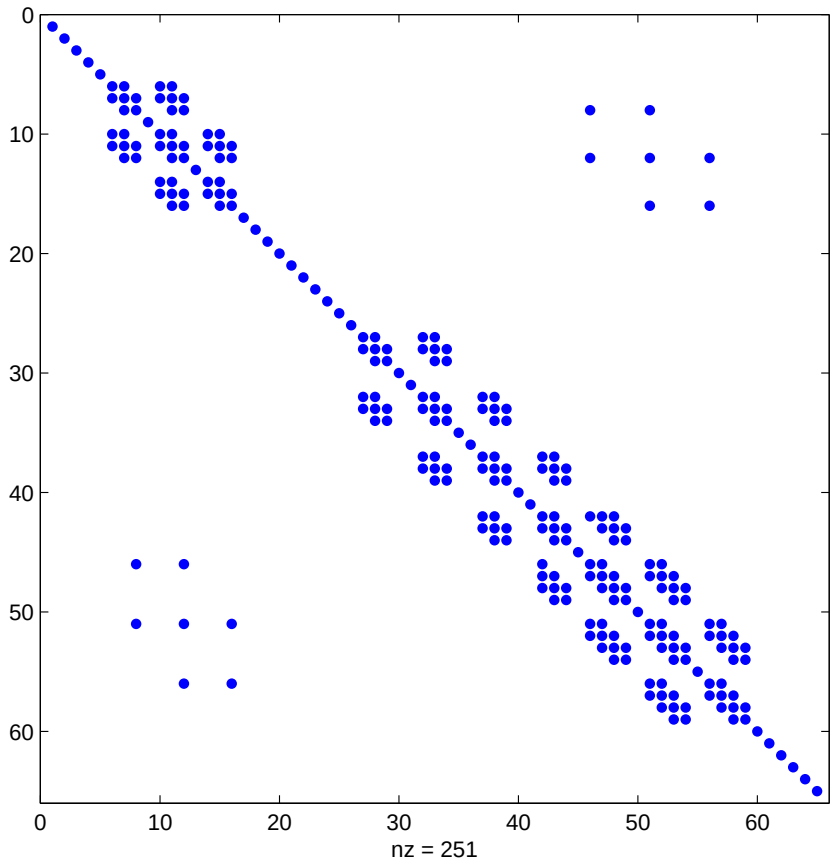
4 Die Lösung der diskreten Poisson-Gleichung

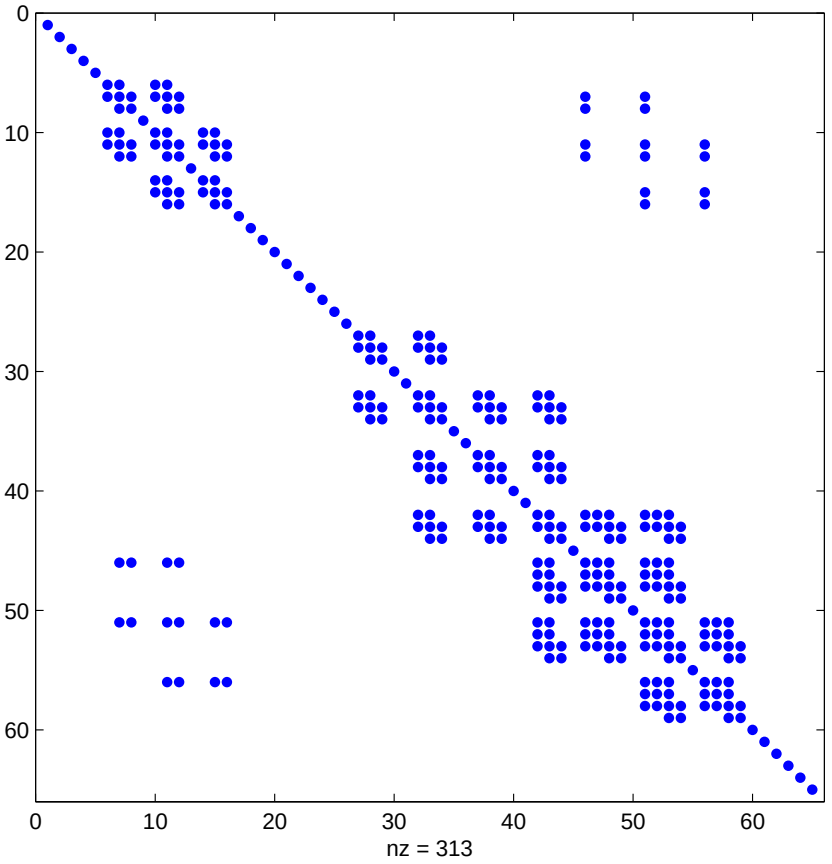
Die Galerkin-Matrix der FE-Diskretisierung der Poisson-RWA (3.1) ist symmetrisch und positiv-definit (gilt nicht notwendig für andere Dgln.)

