

Numerik I Dörfler SS08 - Vorlesungsmitschrieb

Inhaltsverzeichnis

0.1	Aufgaben	3
0.2	Hilfsmittel	3
1	Anwendungsbeispiele	3
1.1	ComputerTomographie	3
1.1.1	Modell	3
1.1.2	Das Tomographie-Problem	3
1.1.3	Ein diskretes Tomographie-Problem	5
1.2	Wärmeleitung	6
1.2.1	Wärmeleitungsgleichung	6
1.2.2	Diskretisierung	7
1.3	Berechnung elektrostatischer Felder	8
1.3.1	Elektrostatische Potenziale und Felder	8
1.3.2	Das Prinzip der virtuellen Arbeit	8
1.3.3	Das Poisson-Problem	9
1.3.4	Diskretisierung des Poissonproblems	10
1.3.5	Konvergenzbetrachtung	11
2	Rundungsfehler und numerische Stabilität	11
2.1	Grenzen der Genauigkeit	11
2.2	Zahldarstellung	12
2.2.1	Zahlssysteme	12
2.2.2	Maschinenzahlen	12
2.2.3	Rundungsfehleranalyse	13
2.3	Konditionen von Abbildungen	15
2.3.1	Norm- und komponentenweise Kondition	15
2.3.2	Beispiele	15
2.4	Stabilität numerischer Algorithmen	16
2.4.1	Vorwärtsanalyse	17
2.4.2	Rückwärtsanalyse	17

3	Lineare Gleichungssysteme	18
3.1	Direkte Verfahren: Gauß-Elimination	18
3.1.1	Das Gaußsche Eliminationsverfahren	18
3.1.2	Die LR-Zerlegung	19
3.1.3	Pivotisierung	22
3.1.4	Rechenaufwand	23
3.1.5	Gauß-Elimination für Bandmatrizen	23
3.1.6	Block-Gauß-Elimination	24
3.1.7	Existenz der LR -Zerlegung ohne Pivotisierung	25
3.1.8	Numerische Stabilität	26
3.1.9	Bemerkungen	26
3.2	Cholesky-Zerlegung	27
3.3	Iterative Verfahren	28
3.3.1	Basisiteration	28
3.3.2	Konvergenz linearer Iterationen	28
3.3.3	Die „klassischen Iterationsverfahren“	30
3.3.4	Konvergenz des Jakobi- und Gauß-Seidel-Verfahrens	32
3.3.5	Konvergenzsatz des SOR-Verfahrens	33
3.3.6	Konvergenz des SSOR	34
3.3.7	Beispiele	34
3.3.8	Konsistent geordnete Matrizen	35
3.3.9	Rechenaufwand	36
3.3.10	Idee Des Mehrgitterverfahrens	38
3.4	Das CG-Verfahren	39
3.4.1	Das Gradientenverfahren	39
3.4.2	Fehlerminimierung auf Unterräumen	40
3.4.3	Krylovräume	41
3.4.4	Das CG-Verfahren nach Hestenes/ Stiefel (1954)	41
3.4.5	Konvergenz des CG-Verfahrens	44
3.4.6	Vorkonditionierung	47
3.5	GMRES (Generalized minimal residuals, 1986)	48
3.5.1	Minmale Residuen	48
3.5.2	Konstruktion des GMRES-Verfahrens	49
4	Nichtlineare Gleichungen	53
4.1	Fixpunkte (Ergänzung 5)	53
4.1.1	Fixpunkte und Nullstellen	53
4.1.2	Banachscher Fixpunktsatz	53
4.1.3	Beispiele	53
4.1.4	Konvergenzordnung	55
4.2	Berechnung von Nullstellen	56
4.2.1	Extrema (Ergänzung 7)	56
4.2.2	Nullstellen reeller Funktionen	56
4.2.3	Lokale Konvergenz des Newtonverfahrens	60

Einteilung der angewandten und numerischen Mathematik

0.1 Aufgaben

- Modellbildung (mathematische Formulierung für physikalische, technische, biologische, ökonomische, ... Prozesse)
- Diskretes Modell (Reduktion auf ein Modell mit endlich vielen zu bestimmenden Parametern)
- Algorithmenentwurf (Befehlsfolge zur Lösung des diskreten Problems)
- Nachweis der „Konvergenz“ und „Stabilität“
- Komplexität und Effizienz

0.2 Hilfsmittel

- Ana I-III, lineare Algebra, Funktionalanalysis, partielle Differentialgleichungen und andere „reine Mathematik“
- Programmiersprachen
- Rechnerarchitekturen
- Kenntnisse im Anwendungsgebiet
- Bandbreite: Numerische Analysis - wissenschaftliches Rechnen

1 Anwendungsbeispiele

1.1 Computertomographie

1.1.1 Modell

Tomographie-Problem:

Rekonstruiere aus den Intensitätsmessungen die innere Struktur von Ω .

1.1.2 Das Tomographie-Problem

x Koordinate längs eines Strahles S ,

$I(x)$ Intensität in x , $I(0) = I_0$, $I_S = I(x_D)$, $S = [0, x_D]$

$\varrho(x)$ Absorptionskoeffizient in x : $\varrho(x) \geq 0$ für $x \in [0, x_D]$ und $\varrho = 0$ außerhalb von Ω

Modell der Absorption

Abnahme der Intensität zwischen x und $x + \Delta x$ (Δx klein) ist proportional zur Intensität

$$I(x + \Delta x) - I(x) \sim -I(x)\Delta x$$

Bildchen

Wir setzen daher $I(x + \Delta x) - I(x) = -\varrho(x)I(x)\Delta x + \underbrace{\mathcal{O}(\Delta x^2)}_{\leq C(\Delta x)^2}$.

Teilen durch Δx und $\Delta x \rightarrow 0$ führt auf

$$\frac{dI}{dx}(x) = I'(x) = -\varrho(x)I(x) \quad \forall x \in S$$

Für $I(x) > 0$ gilt

$$(\log(I(x)))' = \frac{I'(x)}{I(x)} = -\varrho(x)$$

Integration von 0 nach x_D liefert:

$$\log\left(\frac{I_0}{I_S}\right) = \int_0^{x_D} \varrho(x) dx = \int_S \varrho$$

Die Radontransformation

Zu einem Winkel φ betrachten wir ein Bündel von Parallelstrahlen, welche mittels s parametrisiert sind.

$$\omega(\varphi) = [\cos(\varphi); \sin(\varphi)]$$

d.h. $|\omega(\varphi)| = 1$. $\omega(\varphi)^\top$ sei der um $\frac{\pi}{2}$ gedrehte Vektor in mathematisch positiver Richtung (Gegenuhreigersinn)

Zu $\varrho : \mathbb{R}^2 \rightarrow \mathbb{R}$, gegeben, mit Träger in Ω ($\text{supp}(\varrho) := \overline{\{x \in \mathbb{R}^2 : \varrho(x) > 0\}}$) definieren wir die Radontransformierte $R_\varrho : \mathbb{R} \times [0, 2\pi] \rightarrow \mathbb{R}$ wie folgt:

$$R_\varrho(\delta, \varphi) = \int_{\mathbb{R}} \varrho(\delta\omega(\varphi) + t\omega(\varphi)^\top) dt$$

Bemerkung

Die Radontransformierte R ist linear: $R(\lambda\varrho_1 + \varrho_2) = \lambda R_{\varrho_1} + R_{\varrho_2}$ für alle $\lambda \in \mathbb{R}$ und Funktionen ϱ_1, ϱ_2 .

Mathematisches Tomographie-Problem:

Finde zu gegebenem $f : \mathbb{R} \times [0, 2\pi] \rightarrow \mathbb{R}$ ein $\varrho : \mathbb{R}^2 \rightarrow \mathbb{R}$ mit $R_\varrho = f$

Aufgabe:

Existenz und Eindeutigkeit einer Lösung (unter Voraussetzungen). Diskutiere „Stabilität“: Ist Δf eine Störung des Datums f und $\Delta\varrho$ die daraus resultierende Störung der Lösung ϱ , gilt dann $\|\Delta\varrho\| \leq C\|\Delta f\|$ mit nicht zu großem C ($\|\cdot\|$ Abstand)

1.1.3 Ein diskretes Tomographie-Problem

Datenerhebung ist diskret

s_1, \dots, s_n Parameter der Parallelstrahlen

$\varphi_1, \dots, \varphi_m$ Winkeleinstellungen

Problem: Zu gegebenem $f : \mathbb{R} \times [0, 2\pi) \rightarrow \mathbb{R}$ finde $\varrho : \mathbb{R}^2 \rightarrow \mathbb{R}$ mit

$$R_{\varrho}(s_i, \varphi_j) = f(s_i, \varphi_j) \quad i = 1, \dots, n; \quad j = 1, \dots, m$$

So nicht lösbar, denn es gibt unendlich viele ϱ , die dies lösen.

Wir benötigen ein endlich dimensionales Modell für ϱ

Idee: Führe Rasterungen ein (Fernsehen, Zeitung)

lexikographische Anordnung: Charakteristische Funktion einer Zeile $Z_i : \chi_{Z_i} : \mathbb{R}^2 \rightarrow \mathbb{R}$

$$\chi_{Z_i} = \begin{cases} 1, & x \in Z_i \\ 0, & \text{sonst} \end{cases}$$

Ansatz für $\tilde{\varrho}$ (diskretes Modell)

$$\tilde{\varrho}(x) = \sum_{i=1}^M \tilde{\varrho}_i \chi_{Z_i}(x)$$

Die Zahlen $\tilde{\varrho}_i$ sind zu bestimmen aus den Messdaten.

Einsetzen:

$$f(s_i, \varphi_j) \stackrel{!}{=} R_{\tilde{\varrho}}(s_i, \varphi_j) = R\left(\sum_{i=1}^M \tilde{\varrho}_i \chi_{Z_i}\right)(s_i, \varphi_j) \stackrel{R \text{ linear}}{=} \sum_{i=1}^M \tilde{\varrho}_i (R\chi_{Z_i})(s_i, \varphi_j)$$

Lexikographische Anordnung der Punktepaaire $[s_i, \varphi_j]$:

$$\underbrace{[s_1, \varphi_1]}_{=x_1}, \underbrace{[s_2, \varphi_1]}_{=x_2}, \dots, \underbrace{[s_n, \varphi_1]}_{=x_n}, \underbrace{[s_1, \varphi_2]}_{=x_{n+1}}, \dots, \underbrace{[s_n, \varphi_m]}_{=x_N}, \quad N = n \cdot m$$

Eindeutige Zuordnung

$$x_k \leftrightarrow [s_i, \varphi_j], \quad k = (j-1)n + i$$

Wir schreiben: $f_k := f(s_i, \varphi_j)$, $A_{kl} = R\chi_{Z_l}(x_k) = R\chi_{Z_l}(s_i, \varphi_j)$ und erhalten

$$\sum_{l=1}^M A_{kl} \tilde{\varrho}_l = f_k \quad k = 1, \dots, N$$

Dies kann man als lineares Gleichungssystem $Au = b$ schreiben mit $A = [A_{kl}]_{kl} \in \mathbb{R}^{N, M}$; $b = [f_k]_k \in \mathbb{R}^N$; $u = [\tilde{\varrho}_l]_l \in \mathbb{R}^M$

1.2 Wärmeleitung

1.2.1 Wärmeleitungsgleichung

Wärmetransport entlang eines Stabes oder Drahtes (Eindimensionale Struktur)

Bild

$\Omega = (0, 1)$, Variablen: t Zeit, x Ort

$q(t, x)$ Wärmestrom in x zur Zeit t

Erhaltungssatz

Die zeitliche Änderung des Energieinhaltes in $I \subset \mathbb{R}$ ist gleich der Wärmeflussbilanz über dem Rand von I zuzüglich der in I erzeugten oder verbrauchten Energie.

$$\begin{aligned}\partial_t \left(\int_I u(t, x) \, dx \right) &= q(t, x_+) - q(t, x_-) + \int_I \underbrace{\varrho(t, x)}_{\text{Quelldichte}} \, dx \\ &\Leftrightarrow \\ \int_I [\partial_t u(t, x) - \partial_x q(t, x) - \varrho(t, x)] \, dx &= 0\end{aligned}$$

$I = [x_-, x_+]$

Da I beliebig

$$\partial_t u(t, x) - \partial_x q(t, x) = \varrho(t, x) \quad \forall x \in (0, 1), t > 0$$

Fourier: $q(t, x) \sim \partial_x u(t, x)$, also zum Beispiel

$$q(t, x) = \underbrace{a(t, x)}_{\text{Wärmeleitkoeff.}} \partial_x u(t, x)$$

Wir erhalten dann die Wärmeleitungsgleichung:

$$\partial_t u(t, x) - \partial_x (a(t, x) \partial_x u(t, x)) = \varrho(t, x) \quad (*)$$

Ziel: Gegeben $\alpha, \beta \in \mathbb{R}$, $\varrho : \mathbb{R}_{>0} \times [0, 1] \rightarrow \mathbb{R}$, $\varphi : [0, 1] \rightarrow \mathbb{R}$, $a : \mathbb{R}_{>0} \times (0, 1) \rightarrow \mathbb{R}_{>0}$, finde $u : \mathbb{R}_{\geq 0} \times (0, 1) \rightarrow \mathbb{R}$, welches $(*)$ löst und $u(t, 0) = \alpha$, $u(t, 1) = \beta$ und $u(0, x) = \varphi(x)$

Beispiele:

- keine Erzeugung, kein Verbrauch: $\varrho(t, x) = 0$
- Wärmeabstrahlung: $\varrho(t, x) = \sigma u(t, x)^4$ (bei Draht)
- Chemische Reaktion: $\varrho(t, x) = \omega e^{-\lambda/u(t, x)}$ (Arrhenius Gesetz)

Fragestellungen der Analysis

- Formulierung der Gleichung
- Existenz von Lösungen
- Qualitative Eigenschaften der Lösung

stationäres Problem: Wir betrachten das zeitunabhängige Problem und lassen die Variable t weg (und $A = 1, a = 0, b = 1$). Es ergibt sich das RWP

$$\begin{cases} -u''(x) = \varrho(x) = \varphi(x, u(x)), & \forall x \in (0, 1) \\ u(0) = \alpha, \quad u(1) = \beta \end{cases}$$

1.2.2 Diskretisierung

Numerik des stationären Modells. Suchen endliches Modell.

Finite Differenzen: Wähle ein uniformes Gitter, d.h. zu $N \in \mathbb{N}$ wählen wir $h = \frac{1}{N+1}$ und „Gitterpunkte“ $x_i = ih$ für $i = 0, \dots, N+1$ ($N+2$ Punkte). Wir suchen Approximationen u_i an $u(x_i)$. Randbedingungen $u_0 = \alpha, u_{N+1} = \beta$

$$\begin{aligned} u'(x_i) &\approx \frac{u_i - u_{i-1}}{h} \\ u''(x_i) &\approx \frac{u'(x_{i+1}) - u'(x_i)}{h} \approx \frac{u_{i+1} - 2u_i + u_{i-1}}{h^2} \end{aligned}$$

Also:

$$\begin{aligned} u_0 &= \alpha, \\ -u_{i-1} + 2u_i - u_{i+1} &= h^2 \varphi(x_i, u_i) \quad i = 1, \dots, N \\ u_{N+1} &= \beta \end{aligned}$$

Als Gleichungssystem:

$$\begin{bmatrix} 1 & & & & \\ -1 & 2 & -1 & & \\ & \ddots & \ddots & \ddots & \\ & & -1 & 2 & -1 \\ & & & & 1 \end{bmatrix} \cdot \begin{bmatrix} u_0 \\ u_1 \\ \vdots \\ u_N \\ u_{N+1} \end{bmatrix} = h^2 \begin{bmatrix} 0 \\ \varphi(x_1, u_1) \\ \vdots \\ \varphi(x_N, u_N) \\ 0 \end{bmatrix} + \begin{bmatrix} \alpha \\ 0 \\ \vdots \\ 0 \\ \beta \end{bmatrix}$$

$\Leftrightarrow Au_h = \Phi(u_h)$ mit $A \in \mathbb{R}^{N+2, N+2}$, $u_h \in \mathbb{R}^{N+2}$, $\Phi : \mathbb{R}^{N+2} \rightarrow \mathbb{R}^{N+2}$

Besteht rechts keine Abhängigkeit von u_h , so ist dies ein lineares Gleichungssystem. Andernfalls ist es ein Nullstellenproblem:

$$F(u_h) = Au_h - \Phi(u_h) \stackrel{!}{=} 0$$

Fragestellungen der Numerischen Analysis:

1. Gilt $u_n \rightarrow u$ für $N \rightarrow \infty$? In welchem Sinne?
2. Wie findet man Nullstellen von F (N groß)?
3. Wie löst man Gleichungssysteme für große N ?
4. A ist „dünnbesetzt“, d.h. hat nur 3 Nichtnullelemente pro Zeile, unabhängig von N .
5. Lösbarkeit der diskreten Gleichung? Eigenschaften von u_h
6. Verfahren effizient? Wie viele Operationen braucht ein Algorithmus? Was wäre ggf. optimal?
7. Aussagen über die Güte des Resultats

1.3 Berechnung elektrostatischer Felder

Bild

$\Phi : \mathbb{R}^2 \setminus \Omega \rightarrow \mathbb{R}$ „Potenzial“, $\Phi(x) \rightarrow 0$ für $|x| \rightarrow \infty$

Elektrisches Feld: $E = -\nabla\Phi = \begin{bmatrix} -\partial_1\Phi \\ -\partial_2\Phi \\ -\partial_2\Phi \end{bmatrix}$

1.3.1 Elektrostatische Potenziale und Felder

Bild $\partial\Omega = \partial O \cup \Gamma$

Wir suchen Φ mit $\Phi = 0$ auf Γ , $\Phi = 1$ auf ∂O .

Φ heißt Potenzial und $E := -\nabla\Phi$ das elektrische Feld (oder $\text{grad}(\Phi)$)

1.3.2 Das Prinzip der virtuellen Arbeit

Wie sieht Φ in Ω aus? Wir definieren eine Menge von Funktionen:

$$U := \{\varphi \in C^1(\Omega, \mathbb{R}) : \varphi = 0 \text{ auf } \Gamma, \varphi = 1 \text{ auf } \partial O\}$$

die Menge der zulässigen Potenziale. Das gesuchte Potenzial Φ ist dasjenige mit minimaler Feldenergie ε in U , d.h. mit $\varepsilon : U \rightarrow \mathbb{R}_{\geq 0}$ def. durch

$$\varepsilon(\Psi) = \frac{1}{2} \int_{\Omega} |\nabla\Psi|^2 = \frac{1}{2} \int_{\Omega} |\partial_1\Psi|^2 + |\partial_2\Psi|^2$$

gilt $\varepsilon(\Phi) = \min_{\Psi \in U} \varepsilon(\Psi)$

Weiter def. wir $U_0 := \{\xi \in C^1(\Omega, \mathbb{R}) : \xi = 0 \text{ auf } \partial\Omega\}$. Dann gilt: mit $\Phi \in U$ ist auch

$\Phi + t\zeta \in U$, falls $\zeta \in U_0$ und $t \in \mathbb{R}$ ist. Ist Φ ein Minimum von ε , so wird die reellwertige Funktion $t \mapsto \varepsilon(\Phi + t\zeta)$ stationär in $t = 0$ sein.

$$\varepsilon'(\Phi)[\zeta] = \frac{d}{dt}\varepsilon(\Phi + t\zeta)|_{t=0} \stackrel{!}{=} 0$$

Es folgt

$$\begin{aligned} 0 &\stackrel{!}{=} \frac{1}{2} \int_{\Omega} |\nabla(\Phi + t\zeta)|^2 \\ &= \frac{1}{2} \frac{d}{dt} \cdot \left(\int_{\Omega} \{|\nabla\Phi|^2 + 2t\nabla\Phi \cdot \nabla\zeta + t^2 \cdot |\nabla\zeta|^2\} \right) \\ &= \int_{\Omega} \{\nabla\Phi \cdot \nabla\zeta + t|\nabla\zeta|^2\} \end{aligned}$$

d.h. für $t = 0$:

$$0 = \int_{\Omega} \nabla\Phi \cdot \nabla\zeta \quad \forall \zeta \in U_0$$

„Das Prinzip der virtuellen Arbeit“, „Variationsgleichung“ Erfüllt Φ die Variationsgleichung, ist es dann ein Minimum?

Sei $\Phi \in U$ beliebig. Dann ist $\Psi - \Phi \in U_0$. Es gilt:

$$\begin{aligned} \varepsilon(\Psi) &= \varepsilon(\Phi + \underbrace{\Psi - \Phi}_{\in U_0}) \\ &= \varepsilon(\Phi) + \underbrace{\int_{\Omega} \nabla\Phi \cdot \nabla(\Psi - \Phi)}_{=0} + \frac{1}{2} \int_{\Omega} |\nabla(\Psi - \Phi)|^2 \varepsilon(\Phi) \\ &\geq \varepsilon(\Phi) \end{aligned}$$

Sogar: $\varepsilon(\Psi) > \varepsilon(\Phi)$, falls $\Psi \neq \Phi$. Denn: $\int_{\Omega} |\nabla(\Psi - \Phi)|^2 = 0 \Rightarrow \nabla(\Psi - \Phi)(x) = 0 \quad \forall x \in \Omega \Rightarrow (\Psi - \Phi)(x) = \text{const in } \Omega \Rightarrow \Psi = \Phi \text{ in } \Omega$, da $\Psi - \Phi|_{\partial\Omega} = 0$ ist.

