$
$A^{(k)} := (Q^{(k)})^{\mathsf{T}} A^{(k-1)} Q^{(k)}$
Bedeutet: Die Iterierten $A^{(k)}$ sollen ausgehend von $A^{(0)} = A$ diese Eigenschaft haben.

Berechne die QR-Zerlegung von $A^{(k-1)}$ (vgl. Kapitel 5):

$$A^{(k-1)} = Q^{(k)} R^{(k)}$$

Definiere: $A^{(k)} = R^{(k)} Q^{(k)}$

$$\begin{aligned}\Rightarrow \quad A^{(k)} &= R^{(k)} Q^{(k)} \\ &= (Q^{(k)})^{\mathsf{T}} \underbrace{Q^{(k)} R^{(k)}}_{=A^{(k-1)}} Q^{(k)} \\ &= (Q^{(k)})^{\mathsf{T}} A^{(k-1)} Q^{(k)}\end{aligned}$$

Diese Vorgehensweise erzeugt eine Folge von Matrizen $A^{(k)}$, die alle dieselben Eigenwerte haben. Aus der QR-Zerlegung folgt, dass der Aufwand in jedem Iterationsschritt $\mathcal{O}(n^3)$ ist. Dieser Aufwand lässt sich reduzieren, wenn wir zuvor A auf Hessenberggestalt bringen.

Definition 3.2.1 (Hessenbergform):
Eine Matrix $\mathbb{R}^{n\times n}$ hat ***Hessenbergform*** *oder -gestalt, wenn alle Elemente unterhalb der ersten unteren Nebendiagonalen 0 sind. Eine Hessenbergmatrix ist also fast eine obere Dreiecksmatrix bis auf die erste Nebendiagonale.*

$$\begin{pmatrix} * & \cdots & \cdots & \cdots & * \\ * & \ddots & & & \vdots \\ 0 & \ddots & \ddots & & \vdots \\ \vdots & \ddots & \ddots & \ddots & \vdots \\ 0 & \cdots & 0 & * & * \end{pmatrix}$$

Ist A auf Hessenbergform transformiert, so wird diese durch die oben angegebene Iterationsvorschrift nicht zerstört:
Sei H_0 in Hessenbergform und sei $H_0 = Q_1 R_1$ die QR-Zerlegung mit Q_1 Orthogonalmatrix und R_1 obere Dreiecksmatrix.

$$\underbrace{\begin{pmatrix} * & * & * & * & * \\ * & * & * & * & * \\ 0 & * & * & * & * \\ 0 & 0 & * & * & * \\ 0 & 0 & 0 & * & * \end{pmatrix}}_{H_0} = \underbrace{\begin{pmatrix} * & * & * & * & * \\ * & * & * & * & * \\ a & * & * & * & * \\ * & * & * & * & * \\ * & * & * & * & * \end{pmatrix}}_{Q_1} \underbrace{\begin{pmatrix} b & * & * & * & * \\ 0 & * & * & * & * \\ 0 & 0 & * & * & * \\ 0 & 0 & 0 & * & * \\ 0 & 0 & 0 & 0 & * \end{pmatrix}}_{R_1}$$

$\Rightarrow a \cdot b = 0 \overset{b\neq 0}{\Rightarrow} a = 0$
Analog zeigt man induktiv, dass Q_1 auch Hessenbergform hat.
$H_1 = R_1 Q_1 \underset{\text{Dreiecksmatrix}}{\overset{R_1 \text{ obere}}{\Rightarrow}} H_1$ hat obere Hessenbergform. Die QR-Iteration erhält somit die Hessenbergform. Aufwand reduziert sich auf $\mathcal{O}(n^2)$.

Reduktion auf Hessenbergform

Mit Householdermatrizen (vgl. Abschnitt 5.6.2)

WICHTIG: Reduktion muss mit Hilfe von Ähnlichkeitstransformation vorgenommen werden, damit die Eigenwerte nicht verändert werden.