Allen FE-Diskretisierungen gemeinsam: die Galerkin-Matrix ist dünn besetzt (sparse). Wir betrachten die Matrix aus Beispiel 2 (L-Gebiet).

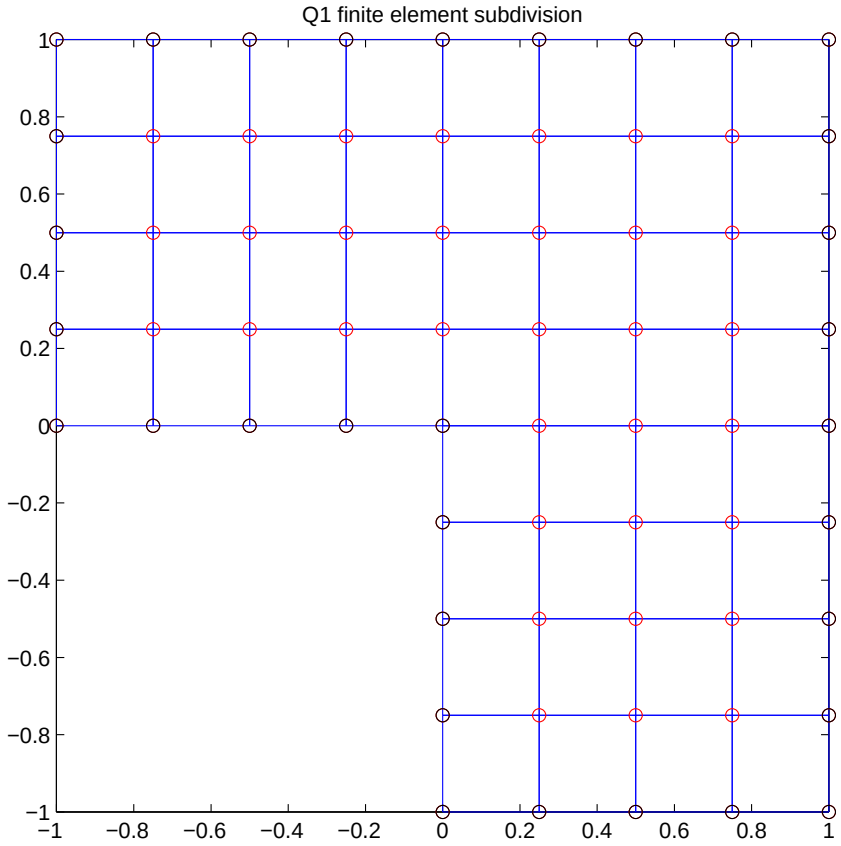
	bilinear			biquadratisch		
N	h	# Einträge $\neq 0$	Anteil	h	# Einträge $\neq 0$	Anteil
65	0.25	251	5.94%	0.5	313	7.4%
225	0.125	1339	2.64%	0.25	2073	4.1%
883	0.0625	6107	0.78%	0.125	10141	1.5%
3201	0.0312	26011	0.25%	0.0625	44767	0.44%
12545	0.0156	107291	0.068%	0.0312	187742	0.12%