1.3.3 Das Poisson-Problem

Gaußscher Integralsatz:

$$\int_{\Omega} \nabla\Phi \cdot \nabla\zeta = - \int_{\Omega} \nabla \cdot \nabla\Phi \zeta, \text{ da } \zeta|_{\partial\Omega} = 0$$

Es gilt $\nabla \cdot \nabla = \text{div}(\text{grad}) = \Delta = \partial_1^2 + \partial_2^2 \Rightarrow \int_{\Omega} \Delta\Phi \zeta = 0 \quad \forall \zeta \in U_0$

$$\Rightarrow \Delta\Phi = \partial_1^2\Phi + \partial_2^2\Phi = 0$$

Allgemein: Poisson-Problem

Zu $\Omega \subset \mathbb{R}^d$ und $f : \Omega \rightarrow \mathbb{R}$, $r : \Omega \rightarrow \mathbb{R}$ finde $u : \Omega \rightarrow \mathbb{R}$ mit

$$\begin{aligned} -\Delta u &= f \text{ in } \Omega \\ u &= r \text{ auf } \Omega \end{aligned}$$

1.3.4 Diskretisierung des Poissonproblems

$\Omega = (0, 1)^2$. Gitter sei $z_{ij} = \left[\frac{i}{n+1}, \frac{j}{n+1}\right] = h \cdot [i, j]$, $n \in \mathbb{N}$, $i, j = 0, \dots, n+1$, $h = \frac{1}{n+1}$
Bild

Ist x_k ein Randpunkt, so gelte $u_k = r(x_k)$ ($u_k \approx u(x_k)$). Für die zweite Ableitung verwenden wir die Formeln aus dem eindimensionalen.

$$\begin{aligned} \partial_1^2 u(x_k) + \partial_2^2 u(x_k) &\approx \frac{1}{h^2}(u_{k+1} - 2u_k + u_{k-1}) + \frac{1}{h^2}((u_{k+(n+2)} - 2u_k + u_{k-(n+2)})) \\ &= \frac{1}{h^2}(u_{k+(n+2)} + u_{k+1} - 4u_k + u_{k-1} - u_{k-(n+2)}) \end{aligned}$$

für x_k im Inneren von Ω . Wir erhalten das Gleichungssystem:

$$\begin{aligned} -u_{k+(n+2)} - u_{k+1} + 4u_k - u_{k-1} - u_{k-(n+2)} &= h^2 f(x_k) \text{ für } x_k \in \Omega \\ u_k &= r(x_k) \text{ für } x_k \in \partial\Omega \end{aligned}$$

Formuliere dies als ein Gleichungssystem in Matrixschreibweise für den Vektor $[u_1, \dots, u_N]$
Differenzenstern (hier „5-Punkte-Stern“)

Bild

Das Gleichungssystem in lexikographischer Anordnung

$$\begin{bmatrix} 1 & & & & & & \\ & \ddots & & & & & \\ & & \ddots & & & & \\ & & & \ddots & & & \\ & & & & 1 & & \\ & & -1 & & -1 & 4 & 1 & -1 \end{bmatrix} \cdot \begin{bmatrix} u_1 \\ \vdots \\ \vdots \\ u_{n+3} \\ u_{n+4} \end{bmatrix} = \begin{bmatrix} r_1 \\ \vdots \\ \vdots \\ r_{n+3} \\ h^2 f_{n+4} \end{bmatrix}$$

Reduktion der Randwerte: Ziel: Eliminiere die trivialen Gleichungen

Beispiel: 1d

1. $u_0 = \alpha$
2. $-u_2 + 2u_1 - u_0 = h^2 f_1$

3. :

Jetzt: Eliminiere 1. und 2. wie folgt

$$u_2 + 2u_1 = \underbrace{h^2 f_1 + \alpha}_{\text{bekannt}}$$

Nun: $Au = b$ mit $A \in \mathbb{R}^{n,n}$, $b \in \mathbb{R}^n$, $u \in \mathbb{R}^n$ und A hat die Gestalt

$$A = \begin{bmatrix} 2 & -1 & & & \\ -1 & 2 & -1 & & \\ & \ddots & \ddots & \ddots & \\ & & \ddots & \ddots & -1 \\ & & & -1 & 2 \end{bmatrix} =: \text{tridiag}(-1, 2, -1)$$

Analog reduzieren wir die Randwerte im 2d-System. Man erhält dann eine Blocktridia-

gonalmatrix $\begin{bmatrix} A & B & & \\ B & A & B & \\ & \ddots & \ddots & \ddots \\ & & \ddots & \ddots & B \\ & & & B & A \end{bmatrix} \in \mathbb{R}^{n^2, n^2}$ mit $A, B \in \mathbb{R}^{n,n}$

1.3.5 Konvergenzbetrachtung

ÜA: Diese diskrete 2. Ableitung approximiert die exakte 2. Ableitung mit $\mathcal{O}(h^2)$ falls $u \in C^4(\mathbb{R})$. Man kann dann zeigen, dass

$$\max_k |u''(x_k) - u''_k| \leq Ch^2$$

mit einem von u unabhängigen C .

2 Rundungsfehler und numerische Stabilität

2.1 Grenzen der Genauigkeit

Wir haben uns in Kapitel I darauf verlassen, dass $\lim_{h \rightarrow 0} \frac{u(x+h) - u(x)}{h} = u'(x)$, falls $u \in C^1(\mathbb{R})$ auch auf dem Computer gilt. Wir rechnen das numerisch nach. Dazu definieren wir

$$\begin{aligned} g^{(1)}(x, h) &= \frac{1}{h} (u(x+h) - u(x)) \\ g^{(2)}(x, h) &= \frac{1}{2h} (u(x+h) - u(x-h)) \end{aligned}$$

Vorwärtsdifferentienquotient bzw Mitteldifferenzenquotient Sei x fest gewählt.

Wir stellen den Wert

$$E^{(i)}(h) := |g^{(i)}(x, h) - u'(x)|$$

als Funktion von h dar. Wir erwarten $E^{(i)}(h) = \mathcal{O}(h^\kappa)$ für ein $\kappa \in \mathbb{N}$. Daraus folgt: $\log(E^{(i)}(h)) = C + \kappa \cdot \log(h)$. Im doppelt logarithmischen Plot erwarten wir eine Gerade mit Steigung κ

2.2 Zahldarstellung

2.2.1 Zahlssysteme

Dezimalbasis: Jede reelle Zahl x hat zur Basis 10 die Darstellung

$$x = x_M \cdot 10^M + x_{M-1} \cdot 10^{M-1} + \dots + x_0 \cdot 10^0 + x_{-1} \cdot 10^{-1} + \dots$$

mit Faktoren $x_l \in \{0, \dots, 9\}$. Die Darstellung ist nicht notwendig endlich und nicht eindeutig ($0.\bar{9} = 1.0$).

Dualbasis: Verwende 2 statt 10.

$$x = x_M \cdot 2^M + x_{M-1} \cdot 2^{M-1} + \dots + x_0 \cdot 2^0 + x_{-1} \cdot 2^{-1} + \dots$$

Hexadezimal: zur Basis 16, Speicheradressen: $0, \dots, 9, A, \dots, F$

Beispiele:

$$\begin{aligned} 9_{10} &= 8 + 1 = 2^3 + 2^0 = 1001_2 \\ 9.25_{10} &= 1001.01_2 \\ 0.000\overline{1100}_2 &= \sum_{k=1}^{\infty} 2^{-4k} + 2^{-4k-1} = \sum_{k=1}^{\infty} \left(\frac{1}{16}\right)^k + \frac{1}{2} \left(\frac{1}{16}\right)^k \\ &= \frac{3}{2} \left(\frac{1}{1 - \frac{1}{16}} - 1 \right) = \frac{1}{10} \end{aligned}$$

Bemerkung: $\frac{1}{10}$ hat im Dezimalsystem eine endliche, im Dualsystem eine unendliche Darstellung. Jedoch gilt: $\frac{1}{2} = 5 \cdot 10^{-1}$. Daher hat jede endliche Darstellung im Dualsystem eine endliche im Dezimalsystem.

2.2.2 Maschinenzahlen

Ein Rechner kennt nur endlich viele Zahlen. Man definiert eine Abbildung $\text{rd} : \mathbb{R} \rightarrow \mathbb{F}$ (Menge der Maschinenzahlen) durch *Bestapproximation* oder *Abschneiden*.. Im Dezimalsystem lautet die allgemeine Darstellung einer Maschinenzahl $y \in \mathbb{F}(10, L, E_{\min}, E_{\max})$:

$$y = \pm 0, \underbrace{* \dots *}_{\substack{\text{Mantisse,} \\ L \text{ Ziffern}}} \cdot 10^e$$

mit $e \in \{E_{min}, \dots, E_{max}\} \subset \mathbb{Z}$

Die *Maschinengenauigkeit* ε hat nach Definition die Eigenschaft

$$\varepsilon := \inf\{x > 0 : \text{rd}(1 - x) < 1\}$$

und es gilt: $\left| \frac{x - \text{rd}(x)}{x} \right| \leq \varepsilon$ für $x \in [\min \mathbb{F}, \max \mathbb{F}] \setminus \{0\}$

In C oder FORTRAN

float, real*4 $\varepsilon \approx 10^{-8}$
double, real*8 $\varepsilon \approx 10^{-16}$

Den arithmetischen Operationen $+$, $-$, \cdot , $/$ entsprechen Operationen in der Rechnerarithmetik $\tilde{+}$, $\tilde{-}$, $\tilde{\cdot}$, $\tilde{/}$ und es gilt für $\circ \in \{+, -, \cdot, /\}$

$$\text{rd}(x) \tilde{\circ} \text{rd}(y) = x \circ y(1 + \varepsilon_{xy}) \text{ mit } |\varepsilon_{xy}| \leq \varepsilon$$

Leider gelten für das Zahlensystem \mathbb{F} viele der üblichen Regeln (z.B. Assoziativgesetz) (\rightarrow ÜA)

2.2.3 Rundungsfehleranalyse

Differenzenquotient: Wir halten in 1.1 die Differenzenquotienten $g^{(1)}(x, h)$ und $g^{(2)}(x, h)$ definiert.

$$\begin{aligned} g^{(1)}(x, h) &= \frac{1}{h} (f(x+h)(1 + \varepsilon_1) - f(x)(1 + \varepsilon_2)) \cdot (1 + \varepsilon_0) \\ &= \left(\frac{f(x+h) - f(x)}{h} + \frac{\varepsilon_1}{h} f(x+h) - \frac{\varepsilon_2}{h} f(x) \right) (1 + \varepsilon_0) \end{aligned}$$

Dann ist $|g^{(1)}(x, h) - f'(x)| = \mathcal{O}(h) + \mathcal{O}\left(\frac{\varepsilon}{h}\right)$

Die Abschätzung ist optimal, wenn beide Summanden vergleichbar sind: $h \approx \frac{\varepsilon}{h} \Rightarrow h^2 \approx \varepsilon \Rightarrow h \approx \sqrt{\varepsilon}$. Der optimale Fehler ist dann $\mathcal{O}(\sqrt{\varepsilon})$. Analog für $g^{(2)} : h \approx \sqrt[3]{\varepsilon}$ und den Fehler $\sqrt[3]{\varepsilon^2}$

Skalarprodukt: Sei $S \equiv S(y) := [1, \dots, 1] \cdot y = \sum_{k=1}^n y_k$ für $y \in \mathbb{R}^n$.

Nun wollen wir $y \in \mathbb{F}^n$ annehmen und die Summe \tilde{S} in Rechnerarithmetik bestimmen.

Algorithmus

```

 $\tilde{S} := y_1$ 
for  $k = 2 : n$ 
 $\tilde{S} = \tilde{S} \tilde{+} y_k$ 
end

```

Beispiel: $n = 3$

$$\tilde{S} = ((y_1 + y_2)(1 + \varepsilon) + y_3)(1 + \varepsilon_2) = (y_1 + y_2)(1 + \varepsilon_1)(1 + \varepsilon_2) + y_3(1 + \varepsilon_2)$$

Induktion:

$$\tilde{S} = (y_1 + y_2) \prod_{i=1}^{n-1} (1 + \varepsilon_i) + \sum_{k=3}^n y_k \prod_{i=k-1}^{n-1} (1 + \varepsilon_i)$$

mit $|\varepsilon_i| \leq \varepsilon$ für $i = 1, \dots, n$

Lemma 1. Seien $\varepsilon_i, \varepsilon$ wie oben, $\sigma_i \in \{\pm 1\}$ ($i = 1, \dots, n$)

Ist $n\varepsilon < 1$, so gilt

$$\prod_{i=1}^n (1 + \varepsilon_i)^{\sigma_i} = 1 + \vartheta_n$$

mit $\vartheta_n \in \mathbb{R}$, $|\vartheta_n| \leq \frac{n\varepsilon}{1 - n\varepsilon} =: \gamma_n$

Bemerkung: $n \approx 10^6$ in einfacher und $n \approx 10^{15}$ in doppelter Genauigkeit.

Beweis. Mit Induktion ÜA □

Theorem 1. Für die Summation von n Zahlen in Rechnerarithmetik gilt die Abschätzung

$$|\tilde{S} - S| \leq |y_1 + y_2| \gamma_{n-1} + \sum_{k=2}^n |y_k| \gamma_{n-k+1}$$

sowie

$$\frac{|\tilde{S} - S|}{|S|} \leq \gamma_{n-1} \left| \frac{\sum_{k=1}^n |y_k|}{\sum_{k=1}^n y_k} \right| = \gamma_{n-1} \frac{S(|y|)}{|S(y)|}$$

wobei $|y|$ hier komponentenweise zu verstehen ist.

Beachte: $\gamma_{n-1} \approx n\varepsilon$, falls $n\varepsilon \ll 1$

Beweis. Direkt aus der Darstellung von \tilde{S} und dem Lemma folgt die erste Abschätzung.

Die γ_k wachsen monoton mit k , d.h. wir können $|\tilde{S} - S| \leq \gamma_{n-1}(|y_1| + |y_2|) + \gamma_{n-1} \sum_{k=3}^n |y_k|$ abschätzen. □

Bemerkungen

- $\gamma_{n-1} \approx n\varepsilon$
- Erst die betraglich kleinen Zahlen addieren
- Schlecht ist der Fall $|S(y)| \ll S(|y|)$, Dies gilt z.B. für Differenzenquotienten

2.3 Konditionen von Abbildungen

Erinnerung: Vektornorm, zugeordnete Operatornorm, verträgliche Operatornorm \rightarrow Ergänzungsblatt

Seien gegeben: Normierte lineare Vektorräume X, Y sowie $f : X \rightarrow Y$ stetige Abbildung.

2.3.1 Norm- und komponentenweise Kondition

Definition. Normweise absolute Kondition ist die kleinste Zahl κ_{abs} mit

$$\|f(\tilde{x}) - f(x)\|_Y \leq \kappa_{\text{abs}} \|\tilde{x} - x\|_X + o(\|\tilde{x} - x\|_X) \quad (\tilde{x} \rightarrow x)$$

Normweise relative Kondition ist die kleinste Zahl κ_{rel} mit

$$\frac{\|f(\tilde{x}) - f(x)\|_Y}{\|f(x)\|_Y} \leq \kappa_{\text{rel}} \frac{\|\tilde{x} - x\|_X}{\|x\|_X} + o(\|\tilde{x} - x\|_X) \quad (\tilde{x} \rightarrow x)$$

für $x \neq 0, f(x) \neq 0$

Komponentenweise relative Kondition ist die kleinste Zahl κ_{rel} mit

$$\left\| \frac{f(\tilde{x}) - f(x)}{f(x)} \right\|_Y \leq \kappa_{\text{rel}} \left\| \frac{\tilde{x} - x}{x} \right\|_X + o(\|\tilde{x} - x\|_X) \quad (\tilde{x} \rightarrow x)$$

Je nach Größenordnung von $\kappa \in \{\kappa_{\text{rel}}, \kappa_{\text{abs}}\}$ nennt man eine Abbildung von f in x gut ($\kappa \approx 1$) oder schlecht ($\kappa \gg 1$) konditioniert

Ist f differenzierbare Abbildung, so setzen wir

$$\begin{aligned} \kappa_{\text{abs}} &:= \|f'(x)\| \\ \kappa_{\text{rel}} &:= \frac{\|f'(x)\| \cdot \|x\|_X}{\|f(x)\|_Y} \quad (\text{normweise}) \\ \kappa_{\text{rel}} &:= \left\| \frac{|f'(x)| \cdot |x|}{|f(x)|} \right\|_Y \quad (\text{komponentenweise}) \end{aligned}$$

Letzteres mit komponentenweiser Definition von $|\cdot|$ und Division. $\|\cdot\|$ Operatornorm zu $\|\cdot\|_X, \|\cdot\|_Y$

2.3.2 Beispiele

- Addition: $f : \mathbb{R}^2 \rightarrow \mathbb{R}, [x_1, x_2] \mapsto x_1 + x_2, \|x\| := |x_1| + |x_2| =: |x|_1$.
Es gilt: $f'(x) = [1, 1]$. Also folgt:

$$\begin{aligned} \kappa_{\text{abs}} &= \max_y \frac{|[1, 1] \cdot y|}{|y|_1} \leq \frac{|y_1| + |y_2|}{|y|_1} = 1 \\ \kappa_{\text{rel}} &= \frac{1 \cdot |x|_1}{\underbrace{|x_1 + x_2|}_{=f(x)}} = \frac{|x_1| + |x_2|}{|x_1 + x_2|} \quad (\text{normweise und komponentenweise}) \end{aligned}$$

Die Addition zweier Zahlen ist „schlecht konditioniert“ falls $x_1 \approx x_2$ (*Stellenauslöschung*). Sie ist „gut konditioniert“ falls $|x_1| + |x_2| = |x_1 + x_2| \Rightarrow \kappa_{\text{rel}} = 1$.

- Multiplikation zweier Zahlen $f : \mathbb{R}^2 \rightarrow \mathbb{R}$, $[x_1, x_2] \mapsto x_1 \cdot x_2$, $|\cdot|_1$.
Es gilt: $f'(x) = [x_2, x_1]$

$$\begin{aligned}\kappa_{\text{abs}} &= \max_y \frac{|f'(x) \cdot y|}{|y|_1} = \frac{|x_2 y_1 + x_1 y_2|}{|y_1| + |y_2|} \leq \max\{|x_1|, |x_2|\} \\ \kappa_{\text{rel}} &= \frac{\left| \frac{f'(x) \cdot |x|}{|f(x)|} \right|}{\left| \frac{f'(x) \cdot |x|}{|f(x)|} \right|} = \frac{|[x_2, x_1] \cdot [x_1, x_2]|}{|x_1 \cdot x_2|} = \frac{2 \cdot |x_1 x_2|}{|x_1 x_2|} = 2\end{aligned}$$

- Lösen eines linearen Gleichungssystems:

Gegeben: A invertierbar in $\mathbb{R}^{n,n}$, $b \in \mathbb{R}^n$

Finde $u \in \mathbb{R}^n$ sodass gilt $Au = b$

1. Störung der rechten Seite b : $f(b) := u = A^{-1}b$

Wir betrachten die normweise Kondition: $f'(b) = A^{-1}$

$$\Rightarrow \kappa_{\text{abs}} = \|A^{-1}\|$$

$\|\cdot\|$ gewählte Vektornorm, $\|\cdot\|$ zugeordnete Operatornorm

$$\begin{aligned}\kappa_{\text{rel}} &= \frac{\|A^{-1}\| \cdot \|b\|}{\|A^{-1}b\|} = \frac{\|A^{-1}\| \cdot \|AA^{-1}b\|}{\|A^{-1}b\|} \leq \frac{\|A^{-1}\| \cdot \|A\| \cdot \|A^{-1}b\|}{\|A^{-1}b\|} \\ &= \|A^{-1}\| \cdot \|A\| =: \text{cond}_{\|\cdot\|}(A) \text{ (Kondition von } A)\end{aligned}$$

2. Einfluss der Störung von A :

Betrachte nun u als Funktion von A : $f : \mathbb{R}^{n,n} \rightarrow \mathbb{R}^n$, $f(A) = u = A^{-1}b$

Es gilt:

$$f'(A)E = -A^{-1}EA^{-1}b = -A^{-1}Eu$$

Daraus folgt:

$$\begin{aligned}\|f'(A)\| &= \sup_E \frac{\|f'(A)E\|}{\|E\|} = \sup_E \frac{\|A^{-1}Eu\|}{\|E\|} \\ &\leq \sup_E \frac{\|A^{-1}\| \cdot \|E\| \cdot \|u\|}{\|E\|} = \|A^{-1}\| \cdot \|u\| \\ \Rightarrow \kappa_{\text{rel}} &\leq \frac{\|A^{-1}\| \cdot \|u\| \cdot \|A\|}{\|u\|} = \text{cond}_{\|\cdot\|}(A)\end{aligned}$$

2.4 Stabilität numerischer Algorithmen

Die Kondition von f in x beschreibt den unvermeidlichen Fehler der Rechenvorschrift $x \mapsto f(x)$.

Es sei $\tilde{f}(x)$ die Vorschrift zur Berechnung von $f(x)$ wir rechnen damit, dass selbst bei exakter Arithmetik auf \mathbb{F} der relative Fehler $\kappa_f(x)\varepsilon$ auftritt.

2.4.1 Vorwärtsanalyse

Definition. Der Stabilitätsindikator des Algorithmus $\tilde{f}(x)$ zur Berechnung von $f(x)$ ist die kleinste Zahl σ , so dass gilt

$$\frac{\|\tilde{f}(\tilde{x})\|_Y}{\|f(\tilde{x})\|_Y} \leq \sigma \underbrace{\kappa_f(\tilde{x})}_{\kappa_{\text{rel normw.}}} \varepsilon + o(\varepsilon) \quad (\varepsilon \rightarrow 0)$$

für alle \tilde{x} mit $\|\tilde{x} - x\|_X \leq \varepsilon \cdot \|x\|_X$

Der Algorithmus \tilde{f} ist stabil im Sinne der Vorwärtsanalyse, falls σ kleiner gleich der Anzahl der elementaren Rechenoperationen ist.

Beispiel: Die Summation:

$$\begin{aligned} \tilde{S}_1 &:= y_1 \\ \text{for } i = 2 : n \quad \tilde{S}_i &= \tilde{S}_{i-1} \oplus y \end{aligned}$$

Es gilt:

$$\frac{|\tilde{S}(y) - S(y)|}{|S(y)|} \leq \gamma_{n-1} \varepsilon \cdot \frac{S(|y|)}{|S(y)|} = (n-1)\varepsilon \kappa_S + o(\varepsilon), \text{ falls } n\varepsilon \ll 1$$

Also $\sigma < n-1$, d.h. die Summation ist vorwärtsstabil.