Sei P_i^T die Orthogonalmatrix (Householdermatrix), die von links an A multipliziert wird, die Einträge der i-ten Spalten unterhalb des ersten unteren Nebendiagonalelements auf 0 transformiert. Dann gilt:

$$\underbrace{\begin{pmatrix} * & * & * & * & * \\ * & * & * & * & * \\ * & * & * & * & * \\ * & * & * & * & * \\ * & * & * & * & * \end{pmatrix}}_{A} \to \underbrace{\begin{pmatrix} * & * & * & * & * \\ * & * & * & * & * \\ 0 & * & * & * & * \\ 0 & * & * & * & * \\ 0 & * & * & * & * \end{pmatrix}}_{P_1^\mathsf{T} A P_1} \to \underbrace{\begin{pmatrix} * & * & * & * & * \\ * & * & * & * & * \\ 0 & * & * & * & * \\ 0 & 0 & * & * & * \\ 0 & 0 & * & * & * \end{pmatrix}}_{P_2^\mathsf{T} P_1^\mathsf{T} A P_1 P_2} \to \underbrace{\begin{pmatrix} * & * & * & * & * \\ * & * & * & * & * \\ 0 & * & * & * & * \\ 0 & 0 & * & * & * \\ 0 & 0 & 0 & * & * \end{pmatrix}}_{P_3^\mathsf{T} P_2^\mathsf{T} P_1^\mathsf{T} A P_1 P_2 P_3}$$

Zur Erinnerung: Abschnitt 5.6.2
$v \in \mathbb{R}^n, \quad v \neq 0$ Gegeben: $P = I - \frac{2}{v^\mathsf{T} v} v v^\mathsf{T}$
Es ergibt sich folgender Algorithmus:

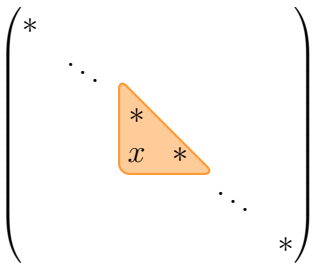
Algorithmus 3.2.3 Reduktion auf Hessenbergform mit Householdermatrizen

1: **for** $k \leftarrow 1$ **to** $n-2$ **do**
2: $\quad x \leftarrow A_{k+1:n,k}$ ▷ k-te Spalte unterhalb der unteren Nebendiagonalen
3: $\quad v_k \leftarrow \operatorname{sign}(x_1)\|x\|_2 e_1 + x$ ▷ $x = (x_1, \ldots, x_n)^\mathsf{T}$
4: $\quad v_k \leftarrow \frac{v_k}{\|v_k\|_2}$
5: $\quad A_{k+1:n,k:n} \leftarrow A_{k+1:n,k:n} - 2v_k(v_k^\mathsf{T} A_{k+1:n,k:n})$
6: $\quad A_{1:n,k+1:n} \leftarrow A_{1:n,k+1:n} - 2(A_{1:n,k+1:n}v_k)v_k^\mathsf{T}$
7: **end for**

$A^{(k)} = (Q^{(k)})^\mathsf{T} A^{(k-1)} Q^{(k)}$
$A^{(k-1)} = Q^{(k)} R^{(k)}$
$A^{(k)} := R^{(k)} Q^{(k)}$

Ziel des QR-Algorithmus ist es, die untere Nebendiagonale der Hessenbergmatrix auf (nahezu) 0 zu transformieren. Zunächst kann man das Problem in kleinere Teilprobleme zerlegen, sofern man schon Nebendiagonaleinträge hat, die (nahezu) 0 sind.
Testkriterium:

$$|h_{i+1,i}| \leq eps(|h_{i,i}| + |h_{i+1,i+1}|)$$



Ist ein solches Element vorhanden, so reduziert sich die Hessenbergmatrix: $H \to \begin{pmatrix} H_{1,1} & H_{1,2} \\ 0 & H_{2,2} \end{pmatrix}$, wobei $H_{i,i}$ mit $i = 1, 2$ Hessenbergform haben.

Satz 3.2.3 (Konvergenz des QR-Verfahrens):
[siehe 10, Kapitel 5.5]
Sei $A \in \mathbb{R}^{n\times n}$ in Hessenbergform mit Eigenwerten λ_i, $i = 1, \ldots, n$ die wie folgt angeordnet sind:

$$|\lambda_1| > |\lambda_2| > \cdots > |\lambda_n|$$

(Diese Bedingung bedeutet, dass keine konjugiert komplexe Eigenwerte auftauchen). Dann gilt für $A^{(k)} := R^{(k)}Q^{(k)}$

$$\lim_{k\to\infty} A^{(k)} = \begin{pmatrix} \lambda_1 & * & \cdots & * \\ 0 & \ddots & \ddots & \vdots \\ \vdots & \ddots & \ddots & * \\ 0 & \cdots & 0 & \lambda_n \end{pmatrix}$$

Für die Konvergenzrate gilt:

$$\left|h_{i+1,i}^{(k)}\right| = \mathcal{O}\left(\left|\frac{\lambda_{i+1}}{\lambda_i}\right|^k\right), \quad i = 1, \ldots, n-1$$

Unter der zusätzlichen Annahme, dass A symmetrisch ist, konvergiert die Folge $(A^{(k)})_{k\in\mathbb{N}}$ gegen eine Diagonalmatrix.