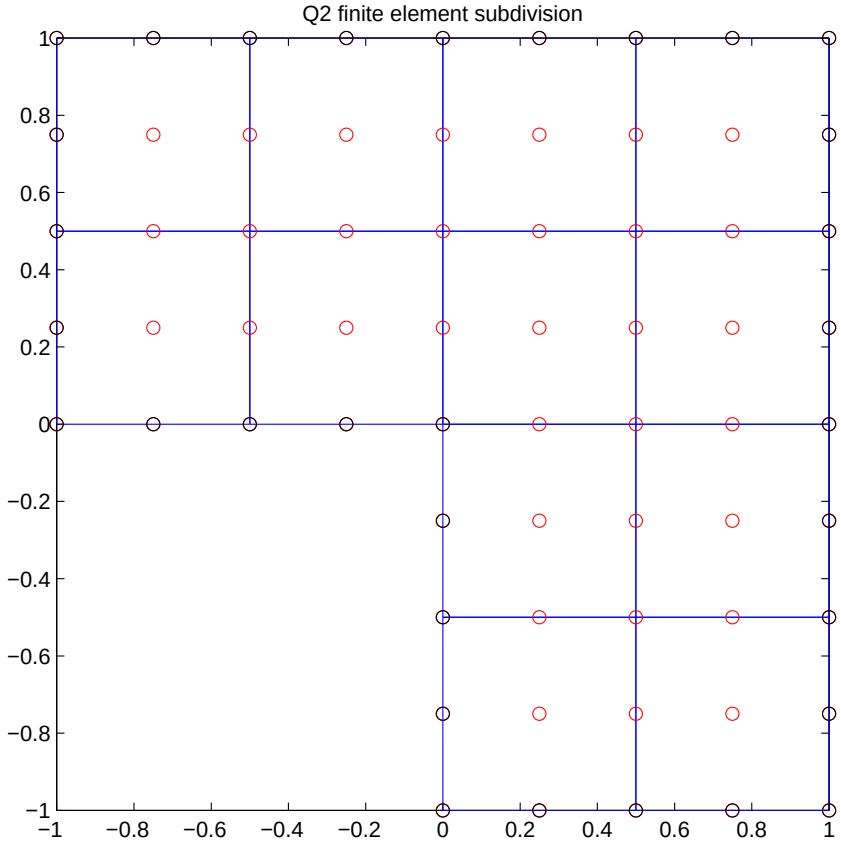
$h = 0.25$, bilineare Elemente



$h = 0.5$, biquadratische Elemente



$h = 0.25$, bilineare Elemente



$h = 0.5$, biquadratische Elemente

Direkte oder iterative Verfahren ?

- In der FE-Praxis wurde lange Zeit mit direkten Verfahren, also Varianten der Gauß-Elimination, gearbeitet.
- Stark verbreitet: frontal solvers. Hier wird die Faktorisierung der Matrix so früh wie möglich während der Assemblierung vorgenommen.
- Weiterentwicklungen auch heute noch relevant, auch als eigenständige Eliminationsverfahren, z.B. UMFPACK.
- Faustregel: in 2D reichen ausgereifte direkte, auf dünn-besetzte Matrizen spezialisierte Löser aus. Für 3D-Diskretisierungen muss auf Iterationsverahren zurückgegriffen werden.
- Vorteil direkter Verfahren: typischerweise als „black box“ einsetzbar. Bei Iterationsverfahren ist deutlich mehr Expertise des Anwenders erforderlich.

4.1 Erste Iterationsverfahren

Zunächst sei allgemein gegeben

$$\boldsymbol{A}\boldsymbol{x} = \boldsymbol{b}, \qquad \boldsymbol{A} \in \mathbb{C}^{n\times n} \text{ invertierbar}, \boldsymbol{b} \in \mathbb{C}^n. \tag{4.1}$$

Grundidee:

Startnäherung $\boldsymbol{x}_0 \approx \boldsymbol{A}^{-1}\boldsymbol{b}$,

Erzeuge rekursiv Folge $\{\boldsymbol{x}_m\}_{m\in\mathbb{N}}$ mit $\lim\limits_{m\to\infty} \boldsymbol{x}_m = \boldsymbol{A}^{-1}\boldsymbol{b}$.