2.4.2 Rückwärtsanalyse

Definition. Der Stabilitätsindikator der Rückwärtsanalyse des Algorithmus $x \mapsto \tilde{f}(x)$, $x \in E$ ist die kleinstmögliche Zahl ϱ , so dass für alle $\tilde{x} \in E$ mit $\|\tilde{x} - x\|_X \leq \varepsilon \|x\|_X$ ein $\hat{x} \in E$ existiert mit $\tilde{f}(\tilde{x}) = f(\hat{x})$, so dass

$$\frac{\|\hat{x} - \tilde{x}\|_X}{\|\tilde{x}\|_X} \leq \varrho \varepsilon + o(\varepsilon) \quad (\varepsilon \rightarrow 0)$$

Der Algorithmus \tilde{f} heißt stabil im Sinne der Rückwärtsanalyse, falls ϱ kleiner gleich der Anzahl der elementaren Rechenoperationen

Lemma 2. (Rückwärtsstabil \Rightarrow Vorwärtsstabil)

$$\sigma \leq \varrho$$

Beweis. Sei $\tilde{x} \in E$ mit $\|x - \tilde{x}\|_X \leq \varepsilon \cdot \|x\|_X$. Dann gilt

$$\begin{aligned} \frac{\|\tilde{f}(\tilde{x}) - f(\tilde{x})\|_Y}{\|f(\tilde{x})\|_Y} &\stackrel{\text{Vor.}}{=} \frac{\|f(\hat{x}) - f(\tilde{x})\|_Y}{\|f(\tilde{x})\|_Y} \\ &\stackrel{\text{Def } \kappa_f}{\leq} \kappa_f(\hat{x}) \frac{\|\hat{x} - \tilde{x}\|_X}{\|\tilde{x}\|_X} + o(\varepsilon) \\ &\stackrel{\text{Vor.}}{\leq} \varrho \varepsilon \cdot \kappa_f(\tilde{x}) + o(\varepsilon) \end{aligned}$$

$\Rightarrow \sigma \leq \varrho$ nach Def. von σ

□

Beispiel: Summation: Wir hatten für $y \in \mathbb{F}^n$

$$\tilde{S}(y) = (y_1 + y_2)(1 + \vartheta_{n-1}) + \sum_{k=3}^n y_k(1 + \vartheta_{n-k+1})$$

Definiere nun

$$\begin{aligned}\hat{y}_1 &:= y_1(1 + \vartheta_{n-1}) \\ \hat{y}_2 &:= y_2(1 + \vartheta_{n-1}) \\ \hat{y}_k &:= y_k(1 + \vartheta_{n-k+1}) \text{ für } k \geq 3\end{aligned}$$

$$\Rightarrow S(\hat{y}) = \tilde{S}(y)$$

Es gilt die Abschätzung

$$|\hat{y} - y|_1 \leq (|y_1| + |y_2|)|\vartheta_{n-1}| + \sum_{k=3}^n |y_k| \cdot |\vartheta_{n-k+1}| \leq \gamma_{n-1}|y|_1$$

Es folgt also

$$\varrho = \gamma_{n-1} = \varrho$$

3 Lineare Gleichungssysteme

3.1 Direkte Verfahren: Gauß-Elimination

3.1.1 Das Gaußsche Eliminationsverfahren

2×2 **Systeme:** Betrachte das Gleichungssystem

$$\begin{aligned}A_{11}u_1 + A_{12}u_2 &= b_1 \\ A_{21}u_1 + A_{22}u_2 &= b_2\end{aligned}$$

wobei die A_{ij} und die b_i gegeben (sodass $A_{11} \neq 0$) und die u_i gesucht sind.

$$\begin{aligned}A_{11}u_1 + A_{12}u_2 &= b_1 & | \cdot L_{21} &= \frac{A_{21}}{A_{11}} \\ A_{21}u_1 + A_{22}u_2 &= b_2 & | - L_{21} \cdot 1. \text{ Zeile}\end{aligned}$$

Äquivalentes System:

$$\begin{aligned}A_{11}u_1 + A_{12}u_2 &= b_1 \\ 0 \cdot u_1 + (A_{22} - L_{21}A_{12})u_2 &= b_2 - L_{21}b_1\end{aligned}$$

$$\tilde{A}_{22} := A_{22} - L_{21}A_{12}, \quad \tilde{b}_2 = b_2 - L_{21}b_1$$

2. Gleichung ist $\tilde{A}_{22}u_2 = \tilde{b}_2$

$$\tilde{A}_{22} \neq 0 \Rightarrow u_2 = \tilde{b}_2 / \tilde{A}_{22} \Rightarrow u_1 = (b_1 - A_{12} \cdot \frac{\tilde{b}_2}{\tilde{A}_{22}}) / A_{11}$$

$n \times n$ Systeme

$$\begin{array}{ccccccc} A_{11}u_1 & + & A_{12}u_2 & + & \dots & + & A_{1n}u_n & = & b_1 \\ A_{21}u_1 & + & A_{22}u_2 & + & \dots & + & A_{2n}u_n & = & b_2 & | - L_{21} \cdot 1. \text{ Zeile} \\ \vdots & & \vdots & & & & \vdots & & \vdots & \\ A_{n1}u_1 & + & A_{n2}u_2 & + & \dots & + & A_{nn}u_n & = & b_n & | - L_{n1} \cdot 1. \text{ Zeile} \end{array}$$

wobei $L_{j1} := \frac{A_{j1}}{A_{11}}$, falls $A_{11} \neq 0$

Mit $\tilde{A}_{22} := A_{22} - L_{21} \cdot A_{12}, \dots, \tilde{A}_{2n} := A_{2n} - L_{21} \cdot A_{1n}, \tilde{A}_{23} := \dots$, allgemein

$$\tilde{A}_{ij} := A_{ij} - L_{i1} \cdot A_{1j} \quad \text{und} \quad \tilde{b}_i := b_i - L_{i1} \cdot b_1$$

ergibt sich das äquivalente System:

$$\begin{bmatrix} A_{11} & A_{12} & \dots & A_{1n} \\ 0 & \tilde{A}_{22} & \dots & \tilde{A}_{2n} \\ \vdots & \vdots & & \vdots \\ 0 & \tilde{A}_{n2} & \dots & \tilde{A}_{nn} \end{bmatrix} \cdot \begin{bmatrix} u_1 \\ u_2 \\ \vdots \\ u_n \end{bmatrix} = \begin{bmatrix} b_1 \\ \tilde{b}_2 \\ \vdots \\ \tilde{b}_n \end{bmatrix}$$

$\tilde{A}_{22} \neq 0$ erlaubt den Algorithmus auf die $n-1 \times n-1$ Untermatrix anzuwenden. Nach $n-1$ Schritten erhalten wir, falls $\tilde{A}_{kk}^{(k)} \neq 0$ gilt

$$\begin{bmatrix} * & \dots & * \\ & \ddots & \vdots \\ 0 & & * \end{bmatrix} \cdot \begin{bmatrix} u_1 \\ \vdots \\ u_n \end{bmatrix} = \begin{bmatrix} * \\ \vdots \\ * \end{bmatrix} \quad (\text{Rechts - obere Dreiecksmatrix})$$

Dieses lässt sich einfach auflösen:

n -te Gleichung: $\tilde{A}_{nn}^{(n)} u_n = \tilde{b}_n^{(n)}$

$$\Rightarrow u_n = \tilde{b}_n^{(n)} / \tilde{A}_{nn}^{(n)} \text{ falls } \tilde{A}_{nn}^{(n)} \neq 0$$

$$(n-1)\text{-te Gleichung: } \underbrace{\tilde{A}_{n-1,n-1}^{(n)}}_{\substack{= \tilde{A}_{n-1,n-1}^{(n-1)} \\ \neq 0 \text{ n. Vor}}} u_{n-1} + \tilde{A}_{n-1,n}^{(n)} \underbrace{u_n}_{\text{bek.}} = \tilde{b}_{n-1}^{(n)}$$

$$\Rightarrow u_{n-1} = \dots$$

usw...

3.1.2 Die LR-Zerlegung

Ziel: formalisiere diesen Algorithmus.

Wir wollen die Elimination in der Form $A \mapsto L \cdot A$ schreiben. Suche L . Sei L_1 die Matrix, die die erste Spalte von A (ab 2. Element) zu 0 mache:

$$(L_1 A)_{ij} = \sum_{k=1}^n L_{1;ik} A_{kj} \stackrel{!}{=} A_{ij} - L_{i1} \cdot A_{1j}$$

$k = i : L_{1;ii} = 1$

$k = 1 : L_{1;i1} = -L_{i1}$ und $L_{1;ik} = 0$ sonst.

L_1 hat also die Gestalt

$$L_1 = \begin{bmatrix} 1 & & & & \\ -L_{21} & \ddots & & & \\ \vdots & & \ddots & & \\ -L_{n1} & & & \ddots & \\ & & & & 1 \end{bmatrix}$$

Genauso folgt:

$$L_2 = \begin{bmatrix} 1 & & & & \\ & 1 & & & \\ & -L_{32} & \ddots & & \\ & \vdots & & \ddots & \\ & -L_{n2} & & & 1 \end{bmatrix}$$

Nach Durchführung von $n - 1$ Schritten erhalten wir die rechts-obere Dreiecksmatrix

$$R = L_{n-1} \cdot \dots \cdot L_2 \cdot L_1 \cdot A$$

sowie

$$\tilde{b} = L_{n-1} \cdot \dots \cdot L_2 \cdot L_1 \cdot b$$

Schreibweise: Zu $a, b \in \mathbb{R}^n$ sei

$$a \otimes b := ab^\top \in \mathbb{R}^{n,n} \text{ (} a \text{ tensor } b \text{)}$$

Achtung: $a \otimes b \stackrel{\text{i.A.}}{\neq} b \otimes a$

Es gilt aber:

$$(a \otimes b) \otimes c = \left(\sum_j a_i \cdot b_j \cdot c_j \right)_i = a(b \cdot c)$$

D.h. $\dim(\text{Bild}(a \otimes b)) = 1$.

Wir definieren

$$\begin{aligned} \vec{i}_k &:= k\text{-ter euklidischer Einheitsvektor} \\ \vec{L}_k &:= [0, \dots, \underbrace{0}_k, L_{k+1,k}, \dots, L_{n,k}] \end{aligned}$$

Damit gilt: $L_k = Id_n - \vec{L}_k \otimes \vec{i}_k =$

$$\begin{bmatrix} 1 & & & & \\ & \ddots & & & \\ & & 1 & & \\ & & -L_{k+1,k} & \ddots & \\ & & \vdots & & \ddots \\ & -L_{n,k} & & & 1 \end{bmatrix}$$

Nun ist

$$\begin{aligned} (Id + \vec{L}_k \otimes \vec{i}_k)L_k &= (Id + \vec{L}_k \otimes \vec{i}_k)(Id - \vec{L}_k \otimes \vec{i}_k) \\ &= Id + \underbrace{\vec{L}_k \otimes \vec{i}_k - \vec{L}_k \otimes \vec{i}_k}_0 - \underbrace{\vec{L}_k \otimes \vec{i}_k \cdot \vec{L}_k \otimes \vec{i}_k}_{=\vec{L}_k(\vec{i}_k \cdot \vec{L}_k)\otimes \vec{i}_k} \\ &= Id \end{aligned}$$

$$\Rightarrow L_k^{-1} = Id + \vec{L}_k \otimes \vec{i}_k$$

Damit erhalten wir ($A = L_1^{-1} \cdot \dots \cdot L_{n-1}^{-1}R$):

$$\begin{aligned} L_1^{-1}L_2^{-1} &= (Id + \vec{L}_1 \otimes \vec{i}_1)(Id + \vec{L}_2 \otimes \vec{i}_2) \\ &= Id + \vec{L}_1 \otimes \vec{i}_1 + \vec{L}_2 \otimes \vec{i}_2 + \underbrace{\vec{L}_1 \otimes \vec{i}_1 \cdot \vec{L}_2 \otimes \vec{i}_2}_{=\vec{i}_1 \cdot \vec{L}_2 \cdot \vec{L}_1 \otimes \vec{i}_2} \\ &= Id + \vec{L}_1 \otimes \vec{i}_1 + \vec{L}_2 \otimes \vec{i}_2 \end{aligned}$$

Mit Induktion folgt:

$$L := L_1^{-1} \cdot \dots \cdot L_{n-1}^{-1} = Id + \sum_{k=1}^{n-1} \vec{L}_k \otimes \vec{i}_k = \begin{bmatrix} 1 & & & \\ L_{21} & \ddots & & \\ \vdots & \ddots & \ddots & \\ L_{n1} & \dots & L_{nn-1} & 1 \end{bmatrix}$$

Satz 1. Ist die Gaußsche Elimination für ein $A \in \mathbb{R}^{n,n}$ durchführbar (d.h. gilt $\tilde{A}_{kk}^{(k)} \neq 0$ für $k = 1, \dots, n-1$), so besitzt A eine LR-Zerlegung, d.h.

$$A = LR$$

mit L links-untere Dreiecksmatrix mit Diagonale 1 und R eine rechts-obere Dreiecksmatrix.

Diese Zerlegung ist für invertierbare Matrizen eindeutig.

Beweis. Beh.: A invertierbar $\Leftrightarrow R$ invertierbar
(unter der Voraussetzung der Existenz von L und R)

$$\begin{aligned} A &= LR \\ \Leftrightarrow A^{-1} &= R^{-1}L^{-1} \end{aligned} \quad (1)$$

$$\Leftrightarrow R^{-1} = A^{-1}L \quad (2)$$

L^{-1} existiert immer. Weiter gilt:

$$(1) : A^{-1} \text{ ex.} \Rightarrow R^{-1} \text{ ex.}$$

$$(2) : R^{-1} \text{ ex.} \Rightarrow A^{-1} \text{ ex.}$$

Wegen der Behauptung folgt im Fall, dass A invertierbar ist die Invertierbarkeit von R .
Eindeutigkeit: Es sei $A = LR = \hat{L}\hat{R}$

$$\Rightarrow \underbrace{R\hat{R}^{-1}}_{\text{r.o. } \Delta\text{matrix}} = \underbrace{L^{-1}\hat{L}}_{\text{l.u. } \Delta\text{matrix}}$$

Beide Seiten sind also gleich der Identität, also folgt $R = \hat{R}$ und $L = \hat{L}$ □

3.1.3 Pivotisierung

Tritt der Fall $\tilde{A}_{kk}^{(k)} = 0$ auf, so wollen wir den Algorithmus modifizieren:

Skizze

Finde nun $l \in \{k+1, \dots, n\}$, so dass

$$|\tilde{A}_{lk}^{(k)}| \geq \max\{|\tilde{A}_{jk}^{(k)}| : j \in \{k+1, \dots, n\}\}$$

Ist $|\tilde{A}_{lk}^{(k)}| = 0$, so hat die Gesamtmatrix keinen maximalen Rang. Ist A invertierbar, so kann dieser Fall nicht auftreten.

Wir vertauschen nun die k -te mit der l -ten Zeile und setzen das Verfahren fort.

Die mathematische Beschreibung dieser Vertauschung ist die Anwendung einer Permutationsmatrix P .

z.B. hier:

$$P = \begin{bmatrix} 1 & & & & & & & & & \\ & \ddots & & & & & & & & \\ & & 1 & & & & & & & \\ & & & 0 & & & 1 & & & \\ & & & & 1 & & & & & \\ & & & & & \ddots & & & & \\ & & & & & & 1 & & & \\ & & & 1 & & & & 0 & & \\ & & & & & & & & 1 & \\ & & & & & & & & & \ddots \\ & & & & & & & & & & 1 \end{bmatrix} \begin{array}{l} \leftarrow k \\ \\ \leftarrow l \end{array} \quad \text{für } k \neq l$$

bzw. $P = Id$ für $k = l$.

Die Elimination liefert also $R = L_{n-1}P_{n-1} \cdots L_1P_1A$. Dabei suchen wir in jedem Schritt das maximale Element. Man kann zeigen, dass man dies in der Form LP schreiben kann, L links-untere Dreiecksmatrix, P Permutationsmatrix

Satz 2. Ist $A \in \mathbb{R}^{n,n}$ invertierbar, dann existiert eine Permutationsmatrix P , so dass PA eine LR -Zerlegung besitzt.

3.1.4 Rechenaufwand

$A \in \mathbb{R}^{n,n}$ vollbesetzt. (Gauß):

Multiplikationen ist $\sum_{k=1}^{n-1} (n-k)^2 = \frac{1}{3}n^3 + \mathcal{O}(n^2) = \mathcal{O}(n^3)$.

Speicher: Wird A nicht mehr benötigt, so kann man die Matrizen L und R auf A abspeichern.

Die explizite Verwendung der Zerlegung LR zur Lösung der gestaffelten Systeme empfiehlt sich, wenn man mehrere GLS der Form $Ax = b$ zu versch. b lösen muss. Einmal $\mathcal{O}(n^3)$ -Aufwand und dann nur noch $\mathcal{O}(n^2)$ für jedes folgende System.

3.1.5 Gauß-Elimination für Bandmatrizen

Schwach besetzte Matrizen und Bandmatrizen

$A \in \mathbb{R}^{n,n}$ heißt *schwachbesetzt* („sparse“), falls gilt:

$$\text{compl}(A) := \#\{[i, j] \in \{1, \dots, n\}^2 : A_{ij} \neq 0\} = \mathcal{O}(n)$$

Wir definieren die *Bandlänge* von A als das maximale $m \in \mathbb{N}$, für das gilt:

$$|i - j| > \left\lfloor \frac{m-1}{2} \right\rfloor \Rightarrow A_{ij} = 0$$

Diskretisierung von $-u''$ in 1d mit „natürlicher Anordnung“ führte auf Tridiag-matrix ($m = 3$). Diskretisierung von $-\Delta u$ in 2d in lexikographischer Anordnung ergab

$$\begin{bmatrix} \ddots & & & & \ddots \\ & \ddots & & & \\ & & \ddots & & \\ & & & \ddots & \\ \ddots & & & & \ddots \end{bmatrix}$$

$\text{compl}(A) = \mathcal{O}(n)$, aber $m = \mathcal{O}(\sqrt{n})$

Die Elimination zerstört die Bandstruktur („fill in“), erhält aber die Bandlänge. Im 1d-Beispiel bleibt es bei einer Tridiagonalmatrix ($\text{compl}(L) = \text{compl}(R) = \mathcal{O}(n)$), aber in 2d folgt $\text{compl}(L) = \text{compl}(R) = \mathcal{O}(n^{\frac{3}{2}})$

Gauß-Elimination für Bandmatrizen

Hier nur Tridiagonalmatrizen:

$$A = \begin{bmatrix} a_1 & a_1^+ & & & \\ a_2^- & a_2 & \ddots & & \\ & \ddots & \ddots & \ddots & \\ & & \ddots & \ddots & a_{n-1}^+ \\ & & & a_n^- & a_n \end{bmatrix}$$

mit $a_1 \neq 0$.

1. Schritt:

$$\tilde{A}^{(2)} = \begin{bmatrix} a_1 & a_1^+ & 0 & \cdots \\ 0 & \underbrace{a_2 - \frac{a_2^-}{a_1} a_1^+}_{\tilde{a}_2^{(2)}} & a_2^+ & \cdots \\ \vdots & & & \end{bmatrix}$$

Induktiv: $L_{i,i-1} = a_i^- / \tilde{a}_{i-1}^{(i-1)}$, $\tilde{a}_i^{(i)} = \tilde{a}_i^{(i-1)} - L_{i,i-1} \cdot a_{i-1}^+$

$$L = \begin{bmatrix} 1 & & & \\ L_{21} & \ddots & & \\ & \ddots & \ddots & \\ & & L_{n,n-1} & 1 \end{bmatrix}, \quad R = \begin{bmatrix} * & a_1^+ & & \\ & \ddots & \ddots & \\ & & \ddots & a_{n-1}^+ \\ & & & * \end{bmatrix}$$

Anzahl Operationen: $\sim 4n$, falls $\tilde{a}_{i-1}^{(i-1)} \neq 0$. Allgemein ist der Aufwand für Bandmatrizen $\mathcal{O}(m^2 n)$ ohne Pivotisierung. Pivotisierung zerstört die Bandstruktur.

3.1.6 Block-Gauß-Elimination

$A \in \mathbb{R}^{n,n}$ mit $n = n_1 + n_2$.

$$A = \begin{bmatrix} A_{11} & A_{12} \\ A_{21} & A_{22} \end{bmatrix}, \quad A_{11} \in \mathbb{R}^{n_1, n_1}; \quad A_{22} \in \mathbb{R}^{n_2, n_2}$$

$u = [u_1, u_2] \in \mathbb{R}^{n_1+n_2}$, $Au = [b_1, b_2] \in \mathbb{R}^{n_1+n_2}$

$$A_{11}u_1 + A_{12}u_2 = b_1$$

$$A_{21}u_1 + A_{22}u_2 = b_2$$

A_{11}^{-1} existiere. Multipliziere 1. Zeile mit $A_{21}A_{11}^{-1} =: L_{21}$ und subtrahiere dies von der 2. Zeile. Wir erhalten das äquivalente System:

$$\begin{aligned} A_{11}u_1 + A_{12}u_2 &= b_1 \\ 0 \cdot u_1 + \underbrace{(A_{22} - A_{21} \cdot A_{11}^{-1} \cdot A_{12})}_{=\tilde{A}_{22}^{(2)}} u_2 &= \underbrace{b_2 - A_{21} \cdot A_{11}^{-1} \cdot b_1}_{=\tilde{b}_2^{(2)}} \end{aligned}$$

Die Block- LR -Zerlegung:

$$A = LR = \begin{bmatrix} Id_{n_2} & 0 \\ L_{21} & Id_{n_2} \end{bmatrix} \cdot \begin{bmatrix} A_{11} & A_{12} \\ 0 & \tilde{A}_{22}^{(2)} \end{bmatrix}$$

3.1.7 Existenz der LR -Zerlegung ohne Pivotisierung

Satz 3. Sei $A \in \mathbb{R}^{n,n}$.