Beweis:
Siehe [12, Seite 62].

□

Shift-Strategie

Berechne statt der QR-Zerlegung von H_0 ($= A^{(0)}$ in Hessenbergform), die einer Matrix

$$H_0 - \mu I = QR$$

Dabei soll μ eine möglichst gute Näherung eines Eigenwertes λ sein. In der Praxis kennt man die Eigenwerte nicht, daher verwendet man als Schätzung die Eigenwerte der rechten unteren $k \times k$ Abschnittsmatrix von H_i in der Hoffnung, dass dann $h^{(i)}_{n-k,n-k-1} \approx \varepsilon$ klein wird.
Als einfachsten Fall verwendet man zur Schätzung von μ und $\overline{\mu}$ die beiden Eigenwerte von

$$\begin{pmatrix} h_{n-1,n-1} & h_{n-1,n} \\ h_{n,n-1} & h_{n,n} \end{pmatrix} = \text{untere rechte } 2 \times 2\text{-Block-Matrix von } H_k \tag{3.2}$$

Grundschema der QR-Iteration mit Shift

Gegeben sei H in Hessenbergform

For $k = 1, 2, 3, \ldots$

- Berechne die Eigenwerte μ_1 und μ_2 von (3.2)
- Bestimme die QR-Zerlegung von $(H - \mu_1 I)(H - \mu_2 I) = QR$
- Ersetze H durch $Q^\mathsf{T} H Q$
- Falls einige der $h_{i+1,i}$ klein genug sind, setze sie auf 0 und mache mit kleineren Blöcken weiter

In der praktischen Implementierung ist es wichtig, dass man in reeller Arithmetik rechnen kann:
Dazu sei $\mu_2 = \overline{\mu_1}$, dann gilt:

$$(H - \mu_1 I)(H - \overline{\mu_1} I) = H^2 - 2Re(\mu_1)H + |\mu_1|^2 I$$

Dieses Produkt muss man nicht explizit ausrechnen, wenn man die Strategie nach Francis benutzt.

29.01.2013
28. Vorlesung

Satz 3.2.4 (Implizites Q-Theorem):
Sei $A, Q \in \mathbb{R}^{n\times n}$, Q orthogonal, so dass $H = Q^\mathsf{T} A Q$ eine nicht reduzierte Hessenbergmatrix ist, d. h. $h_{i+1,i} \neq 0, \quad i = 1, \ldots, n-1$. Dann ist H bis auf Multiplikation von links und rechts mit einer Vorzeichenmatrix $\Sigma := \operatorname{diag}(\pm 1, \ldots, \pm 1)$ eindeutig bestimmt, falls die erste Spalte von Q fest vorgegeben wird.

Beweis (per vollständiger Induktion):
Sei $H = Q^\mathsf{T} A Q$ wie im Satz und sei $W \in \mathbb{R}^{n\times n}$ orthogonal, so dass $W^\mathsf{T} A W = \tilde{H}$ eine weitere nicht reduzierte Hessenbergmatrix ist. Zum Beweis der Eindeutigkeit genügt zu zeigen

$$Qe_1 = We_1 \Rightarrow W \overset{!}{=} Q\Sigma$$

Induktion über die ersten k Spalten von Q und W:
$k = 1$: ✓
$k \mapsto k+1$:
Sei $We_i = \sigma_i Q e_i,\ i = 1, \dots, k$ erfüllt mit $\sigma \in \{\pm 1\}$
$\Rightarrow \tilde{h}_{ij} = w_i^\mathsf{T} A w_j = \sigma_i \sigma_j q_i^\mathsf{T} A q_j = \sigma_i \sigma_j h_{i,j} \quad \forall i, j = 1, \dots, k$
Diese Aussage ist äquivalent zu:
Die führende $k \times k$ Abschnittsmatrix von H stimmt auf Vorzeichenskalierung mit $\tilde{H}$ überein. H und $\tilde{H}$ nicht reduziert $\Rightarrow h_{k+1,k} \neq 0 \neq \tilde{h}_{k+1,k}$
Durch Vergleich der k-ten Spalten von AQ und AW, d. h. $AQe_k = QHe_k$ und $AWe_k = W\tilde{H}e_k$ erhalten wir für w_{k+1} und analog für q_{k+1}