Bezeichnungen:

$\boldsymbol{r}_m := \boldsymbol{b} - \boldsymbol{A}\boldsymbol{x}_m$	Residuum, Defekt,
$\boldsymbol{x}^\star := \boldsymbol{A}^{-1}\boldsymbol{b}$	(exakte) Lösung,
$\boldsymbol{e}_m := \boldsymbol{x}^\star - \boldsymbol{x}_m$	Fehler.

Die ersten Iterationsverfahren (**Handrechnung!**) beruhen auf Umstellung einzelner Gleichungen, Auflösen nach den entsprechenden Unbekannten und Wiedereinsetzen (Relaxation).

- Carl-Friedrich Gauß ($\sim$ 1832), Normalgleichungen aus Astronomie, Landesvermessung, 20–40 Unbekannte

  „Ich empfehle Ihnen diesen Modus zur Nachahmung. Schwerlich werden Sie je wieder direkt eliminieren, wenigstens nicht, wenn Sie mehr als 2 Unbekannte haben. Das indirekte Verfahren lässt sich halb im Schlafe ausführen, oder man kann während desselben an andere Dinge denken.“ *Brief von Gauß an Gerling, 1823*

- Carl Gustav Jacobi ($\sim$ 1846)
- Ludwig Seidel ($\sim$ 1862)
- Christian August Nagel (Sächsische Triangulation, 1867-1886) 159 Unbekannte

Formale Beschreibung von Relaxationsverfahren durch Zerlegungen (splittings):

$$\boldsymbol{A} = \boldsymbol{M} - \boldsymbol{N}, \qquad \boldsymbol{M} \text{ invertierbar}, \tag{4.2a}$$

überführt (4.1) in Fixpunktform $\boldsymbol{Mx} = \boldsymbol{Nx} + \boldsymbol{b}$, zugehörige Fixpunktiteration

$$\boldsymbol{x}_{m+1} = \boldsymbol{T}\boldsymbol{x}_m + \boldsymbol{c}, \qquad \boldsymbol{T} = \boldsymbol{M}^{-1}\boldsymbol{N}, \quad \boldsymbol{c} = \boldsymbol{M}^{-1}\boldsymbol{b}. \tag{4.2b}$$

Typische algorithmische Umsetzung: (beachte: $\boldsymbol{T}\boldsymbol{x}_m + \boldsymbol{c} = \boldsymbol{x}_m + \boldsymbol{M}^{-1}\boldsymbol{r}_m$)

Algorithmus 1 : Relaxationsverfahren.

Gegeben : $\boldsymbol{A}$ invertierbar, $\boldsymbol{b}$, Startnäherung $\boldsymbol{x}_0$

1 **for** $m = 0, 1, 2 \ldots$ *bis Abbruchkriterium erfüllt* **do**

2 $\quad \boldsymbol{r}_m \leftarrow \boldsymbol{b} - \boldsymbol{A}\boldsymbol{x}_m$

3 $\quad$ Löse $\boldsymbol{M}\boldsymbol{h} = \boldsymbol{r}_m$

4 $\quad \boldsymbol{x}_{m+1} \leftarrow \boldsymbol{x}_m + \boldsymbol{h}$

Klassische Zerlegungen

- $\boldsymbol{M} = \boldsymbol{I}$
  Richardson-Verfahren
- $\boldsymbol{M} = \boldsymbol{D} := \mathrm{diag}(\boldsymbol{A})$
  Jacobi-Verfahren, Einzelschrittverfahren
- $\boldsymbol{M} = \boldsymbol{D} - \boldsymbol{L}$, $\boldsymbol{L} = \mathrm{tril}(\boldsymbol{A})$
  Gauß-Seidel-Verfahren, Einzelschrittverfahren, Liebmann-Verfahren
- $\boldsymbol{M} = \boldsymbol{D} - \omega\boldsymbol{L}$, $\omega \in \mathbb{R}$
  SOR-Verfahren (successive-overrelaxation)
- $\boldsymbol{M} = \frac{\omega}{2-\omega}(\frac{1}{\omega}\boldsymbol{D} - \boldsymbol{L})\boldsymbol{D}^{-1}(\frac{1}{\omega}\boldsymbol{D} - \boldsymbol{U})$, $\boldsymbol{U} := \mathrm{triu}(\boldsymbol{A})$, $\omega \in \mathbb{R}$
  SSOR-Verfahren (symmetric SOR)