(i) A heißt diagonaldominant, falls

$$\sum_{\substack{j=1 \\ j \neq i}}^n |A_{ij}| < |A_{ii}| \quad i = 1, \dots, n$$

(ii) A heißt symmetrisch und positiv definit, falls

$$A_{ij} = A_{ji} \text{ und } v \cdot Av > 0 \quad \forall v \in \mathbb{R}^n \setminus \{0\}$$

Für Matrizen A mit (i) oder (ii) ist die Elimination ohne Pivotisierung durchführbar.

Beweis. (i) $A_{11} \neq 0$ nach Voraussetzung

z.z.: $\tilde{A} = L_1 A$ ist wieder diagonaldominant.

Elimination: $\tilde{A}_{ij} = A_{ij} - L_{i1} A_{1j}$ für $i, j = 2, \dots, n$

Für $i = 2, \dots, n$ gilt

$$\begin{aligned} \sum_{\substack{j=2 \\ j \neq i}}^n |\tilde{A}_{ij}| &\leq \sum_{\substack{j=2 \\ j \neq i}}^n \{|A_{ij}| + |L_{i1}| \cdot |A_{1j}|\} \\ &= \sum_{\substack{j=1 \\ j \neq i}}^n |A_{ij}| - |A_{i1}| + |L_{i1}| \cdot \left(\sum_{j=2}^n |A_{1j}| - |A_{1i}| \right) \\ &< |A_{ii}| - |A_{i1}| + |L_{i1}| \cdot (|A_{11}| - |A_{1i}|) \\ &= |A_{ii}| - |A_{i1}| + \left| \frac{A_{i1}}{A_{11}} \right| \cdot (|A_{11}| - |A_{1i}|) \\ &= |A_{ii}| - |L_{i1}| \cdot |A_{1i}| \\ &\leq |A_{ii} - L_{i1} A_{1i}| = |\tilde{A}_{ii}| \end{aligned}$$

(ii) Reicht zu zeigen $\tilde{A} := L_1 A$ wieder symmetrisch und positiv definit.

$$A_{11} = \vec{i}_1 \cdot A \vec{i}_1 > 0.$$

\tilde{A} symmetrisch:

$$\tilde{A}_{ij} = A_{ij} - \frac{1}{A_{11}} \underbrace{A_{i1} \cdot A_{1j}}_{=A_{1j} \cdot A_{i1}=A_{j1} \cdot A_{1i}} = A_{ji} - \frac{1}{A_{11}} A_{j1} A_{1i} = \tilde{A}_{ji}$$

\tilde{A} positiv definit:

wir schreiben $A = \begin{bmatrix} A_{11} & a_1^\top \\ a_1 & A' \end{bmatrix}$, $v = \begin{bmatrix} v_1 \\ v' \end{bmatrix} \in \mathbb{R} \times \mathbb{R}^{n-1}$. Sei $v \neq 0$.

$$\begin{aligned}
0 < v \cdot Av &= \begin{bmatrix} v_1 \\ v' \end{bmatrix} \cdot \begin{bmatrix} A_{11} & a_1^\top \\ a_1 & A' \end{bmatrix} \begin{bmatrix} v_1 \\ v' \end{bmatrix} \\
&= \begin{bmatrix} v_1 \\ v' \end{bmatrix} \cdot \begin{bmatrix} A_{11}v_1 + a_1 \cdot v' \\ v_1 a_1 + A'v' \end{bmatrix} \\
&= A_{11}v_1^2 + 2v_1 a_1 \cdot v' + v' \cdot A'v' + \frac{1}{A_{11}}(a_1 \cdot v')^2 - \frac{1}{A_{11}}(a_1 \cdot v')^2 \\
&= A_{11}(v_1 - \frac{1}{A_{11}}a_1 \cdot v')^2 + v' \cdot A'v' - \frac{1}{A_{11}} \underbrace{(a_1 \cdot v')^2}_{\substack{v' \cdot a_1 a_1 \cdot v' \\ = v' \hat{a}_1 \otimes a_1 v'}} \\
&= A_{11}(v_1 - \frac{1}{A_{11}}a_1 \cdot v')^2 + v' \cdot (A' - \frac{1}{A_{11}}a_1 \otimes a_1)v'
\end{aligned}$$

zu $v' \in \mathbb{R}^{n-1}$ beliebig, wähle $v_1 = -\frac{1}{A_{11}}a_1 \cdot v'$ und erhalten

$$0 < v' \cdot \underbrace{(A' - \frac{1}{A_{11}}a_1 \otimes a_1)v'}_{ij\text{-Komponente ist } A_{ij} - \frac{1}{A_{11}}A_{1i} \cdot A_{1j} = \tilde{A}_{ij}}$$

\Rightarrow Für alle $v' \in \mathbb{R}^{n-1} \setminus \{0\}$ gilt $0 < v' \cdot [\tilde{A}_{ij}]_{\substack{i=2,\dots,n \\ j=2,\dots,n}} v' \Rightarrow \text{Beh.}$

□

3.1.8 Numerische Stabilität

Satz 4. Zu $A \in \mathbb{R}^{n,n}$ sei $\tilde{L}\tilde{R}$ die numerisch berechnete LR-Zerlegung. Dann gilt:

$$\frac{\|\tilde{L}\tilde{R} - A\|_\infty}{\|A\|_\infty} \leq 2n^3 f(A)\varepsilon + o(\varepsilon)$$

mit $f(A) = \frac{\max\{|\tilde{a}_{ij}^{(k)}| : k, i, j\}}{\max\{|a_{ij}| : i, j\}}$.

D.h. Stabilität liegt vor, falls $f(A) \in \mathcal{O}(1)$ ist.

z.B. für diagonaldominante Matrizen $f(A) \leq 2$, aber Beispiele mit $f(A) = 2^n$ sind explizit bekannt.

3.1.9 Bemerkungen

- Man kann mit der Gauß-Elimination auch die Inverse einer Matrix berechnen (Gauß-Jordan-Algorithmus)
- Mit der Gauß-Elimination kann man $\det(A)$ berechnen:

$$\det(A) = \det(LR) = \det(L) \cdot \det(R) = \det(R) = \prod_{k=1}^n R_{kk}$$

3.2 Cholesky-Zerlegung

Satz 5. Sei A spd (symmetrisch, positiv definit) aus $\mathbb{R}^{n,n}$.

Dann ex. eine untere Dreiecksmatrix L mit positiven Diagonaleinträgen, so dass

$$A = L \cdot L^T \quad (\text{Cholesky - Zerlegung})$$

Beweis. Mit Induktion über n :

$$n = 1 : 0 < A_{11} = \sqrt{A_{11}} \cdot \sqrt{A_{11}}$$

$$n - 1 \hookrightarrow n : \text{Sei } A = \begin{bmatrix} A' & a_1 \\ a_1^\top & A_{nn} \end{bmatrix} \text{ mit } A' \in \mathbb{R}^{n-1, n-1}, a_1 \in \mathbb{R}^{n-1}$$

Mit $v = [v', 0]$ sieht man: A' spd.

A' hat nach I.V. eine Zerlegung $A' = L'(L')^\top$

Ansatz:

$$\begin{aligned} A = \begin{bmatrix} A' & a_1 \\ a_1^\top & A_{nn} \end{bmatrix} &\stackrel{!}{=} \underbrace{\begin{bmatrix} L' & 0 \\ r^\top & \alpha \end{bmatrix}}_L \cdot \underbrace{\begin{bmatrix} (L')^\top & r \\ 0 & \alpha \end{bmatrix}}_{L^\top} \\ &= \begin{bmatrix} L'(L')^\top & L'r \\ (L'r)^\top & |r|^2 + \alpha^2 \end{bmatrix} \end{aligned}$$

Ziel: Gib r, α an.

$$a_1 = L'r \Rightarrow r = (L')^{-1} \cdot a_1$$

Aber:

$$\begin{aligned} |r|^2 + \alpha^2 &\stackrel{!}{=} A_{nn} > 0 \\ \alpha^2 &= A_{nn} - |r|^2 \stackrel{?}{>} 0 \end{aligned}$$

$$\text{Falls ja: } \alpha := \sqrt{A_{nn} - |r|^2}$$

$$\text{Wähle } \tilde{r} := ((L')^\top)^{-1} r \in \mathbb{R}^{n-1} \text{ und nutze } 0 < \begin{bmatrix} \tilde{r} \\ -1 \end{bmatrix} \cdot A \begin{bmatrix} \tilde{r} \\ -1 \end{bmatrix}$$

□

Algorithmus:

$$\text{Ansatz: } A_{ik} = \sum_{j=1}^k L_{ij} \cdot L_{kj}, \quad i \geq k$$

Spaltenweise auflösen:

$$\begin{aligned} k = 1, \quad i = 1, \dots, n \quad & A_{i1} = L_{i1} \cdot L_{11} \\ & i = 1 \quad A_{11} = L_{11}^2 \Rightarrow L_{11} = A_{11}^{1/2} \\ & i > 1 : \quad L_{i1} = A_{i1}/L_{11} = A_{i1}/\sqrt{A_{11}} \\ k = 2, \quad i = 2, \dots, n \quad & A_{i2} = L_{i1}L_{21} + L_{i2}L_{22} \\ & i = 2 \quad A_{22} = L_{21}^2 + L_{22}^2 \Rightarrow L_{22} = \sqrt{A_{22} - L_{21}^2} \\ & i > 2 : \quad L_{i2} = \dots \end{aligned}$$

Satz 6. Sei $A \in \mathbb{R}^{n,n}$ spd. Algorithmus liefere $A = \tilde{L}(\tilde{L})^\top$.

Dann gilt

$$\frac{\|\tilde{L}(\tilde{L})^\top - A\|_2}{\|A\|_2} \leq 8n(n+1)\varepsilon + o(\varepsilon)$$

Der Algorithmus ist also rückwärtsstabil.

3.3 Iterative Verfahren

3.3.1 Basisiteration

Ziel: Schreibe $Au = b$ als Fixpunktiteration zur Lösung $u = Tu + d$ mit geeignetem $T \in \mathbb{R}^{n,n}$, $d \in \mathbb{R}^n$

Sei $B \in \mathbb{R}^{n,n}$ invertierbar. Dann gilt:

$$Au = b \Leftrightarrow B Au = B b \Leftrightarrow u = u - B Au - B b \Leftrightarrow u = \underbrace{(Id - BA)}_{=:T} u + \underbrace{Bb}_{=:d}$$

B nennt man *Vorkonditionierung* (dient der Beschleunigung des folgenden Algorithmus)

Basisiteration:

$$u_{i+1} = (Id - BA)u_i + Bb \quad (\text{Fixpunktiteration})$$

Falls $u_i \rightarrow u$ ($i \rightarrow \infty$), so löst u die Gleichung $Au = b$

Einschub zu 3.1.: Zerlegungen: Idee um an geeignetes B zu kommen: $A = M - N$ („Hauptteil“ M (invertierbar), „Nebenteil“ N)

$$Au = b \Leftrightarrow Mu - Nu = b \Leftrightarrow Mu = Nu + b \Leftrightarrow u = \underbrace{M^{-1}N}_{=:T} u + \underbrace{M^{-1}b}_{=:d}$$

bzw.. $B = M^{-1}$

In 3.3.2.: $M = D$, $N = D(L + R)$

Bemerkung

Optimal wäre $B = A^{-1}$, aber B sollte nur so komplex wie A sein. Widerspricht sich.

3.3.2 Konvergenz linearer Iterationen

Satz 7. Zu $u_0 \in \mathbb{R}^n$ definieren wir $u_{i+1} := Tu_i + d$. Ist u Lösung zu $u = Tu + d$, dann gilt:

- 1.) Gilt in einer Operatornorm $\|T\| < 1$, so konv die Folge $\{u_i\}_{i \geq 0}$ gegen u und es gilt:

$$\begin{aligned} |u_i - u| &\leq \|T\|^i |u_0 - u| \quad (\text{A priori - Abschätzung}) \\ |u_i - u| &\leq \frac{\|T\|}{1 - \|T\|} |u_i - u_{i-1}| \quad (\text{A posteriori}) \end{aligned}$$

- 2.) Es gilt $u_i \rightarrow u$ ($i \rightarrow \infty$) für alle $u_0 \in \mathbb{C}^n \Leftrightarrow \varrho(T) < 1$

Beweis. 1.) *Banachscher Fixpunktsatz:*

$$\begin{aligned} (u_i - u &= Tu_{i-1} + d - Tu - d = T(u_{i-1} - u) \\ |u_i - u| &\leq \|T\| \cdot |u_{i-1} - u| \leq \dots \leq \|T\|^i |u_0 - u|) \\ q &:= \|T\| < 1 \end{aligned}$$

$$\begin{aligned} |u - u_i| &\leq |u - u_{i+1}| + |u_{i+1} - u_i| \\ &\leq |u - u_{i+1}| + q|u_i - u_{i-1}| \\ &\leq |u - u_{i+2}| + q^2|u_i - u_{i-1}| + q|u_i - u_{i-1}| \leq \dots \leq \\ &\leq q \cdot \sum_{j=0}^{\infty} q^j |u_i - u_{i-1}| \\ &= \frac{q}{1-q} \cdot |u_i - u_{i-1}| \end{aligned}$$

2.) „ \Rightarrow “: Definiere $e_i := u - u_i$, dann gilt:

$$e_{i+1} = Te_i$$

Mit Induktion folgt:

$$e_i = T^i e_0$$

Beachte: Es gilt

$$u_i \rightarrow 0 \ (i \rightarrow \infty) \Leftrightarrow e_i \rightarrow 0 \ (i \rightarrow \infty)$$

Es sei $\lambda \in \mathbb{C}$ ein Eigenwert von T und z ein zugehöriger normierter Eigenvektor (also $|z| = 1$)

$$Tz = \lambda z$$

Wähle $u_0 := u - z$

$$\Rightarrow e_i = T^i e_0 = T^i z = \lambda^i z$$

Nach Vor. gilt $|\lambda|^i = |\lambda|^i |z| = |e_i| \rightarrow 0 \ (i \rightarrow \infty)$, also gilt

$$|\lambda| < 1$$

Da λ beliebiger Eigenwert war folgt $\varrho(T) < 1$.

„ \Leftarrow “: Da $\varrho(T) < 1$, gibt es ein $\varepsilon > 0$ mit $\varrho(T) < 1 - \varepsilon$. Mit ÜA gilt: $\exists \|\cdot\|_\varepsilon$, induzierte Matrixnorm, sodass gilt

$$\|T\|_\varepsilon \leq \underbrace{\varrho(T) + \varepsilon}_{<1}$$

□

Eigenwerte von Tridiagonalmatrizen

Es seien $a, b, c \in \mathbb{R}$ mit $ac > 0$ und $A := \text{tridiag}_N[a, b, c]$ eine reelle Tridiagonalmatrix. Dann sind die Eigenvektoren von A gegeben durch

$$s^k = \left[\left(\frac{a}{c} \right)^{\frac{j-1}{2}} \sin \left(\frac{k\pi j}{N+1} \right) \right]_{j=1, \dots, N}, \quad k = 1, \dots, N$$

Die zugehörigen Eigenwerte sind

$$\lambda_k = b + 2 \operatorname{sgn}(a) \sqrt{ac} \cdot \cos \left(\frac{k\pi}{N+1} \right), \quad k = 1, \dots, N$$

3.3.3 Die „klassischen Iterationsverfahren“

Richardson-Verfahren $B = \omega \cdot Id$ für ein geeignetes $\omega \in \mathbb{R}$. D.h.

$$u_{i+1} = u_i - \omega(Au_i - b) \quad (i > 0)$$

Die Iterationsmatrix ist

$$T_R = Id - \omega A$$

Jakobi-Verfahren (Gesamtschrittverfahren) Zerlegung von $A = D(Id - L - R)$ mit $D := \text{diag}(A)$, $-DL$ der links-untere, $-DR$ der rechts-obere Anteil von A (mit Diagonale 0)

$$\underbrace{\begin{bmatrix} * & * & * \\ * & * & * \\ * & * & * \end{bmatrix}}_A = \underbrace{\begin{bmatrix} * & & \\ & * & \\ & & * \end{bmatrix}}_D + \underbrace{\begin{bmatrix} * & & \\ * & * & \\ * & * & \end{bmatrix}}_{-DL} + \underbrace{\begin{bmatrix} & * & * \\ & & * \\ & & \end{bmatrix}}_{-DR}$$

Iteration:

$$u_{i+1} = u_i - \underbrace{D^{-1}}_{=B}(Au_i - b)$$

Die Iterationsmatrix lautet also:

$$T_J = Id - D^{-1}A$$

In Komponenten:

$$\begin{aligned} u_{i+1,l} &= u_{i,l} - \frac{1}{A_{ll}} \left(\sum_{m=1}^n A_{lm} u_{i,m} - b_l \right) \\ &= -\frac{1}{A_{ll}} \left(\sum_{\substack{m=1 \\ m \neq l}}^n A_{lm} u_{i,m} - b_l \right) \end{aligned}$$

Oft verwendet man noch einen „Dämpfungsfaktor“ $\omega \in \mathbb{R}$

$$u_{i+1} = u_i - \omega D^{-1}(Au_i - b) \quad \text{„Gedämpftes Jakobi-Verfahren“}$$

Hier ist die Iterationsmatrix also

$$T_{J,\omega} = (Id - \omega D^{-1}A)$$

Bemerkung $u = [u_{i,l}]_l, V \subset \mathbb{R}^n$
 $V \leftarrow AU, V \leftarrow V - b, V \leftarrow D^{-1}V$
 $U \leftarrow U - \omega V$

Gauß-Seidel-Verfahren (Einzelschrittverfahren) und das SOR-Verfahren

Einzelschrittverfahren: Idee: nutze schon die neu berechneten Komponenten, um Au zu berechnen.

$$u_{i+1,l} = -\frac{1}{A_{ll}} \left(\sum_{m=1}^{l-1} A_{l,m} u_{i+1,m} + \sum_{m=l+1}^n A_{l,m} u_{i,m} - b_l \right) \quad (*)$$

In Matrix-Schreibweise:

$$\begin{aligned} Du_{i+1} - DLu_{i+1} - DRu_i &= b \\ D(Id - L)u_{i+1} &= b + DRu_i \\ \Rightarrow u_{i+1} &= (Id - L)^{-1}D^{-1}(b + DRu_i) \end{aligned}$$

Die Iterationsmatrix ist also:

$$T_{GS} = (Id - L)^{-1} \cdot R$$

(Formel! Die Implementierung ist die Formel (*))

Hier ist $M = D(Id - L)$ oder $B = (Id - L)^{-1}D^{-1}$

SOR-Verfahren (successive overrelaxation) Mit Dämpfungsparameter $\omega \in \mathbb{R}$

$$u_{i+1} = u_i - \omega D^{-1}(Du_i - DLu_{i+1} - DRu_i - b)$$

bzw.

$$u_{i+1} = u_i - \omega (Id - \omega L)^{-1} D^{-1} (Au_i - b)$$

Die Iterationsmatrix ist

$$T_{\omega}^{\text{SOR}^+} = Id - \omega (Id - \omega L)^{-1} D^{-1} A$$

Implementierung:

Mit Hilfe eines Unterprogramms $(U, l) \rightarrow (AU - b)_l$ spart man sich den Vektor V gegenüber 3.3.2. Das Verfahren ist aber abhängig vom gewählten Durchlauf

SSOR-Verfahren (Symmetrisches SOR) Erst SOR mit Durchlauf $1, \dots, n$ dann $n, \dots, 1$.
Außerdem erhält man damit eine symmetrische Iteration.

Die Iterationsmatrix T_ω^{SSOR} setzt sich zusammen aus

$$\begin{aligned} T_\omega^{\text{SOR}^+} &:= Id - \omega(Id - \omega L)^{-1} D^{-1} A \quad \text{und} \\ T_\omega^{\text{SOR}^-} &:= Id - \omega(Id - \omega R)^{-1} D^{-1} A \end{aligned}$$

zu deren Produkt:

$$T_\omega^{\text{SSOR}} = T_\omega^{\text{SOR}^-} \cdot T_\omega^{\text{SOR}^+}$$

3.3.4 Konvergenz des Jakobi- und Gauß-Seidel-Verfahrens

Satz 8. Sei $A \in \mathbb{R}^{n,n}$ mit $A_{ii} \neq 0$ für alle i .

(i) (Starkes Zeilensummenkriterium:) Gilt

$$\|L + R\|_\infty < 1 \quad (\text{Zeilensummennorm})$$

dann konvergieren Jakobi und GS und es gilt

$$\varrho(T_{GS}) \leq \varrho(T_J) < 1$$

(ii) (Schwachere Zeilensummenkriterium:) Es gelte

$$\sum_{\substack{j=1 \\ j \neq i}}^n \left| \frac{A_{ij}}{A_{ii}} \right| \leq 1, \quad i = 1, \dots, n$$

aber „<“ gelte für wenigstens einen Index. Weiter sei A unzerlegbar (irreduzibel), d.h. gibt es Mengen $M_1, M_2 \subset I = \{1, \dots, n\}$ mit $M_1 \cup M_2 = I$, aber $M_1 \cap M_2 = \emptyset$ und gilt $A_{ij} = 0$ für alle $(i, j) \in M_1 \times M_2$, so folgt $M_1 = \emptyset$ oder $M_2 = \emptyset$.

(Keine Permutation P führt auf $PA = \begin{bmatrix} * & 0 \\ * & * \end{bmatrix}$)

Dann konvergieren Jakobi- und GS-Verfahren.

Beweis. (ii) z.z. Jakobiverfahren: $T \equiv T_J = Id - D^{-1}A \Rightarrow \varrho(T) < 1$.

Sei $\lambda \in \mathbb{C}$, $v \in \mathbb{C}^n$ mit $|v|_\infty = 1$ und $Tv = \lambda v$.

Annahme: $|\lambda| \geq 1$

Dann gilt für jedes $i \in I = \{1, \dots, n\}$

$$|v_i| \leq |\lambda v_i| = |(Tv)_i| \leq \sum_{\substack{j=1 \\ j \neq i}} \left| \frac{A_{ij}}{A_{ii}} \right| \underbrace{|v_j|}_{\leq 1} \leq \sum_{\substack{j=1 \\ j \neq i}} \left| \frac{A_{ij}}{A_{ii}} \right| \leq 1$$

Sei i_0 ein Index mit „<“.