$$
\begin{aligned}
w_{k+1} &= \frac{1}{\tilde{h}_{k+1,k}} \left(A w_k - \sum_{i=1}^{k} \tilde{h}_{i,k} w_i \right) \\
&\overset{\text{I.V.}}{=} \frac{1}{\tilde{h}_{k+1,k}} \left(\sigma_k A q_k - \sum_{i=1}^{k} \sigma_i \sigma_k h_{i,k} \sigma_i q_i \right) \\
&\overset{\sigma_i^2 = 1}{=} \frac{\sigma_k}{\tilde{h}_{k+1,k}} \left(A q_k - \sum_{i=1}^{k} h_{i,k} q_i \right) \\
&\overset{Q^\mathsf{T} A Q = H}{=} \underbrace{\frac{\sigma_k}{\tilde{h}_{k+1,k}} h_{k+1,k}}_{=c} q_{k+1}
\end{aligned}
$$

$\Rightarrow w_{k+1}$ und q_{k+1} stimmen bis auf ein skalares Vielfaches überein.
w_{k+1} und q_{k+1} sind Spalten einer orthogonalen Matrix.
Also: $1 = \|w_{k+1}\| = |c| \|q_{k+1}\| = |c|$. Somit unterscheidet sich q_{k+1} und w_{k+1} nur um einen Faktor vom Betrag Eins.

□

Berechnung der ersten Spalten der Transformationsmatrix Q

1. Berechne den ersten Spaltenvektor von $(H - \mu_1 I)(H - \mu_2 I)$.

$$
x = (H - \mu_1 I)(H - \mu_2 I) = \begin{pmatrix} * \\ * \\ * \\ 0 \\ \vdots \\ 0 \end{pmatrix}
$$

2. Berechne die Householdertransformation P, so dass $Px = r_{1,1} e_1$. Dies ist der erste Schritt der QR-Zerlegung: $(H - \mu_1 I)(H - \mu_2 I) = QR$

3. Man ersetze H durch $P^\mathsf{T}HP$:

$$H \to P^\mathsf{T}HP = \begin{pmatrix} * & * & * & * & * & * \\ * & * & * & * & * & * \\ + & * & * & * & * & * \\ + & + & * & * & * & * \\ 0 & 0 & 0 & * & * & * \\ 0 & 0 & 0 & 0 & * & * \end{pmatrix}$$

4. Man bringe $P^\mathsf{T}HP$ wie zuvor mit einer Householdertransformation G auf Hessenbergform:

$$P^\mathsf{T}HP \to G^\mathsf{T}P^\mathsf{T}HPG$$

Dabei nutzt man aus, dass $P^\mathsf{T}HP$ schon fast Hessenbergform hat. Diese Aktion wird manchmal Buckelschieben genannt.

Beispiel:

$$P^\mathsf{T}HP \to \begin{pmatrix} * & * & * & * & * & * \\ * & * & * & * & * & * \\ 0 & * & * & * & * & * \\ 0 & + & * & * & * & * \\ 0 & + & + & * & * & * \\ 0 & 0 & 0 & 0 & * & * \end{pmatrix}$$

$$\to \begin{pmatrix} * & * & * & * & * & * \\ * & * & * & * & * & * \\ 0 & * & * & * & * & * \\ 0 & 0 & * & * & * & * \\ 0 & 0 & + & * & * & * \\ 0 & 0 & + & + & * & * \end{pmatrix} \to \begin{pmatrix} * & * & * & * & * & * \\ * & * & * & * & * & * \\ 0 & * & * & * & * & * \\ 0 & 0 & * & * & * & * \\ 0 & 0 & 0 & * & * & * \\ 0 & 0 & 0 & + & * & * \end{pmatrix}$$

$$\to \begin{pmatrix} * & \cdots & \cdots & \cdots & * \\ * & \ddots & & & \vdots \\ 0 & \ddots & \ddots & & \vdots \\ \vdots & \ddots & \ddots & \ddots & \vdots \\ 0 & \cdots & 0 & * & * \end{pmatrix}$$

Die erste Zeile und Spalte der gegebenen Matrix $P^\mathsf{T}HP$ bleibt unberührt. Also ist die erste Spalte von PG dieselbe wie die von P. Wegen Satz 3.2.4 (Implizites Q-Theorem) muss $Q = PG$ dann bereits die richtige Transformation gewesen sein.