Dann ist für ein $j \in I$: $A_{i_0 j} \neq 0$, denn sonst wäre A reduzibel mit $M_1 = \{i_0\}$, $M_2 = I \setminus \{i_0\}$. Für $i = i_0$ folgt dann also $|v_{i_0}| < 1$.

Nun sei $M_1 = \{i \in I : |v_i| = 1\}$ und $M_2 = I \setminus M_1$.

Dann ist $M_1 \neq \emptyset$ nach Vor. $|v|_\infty = 1$. Sei $i \in M_1$. Weil nicht $A_{ij} = 0$ für alle $j \in M_2$ gelten kann, kann man $|v_i| < 1$ wie oben zeigen. Wid.

Gauß-Seidel-Verfahren: Wähle λ, v wie oben für $T = T_{GS}$
Für T gilt:

$$(Tv)_i = \sum_{j=1}^{i-1} \frac{A_{ij}}{A_{ii}} (Tv)_j + \sum_{j=i+1}^n \frac{A_{ij}}{A_{ii}} \cdot v_j$$

Mit Induktion folgt: $|(Tv)_j| \leq 1$ für $j \in I$.

Damit $|(Tv)_j| = |\lambda v_j| = |\lambda| \cdot |v_j| \leq |v_j| \Rightarrow |\lambda| \leq 1$

Somit

$$|(Tv)_i| \leq \sum_{j=1}^{i-1} \left| \frac{A_{ij}}{A_{ii}} \underbrace{(Tv)_j}_{\leq 1} \right| + \sum_{j=i+1}^n \left| \frac{A_{ij}}{A_{ii}} \right| \underbrace{|v_j|}_{\leq 1} \leq \sum_{\substack{j=1 \\ j \neq i}}^n \left| \frac{A_{ij}}{A_{ii}} \right|$$

Dann geht der Beweis wie oben.

□

3.3.5 Konvergenzsatz des SOR-Verfahrens

Satz 9. Es sei $A \in \mathbb{R}^{n,n}$ mit $A_{ii} \neq 0$, $i = 1, \dots, n$ mit $T_{GS,\omega}$ sei die Iterationsmatrix des SOR-Verfahrens. Dann gilt:

1.)

$$\varrho(T_{GS,\omega}) \geq |\omega - 1|$$

D.h. SOR konvergiert höchstens für $\omega \in (0, 2)$

2.) Ist A spd, so gilt:

$$\varrho(T_{GS,\omega}) < 1 \quad \text{für } \omega \in (0, 2)$$

Beweis. 1.) $T = T_{GS,\omega}$ hat die Form

$$\begin{aligned} T &= Id - \omega(Id - \omega L)^{-1} D^{-1} A \\ &= (Id - \omega L)^{-1} (Id - \omega L - \omega D^{-1} A) \\ &= (Id - \omega L)^{-1} ((1 - \omega)Id + \omega R) \end{aligned}$$

$$\det(T) = \det((Id - \omega L)^{-1}) \cdot \det((1 - \omega)Id + \omega R) = (1 - \omega)^n.$$

Wegen $\det(T) = \prod_{i=1}^n \lambda_i$ folgt: es ex ein $i_0 \in I$ mit $\varrho(T) \geq |\lambda_{i_0}| \geq |\omega - 1|$.

2.) Aufwändig.

□

Bemerkung Konvergenzkriterium ist unabhängig von der Nummerierung.

3.3.6 Konvergenz des SSOR

Satz 10. Sei $A \in \mathbb{R}^{n,n}$ spd. Zu $\omega \in \mathbb{R}$ sei

$T_{GS,\omega}^+$: SOR-Operator mit Durchlauf $i = 1, \dots, n$

$T_{GS,\omega}^-$: SOR-Operator mit Durchlauf $i = n, \dots, 1$

Dann ist die Iterationsmatrix des SSOR-Verfahrens durch $\delta_\omega = T_{GS,\omega}^- \cdot T_{GS,\omega}^+$ gegeben.

Es gilt:

$$\varrho(\delta_\omega) \geq |\omega - 1|^2 \text{ und } \varrho(\delta_\omega) < 1 \text{ für } \omega \in (0, 2)$$

Beweis. Korollar zum letzten Theorem □

3.3.7 Beispiele

$A := \text{tridiag}(-1, 2, -1)$.

Mit $h := \frac{1}{n+1}$ erhalten wir die Eigenwerte $\lambda_k = 2(1 - \cos(k\pi h))$. Wir suchen $\varrho(T)$ für verschiedene Verfahren:

1.) Jakobi-Verfahren:

$$T_J = Id - D^{-1}A = Id - \frac{1}{2}A$$

$$\text{Eigenwerte: } \lambda_{J,k} = 1 - \frac{1}{2}\lambda_k = \cos(k\pi h)$$

Bild

$$\Rightarrow \varrho(T_J) = \cos(\pi h) = 1 - \frac{1}{2}(\pi h)^2 + \mathcal{O}(h^4) = 1 - \frac{1}{2}\frac{\pi^2}{n^2} + \mathcal{O}(n^{-3})$$

2.) Gauß-Seidel-Verfahren:

Für die Komponenten von $T_{GS} \cdot u$ gilt:

$$(T_{GS}u)_l = \frac{1}{2}((T_{GS}u)_{l-1} + u_{l+1})$$

Ist $T_{GS}u = \lambda_{GS}u$, so folgt:

$$\begin{aligned} \lambda_{GS}u_l &= \frac{1}{2}(\lambda_{GS}u_{l-1} + u_{l+1}) \quad | \cdot \lambda_{GS}^{-\frac{l+1}{2}} = \sqrt{\lambda_{GS}}^{-(l+1)} \\ \Leftrightarrow \sqrt{\lambda_{GS}}^{-l+1}u_l &= \frac{1}{2}(\sqrt{\lambda_{GS}}^{-l+1}u_{l-1} + \sqrt{\lambda_{GS}}^{-l-1}u_{l+1}) \\ \Leftrightarrow \sqrt{\lambda_{GS}}v_l &= \frac{1}{2}(v_{l-1} + v_{l+1}) = (T_Jv)_l \end{aligned}$$

Ist λ_J Eigenwert von T_J , so ist λ_J^2 Eigenwert von T_{GS}

$$\varrho(T_{GS}) = \cos(\pi h)^2 = (1 - \pi h + \mathcal{O}(h^4))^2 = 1 - \pi^2 h^2 + \mathcal{O}(n^3)$$

3.) SOR-Verfahren:

$$T_\omega \equiv T_{SOR,\omega} \quad (\omega = 1 : T_1 = T_{GS})$$

$$(T_\omega u)_l = (1 - \omega)u_l + \frac{1}{2}\omega(\lambda_\omega u_{l-1} + u_{l+1})$$

$$\Rightarrow (1 - \omega)u_l + \frac{1}{2}\omega\sqrt{\lambda_\omega}(\sqrt{\lambda_\omega}u_{l-1} + \frac{1}{\sqrt{\lambda_\omega}}u_{l+1}) = \lambda_\omega u_l$$

Multiplikation der Gleichung mit $\sqrt{\lambda_\omega}^{-l}$ und Substitution von $v_l = \sqrt{\lambda_\omega}^{-l} \cdot u_l$ ergibt:

$$\begin{aligned} (1 - \omega)v_l + \frac{1}{2}\omega\sqrt{\lambda_\omega}(v_{l-1} + v_{l+1}) &= \lambda_\omega v_l \\ \Rightarrow \frac{1}{\omega\sqrt{\lambda_\omega}}(\lambda_\omega + \omega - 1)v_l &= \frac{1}{2}(v_{l-1} + v_{l+1}) \end{aligned}$$

D.h. $\lambda_\omega \in \text{spec}(T_\omega) \Rightarrow \frac{1}{\omega\sqrt{\lambda_\omega}}(\lambda_\omega + \omega - 1) \in \text{spec}(T_J) = \{\cos(k\pi h) : k = 1, \dots, n\}$

$$\Rightarrow (\sqrt{\lambda_\omega})^2 - \omega \cos(k\pi h)\sqrt{\lambda_\omega} + \omega - 1 = 0$$

Lösung der quadratischen Gleichung ergibt die Eigenwerte des SOR-Verfahrens.

($\omega = 1$: Eigenwerte des GS-Verfahrens: $\lambda_{\omega=1} = \cos(k\pi h)^2$)

Wir berechnen nun ω , so dass $\varrho(T_\omega)$ minimal ist.

Bild

Es folgt:

$$\begin{aligned} \omega_{\text{opt}} &= \frac{2}{1 + \sqrt{1 - \varrho(T_J)^2}} \geq 1 \\ \varrho_{\text{opt}} &= \omega_{\text{opt}} - 1 \end{aligned}$$

Für unser Beispiel und $h \rightarrow 0$:

$$\begin{aligned} 1 - \varrho(T_J)^2 &\approx 1 - (1 - \frac{1}{2}\pi^2 h^2)^2 \approx \pi^2 h^2 \\ \omega_{\text{opt}} &= \frac{2}{1 + \pi h} \approx 2(1 - \pi h) \\ \varrho_{\text{opt}} &\approx 1 - 2\pi h \end{aligned}$$

3.3.8 Konsistent geordnete Matrizen

Definition. $A \in \mathbb{R}^{n,n}$ heißt konsistent geordnet, wenn gilt: bzgl. der Zerlegung $A = D(Id - L - R)$ sind die Eigenwerte von $\alpha L + \frac{1}{\alpha}R$ unabhängig von $\alpha \in \mathbb{C} \setminus \{0\}$

Satz 11. Für konsistent geordnete Matrizen A mit $A_{ii} \neq 0$ und $\text{spec}(T_J) \subset (-1, 1)$ gilt

$$\varrho(T_{GS}) = \varrho(T_J)^2$$

und für das SOR-Verfahren gilt

$$\begin{aligned} \omega_{\text{opt}} &= \frac{2}{1 + \sqrt{1 - \varrho(T_J)^2}} \in (1, 2) \\ \varrho_{\text{opt}} &= \omega_{\text{opt}} - 1 \end{aligned}$$

Beweis. ÜA

□

Beispiele konsistent geordneter Matrizen:

- Tridiagonalmatrizen
- Block-Tridiagonalmatrizen
- Zwei-zyklische oder Red-Black-Matrizen

A heißt *zwei-zyklisch* oder *red-black-Matrix*, falls es eine Permutation gibt, so dass A auf die Form $\begin{bmatrix} D_1 & * \\ * & D_2 \end{bmatrix}$ mit Diagonalmatrizen D_1, D_2 gebracht werden kann.

3.3.9 Rechenaufwand

- 1.) Es sei $\varrho = \varrho(T) < 1$ und $\text{compl}(T) \approx \text{compl}(A)$. Der Aufwand zur Fehlerreduktion um den Faktor $\tau \in (0, 1)$ sei die Anzahl der Rechenoperationen um u_m mit

$$|u_m - u_*| \leq \tau |u_0 - u_*|$$

($Au_* = b$) zu erhalten.

Wir erhalten $\frac{|u_m - u_*|}{|u_0 - u_*|} \leq \varrho^m \stackrel{!}{\leq} \tau$.

$$\Rightarrow m \cdot \log(\varrho) \leq \log(\tau) \Rightarrow m \geq \frac{\log(\tau)}{\log(\varrho)} = \frac{\log(1/\tau)}{\log(1/\varrho)}$$

Aufwand := $m \cdot \text{compl}(T) \approx m \cdot \text{compl}(A)$

$\text{compl}(A) \sim n$

$$\log(1/\varrho) = |\log(\varrho)| \approx |\log(1 - \frac{1}{2}\pi^2 h^2)| \approx \frac{1}{2}\pi^2 h^2 \approx \frac{1}{2} \frac{\pi^2}{n^2}$$

für das Beispiel aus 3.7

$$\Rightarrow \text{Aufwand}_J \sim n \cdot n^2 \cdot \log(1/\tau) \sim n^3 \cdot \log(1/\tau)$$

$$\text{Aufwand}_{GS} \approx \frac{1}{2} \cdot \text{Aufwand}_J \sim n^3 \cdot \log(1/\tau)$$

$$\text{Aufwand}_{SOR} \sim n \cdot \frac{\log(1/\tau)}{h} \sim n^2 \cdot \log(1/\tau) \sim \frac{1}{n} \cdot \text{Aufwand}_{GS}$$

- 2.) SSOR-Verfahren ist nicht schneller als das Gauß-Seidel-Verfahren:

$$\varrho(\delta_\omega) = \varrho(T_{GS, \omega(2-\omega)}) \geq \varrho(T_{GS, 1})$$

- 3.) Diagonaldominante A , $A = \text{tridiag}(-1, a, -1)$ mit $a > 2$. Dann wird $\varrho(T_J) = 2/a < 1$ unabhängig von n . Der Aufwand ist dann $\sim n \cdot \log(1/\tau)$

Beispiel:

$$\begin{aligned}
\partial_t u - u'' &= 0 \text{ in } (0, 1) \\
u(t, 0) = u(t, 1) &= 0 \quad \forall t > 0 \\
u(0, x) &= \varphi(x) \quad \forall x \in (0, 1)
\end{aligned}$$

Wir diskretisieren:

$$\partial_t u(t, x) \approx \frac{u(t, x) - u(t - \Delta t, x)}{\Delta t}$$

Mit $u_i^k \approx u(t_k, x_i)$, $t_k = k\Delta t$, $x_i = ih$

$$\frac{u_i^{k+1} - u_i^k}{\Delta t} + \frac{1}{h^2}(-u_{i+1}^{k+1} + 2u_i^{k+1} - u_{i-1}^{k+1}) = 0$$

In Matrixschreibweise:

$$\left(Id_n + \frac{\Delta t}{h^2} \text{tridiag}_n(-1, 2, -1) \right) u^{k+1} = u^k \quad (*)$$

wobei $u^k = [u_i^k]_{i=1, \dots, n}$
 u^0 (Startwert: $u_i^0 = \varphi(x_i)$)
 $\rightarrow u^1$ (Löse * für $k = 0$)
 $\rightarrow u^2$ (Löse * für $k = 1$)
 $\rightarrow \dots$

Matrix in (*) (Mult mit $\frac{h^2}{\Delta t}$)

$$\text{tridiag}_n(-1, 2 + \underbrace{\frac{h^2}{\Delta t}}_{=: a > 2}, -1)$$

Zusatz: 2D-Fall Bsp.: $\begin{bmatrix} -1 & \frac{1}{4} & -1 \\ & -1 & \end{bmatrix}$ auf $[0, 1]^2$ mit lexikographischer Anordnung.

Sei n Anzahl der Punkte in einer Raumrichtung, $N = n^2$, $h = \frac{1}{n+1} \approx \frac{1}{n} = \frac{1}{\sqrt{N}}$

$$\begin{aligned}
\varrho_J &= 1 - \mathcal{O}(h^2) = 1 - \mathcal{O}\left(\frac{1}{N}\right) \\
\varrho_{\text{GS}} &= 1 - \mathcal{O}(h^2) = 1 - \mathcal{O}\left(\frac{1}{N}\right) \\
\varrho_{\text{opt}} &= 1 - \mathcal{O}(h) = 1 - \mathcal{O}\left(\frac{1}{\sqrt{N}}\right)
\end{aligned}$$

Weiter gilt:

$$\begin{aligned}
\text{Aufwand}_J &\sim N^2 \\
\text{Aufwand}_{\text{GS}} &\sim N^2 \\
\text{Aufwand}_{\text{opt}} &\sim N \cdot \sqrt{N} = N^{3/2}
\end{aligned}$$

Gaußelimination: Bandmatrix der Breite $m = n \approx \sqrt{N}$

$$\text{Aufwand}_{\text{GE}} = m^2 N \approx N^2$$

Abbruchkriterium für Iterationen:

$$|u_i - u_{i+1}| \leq \text{Tol} \quad \text{oder} \quad \underbrace{|Au_i - b|}_{\text{Residuum}} \leq \text{Tol} \cdot |b|$$

wobei $\text{Tol} \in \mathbb{R}_+$ die „Toleranz“ ist.

3.3.10 Idee Des Mehrgitterverfahrens

Problem: Aufwand ist noch $\mathcal{O}(n^\kappa)$ mit $\kappa > 1$.

Wir suchen schnelle Löser: $\kappa = 1$.

Gedämpftes Jakobi-Verfahren mit $\omega = 1/2$.

$$T_J = Id - \frac{1}{2} \left(\frac{1}{2} \right) A = Id - \frac{1}{4} A$$

im Beispiel aus 3.7. Dann ist

$$\text{spec}(T_{J,1/2}) = \left\{ 1 - \frac{1}{4} \lambda : \lambda \in \text{spec}(A) \right\} = \left\{ \frac{1}{2} (1 + \cos(k\pi n)) : k = 1, \dots, n \right\}$$

Für den Fehlervektor e_{i+1} gilt: $e_{i+1} = T_{J,1/2} e_i$. Sei $\{s_l\}_{l=1,\dots,n}$ Eigenbasis von A . Stelle e_i als Linearkombination der s_i dar:

$$e_i = \sum_{l=1}^n \alpha_l^{(i)} s_l$$

Dann folgt für $i + 1$:

$$e_{i+1} = T_{J,1/2} e_i = \sum_{l=1}^n \lambda_l \alpha_l^{(i)} s_l$$

Sei nun n gerade. Wir definieren

$$e_i^{\text{NF}} := \sum_{l=1}^{n/2} \alpha_l^{(i)} s_l \quad (\text{Niederfrequenter Anteil})$$

$$e_i^{\text{HF}} := \sum_{l=\frac{n}{2}+1}^n \alpha_l^{(i)} s_l \quad (\text{Hochfrequenter Anteil})$$

Bilder

Es gilt:

$$\begin{aligned} |T_J e_i^{\text{NF}}| &\leq |e_i^{\text{NF}}| \\ |T_J e_i^{\text{HF}}| &\leq \frac{1}{2} |e_i^{\text{HF}}| \end{aligned}$$

Idee: Verwende 2 Löser, einen für den NF-Anteil und gedämpftes Jakobi-Verfahren für den HF-Anteil

Bildchen

NF-Löser ist ein direktes Verfahren auf den Knoten echt unterhalb des feinsten Levels.

Trick: Verfähre analog für das Grobgitterproblem. Hierzu wird Jakobi heute noch verwendet.

Theorie: N -unabhängige Konvergenzrate.

Entwicklung des Mehrgitter-Verfahrens: 1965-1990.

3.4 Das CG-Verfahren

3.4.1 Das Gradientenverfahren

Definition. Sei $A \in \mathbb{R}^{n,n}$ spd und $b \in \mathbb{R}^n$ beliebig. Dann heißt die Abbildung $\varepsilon : \mathbb{R}^n \rightarrow \mathbb{R}$ def. durch

$$\varepsilon(v) := v \cdot Av - b \cdot v$$

die Energie.

ε ist strikt konvexe, nach unten beschränkte Funktion mit $\lim_{|v| \rightarrow \infty} \varepsilon(v) = \infty$. Weiter gilt

$$\varepsilon'' = A$$

Bildchen

Also hat ε ein eindeutiges Minimum in u_* und dies ist charakterisiert durch $\varepsilon'(u_*)[d] = 0 \quad \forall d \in \mathbb{R}^n$. Es gilt:

$$\varepsilon'(v)d = (Av - b) \cdot d \quad \forall d \in \mathbb{R}^n$$

Also folgt:

$$\varepsilon'(u_*) = 0 \Leftrightarrow Au_* = b$$

Idee: konstruiere Folge $\{u_k\}_k$, so dass $\varepsilon(u_{k+1}) < \varepsilon(u_k)$ ist mit $\lim_{k \rightarrow \infty} \varepsilon(u_k) = \min_{v \in \mathbb{R}^n} \varepsilon(v) = \varepsilon(u_*)$

Der steilste Anstieg in u_k ist

$$-\nabla \varepsilon(u_k) = -(Au_k - b) =: -r_k$$

Ansatz für k -ten Schritt:

$$u_{k+1} = u_k - \alpha_k r_k$$

Mit $\alpha_k \in \mathbb{R}$. Bestimme α_k wie folgt: Def.

$$\Phi(\alpha) := \varepsilon(u_k - \alpha r_k) \quad (\alpha \in \mathbb{R})$$

Φ ist nach unten beschränkt und strikt konvex mit $\lim_{|\alpha| \rightarrow \infty} \Phi(\alpha) = \infty$ Daher ex. α_k mit

$\Phi(\alpha_k) = \min_{\alpha \in \mathbb{R}} \Phi(\alpha)$ und es gilt: $\Phi'(\alpha_k) = 0$.

$$\begin{aligned} 0 &\stackrel{!}{=} \Phi'(\alpha_k) &= \varepsilon'(u_k - \alpha_k r_k) \cdot (-r_k) \\ &= (A(u_k - \alpha_k r_k) - b) \cdot (-r_k) \\ &= -(r_k - \alpha_k A r_k) \cdot r_k \\ &= -r_k \cdot r_k + \alpha_k A r_k \cdot r_k \end{aligned}$$

$$\Rightarrow \alpha_k = \frac{|r_k|^2}{Ar_k \cdot r_k}$$

Denn: $r_k \cdot Ar_k \stackrel{!}{=} 0 \Rightarrow r_k = 0 \Rightarrow Au_k = b$. Fertig!

Satz 12. Sei A spd, $(v, w)_A := Av \cdot w$, $\|v\|_A := (v, v)_A^{1/2}$
Ist $Au = b$ und $u_0 \in \mathbb{R}^n$, so konvergiert die Folge $\{u_k\}_k$ mit

$$u_{k+1} = u_k - \frac{|r_k|^2}{\|r_k\|_A^2} r_k, \quad k \geq 0, \quad r_k = Au_k - b$$

gegen u und es gilt:

$$\|u_{k+1} - u\|_A \leq \frac{\kappa - 1}{\kappa + 1} \|u_k - u\|_A = \left(1 - \frac{2}{\kappa + 1}\right) \|u_k - u\|_A$$

wobei $\kappa = \text{cond}_2(A) = \frac{\lambda_{\max}}{\lambda_{\min}}$. Bea.: $\|\cdot\|$ heißt Energienorm.

3.4.2 Fehlerminimierung auf Unterräumen

Algorithmus in 4.1 (CG-Verfahren) ist zu langsam.

Idee: $\{V_k\}_{k=1, \dots, n}$ sei eine Folge von Unterräumen des \mathbb{R}^n mit $\dim V_k = k$.

Ausgehend von $u_0 \in \mathbb{R}^n$ machen wir den Ansatz

$$u_{k+1} = u_k + p_{k+1} \quad \text{mit } p_{k+1} \in V_{k+1}$$

Wir definieren p_{k+1} durch

$$\|e_{k+1}\|_A = \|e_k + p_{k+1}\|_A \stackrel{!}{=} \min_{p \in V_{k+1}} \|e_k + p\|_A.$$

Wegen $0 \in V_{k+1}$ gilt $\|e_{k+1}\|_A \leq \|e_k\|_A$ und mit $V_n = \mathbb{R}^n$ ist $u_n = u$ die Lösung.

Wir definieren $\Phi : V^{k+1} \rightarrow \mathbb{R}$ durch

$$\Phi(p) := \|e_k + p\|_A^2 \quad (p \in V^{k+1})$$

Φ ist strikt konvex und es gilt $\Phi(p) \rightarrow \infty$ ($|p| \rightarrow \infty$). Das Minimum in p_{k+1} ist vollständig charakterisiert durch

$$\begin{aligned} 0 &\stackrel{!}{=} (\nabla \Phi(p_{k+1}), q)_A \\ &= 2(e_k + p_{k+1}, q)_A \\ &= 2(e_{k+1}, q)_A \quad \forall q \in V_{k+1} \end{aligned}$$

$\Rightarrow (e_{k+1}, q)_A = 0$ für alle $q \in V_{k+1}$. Wir nennen diese Eigenschaft von e_{k+1} *A-Orthogonalität* von e_{k+1} und V_{k+1} (Schreibweise $e_{k+1} \perp_A V_{k+1}$)

3.4.3 Krylovräume

Für die Idee aus 4.2 wählen wir zu $d_0 \in \mathbb{R}^n \setminus \{0\}$ die Räume

$$V_k \equiv V_k(A, d_0) = \text{span}\{d_0, Ad_0, \dots, A^{k-1}d_0\} \quad (k \geq 1)$$

Wir nehmen erstmal an, dass $\dim V_k = k$ ist. Wir errichten nun auf V_k eine orthogonale Basis mit dem Gram-Schmidt-Verfahren ausgehend von d_0 . Ansatz:

$$d_{k+1} = Ad_k - \sum_{l=0}^k \sigma_{kl} d_l$$

Bestimme die σ_{kl} durch die Forderung $(d_{k+1}, d_j)_A = 0 \quad j = 0, \dots, k$. (Tatsächlich benötigt man nur σ_{kk} und $\sigma_{k,k-1}$). Es gilt

$$V_k = \text{span}\{d_0, \dots, d_{k-1}\}$$

und Ad_k ist genau dann linear unabhängig von $\{d_0, \dots, d_k\}$ solange $A^{k+1}d_0 \notin \text{span}\{d_0, \dots, A^k d_0\}$ ist.

Beweis. Dazu: $d_1 \in Ad_0 + \text{span}\{d_0\} = Ad_0 + V_1 \subset V_2$

I. V.: $d_k \in A^k d_0 + \text{span}\{d_0, \dots, d_{k-1}\} \subset A^k d_0 + V_k \Rightarrow Ad_k \in A^{k+1} d_0 + AV_k \subset V_{k+1} \quad \square$

3.4.4 Das CG-Verfahren nach Hestenes/ Stiefel (1954)

Idee aus 4.3 aber mit einer Modifikation, die die Zahl der Koeffizienten reduziert.

$u_0 \in \mathbb{R}^n, r_0 = Au_0 - b =: d_0$,

$$d_{k+1} = r_{k+1} + \sum_{l=0}^k \sigma_{kl} d_l \quad (k \geq 0)$$

Lemma 3.

$$\text{span}\{d_l : l = 0, \dots, k\} \subset V_{k+1}(A, r_0) \equiv V_{k+1}$$

Beweis. $k = 1$: $\text{span}\{d_0\} = \text{span}\{r_0\} = V_1$

Ann.: $\text{span}\{d_l : l = 0, \dots, k\} \subset V_{k+1} \xrightarrow{!} d_{k+1} \in V_{k+2}$

$$\begin{aligned} d_{k+1} \in r_{k+1} + \text{span}\{d_0, \dots, d_k\} &\stackrel{\text{I.V.}}{=} Au_{k+1} - b + V_{k+1} \\ &\subset A(u_k + V_{k+1}) - b + V_{k+1} \\ &= \underbrace{r_k}_{\in V_{k+1}} + AV_{k+1} + V_{k+1} \\ &= AV_{k+1} + V_{k+1} \subset V_{k+2} \end{aligned}$$

\square

Konstruktion des Verfahrens

Es gelte $e_k \perp_A V_k$, $(d_i, d_j)_A = 0$ für $i, j \leq k, i \neq j$.

Geforderte Minimalität des Fehlers:

$$\begin{aligned} 0 &\stackrel{!}{=} (e_{k+1}, d_j)_A = A e_{k+1} \cdot d_j \\ &= A(u_{k+1} - u) \cdot d_j \\ &= r_{k+1} \cdot d_j \quad (j = 0, \dots, k) \end{aligned}$$

Weiter

$$0 = r_{k+1} \cdot A d_i = (r_{k+1}, d_i)_A \quad (i = 0, \dots, k-1)$$

Berechnung der σ_{kl} für $j = 0, \dots, k-1$:

$$0 \stackrel{!}{=} (d_{k+1}, d_j)_A \stackrel{\text{orth.}}{=} \underbrace{(r_{k+1}, d_j)_A}_{=0} + \sigma_{kj} \|d_j\|_A^2$$

$\Rightarrow \sigma_{kj} = 0$ für $j = 0, \dots, k-1$.

Es bleibt $j = k$:

$$0 \stackrel{!}{=} (d_{k+1}, d_k)_A = (r_{k+1}, d_k)_A + \sigma_{kk} \|d_k\|_A^2$$

$$\Rightarrow \beta_k := \sigma_{kk} = -\frac{(r_{k+1}, d_k)_A}{\|d_k\|_A^2} \Rightarrow d_{k+1} = r_{k+1} + \beta_k d_k$$

Aus $e_k \perp_A V_k$ folgt

$$\begin{aligned} (e_k, d_k)_A &= (e_k, r_k)_A + \beta_{k-1} \underbrace{(e_k, d_{k-1})_A}_{\in V_k} \\ &= (e_k, r_k)_A = A e_k \cdot r_k = |r_k|^2 \end{aligned}$$

Orthogonalisierung des Fehlers e_{k+1}

$$0 = (e_{k+1}, d_j)_A \text{ für } j < k$$

$\Rightarrow (e_k + p_{k+1}, \underbrace{d_j}_{\in V_k})_A = (p_{k+1}, d_j)_A$ für $j < k$. Also $p_{k+1} \sim d_k$, etwa $p_{k+1} = \alpha_k d_k$ und

damit

$$(*) \quad u_{k+1} = u_k - \alpha_k d_k$$

α_k folgt aus

$$\begin{aligned} (e_{k+1}, d_k)_A &= (e_k - \alpha_k d_k, d_k)_A \\ &= (e_k, d_k)_A - \alpha_k \|d_k\|^2 \\ &= |r_k|^2 - \alpha_k \|d_k\|_A^2 \end{aligned}$$

$$\Rightarrow \alpha_k = \frac{|r_k|^2}{\|d_k\|_A^2}$$

Damit lässt sich β_k eleganter schreiben: aus $(*)$ folgt:

$$r_{k+1} = r_k - \alpha_k A d_k$$

Dann ist

$$\begin{aligned}
(r_{k+1}, d_k)_A &= r_{k+1} \cdot Ad_k \\
&= r_{k+1} \cdot \left(-\frac{1}{\alpha_k} (r_{k+1} - r_k) \right) \\
&= -\frac{\|d_k\|_A^2}{|r_k|^2} (|r_{k+1}|^2 - \underbrace{r_{k+1} \cdot r_k}_{=0(z.z.)})
\end{aligned}$$

und es folgt $\beta_k = -\frac{(r_{k+1}, d_k)_A}{\|d_k\|_A^2} = \frac{|r_{k+1}|^2}{|r_k|^2}$

Noch z.z.: $r_{k+1} \cdot r_k = 0$:

$$\begin{aligned}
r_{k+1} \cdot r_k &= (r_k - \alpha_k Ad_k) \cdot r_k \\
&= |r_k|^2 - \alpha_k Ad_k \cdot r_k \\
&= |r_k|^2 - \frac{|r_k|^2}{\|d_k\|_A^2} Ad_k \cdot (d_k - \beta_{k-1} d_{k-1}) \\
&= |r_k|^2 - |r_k|^2 = 0
\end{aligned}$$

Der Algorithmus

Initialisierung $u_0 \in \mathbb{R}^n$, $r_0 = Au_0 - b$, $d_0 = r_0$

Iteration $k \geq 0$

$$\begin{aligned}
\alpha_k &= \frac{|r_k|^2}{d_k \cdot Ad_k} = \frac{|r_k|^2}{\|d_k\|_A^2} \\
u_{k+1} &= u_k - \alpha_k d_k \\
r_{k+1} &= r_k - \alpha_k Ad_k \\
\beta_k &= \frac{|r_{k+1}|^2}{|r_k|^2} \\
d_{k+1} &= r_{k+1} + \beta_k d_k
\end{aligned}$$

Wohldefiniert?

$r_k = 0 \Leftrightarrow Au_k = b \checkmark$

$d_{k+1} = 0$?

Dann wäre $\sum_{j=0}^{k+1} \gamma_j A^j d_0 = 0$ für $\gamma \in \mathbb{R}^{k+2} \setminus \{0\}$

$\gamma_0 \neq 0$: $\underbrace{d_0}_{=r_0} = \sum_{j=1}^{k+1} \frac{\gamma_j}{\gamma_0} A^j d_0$

$$\Rightarrow e_0 = A^{-1} r_0 = A^{-1} d_0 = - \sum_{j=0}^k \frac{\gamma_{j-1}}{\gamma_0} A^j d_0 \in V_{k+1}$$

Folgt genauso, falls $\gamma_0 = 0$ und $\gamma_1 \neq 0$ wäre.

$$e_{k+1} = e_k + p_{k+1} = e_{k-1} + p_k + p_{k+1} = \dots \in e_0 + V_{k+1} \subseteq V_{k+1} \text{ da } e_0 \in V_{k+1}$$

$$\Rightarrow e_{k+1} = 0, \text{ da } e_{k+1} \perp_A V_{k+1}$$

Aufwand	MV	VV	SV	Speicher
	1	2	3	$3N$

- MV $\hat{=}$ Matrix * Vektor
- VV $\hat{=}$ Skalarprodukte
- SV $\hat{=}$ Skalar * Vektor
- Speicher: zusätzlicher Speicher

3.4.5 Konvergenz des CG-Verfahrens

Ausgangspunkt:

$$\|e_k\|_A = \min_{p \in V_k} \|e_{k-1} + p\|_A$$

$V_k = \text{span}\{d_0, \dots, A^{k-1}d_0\}$. Aus $u_k \in u_0 + V_k$ folgt $e_k \in e_0 + V_k$

$$\Rightarrow e_k = e_0 + \sum_{j=0}^{k-1} u_{kj} A^j d_0$$

für geeignete u_{kj} . Es ist $d_0 = r_0 = Ae_0$, also gilt

$$e_k = e_0 + A \cdot \sum_{j=0}^{k-1} u_{kj} \cdot A^j e_0$$

Es gibt also ein Polynom $q_k \in \mathbb{P}_k^* = \{q \in \mathbb{P}_k : q(0) = 1\}$ mit $e_k = q_k(A)e_0$. D.h. wir können auch schreiben

$$\|e_k\|_A = \min_{q \in \mathbb{P}_k^*} \{\|q(A)e_0\|_A\}$$

A spd $\Rightarrow \exists$ ONB $\{z_l\}_l$ mit $Az_l = \lambda_l z_l$, λ_l die Eigenwerte von A . Dann gilt etwa

$$q(A)e_0 = q(A) \sum_{l=1}^n \alpha_l z_l = \sum_{l=1}^n \alpha_l q(\lambda_l) z_l$$

Für den Fehler e_k gilt:

$$\begin{aligned} \|e_k\|_A^2 &= \sum_{l=1}^n \alpha_l^2 q_k(\lambda_l)^2 \\ &\leq \max\{|q_k(\lambda_l)|^2\} \cdot \sum_{l=1}^n \alpha_l^2 \\ &\leq \max_{\lambda \in \text{spec}(A)} \{|q_k(\lambda)|^2\} \cdot \|e_0\|_A^2 \end{aligned}$$

Wir nehmen an, dass $\text{spec}(A) \subset [a, b] \subset \mathbb{R}_+$ ist. Dann ist

$$\max_{\lambda \in \text{spec}(A)} \{|q_k(\lambda)|^2\} \leq \max_{\lambda \in [a, b]} \{|q_k(\lambda)|^2\}$$

Insgesamt ist

$$\|e_k\|_A^2 \leq \min_{q \in \mathbb{P}_k^*} \max_{\lambda \in [a, b]} |q(\lambda)|^2 \cdot \|e_0\|_A^2$$

Den Vorfaktor nennen wir $\varrho_{a,b,k}^2$

Bildchen

Die Lösung ist lange bekannt, es gilt:

$$\varrho_{a,b,k} \leq 2 \left(\frac{\sqrt{\kappa} - 1}{\sqrt{\kappa} + 1} \right)^k \quad \text{mit } \kappa = b/a > 1$$

$\kappa = 1 \Rightarrow b = a \Rightarrow A \sim Id.$

Optimal: $a = \lambda_{\min}(A)$, $b = \lambda_{\max}(A)$

$\Rightarrow \kappa$ ist die Kondition $\text{cond}_2(A)$

Satz 13. Das CG-Verfahren für eine symmetrisch positive Matrix A konvergiert für alle Startwerte wenigstens linear, d.h.

$$\|u_k - u\|_A \leq 2 \left(\frac{\sqrt{\kappa} - 1}{\sqrt{\kappa} + 1} \right)^k \|u_0 - u\|_A = 2 \left(1 - \frac{2}{\sqrt{\kappa} + 1} \right)^k \|u_0 - u\|_A$$

Beweis. A : Das Problem wird gelöst von $q_k(x) = \frac{T_k\left(\frac{b+a-2x}{b-a}\right)}{T_k\left(\frac{b+a}{b-a}\right)}$, d.h.

$$\max_{\lambda \in [a, b]} |q_k(\lambda)|^2 = \min_{q \in \mathbb{P}_k^*} \min_{\lambda \in [a, b]} |q(\lambda)|^2$$

T_k ist das k -te Tschebyscheff-Polynom:

$$T_k(t) = \cos(k \cdot \arccos(t))$$

Das Argument ist die Transformation $[a, b] \rightarrow [-1, 1]$ z.z.: T_k ist ein Polynom:

Sei $\theta := \arccos(t)$. Dann gilt:

$$\begin{aligned}
T_k(t) &= \cos(k\theta) \\
&= \frac{1}{2} (e^{ik\theta} + e^{-ik\theta}) \\
&= \frac{1}{2} \left((e^{i\theta})^k + (e^{-i\theta})^k \right) \\
&= \frac{1}{2} \left((\cos(\theta) + i \cdot \sin(\theta))^k + (\cos(\theta) - i \cdot \sin(\theta))^k \right) \\
&= \frac{1}{2} \cdot \sum_{l=0}^k \binom{k}{l} \cos(\theta)^{k-l} ((i \cdot \sin(\theta))^2 + (-i \cdot \sin(\theta))^2) \\
&= \sum_{\substack{l=0 \\ l \text{ gerade}}}^k \binom{k}{l} \underbrace{\cos(\theta)^{k-l}}_{=t} \cdot \underbrace{(i \cdot \sin(\theta))^2}_{=\sqrt{1-t^2}} \\
&\stackrel{l=2l'}{=} - \sum_{l'=0}^{\lfloor k/2 \rfloor} \binom{k}{2l'} t^{k-2l'} (1-t^2)^{l'} \in \mathbb{P}_k
\end{aligned}$$

Also $q_k \in \mathbb{P}_k$, $q_k(0) = 1$. Für $t \in [-1, 1]$ ist $|T_k(t)| \leq 1$ und mit $\kappa = \frac{b}{a}$ gilt

$$\max_{x \in [a, b]} |q_k(x)| \leq \frac{1}{T_k\left(\frac{\kappa+1}{\kappa-1}\right)}$$

Aus der obigen Rechnung:

$$T_k(t) = \frac{1}{2} \left((t + \sqrt{t^2 - 1})^k + (t - \sqrt{t^2 - 1})^k \right)$$

Weiter gilt:

$$\left(\frac{\kappa+1}{\kappa-1} \right)^2 - 1 = \frac{\kappa^2 + 2\kappa + 1 - (\kappa^2 - 2\kappa + 1)}{(\kappa-1)^2} = \frac{4\kappa}{(\kappa-1)^\kappa}$$

Insgesamt folgt:

$$\begin{aligned}
T_k\left(\frac{\kappa+1}{\kappa-1}\right) &\geq \frac{1}{2} \left(\frac{\kappa+1}{\kappa-1} + \sqrt{\left(\frac{\kappa+1}{\kappa-1}\right)^2 - 1} \right)^k \\
&\geq \frac{1}{2} \left(\frac{\kappa+1}{\kappa-1} + \frac{2\sqrt{\kappa}}{\kappa-1} \right)^k \\
&= \frac{1}{2} \left(\frac{(\sqrt{\kappa}+1)^2}{(\sqrt{\kappa}+1)(\sqrt{\kappa}-1)} \right)^k \\
&= \frac{1}{2} \left(\frac{\sqrt{\kappa}+1}{\sqrt{\kappa}-1} \right)^k
\end{aligned}$$

□

3.4.6 Vorkonditionierung

In der Praxis: $\kappa = \kappa_n \rightarrow \infty (n \rightarrow \infty)$ liefert zu langsame Konvergenz.

C sei spd. Dann schreiben wir

$$CAu = Cb.$$

Wir wenden das CG-Verfahren auf dieses System an. CA ist i.A. nicht symmetrisch. Wir benötigen die Symmetrie aber nur im $(\cdot, \cdot)_A$ -Skalarprodukt. Dies gilt: Seien $x, y \in \mathbb{R}^n$:

$$(CAx, y)_A = A(CA)x \cdot y = CAx \cdot Ay = Ax \cdot CAy = (x, CAy)_A$$

$$\Rightarrow \text{adj}_A(CA) = CA.$$

Damit schreibt sich das CG-Verfahren wie folgt:

Initialisierung:

$$u_0, r_0 = Au_0 - b, d_0 = Cr_0 = h_0$$

Iteration für $k \geq 0$:

$$\begin{aligned}\alpha_k &= \frac{r_k \cdot h_k}{d_k \cdot Ad_k} \\ u_{k+1} &= u_k - \alpha_k d_k \\ r_{k+1} &= r_k - \alpha_k Ad_k \\ h_{k+1} &= Cr_{k+1} \\ \beta_k &= \frac{r_{k+1} \cdot h_{k+1}}{(r_k \cdot h_k)} \\ d_{k+1} &= h_{k+1} + \beta_k d_k\end{aligned}$$

$C = Id$: CG wie vorher.

$h_{k+1} = h_k - \alpha_k (Ad_k)$ (Ad_k ist das Residuum der neuen Gleichung.)

Der Krylorraum ist $V_k(CA, d_0)$

Abbruch:

$$\sqrt{\frac{|r_k \cdot h_k|}{b \cdot cb}} \leq \text{Tol}$$

In der Fehlerabschätzung steht dann $\kappa = \kappa(CA)$.

Am besten: $C \approx A^{-1}$, aber auch $\text{compl}(C) \approx \text{compl}(A)$ - Widerspricht sich!

Beispiele:

- $C = \text{diag}(A)^{-1}$
Billig, aber nur sinnvoll, wenn die Diagonale stark variiert.

- $C = T$, T ein Schritt eines konvergenten iterativen Verfahrens. Etwa T_{SSOR} (symmetrisch!)
Man erhält $\kappa = \mathcal{O}(\sqrt{N})$ statt $\mathcal{O}(N)$ für das Poissonproblem auf $[0, 1]^2$
oder $C = T_{\text{Multigrid}} \Rightarrow \kappa(CA) = \mathcal{O}(1)$

Bemerkungen

- Die Konvergenz des CG-Verfahrens beschleunigt im Laufe der Iteration
Bildchen
- Die Konvergenz des CG-Verfahrens hängt von der Eigenwertverteilung ab.
Bildchen

3.5 GMRES (Generalized minimal residuals, 1986)

3.5.1 Minimale Residuen

Problem: CG funktioniert nur für symmetrisch positiv definite Matrizen A
In vielen Problemen ist A weder symmetrisch noch positiv definit:

$$\begin{aligned} -u'' + \beta u' &= f && (\text{in } \mathbb{R}) \\ -\Delta u + \underbrace{b \cdot \nabla u}_{\text{Transportterm}} &= f && (\text{in } \mathbb{R}^d) \end{aligned}$$

Ziel: Nutze Prinzipien aus 4

Idee: A invertierbar $\Rightarrow A^\top A$ ist spd.

e_k Fehler $\Rightarrow \|e_k\|_{A^\top A} = \|Ae_k\|_2 = \|r_k\|_2 = \|Au_k - b\|_2$ (i.F.: $|\cdot| = \|\cdot\|_2$)

$$Au = b \Rightarrow A^\top Au = A^\top b$$

„CG-Verfahren für Normalengleichungen“ (ÜA)

Die Konvergenz, die sich aus den Fehlerabschätzungen von 4,5 ergibt, ist meist viel zu langsam: $\kappa(A^\top A) \stackrel{\text{i.A.}}{\gg} \kappa(A)$. (Wir arbeiten hier auf $V_k(A^\top A)$!)