QR-Algorithmus in zwei Schritten

1. Algorithmus 3.2.4 (QR-Schritt nach Francis)
2. Algorithmus 3.2.5 (QR-Algorithmus)

Material richtet sich nach [11]

Algorithmus 3.2.4 QR-Schritt nach FRANCIS [vgl. 11, Seite 206]

Berechnet für eine nicht-reduzierte-Hessenbergmatrix $H \in \mathbb{R}^{n \times n}$ die Transformierte $Q^\mathsf{T} H Q$ mit hoffentlich kleinem $h_{i+1,i}$

1: $m \leftarrow n - 1$

Berechne für die Eigenwerte μ_1 und μ_2 von $\begin{pmatrix} h_{n-1,n-1} & h_{n-1,n} \\ h_{n,n-1} & h_{n,n} \end{pmatrix}$ die Werte $s = \mu_1 + \mu_2$ und $d = \mu_1 * \mu_2$

2: $s \leftarrow H(m,m) + H(n,n)$ ▷ Spur

3: $d \leftarrow H(m,m) * H(n,n) - H(m,n) * H(n,m)$ ▷ Determinante

Berechne die erste Spalte von $(H - \mu_1 I)(H - \mu_2 I)$

4: $x \leftarrow H(1,1) * H(1,1) + H(1,2) * H(2,1) - s * H(1,1) + d$

5: $y \leftarrow H(2,1) * (H(1,1) + H(2,2) - s)$

6: $z \leftarrow H(2,1) * H(3,2)$

7: **for** $k \leftarrow 0$ **to** $n - 3$ **do**

Wende Householdertransformationen auf H an. Leitet Buckel bzw. schiebt Buckel weiter nach rechts unten.

8: $H(k{+}1 : k{+}3, k{+}1 : n) \leftarrow \mathit{householdermultlinks}(H(k{+}1 : k{+}3, k{+}1 : n), v, \beta)$

9: $r \leftarrow \min\{k + 4, n\}$

10: $H(r + 1, k + 1 : k + 3) \leftarrow \mathit{householdermultrechts}(H(r + 1, k + 1 : k + 3), v, \beta)$

Berechne die erste Spalte der Restmatrix

11: $x \leftarrow H(k + 2, k + 1)$

12: $y \leftarrow H(k + 3, k + 1)$

13: **if** $k \neq n - 3$ **then**

14: $z \leftarrow H(k + 4, k + 1)$

15: **end if**

16: **end for**

Berechne die letzte Householdertransformation

17: $[v, \beta] \leftarrow \mathit{householder}([x, y]^\mathsf{T})$

18: $H(n - 1 : n, n - 2 : n) \leftarrow \mathit{householdermultlinks}(H(n - 1 : n, n - 2 : n), v, \beta)$

19: $H(1 : n, n - 1 : n) \leftarrow \mathit{householdermultrechts}(H(1 : n, n - 1 : n), v, \beta)$

Die Routine $B = \mathit{householdermultlinks}(B, v, \beta)$ berechnet für eine Matrix B und Householdervektoren v, β die Form $B = (I + \beta v v^\mathsf{T})B$
Analog für $\mathit{householdermultrechts} : B = B(I + \beta v v^\mathsf{T})$

31.01.2013
29. Vorlesung

Algorithmus 3.2.5 QR-Algorithmus [vgl. 11, Seite 207]

Berechnet für $A \in \mathbb{R}^{n\times n}$ die reelle Schurform $Q^\mathsf{T} AQ = R$. Dabei wird A mit der reellen Schurform von A überschrieben.

1: Verwende Algorithmus 3.2.3 zur Reduktion auf Hessenbergform
2: Initialisiere $Q = I$

3: **repeat**
4: - Setze alle $h_{i+1,i} = 0$ für die $|h_{i+1,i}| \leq eps(|h_{i,i}| + |h_{i+1,i+1}|)$ gilt.
5: - Finde das größte $q \geq 0$ und das kleinste $p \geq 0$, so dass

$$H = \begin{pmatrix} H_{1,1} & H_{1,2} & H_{1,3} \\ 0 & H_{2,2} & H_{2,3} \\ 0 & 0 & H_{3,3} \end{pmatrix} \begin{matrix} p \\ n-p-q \\ q \end{matrix}$$
$$\begin{matrix} p & n-p-q & q \end{matrix}$$

wobei $H_{3,3}$ = Block-Dreiecksform und $H_{2,2}$ nicht-reduziert ist.