Idee: Nutze $\|\cdot\|_{A^\top A}$ für den Fehler, aber minimiere auf $V_k = V_k(A, d_0)$. Finde $u_k \in u_0 + V_k$ mit

$$|r_k| = \|Au_k - b\| = \min_{v_k \in u_0 + V_k} \|Av_k - b\| \quad (*)$$

$$V_k \in u_0 + V_k \Rightarrow v_k = u_0 + \sum_{l=0}^{k-1} \alpha_l A^l r_0, \text{ falls } d_0 \sim r_0$$

$$\begin{aligned} \Rightarrow Av_k - b &= \underbrace{Au_0 - b}_{=r_0} + A \cdot \sum_{l=0}^{k-1} \alpha_l A^l r_0 \\ &= \left(Id + A \cdot \sum_{l=0}^{k-1} \alpha_l A^l \right) r_0 \\ &= q(A) r_0 \quad \text{mit einem } q \in \mathbb{P}_k^* \end{aligned}$$

Für das Minimum gilt daher:

$$|r_k| = \min_{q \in \mathbb{P}_k^*} |q(A) \cdot r_0| \leq \min_{q \in \mathbb{P}_k^*} \|q(A)\|_2 |r_0| \quad (**)$$

Daraus gewinnen wir Fehlerabschätzungen

Satz 14. (Fehlerabschätzung für GMRES)

Sei $A \in \mathbb{R}^{n,n}$ regulär, u_k Lösung von

$$|Au_k - b| = \min_{v_k \in u_0 + V_k} |Av_k - b|$$

1.) A diagonalisierbar mit $A = XDX^{-1}$, D diagonal, $X, D \in \mathbb{C}^{n,n}$, so gilt:

$$|r_k| \leq \text{cond}_2(X) \cdot \max_{\lambda \in \text{spec}(A)} |q(\lambda)| |r_0| \quad \forall q \in \mathbb{P}_k^*$$

2.) A normal ($AA^\top = A^\top A$). Dann gilt 1.) mit $\text{cond}_2(X) = 1$

3.) $\|Id - A\|_2 \leq \varrho < 1 \Rightarrow |r_k| \leq |r_0| \varrho^k$

Beweis. 1.) $q(A) = q(XDX^{-1}) = Xq(D)X^{-1}$

Also ist

$$\begin{aligned} \|q(A)\|_2 &\leq \|X\|_2 \cdot \|X^{-1}\|_2 \cdot \|\text{diag}(q(\lambda_1), \dots, q(\lambda_n))\|_2 \\ &\leq \text{cond}_2(X) \cdot \max_{\lambda \in \text{spec}(A)} |q(\lambda)| \end{aligned}$$

Behauptung folgt aus (**)

2.) A normal $\Rightarrow X$ orthonormal $\Rightarrow \text{cond}_2(X) = 1$.

3.) Wähle $q(t) := (1 - t)^k$. Dann $q \in \mathbb{P}_k^*$.

$$\|q(A)\|_2 = \|(Id - A)^k\|_2 \leq \|Id - A\|_2^k \leq \varrho^k$$

$$\Rightarrow \min_{q \in \mathbb{P}_k^*} \|q(A)\|_2 \leq \varrho^k \text{ Behauptung folgt mit (**)}$$

□

Bemerkung Ist $\text{spec}(A) \subset [a, b] \subset \mathbb{R}$ für $0 < a < b$, so kann man die Abschätzung aus 4.5 verwenden mit $\kappa = b/a$

3.5.2 Konstruktion des GMRES-Verfahrens

Schritt 1: Konstruiere eine euklidisch orthonormale Basis des Krylovraumes (Gram-Schmidt)

Start: $d_0 = \frac{r_0}{|r_0|}$ (o.B.d.A. $|r_0| \neq 0$)

Iteration: für $k \geq 0$:

$$\begin{aligned}\sigma_{kj} &= Ad_k \cdot d_j \quad j = 0, \dots, k \\ v_{k+1} &= Ad_k - \sum_{j=0}^k \sigma_{kj} d_j \\ \sigma_{k,k+1} &= |v_{k+1}| \\ d_{k+1} &= \frac{v_{k+1}}{|v_{k+1}|}\end{aligned}$$

Aufwand im k -ten Schritt: $\frac{MV}{1} \mid \frac{VV}{k+3} \mid \frac{SV}{k+2}$ Speicher: $(k+2)n + \mathcal{O}(k^2)$ insgesamt

Der Aufwand über K Schritte ist $\mathcal{O}(K^2) \sim \sum_{k=1}^K \mathcal{O}(k)$.

Speicher: $\mathcal{O}(K)n + \mathcal{O}(K^2)$

Schritt2: Minimierung des Residuums

$$\begin{aligned}|r_k| &= \min_{v_k \in u_0 + V_k} |Av_k - b| \\ &= \min_{z_k \in V_k} \underbrace{|Au_0 - b|}_{=r_0} + |Az_k| \\ &= \min_{z_k \in V_k} |Az_k - \beta_0 d_0| \quad \text{mit } \beta_0 = -|r_0|\end{aligned}$$

Es sei $P_k : V_k \rightarrow \mathbb{R}^k$ die orthonormale Projektion mit $P_k d_{l-1} = \vec{i}_l$

Aus $Ad_k = \sigma_{k,k+1} d_{k+1} + \sum_{j=0}^k \sigma_{kj} d_j$ folgt $A|_{V_{k+1}} \rightarrow V_{k+2}$, d.h. in der Basis $\{d_0, \dots, d_k\}$ hat A „Hessenberggestalt“:

$$A|_{V_k} = \begin{bmatrix} \sigma_{00} & \sigma_{10} & & & \\ \sigma_{01} & \sigma_{11} & & & * \\ & \ddots & \ddots & & \\ & & \ddots & \sigma_{k+1,k+1} & \\ & & & \sigma_{k,k+1} & \end{bmatrix} \in \mathbb{R}^{k+1}$$

Mit $A_k := P_{k+1} A P_k$ folgt

$$|r_k| = \min_{z_k \in V_k} |P_{k+1} (A \underbrace{P_k^\top P_k}_{=Id_{V_k}} z_k - \beta d_0)| = \min_{\omega_k \in \mathbb{R}^k} |A_k \omega_k - \beta_0 \vec{i}_1|$$

Trick: Mittels orthonormaler Matrizen $L_1, \dots, L_K \in \mathbb{R}^{k+1, k+1}$ kann man erreichen, dass $L_k \cdot \dots \cdot L_1 A_k = \begin{bmatrix} R_k & \\ & 0 \end{bmatrix} \in \mathbb{R}^{k+1, k}$ und R_k ist r.o. Dreiecksmatrix. Dann

$$\begin{aligned} |r_k| &= \min_{\omega_k \in \mathbb{R}^k} \left| \underbrace{L_k \cdot \dots \cdot L_1}_{\text{orthonormal}} (A_k \omega_k - \beta_0 \vec{i}_1) \right| \\ &= \min_{\omega_k \in \mathbb{R}^k} \left\| \begin{bmatrix} R_k \omega_k \\ 0 \end{bmatrix} - \begin{bmatrix} b_k \\ \varrho_k \end{bmatrix} \right\| \quad \text{mit } b_k \in \mathbb{R}^k, \varrho_k \in \mathbb{R} \end{aligned}$$

$$\Rightarrow |r_k| = \min_{\omega_k \in \mathbb{R}^k} (|R_k \omega_k - b_k|^2 + \varrho_k^2)^{1/2}$$

Das Minimum wird für $\omega_k = R_k^{-1} b_k$ angenommen ($\text{Rang}(R_k) = \text{Rang}(A_k) = \text{Rang}(A|v_k) = k$, falls $\dim(V_k) = k$) und dann ist $|r_k| = \varrho_k$.

Zwar ist ω_k billig berechenbar ($\mathcal{O}(k^2)$ Multiplikationen, da Dreiecksmatrix), aber ϱ_k ist bekannt ohne ω_k zu kennen! Wir berechnen ω_k erst, wenn ϱ_k klein genug ist oder $k = k_{\max}$ erreicht ist.

Bemerkung:

$$L_j = \begin{bmatrix} 1 & & & & & \\ & \ddots & & & & \\ & & 1 & & & \\ & & & c & s & \\ & & & -s & c & \\ & & & & & 1 \\ & & & & & & \ddots & \\ & & & & & & & 1 \end{bmatrix} \in \mathbb{R}^{k+1, k+1}, \quad j \leq k, \quad c^2 + s^2 = 1$$

Man kann c bestimmen aus der Bedingung, dass der $k+1, k$ -te Eintrag von $L_k(L_{k-1} \cdot \dots \cdot L_1 A_k) = 0$ wird.

Speicher und Anwendung der Matrizen L_j sind $\mathcal{O}(k)$

Algorithmus: Start: $u_0 \in \mathbb{R}^n$, $r_0 = Au_0 - b \neq 0$, $d_0 := r_0/|r_0|$, $b_0 := -|r_0|\vec{i}_1$

Iteration für $k \geq 0$

- Stopp, falls $\varrho_k = |b_{k+1, k+1}| < \text{Tol}$, sonst $k \rightarrow k+1$
- Berechne σ_{kj} für $j = 0, \dots, k$, $d_{k+1}, \sigma_{k, k+1}$
- Berechne $\left[\widetilde{R_{k+1} j} \right]_{j=0, \dots, k} = L_k \cdot \dots \cdot L_1 [\sigma_{kj}]_{j=0, \dots, k}$
 $(L_j \in \mathbb{R}^{k, k} \rightarrow \begin{bmatrix} L_j & 0 \\ 0 & 1 \end{bmatrix} \in \mathbb{R}^{k+1, k+1})$
- Berechne die Rotation L_{k+1}
- Berechne $[R_{k+1} j]_{j=0, \dots, k} = L_{k+1} [\widetilde{R_{k+1} j}]_{j=0, \dots, k}$

- Berechne $\omega_k = R_{k+1}^{-1} b_{k+1}$
 $u = u_0 + \sum_{j=0}^k \omega_{k+1,j} \cdot d_j$

$$\begin{bmatrix} R_k \\ 0 \dots 0 \end{bmatrix} \rightarrow \left[\begin{array}{c|c} R_k & \begin{matrix} \vdots \\ \sigma_{kj} \\ \vdots \end{matrix} \end{array} \right] \xrightarrow[\text{auf letzte Spalte}]{L_k \dots L_1} \left[\begin{array}{c|c} R_k & \begin{matrix} \vdots \\ \tilde{\sigma}_{kj} \\ * \end{matrix} \end{array} \right] \xrightarrow{L_{k+1}} \begin{bmatrix} R_{k+1} \\ \vdots \\ 0 \end{bmatrix}$$

$$\text{Rechte Seite: } \begin{bmatrix} b_k \\ 0 \end{bmatrix} \xrightarrow{L_{k+1}} \begin{bmatrix} \vdots \\ (*) \end{bmatrix}$$

Satz 15. DAS GMRES-Verfahren (in exakter Arithmetik) ist für invertierbare Matrizen durchführbar und erzeugt eine Folge abnehmender Residuen

$$|r_{k+1}| \leq |r_k|$$

(wobei $(r_k = Au_k - b)$ und $r_n = 0$)

Unter geeigneten Voraussetzungen fällt $|r_k|$ streng monoton (siehe Theorem 5.1).

Der Aufwand für k Schritte ist $\mathcal{O}(k^2 N)$

Speicher: $\mathcal{O}(kN) + \mathcal{O}(k^2)$

Beweis. Fehlt: $\dim(V_k) = k$, bzw. $v_{k+1} \neq 0$ im 1. Schritt (GS). Dann ist $Ad_k \in \text{span}\{d_0, \dots, d_k\}$

$$\Rightarrow e_0 = A^{-1}r_0 \sim A^{-1}d_0 \stackrel{\text{wie 4.4}}{\in} \text{span}\{d_0, \dots, d_k\} = V_{k+1}$$

$$\Rightarrow e_{k+1} \in V_{k+1}, e_{k+1} \perp_{A^T A} V_{k+1} \Rightarrow e_{k+1} = 0$$

$$0 \stackrel{!}{=} v_{k+1} = Ad_k + \sum_{j=0}^k \sigma_{kj} \cdot d_j \quad \square$$

Bemerkung

- 1.) In der Praxis darf k nicht zu groß werden GMRES(k_{\max}) bricht nach k_{\max} Schritten ab und startet mit der bis dahin erhaltenen Lösung neu (Restart). Typisch: GMRES(5) bzw. GMRES(25)

- 2.) Es sei $A = \begin{bmatrix} 0 & & & 1 \\ 1 & \ddots & & \\ & \ddots & \ddots & \\ & & 1 & 0 \end{bmatrix}$. Man kann $b, u_0 \in \mathbb{R}^n$ wählen mit $u_0 = u_1 = \dots u_{n-1}$, $u_n = u$ (u die exakte Lösung)

$$\text{Also } |r_0| = \dots = |r_{n-1}| \neq 0, r_n = 0$$

$$\text{spec}(A) = \{\lambda \in \mathbb{C} : \lambda^n = 1\}$$

RESTART-GMRES konv. nicht.

4 Nichtlineare Gleichungen

Sei $D \subset \mathbb{R}^N$ und $F : D \rightarrow \mathbb{R}^N$ beliebig. Gesucht wird $U \in \mathbb{R}^N$ mit

$$F(U) = 0$$

Speziell: $F(U) = AU - b$, $A \in \mathbb{R}^{N,N}$, $b \in \mathbb{R}^N$ lineares Problem

4.1 Fixpunkte (Ergänzung 5)

4.1.1 Fixpunkte und Nullstellen

U Fixpunkt von G : $U = G(U)$

U Nullstelle von F : $F(U) = 0$

U Fixpunkt von $G \Leftrightarrow U$ Nullstelle von $F(X) := X - G(X)$

4.1.2 Banachscher Fixpunktsatz

Sei V ein Banach-Raum, $D \subseteq V$ abgeschlossen, $f : D \rightarrow D$ eine Kontraktion, d.h. $\exists q \in (0, 1)$ mit

$$\|f(x) - f(y)\| \leq q\|x - y\| \quad (x, y \in D)$$

Dann gilt:

- (i) f besitzt genau einen Fixpunkt x_* in D
- (ii) Zu jedem $x_0 \in D$ konvergiert die durch $x_{i+1} := f(x_i)$ definierte Folge gegen x_* und es gelten die Abschätzungen

$$\|x_i - x_*\| \leq q^i \|x_0 - x_*\| \quad (\text{A priori Abschätzung})$$

$$\|x_i - x_*\| \leq \frac{q}{1-q} \|x_i - x_{i-1}\| \quad (\text{A posteriori Abschätzung})$$

4.1.3 Beispiele

- 1.) $f : [a, b] \subseteq \mathbb{R} \rightarrow [a, b]$ differenzierbar mit $|f'(x)| \leq q < 1 \forall x \in [a, b]$ für ein $q \in (0, 1)$
 $\Rightarrow \exists! x_* \in [a, b] : f(x_*) = x_*$ und die Fixpunktiteration $x_{i+1} := f(x_i)$ konvergiert für die Startwerte $x_0 \in [a, b]$.

- 2.) Suche Lösung von $x = \cos(x)$:

$$x_0 \in \mathbb{R}, \quad x_{i+1} = \cos(x_i)$$

Bildchen

Wende (1) an

$$\max_{x \in \mathbb{R}} |\cos'(x)| = \max_{x \in \mathbb{R}} |\sin(x)| = 1$$

So geht es noch nicht.

Aber: $V = [0, 1]$. Dann

$$\max_{x \in [0, 1]} |\sin(x)| = \sin(1) < 1$$

$\cos(V) \subset V$. Anwendung von (1) ist OK.

$x_0 \in \mathbb{R} \Rightarrow x_1 = \cos(x_0) \in [-1, 1] \Rightarrow x_2 = \cos(x_1) \in [0, 1]$

Jetzt weiter wie eben. Konvergenz für alle $x_0 \in \mathbb{R}$

Satz 16. V Banach-Raum, $D \subset V$ abgeschlossen, $f : D \rightarrow D$ eine Kontraktion mit Rate q der Fixpunktiteration und Fixpunkt v_x . $g : D \rightarrow D$ sei eine Störung von f mit

$$\|f(v) - g(v)\|_V \leq \varepsilon \quad \forall v \in D$$

Definiere $\{v_i\}_i, \{w_i\}$ durch $v_{i+1} := f(v_i)$, $w_{i+1} := g(w_i)$ für $v_0, w_0 \in D$ und $\|v_0 - w_0\|_V \leq \varepsilon$. Dann gilt:

$$\begin{aligned} \|v_i - w_i\|_V &\leq \frac{\varepsilon}{1-q} \\ \|v_* - w_i\|_V &\leq \frac{1}{1-q} (\varepsilon(1+3q^i) + q^i \|w_0 - g(w_0)\|_V) \end{aligned}$$

Bildchen

Beweis. $v_0 \in D \Rightarrow v_1 \in D \Rightarrow \dots$

$w_0 \in D \Rightarrow w_1 \in D \Rightarrow \dots$

Folgen sind wohldefiniert

$$\begin{aligned} \|v_{i+1} - w_{i+1}\|_V &= \|f(v_i) - g(w_i)\|_V \\ &\leq \|f(v_i) - f(w_i)\|_V + \|f(w_i) - g(w_i)\|_V \\ &\leq q \cdot \|v_i - w_i\|_V + \varepsilon \\ &\leq q^2 \cdot \|v_{i-1} - w_{i-1}\|_V + (1+q)\varepsilon \\ &\leq \dots \leq q^{i+1} \underbrace{\|v_0 - w_0\|_V}_{\leq \varepsilon} + \sum_{j=0}^i q^j \varepsilon \\ &\leq \sum_{j=0}^{i+1} q^j \varepsilon \leq \sum_{j=0}^{\infty} q^j \varepsilon = \frac{1}{1-q} \varepsilon. \end{aligned}$$

Mit dem Fixpunktsatz von Banach:

$$\begin{aligned} \|v_* - w_i\|_V &\leq \|v_* - v_i\|_V + \|v_i - w_i\|_V \\ &= \frac{q^i}{1-q} \|v_0 - f(v_0)\|_V + \frac{\varepsilon}{1-q} \\ &\leq \frac{q^i}{1-q} (\underbrace{\|v_0 - w_0\|_V}_{\leq \varepsilon} + \|w_0 - g(w_0)\|_V + \underbrace{\|g(w_0) - f(v_0)\|_V}_{\substack{= \|w_1 - v_1\|_V \\ \leq (1+q)\varepsilon \leq 2\varepsilon}}) + \frac{\varepsilon}{1-q} \end{aligned}$$

□

Problem: Wie schnell sind Fixpunktverfahren?

4.1.4 Konvergenzordnung

V Banach-Raum, $\{v_i\}_i$ eine iterative erzeugte Folge mit $\lim_{i \rightarrow \infty} v_i = v_*$. Die Iteration hat *Konvergenzordnung* $p \geq 1$, falls für den Fehler $e_i := v_i - v_*$ gilt:

$$\lim_{i \rightarrow \infty} \frac{\|e_i\|_V}{\|e_{i-1}\|_V^p} = c \in \mathbb{R}$$

Falls $c \neq 0$, so heißt p die *genaue Konvergenzordnung* und c heißt *asymptotischer Fehlerkoeffizient*.

Beispiele

$p = 1$: Geometrische oder lineare Konvergenz

$p = 2$: Quadratische Konvergenz.

Satz 17. $I \subseteq \mathbb{R}$, $\Phi : I \rightarrow \mathbb{R}$ habe einen Fixpunkt $x_* \in I$ und sei p -mal stetig db. mit

$$\Phi'(x_*) = \dots = \Phi^{(p-1)}(x_*) = 0 \text{ falls } p > 1$$

oder

$$|\Phi'(x_*)| < 1 \text{ falls } p = 1 \text{ ist}$$

Dann konvergiert das Iterationsverfahren

$$x_{i+1} = \Phi(x_i)$$

für die Startwerte x_0 nahe x_* und hat bzgl. $|\cdot|$ die Konvergenzordnung p .

Ist $\Phi^{(p)}(x_*) \neq 0$, so ist p die genaue Konvergenzordnung.

Beweis. Nach Voraussetzung gibt es für alle $p \geq 1$ eine Umgebung von x_* , in der $|\Phi'| < 1$ gilt. Nach 1.3(1) konvergiert die Fixpunktiteration für alle Startwerte dieser Umgebung gegen x_* .

Mit Taylorentwicklung:

$$x_{i+1} = \Phi(x_i) = \sum_{l=0}^{p-1} \frac{1}{l!} \Phi^{(l)}(x_*) (x_i - x_*)^l + \frac{1}{p!} \Phi^{(p)}(\xi_i) (x_i - x_*)^p$$

(ξ_i zwischen x_* und x_i).

Einsetzen der Voraussetzung:

$$x_{i+1} = x_* + \frac{1}{p!} \Phi^{(p)}(\xi_i) (x_i - x_*)^p$$

und somit

$$\lim_{i \rightarrow \infty} \frac{|x_{i+1} - x_*|}{|x_i - x_*|^p} = \lim_{i \rightarrow \infty} \frac{1}{p!} |\Phi^{(p)}(\xi_i)| = \frac{1}{p!} |\Phi^{(p)}(x_*)|$$

□

Bemerkung: Lineare vs. Quadratische Konvergenz.

$$e_0 = 10^{-1}$$

Lineare Konvergenz: $q = 1/2$, $e_k = \left(\frac{1}{\alpha}\right)^k e_0 \approx 10^{-0.3k} e_0$

1 Stelle \rightsquigarrow 3 Iterationen

8 Stellen \rightsquigarrow 24 Iterationen

Quadratische Konvergenz: $c = 1$

$$e_0 = \frac{1}{10}, e_1 = e_0^2 = 10^{-2}, e_2 = 10^{-4}, e_3 = 10^{-8}$$

4.2 Berechnung von Nullstellen

4.2.1 Extrema (Ergänzung 7)

x_* Extremum von f und f db $\Rightarrow f'(x_*) = 0$

\rightsquigarrow Nullstellenproblem

4.2.2 Nullstellen reeller Funktionen

Im Folgenden sei $I = [a, b] \subset \mathbb{R}$, $a < b$, f mindestens stetig.