6: **if** $q < n$ **then**
7: - Mache einen QR-Schritt nach Francis (Algorithmus 3.2.4) mit
$H_{2,2} \mapsto Z^\mathsf{T} H_{2,2} Z$
$H_{1,2} \mapsto H_{1,2} Z$
$H_{2,3} \mapsto Z^\mathsf{T} H_{2,3}$
8: **end if**

Falls Q explizit benötigt wird:
9: $Q = Q\,\mathrm{diag}(I_p, Z, I_q)$
10: **until** $q = n$

3.3. Eigenwertabschätzungen

Zur Erinnerung: $\|\cdot\|$ sei submultiplikative Matrixnorm.
$A \in \mathbb{K}^{n\times n}$, λ Eigenwert von A

$$\Rightarrow |\lambda| \leq \underbrace{\rho(A)}_{\text{Spektralradius}} \leq \|A\| \quad \forall \lambda \in \underbrace{\sigma(A)}_{\text{Spektrum}}$$

Satz 3.3.1:
Alle Eigenwerte einer Matrix $A \in \mathbb{K}^{n\times n}$ liegen im Kreis um den Nullpunkt mit Radius $\|A\|$. Diese Abschätzung lässt sich noch verfeinern. Dazu führen wir den Wertevorrat oder -bereich $G(A)$ ein:

$$G(A) := \left\{ \frac{x^* Ax}{x^* x} : x \neq 0 \right\} = \textit{Menge aller Rayleigh-Quotienten}$$

Wähle x als Eigenvektor zum Eigenwert $\lambda \Rightarrow \sigma(A) \subset G(A)$.

Satz 3.3.2:
$n \times n$-Matrix A, $H := \frac{1}{2}(A + A^*)$ (hermitescher Anteil von A), $S := \frac{1}{2i}(A - A^*)$ (schief-hermitescher Anteil von A). Dann gilt für jeden Eigenwert $\lambda \in \sigma(A)$

1. $\lambda_{min}(H) \leq Re(\lambda) \leq \lambda_{max}(H)$
2. $\lambda_{min}(S) \leq Im(\lambda) \leq \lambda_{max}(S)$

Beweis:
$\sigma(A) \subset G(A)$, sei $\lambda \in G(A)$ beliebig.

$$\begin{aligned} Re(\lambda) &\leq \max_{x \neq 0} Re\left(\frac{x^*Ax}{x^*x}\right) \\ &= \max_{x \neq 0} \frac{1}{2}\left(\frac{x^*Ax + x^*A^*x}{x^*x}\right) \\ &= \max_{x \neq 0}\left(\frac{x^*Hx}{x^*x}\right) \\ &\overset{H \text{ hermitesch}}{=} \lambda_{\max}(H) \end{aligned}$$

Analog: $Im(\lambda) \leq \lambda_{\max}(S)$.
Die unteren Abschätzungen funktionieren ebenfalls analog.

□

Satz 3.3.3 (1. Satz von Gerschgorin):
Sei $A \in \mathbb{K}^{n \times n}$, $r_i := \sum_{\substack{j=1 \\ j \neq i}}^{n} |a_{i,j}|$ und $k_i := \{z \in \mathbb{C} \colon |z - a_{ii}| \leq r_i\}$. Für die ***Gerschgorin-konstante*** *k_i gilt dann, dass alle Eigenwerte von A in $\bigcup_{i=1}^{n} k_i$ enthalten sind.*

$$\sigma(A) \subset \bigcup_{i=1}^{n} k_i$$

Beweis:
Betrachte die Zerlegung $A = D + E$ mit $D := \operatorname{diag}(A)$, $E := A - D =: (e_{ij})_{i,j}$.

$$\Rightarrow \operatorname{diag}(E) = 0$$

Sei nun $\lambda \in \sigma(A)$ und x der zugehörige Eigenvektor und $B_\lambda := A - \lambda I = (D - \lambda I) + E$.

$$\begin{aligned} B_\lambda x = 0 &\Leftrightarrow (D - \lambda I)x + Ex = 0 \\ &\Leftrightarrow x = -(D - \lambda I)^{-1}Ex \\ \Rightarrow \quad \cancel{\|x\|_\infty} &\leq \|(D - \lambda I)^{-1}\|_\infty \cdot \|E\|_\infty \cdot \cancel{\|x\|_\infty} \\ \Rightarrow \quad 1 &\leq \|(D - \lambda I)^{-1}\|_\infty \cdot \|E\|_\infty \\ &= \sum_{\substack{j=1 \\ j \neq k_0}}^{n} \frac{|e_{k_0,j}|}{|a_{k_0,k_0} - \lambda|} \\ &= \sum_{\substack{j=1 \\ j \neq k_0}}^{n} \frac{|a_{k_0,j}|}{|a_{k_0,k_0} - \lambda|} \end{aligned}$$

für ein $k_0 \in \{1, \ldots, n\}$.
$\Rightarrow |a_{k_0,k_0} - \lambda| \leq \sum_{\substack{j=1 \\ j \neq k_0}}^{n} |a_{k_j,j}| = r_{k_0}$
$\Rightarrow \lambda \in k_{k_0}$