Bisektionsverfahren Es gelte $f(a)f(b) < 0$ („ $\neq 0$ “ $\Rightarrow f(a) = 0$ oder $f(b) = 0$).

Wir konstruieren Intervalle $\{I_k\}_k$ wie folgt:

Start:

$$a_0 := a, b_0 := b, I_0 := [a_0, b_0]$$

Iteration: $L \geq 0$

$$1.) \quad \bar{x} := \frac{1}{2}(a_k + b_k)$$

$$2.) \quad \text{Stop: } f(\bar{x}) = 0$$

$$3.) \quad f(a_k) \cdot f(\bar{x}) \stackrel{?}{<} 0 : a_{k+1} = a_k, b_{k+1} = \bar{x} \\ \text{sonst: } a_{k+1} = \bar{x}, b_{k+1} = b_k$$

$$4.) \quad k \mapsto k + 1, I_{k+1} = [a_{k+1}, b_{k+1}]$$

Abbruch: $\text{Tol}_X, \text{Tol}_f \geq 0$ gegeben, $\text{Tol}_x + \text{Tol}_f > 0$

$k_{\max} \in \mathbb{N}$. Rückgabe x und $f(x)$ mit

x Approximation der Nullstelle mit $|x - x_*| \leq \text{Tol}_x$ oder $|f(x)| \leq \text{Tol}_f$ $f(x)$: Funktionswert in x

Modifikation der Iteration:

$$|f(\bar{x})| \leq \text{Tol}_f :$$

return($\bar{x}, f(\bar{x})$);

$$|b_k - a_k| \leq \text{Tol}_x :$$

falls $|f(a_k)| < |f(b_k)|$ return ($a_k, f(a_k)$), sonst return($b_k, f(b_k)$)

Satz 18. $f : [a, b] \rightarrow \mathbb{R}$ stetig mit $f(a) \cdot f(b) < 0$. $\text{Tol}_x, \text{Tol}_f, k_{\max}$ wie oben gegeben.
Dann bricht das Bisektionsverfahren nach endlich vielen Schritten ab, auch falls $k_{\max} = \infty$

Beweis. Das Verfahren ist wohldefiniert aufgrund des Zwischenwertsatzes.
Die Existenz einer Nullstelle in I_k ist für jedes k gesichert.

$$\text{Tol}_x > 0 : |I_k| = \left(\frac{1}{2}\right)^k |b - a| \stackrel{!}{\leq} \text{Tol}_x \Rightarrow k \leq \left\lceil \frac{\log_2(b-a)}{\text{Tol}_x} \right\rceil$$

$$\text{Tol}_f > 0 : b_k - a_k \rightarrow 0$$

Da f stetig ist und eine Nullstelle in $[a_k, b_k]$ hat, gilt $\lim_{k \rightarrow \infty} f(a_k) = \lim_{k \rightarrow \infty} f(b_k) = 0$

$$\Rightarrow \exists k_f \in \mathbb{N} : \min\{|f(a_{k_f})|, |f(b_{k_f})|\} \leq \text{Tol}_f$$

(Gilt $|f'(x)| \leq C \forall x \in [a, b]$, so gilt z.B.: $|f(a_k)| = |f(a_k) - f(x_k)| \leq |I_k| \max_{x \in [a, b]} |f'(x)| \leq$

$$C \left(\frac{1}{2}\right)^k \stackrel{!}{\leq} \text{Tol}_f)$$

□

Probleme

- a, b zu finden mit $f(a) \cdot f(b) < 0$ kann sehr schwierig sein.
- Die Konvergenz ist in der Praxis zu langsam.
(Siehe 1.5: Konvergenzordnung ist 1 mit $c = \frac{1}{2}$)
- Die Methode ist auf \mathbb{R} beschränkt

Regula Falsi Wie in 2.2.1 aber mit \bar{x} wie folgt:
Bildchen

$$\bar{x} = a_k - \frac{f(a_k)(b_k - a_k)}{f(b_k) - f(a_k)}$$

Keine Auslöschung im Nenner wegen $f(a_k) \cdot f(b_k) < 0$. Weiteres Vorgehen wie in 2.2.1
Konvergenz: Konvergiert wie in 2.2.1 im Fall $\text{Tol}_f > 0$. Die Konvergenz kann beliebig langsam sein. Im „besten“ Fall ist die Konvergenz linear (unter noch allgemeinen Voraussetzungen)

Das Sekantenverfahren Bildchen

$f : \mathbb{R} \rightarrow \mathbb{R}$ stetig.

x_1, x_2 gegeben, $x_1 \neq x_2$ und $f(x_1) \neq f(x_2)$

x_3 ist dann die Nullstelle der Sekante

Initialisierung: $x_1 \neq x_2, f(x_1) \neq f(x_2)$

Iteration für $k \geq 0$:

1.) Falls $f(x_{k-1}) \neq f(x_k)$

$$x_{k+1} = x_k - \frac{f(x_k)(x_k - x_{k-1})}{f(x_k) - f(x_{k-1})}$$

2.) $k \leadsto k + 1$

Abbruch: $\text{Tol}_x, \text{Tol}_f, \text{Tol}_{f'}, k_{\max}$

Wie in 2.2.1 aber mit

$$\begin{aligned} |x_k - x_{k-1}| &\leq \text{Tol}_x? \\ |f(x_k)| &\leq \text{Tol}_f? \\ k &\leq k_{\max} \\ \text{und } |f(x_k) - f(x_{k-1})| &\leq \text{Tol}_{f'}? \end{aligned}$$

Die letzten beiden Bedingungen führen zu einem erfolglosen Abbruch.

Bemerkungen

- Keine Erfolgsgarantie für allgemeine Startwerte
- Kleine f -Differenzen erzeugen große Fehler

Aber:

- Günstiger Aufwand (1 f -Auswertung pro Schritt) bei schneller Konvergenz, falls es konvergiert.
- Gewisse Verallgemeinerung auf \mathbb{R}^N möglich

Satz 19. $f \in C^2(\mathbb{R})$, $f(x_*) = 0$, $f'(x_*) \neq 0$, $f''(x_*) \neq 0$.

Dann ex. eine Umgebung U von x_* , sodass das Sekantenverfahren für alle Startwerte aus U konvergiert und die Konvergenzordnung ist genau $\frac{1}{2}(1 + \sqrt{5}) \approx 1.6$

Newton-Verfahren $f : \mathbb{R} \rightarrow \mathbb{R}$ stetig db.

Idee: Verwende Tangente statt Sekante

Bildchen

Initialisierung: x_1 mit $f'(x_1) \neq 0$

Iteration: für $k \geq 0$

1.) Falls $f'(x_k) \neq 0$

$$x_{k+1} = x_k - \frac{f(x_k)}{f'(x_k)}$$

2.) $k \leadsto k+1$

Abbruch: $\text{Tol}_x, \text{Tol}_f, k_{\max}, \text{Tol}_{f'}$

$$\begin{aligned} |x_k - x_{k-1}| &\leq \text{Tol}_x \\ |f(x_k)| &\leq \text{Tol}_f \\ k &\leq k_{\max} \\ |f'(x_k)| &\leq \text{Tol}_{f'} \end{aligned}$$

In den letzten beiden Fällen ist der Abbruch erfolglos

Bemerkungen

- Keine Garantie eines erfolgreichen Abbruchs (im Allgemeinen)
- Kleine Werte von f' führen zu großen Fehlern

Aber:

- sehr schnell, falls konvergent
- Verallgemeinerung auf \mathbb{R}^N bzw. Banachräume möglich

Konvergenzordnung des Newton-Verfahrens

$f \in C^3$, $f(x_*) = 0$, $f'(x_*) \neq 0$

Die Iterationsfunktion des Newton-Verfahrens ist

$$\Phi(x) := x - \frac{f(x)}{f'(x)}$$

Nach 1.5 bilden wir $\Phi'(x_*)$, $\Phi''(x_*)$

$$\begin{aligned} \Phi'(x) &= 1 - \left(1 - \frac{f(x)f''(x)}{f'(x)^2}\right) = \frac{f(x)f''(x)}{f'(x)^2} \stackrel{x=x_*}{=} 0 \\ \Phi''(x) &= \frac{f''(x)}{f'(x)} + f(x)(\dots) \stackrel{x=x_*}{=} \frac{f''(x_*)}{f'(x_*)} + 0 \end{aligned}$$

Die Konvergenz ist quadratisch und sie ist genau quadratisch, falls $f''(x_*) \neq 0$

4.2.3 Lokale Konvergenz des Newtonverfahrens

Es sei $(V, \|\cdot\|_V)$ ein Banachraum, $\emptyset \neq U \subset V$, $f : U \rightarrow V$ eine stetig db. Funktion mit $f'(v)^{-1} \in \mathbb{L}(V, V)$ für alle $v \in U$ sowie

$$\sup_{v \in U} \|f'(v)^{-1}\|_{\mathbb{L}(V, V)} \leq K < \infty$$

und

$$\|f'(v) - f'(w)\|_{\mathbb{L}(V, V)} \rightarrow 0 \quad (\|v - w\|_V \rightarrow 0) \text{ glm. für } v, w \in U.$$

Weiter sei $u_* \in U$ eine Nullstelle von f . Dann gibt es zu jedem $q \in (0, 1)$ ein $\delta > 0$, so dass für jeden Startwert $u_0 \in B_\delta(u_*)$ die Newton-Iteration $u_{i+1} = u_i - f'(u_i)^{-1}f(u_i)$ wohldefiniert ist und für $i \geq 0$ gilt

$$\|u_i - u_*\|_V \leq q \|u_0 - u_*\|_V$$

Ist f zweimal stetig db, so ist die Konvergenz quadratisch:

$$\|u_{i+1} - u_*\|_V \leq C \|u_i - u_*\|_V^2$$

für $i \geq 0$ und ein $C > 0$. C hängt von f ab.

Insbesondere bricht das Verfahren nach endlich vielen Schritten bzgl. der Kriterien

$$\begin{aligned} \|u_i - u_{i-1}\|_V &\stackrel{!}{\leq} \text{Tol}_x \quad \text{oder} \\ \|f(u_i)\|_V &\leq \text{Tol}_f \end{aligned}$$

für $\text{Tol}_x, \text{Tol}_f \geq 0$, $\text{Tol}_x + \text{Tol}_f > 0$ ab

Bemerkung $T : V \rightarrow V$ linear, stetig ($:\Leftrightarrow T \in \mathbb{L}(V, V)$),

$$\|T\|_{\mathbb{L}(V, V)} := \sup_{v \in V} \frac{\|Tv\|_V}{\|v\|_V}$$

Beweis. Sei $r_0 > 0$ mit $\overline{B_{r_0}(u_*)} \subset U$. Dann gilt für $u \in B_r(u_*)$ ($0 < r < r_0$)

$$f(u) = f(u_*) + \int_0^1 f'(u_* + t(u - u_*))(u - u_*) dt$$

Die Iterationsfunktion des Newton-Verfahrens ist

$$G(u) := u - f'(u)^{-1} \cdot f(u)$$

G ist auf $B_{r_0}(u_*)$ wohldefiniert und mit $u(t) := u_* + t(u - u_*)$ gilt

$$\begin{aligned} G(u) - u_* &= u - u_* - f'(u)^{-1} \int_0^1 f'(u(t))(u - u_*) dt \\ &= \int_0^1 f'(u)^{-1} (f'(u) - f'(u(t)))(u - u_*) dt \end{aligned}$$

Daher:

$$\|G(u) - u_*\|_V \leq \sup_{v \in B_r(u_*)} \|f'(v)^{-1}\|_{\mathbb{L}(V,V)} \cdot \sup_{t \in (0,1)} \|f'(u) - f'(u(t))\|_{\mathbb{L}(V,V)} \cdot \|u - u_*\|_V$$

Zu $q \in (0,1)$ wähle also δ , so dass

$$\|G(u) - u_*\|_V \leq q \cdot \|u - u_*\|_V \text{ für alle } u \in B_\delta(u_*)$$

Für $u_0 \in B_\delta(u_*)$ folgt also induktiv

$$\|u_{i+1} - u_*\|_V = \|G(u_i) - u_*\|_V \leq q \cdot \|u_i - u_*\|_V \leq \delta$$

d.h. $\{u_i\}_i \in B_\delta(u_*)$ und $\lim_{i \rightarrow \infty} u_i = u_*$. Insbesondere

$$\|u_i - u_*\|_V \leq q^i \|u_0 - u_*\|_V$$

Ist f zweimal stetig db, so gilt:

$$\begin{aligned} \sup_{t \in (0,1)} \|f'(u) - f'(u(t))\|_{\mathbb{L}(V,V)} &\leq C' \|u - u(t)\|_V \\ &\leq C' \|u - u_*\|_V \quad \text{mit } C' = C'(f'') \end{aligned}$$

Also

$$\|G(u) - u_*\|_V \leq KC' \|u - u_*\|_V^2 = C \|u - u_*\|_V^2$$

$$\Rightarrow \|u_{i+1} - u_*\|_V \leq C \|u_i - u_*\|_V^2$$

Mit $\|u_{i+1} - u_i\|_V \leq \|u_{i+1} - u_*\|_V + \|u_i - u_*\|_V \leq 2 \cdot \|u_i - u_*\|_V$.

Also $\|u_{i+1} - u_i\|_V \rightarrow 0$ und mit Stetigkeit $\|f(u_i)\|_V \rightarrow 0$ für $i \rightarrow \infty$. Daraus folgt der Abbruch nach endlich vielen Schritten. \square

Bemerkungen:

- f' invertierbar heißt, dass u_* eine einfache Nullstelle ist
- u Nullstelle von f . Dann sei $\varepsilon(u)$ der Einzugsbereich von u , d.h. $u_0 \in \varepsilon(u) \Rightarrow$ das Newton-Verfahren ist wohldefiniert für u_0 und die Folge $\{u_i\}_{i \geq 0}$ konvergiert gegen u .
Der vorherige Satz sagt: $B_\delta(u) \subseteq \varepsilon(u)$ für δ klein (unter genannten Voraussetzungen)

Beispiel $V = \mathbb{R}$, $f(x) = \arctan(x)$

Bildchen

$$\begin{aligned} f(0) &= 0 \\ |x_0| < X_0 &\Rightarrow x_i \rightarrow 0 \\ |x_0| > X_0 &\Rightarrow |x_i| \rightarrow \infty \\ x_0 = X_0 &\Rightarrow x_i = (-1)^i \cdot x_0 \end{aligned}$$

Für $V = \mathbb{R}^n$, $n \geq 2$ ist $\varepsilon(u)$ sehr kompliziert.

Wir berechnen für große Raumdimension n $f(u_i)^{-1}$ nicht explizit. Stattdessen lösen wir

$$\begin{aligned} f'(u_i)d_i &= -f(u_i) \\ u_{i+1} &= u_i + d_i \end{aligned}$$

Newton-Kantorovich-Theorem $F : D \subset V \rightarrow V$, V Banachraum, D offen und konvex, F stetig db, $x_0 \in D$ und $F'(x_0)$ invertierbar sowie

$$\begin{aligned} \|F'(x_0)^{-1}F(x_0)\| &\leq \alpha \\ \|F'(x_0)^{-1}(F'(y) - F'(x))\|_{\mathbb{L}(V,V)} &\leq \omega_0 \cdot \|x - y\|_V \quad \forall x, y \in D \\ h_0 := \alpha\omega_0 &< 1/2 \\ B_\delta(x_0) &\subset D, \quad \delta := \frac{1}{\omega_0}(1 - (1 - 2h_0)^{1/2}) \end{aligned}$$

Dann ist die Folge $\{x_k\}_k$ der Newton-Iteration wohldefiniert, sie bleibt in $B_\delta(x_0)$ und konvergiert gegen ein x_* mit $F(x_*) = 0$. Die Konvergenz ist quadratisch.

Bemerkung

- Die Existenz der Nullstelle wird garantiert. Daher sind solche Theoreme auch in der Analysis interessant.
- Man kann (wie bei Banach) a priori Schranken oder a posteriori Schranken betrachten
- Beachte: $F(u) = 0 \Leftrightarrow AF(u) = 0$, falls A invertierbar ist. Wie in 2.4.1, 2.4.2 hängen die Konstanten von A ab. Die Größe $F'^{-1}F$ ist invariant gegenüber der Transformation $F \mapsto AF$

4.2.4 Globale Konvergenz

Idee: Definiere eine „Energie“, die in jedem Schritt verkleinert wird:
für ein $E : V = \mathbb{R}^n \rightarrow \mathbb{R}$ gelte

$$|u_{i+1}| = |u_i - f'(u_i)^{-1}f(u_i)| = E(u_{i+1}) < E(u_i)$$

Problem: u_{i+1} sollte nicht zu weit weg sein von u_i . Ausweg (siehe Jakobi- oder SOR-Verfahren): Dämpfung.

Für $\tau_i > 0$ ist $u_{i+1} = u_i - \tau_i f'(u_i)^{-1}f(u_i)$ das gedämpfte Newton-Verfahren.

„ i klein“: $\tau_i \in (0, 1)$ klein

„ i groß“: $\tau_i \rightarrow 1$ um von der quadratischen Konvergenz zu profitieren. ($\tau \neq 1$: gedämpftes Newton-Verfahren konvergiert nur linear)

Lemma 4. $\emptyset \neq D \subset \mathbb{R}^n$ abgeschlossen und beschränkt. $f \in C^1(D, \mathbb{R}^n)$ und $f'(u)^{-1}$ existiere für alle $u \in D$. $|\cdot|$ eine Vektornorm.

Definiere $E : D \rightarrow \mathbb{R}$, $u \mapsto E(u) = |f(u)|$ mit $d(u) := -f'(u)^{-1} \cdot f(u)$. Dann gilt:
Für alle $\varepsilon > 0$ existiert ein $\delta > 0$ mit

$$E(u + \tau d(u)) \leq (1 - \tau + \varepsilon \tau) E(u) \quad \text{für alle } u \in D, \tau \in (0, \delta)$$

Beweis. Für $u \in D$:

$$\begin{aligned} f(u + \tau d(u)) &= f(u) + \int_0^\tau f'(u + sd(u)) d(u) \, ds \\ &= \left(Id - \int_0^\tau f'(u + sd(u)) f'(u)^{-1} \, ds \right) f(u) \\ &= \left((1 - \tau) Id - \int_0^\tau (f'(u + sd(u)) - f'(u)) f'(u)^{-1} \, ds \right) f(u) \end{aligned}$$

τ genügend klein:

$$|f(u + \tau d(u))| \leq (1 - \tau + \underbrace{\tau \sup_{s \in (0, \tau)} \|f'(u + sd(u)) - f'(u)\|_2}_{\leq C^{-1} \cdot \varepsilon, \text{ falls } \tau \leq \delta} \cdot \underbrace{\|f'(u)^{-1}\|_2}_{\leq C}) \cdot |f(u)|$$

$$\Rightarrow E(u + \tau d(u)) \leq (1 - \tau + \varepsilon \tau) E(u) \quad \square$$

Schrittweitensteuerung f wie in 2.3, E wie oben. Wähle ein $\sigma \in (0, 1)$ und $u_0 \in D$.
Newton-Verfahren mit Schrittweitensteuerung

Initialisierung: $u_0 \in D$

Iteration: für $k \geq 0$

- 1.) Löse $f'(u_k) d_k = -f(u_k)$ für d_k
- 2.) Bestimme $\tau_k = 2^{-q_k}$ und $q_k \in \mathbb{N}$ minimal mit $B_{\tau_k |d_k|}(u_k) \subset D$ und $E(u_k + \tau_k d_k) \leq (1 - \sigma \tau_k) E(u_k)$
- 3.) $u_{k+1} = u_k + \tau_k d_k$, gehe zu (1)

Wahl des Wertes q_k

$k = 0$: $q = 0, 1, \dots$ bist die Bedingung in (2) für ein q_0 zum ersten Mal erfüllt ist.

$k > 0$: Probiere $q = q_{k-1} - 1, q_{k-1}, \dots$ bist (2) für ein q_k zum ersten Mal erfüllt ist.

Globale Konvergenz

Satz 20. *f wie im Lemma in 2.4.1 bzgl. eines D_α .*

Zu $\alpha > 0$ sei $D_\alpha := \{v \in D : |f(v)| \leq \alpha\}$ nichtleer und kompakt. (f darf nur eine Nullstelle haben und muss glm konvergieren)

Dann konvergiert das Verfahren aus 2.4.1 für alle Startwerte $u_0 \in D_\alpha$ gegen eine Nullstelle von f in D_α .

Insbesondere folgt der Abbruch nach endlich vielen Schritten bzgl. des Kriteriums $E(u_k) \leq \text{Tol}_f$ für ein $\text{Tol}_f > 0$

Beweis. *Nach Konstruktion gilt:*

$$E(u_{[k+1]}) \leq E(u_k) \leq \dots \leq E(u_0) = \alpha$$

und $\{u_k\}_k \subseteq D_\alpha$.

Die Folge konvergiert daher, weil D_α kompakt ist, etwa $u_k \rightarrow u_(k \rightarrow \infty)$ für eine Teilfolge. Nach dem Lemma gibt es zu jedem $\varepsilon > 0$ ein $\delta > 0$, so dass*

$$|f(u_k + \tau d(u_k))| \leq (1 - (1 - \varepsilon)\tau)|f(u_k)|$$

für $0 \leq \tau \leq \delta$, gleichmäßig in D_α .

Nun sei $\varepsilon := 1 - \sigma$, d.h.

$$|f(u_k + \tau d(u_k))| \leq (1 - \sigma\tau)|f(u_k)|$$

Diese Ungleichung gilt für $\tau = \delta$, d.h. nach Konstruktion gilt $\tau_k \geq \delta/2$. Insbesondere erhalten wir nach endl. vielen Schritten

$$|f(u_{k+1})| = |f(u_k + \tau_k d_k)| \leq (1 - \frac{1}{2}\delta\sigma)|f(u_k)|,$$

also $E(u_{k+1}) \leq \kappa E(u_k)$ für ein $\kappa \in (0, 1)$, so dass $\lim_{k \rightarrow \infty} E(u_k) = 0$. Insbesondere wird

$E(u_k) = |f(u_k)| \stackrel{!}{\leq} \text{Tol}_f$ nach endlich vielen Schritten erreicht. □