□

$\sigma(A) = \sigma(A^\mathsf{T}) \overset{\text{Satz 3.3.3}}{\Rightarrow} \sigma(A) \subset \bigcup_{j=1}^n \tilde{k}_j$ mit $\tilde{k}_j := \left\{ z \in \mathbb{C} \colon |z - a_{jj}| \leq \sum_{\substack{i=1 \\ i \neq j}}^n |a_{i,j}| \right\}$

$\Rightarrow$ Satz 3.3.4 (2. Satz von Gerschgorin).

Satz 3.3.4 (2. Satz von Gerschgorin):
$A \in \mathbb{K}^{n \times n}$, dann gilt:

$$\sigma(A) \subset \bigcup_{j=1}^n k_j \cap \bigcup_{j=1}^n \tilde{k}_j$$

Satz 3.3.5 (3. Satz von Gerschgorin):
Für ein $m \in \{1, \ldots, n\}$ seien

$$D_1 := \bigcup_{i=1}^m k_i \text{ und } D_2 := \bigcup_{i=m+1}^n k_i$$

Falls $D_1 \cap D_2 = \emptyset$, dann enthält D_1 genau m Eigenwerte von A jeweils nach der algebraischen Vielfachheit gezählt. Die übrigen Eigenwerte sind in D_2 enthalten.

Beweis:
Sei E wie im Beweis von Satz 3.3.3.

$$A_\varepsilon := D + \varepsilon E$$
$$A_0 = D_1 A_1 = A$$
$$p(\lambda, \varepsilon) := \det(A_\varepsilon - \lambda I) \text{ charakteristisches Polynom von } A_\varepsilon$$

$\Rightarrow$ Eigenwerte $\lambda_i(\varepsilon)$ von A_ε hängen stetig von ε ab. Da die Koeffizienten von $p(\lambda, \varepsilon)$ stetig abhängig von ε sind, gilt das auch für die Nullstellen. Die Werte $\lambda_i(\varepsilon)$ bilden für $0 \leq \varepsilon \leq 1$ eine zusammenhängende Kurve in $\mathbb{C}$. Für $\varepsilon = 0$ ist $A_0 = D$ und dementsprechend $\lambda_i(0) = a_{ii}$

$$k_i = k_i(0) = a_{ii}$$

Die m Eigenwerte $\lambda_1(\varepsilon), \ldots, \lambda_m(\varepsilon)$ mit denen $D_1(\varepsilon)$ gebildet wird, bleiben in den entsprechenden Kreisen $k_i(\varepsilon)$, $i = 1, \ldots, m$ für wachsendes ε. Sie können D_1 nicht verlassen. Da $D_1(\varepsilon)$ disjunkt von $D_2(\varepsilon)$ bleibt $\forall \varepsilon \leq 1$, gilt $\lambda_i(1) \in D_1$ für $i = 1, \ldots, m$.

□

Beispiel:

$$A = \begin{pmatrix} 2 & -1 & 0 & \cdots & 0 \\ -1 & 2 & -1 & \ddots & \vdots \\ 0 & \ddots & \ddots & \ddots & 0 \\ \vdots & \ddots & -1 & 2 & -1 \\ 0 & \cdots & 0 & -1 & 2 \end{pmatrix}$$

$$A = A^\mathsf{T} \Rightarrow \sigma(A) \subset \mathbb{R} \tag{3.3}$$

Die Radien der Gerschgorinkreise sind Eins und Zwei, alle haben den Mittelpunkt Zwei.

$$\overset{\text{Satz 3.3.3}}{\Rightarrow} \quad \sigma(A) \subset B_2(2) \overset{(3.3)}{\Rightarrow} \sigma(A) \subset [0, 4]$$
$$\Rightarrow \quad A \text{ ist symmetrisch positiv semidefinit}$$

A. Bibliografie

A.a. Literaturverzeichnis

[1] Harro Heuser. *Lehrbuch der Analysis 2 - Mit 632 Aufgaben, zum Teil mit Lösungen*. 12. Auflage. Berlin: Teubner B.G. GmbH, 2002. ISBN: 978-3-519-52232-4 (siehe S. 4, 14, 15).

[2] Otto Forster. *Analysis 1*. Berlin: Springer, 2006. ISBN: 978-3-834-89089-4 (siehe S. 5).

[3] Otto Forster. *Analysis 2 - Differentialrechnung im $\mathbb{R}^n$, Gewöhnliche Differentialgleichungen*. Berlin: Springer, 2008. ISBN: 978-3-834-89541-7 (siehe S. 12).

[4] Peter Deuflhard. *Newton Methods for Nonlinear Problems - Affine Invariance and Adaptive Algorithms*. Berlin, Heidelberg: Springer, 2011. ISBN: 978-3-642-23899-4 (siehe S. 16, 17).

[5] Michael L. Overton. *Numerical Computing with IEEE Floating Point Arithmetic - Including One Theorem, One Rule of Thumb, and One Hundred and One Exercises*. Cambridge: Cambridge University Press, 2001. ISBN: 978-0-898-71571-2 (siehe S. 30, 32, 34).

[6] Lloyd N. Trefethen und David Bau III. *Numerical Linear Algebra*. Philadelphia: SIAM, 1997. ISBN: 978-0-898-71361-9 (siehe S. 46, 56, 57, 68).

[7] Roland W. Freund und Ronald H.W. Hoppe. *Stoer/Bulirsch: Numerische Mathematik 1*. Berlin: Springer DE, 2007. ISBN: 978-3-540-45390-1 (siehe S. 69).

[8] Wladimir I. Smirnow. *Lehrgang der höheren Mathematik - 3,2*. 13. unveränd. Aufl. Thun, Frankfurt am Main: Harri Deutsch Verlag, 1987. ISBN: 978-3-817-11300-2 (siehe S. 78).

[9] Peter Deuflhard und Andreas Hohmann. *Numerische Mathematik 1 - Eine algorithmisch orientierte Einführung*. Berlin: Walter de Gruyter, 2008. ISBN: 978-3-110-20355-4 (siehe S. 85).

[10] Alfio Quarteroni, Riccardo Sacco und Fausto Saleri. *Numerische Mathematik 2*. Berlin: Springer-Verlag, 2002. ISBN: 978-3-642-56191-7 (siehe S. 94, 181).

[11] Matthias Bollhöfer und Volker Mehrmann. *Numerische Mathematik - Eine Projektorientierte Einführung Für Ingenieure, Mathematiker und Naturwissenschaftler*. Wiesbaden: Vieweg+Teubner Verlag, 2004. ISBN: 978-3-528-03220-3 (siehe S. 104, 114, 118, 184–186, VIII).

[12] Josef Stoer und Roland Bulirsch. *Numerische Mathematik 2 - Eine Einführung - Unter Berücksichtigung Von Vorlesungen Von F.L.Bauer*. Berlin: Springer-Verlag, 2007. ISBN: 978-3-540-26268-8 (siehe S. 112, 114, 115, 182).

[13] Richard Courant, David Hilbert und Peter D. Lax. *Methoden Der Mathematischen Physik*. Berlin: Springer-Verlag, 1993. ISBN: 978-3-642-58039-0 (siehe S. 131).

[14] R. Courant, K. Friedrichs und H. Lewy. „Über die partiellen Differenzengleichungen der mathematischen Physik“. German. In: 100 (1 1928), S. 32–74. ISSN: 0025-5831. DOI: 10.1007/BF01448839. URL: http://dx.doi.org/10.1007/BF01448839 (siehe S. 135).

[15] Alfio M. Quarteroni und Alberto Valli. *Numerical Approximation of Partial Differential Equations*. 1st ed. 1994. 2nd printing 2008. Berlin, Heidelberg: Springer-Verlag, 2008. ISBN: 978-3-540-85267-4 (siehe S. 139).

[16] Murray H. Protter und Hans F. Weinberger. *Maximum Principles in Differential Equations*. Softcover reprint of the original 1st ed. 1984. London: Springer London, Limited, 2011. ISBN: 978-1-461-29769-7 (siehe S. 144).

[17] Aslak Tveito und Ragnar Winther. *Einführung in partielle Differentialgleichungen - Ein numerischer Zugang*. Berlin: Springer-Verlag, 2002. ISBN: 978-3-540-42404-8 (siehe S. 148).

B. Index

-M-

-N-

-O-

-P-

-R-

-S-

-T-

-U-

-V-

-W-

-Z-

C. Vorlesungsverzeichnis

D. Algorithmen

E. Theoreme

F. Definitionen

G. Bemerkungen