Für den Fehler der m-ten Iterierten aus (4.2b) gilt

$$\boldsymbol{e}_{m+1} = \boldsymbol{x}^\star - \boldsymbol{x}_{m+1} = (\boldsymbol{I} - \boldsymbol{M}^{-1}\boldsymbol{A})(\boldsymbol{x}^\star - \boldsymbol{x}_m) = \boldsymbol{T}\boldsymbol{e}_m = \cdots = \boldsymbol{T}^m \boldsymbol{e}_0.$$

Lemma 4.1 *Für eine beliebige Matrix $\boldsymbol{T} \in \mathbb{C}^{n\times n}$ gilt*

$$\lim_{m\to\infty} \boldsymbol{T}^m = \boldsymbol{O} \quad \Leftrightarrow \quad \rho(\boldsymbol{T}) < 1.$$

Satz 4.2 *Das durch die Zerlegung (4.2a) definierte Relaxationsverfahren (4.2b) konvergiert genau dann für alle Startnäherungen x_0, wenn für die Iterationsmatrix $\boldsymbol{T} = \boldsymbol{M}^{-1}\boldsymbol{N}$ gilt $\rho(\boldsymbol{T}) < 1$.*

4.2 Das CG-Verfahren

Satz 4.3 *Die Iterierten, Residuen und Fehler des durch die Zerlegung* (4.2a) *definierten Relaxationsverfahrens* (4.2b) *sind gegeben durch*

$$\begin{aligned} \boldsymbol{x}_m &= \boldsymbol{x}_0 + \sum_{j=0}^{m-1} (\boldsymbol{I} - \boldsymbol{M}^{-1}\boldsymbol{A})^j \boldsymbol{M}^{-1}\boldsymbol{r}_0, \\ \boldsymbol{M}^{-1}\boldsymbol{r}_m &= (\boldsymbol{I} - \boldsymbol{M}^{-1}\boldsymbol{A})^m \boldsymbol{M}^{-1}\boldsymbol{r}_0, \\ \boldsymbol{e}_m &= (\boldsymbol{I} - \boldsymbol{M}^{-1}\boldsymbol{A})^m \boldsymbol{e}_0 \qquad\qquad m = 1, 2, \ldots \end{aligned}$$

also durch

$$\boldsymbol{x}_m = \boldsymbol{x}_0 + q_{m-1}(\boldsymbol{T})\boldsymbol{M}^{-1}\boldsymbol{r}_0, \quad \boldsymbol{M}^{-1}\boldsymbol{r}_m = p_m(\boldsymbol{T})\boldsymbol{M}^{-1}\boldsymbol{r}_0, \quad \boldsymbol{e}_m = p_m(\boldsymbol{T})\boldsymbol{e}_0$$

mit Polynomen

$$q_{m-1}(\lambda) = 1 + \lambda + \cdots + \lambda^{m-1} \in \mathscr{P}_{m-1} \quad \textit{bzw.} \quad p_m(\lambda) = \lambda^m \in \mathscr{P}_m \ .$$

Das CG-Verfahren (conjugate gradients), oder allgemein polynomiale Beschleunigungsverfahren, auch Krylov-Unterraumverfahren genannt, können als Iterationsverfahren interpretiert werden, bei denen die Iterierten bzw. Residuen die Form

$$\boldsymbol{x}_m = \boldsymbol{x}_0 + q_{m-1}(\boldsymbol{A})\boldsymbol{r}_0, \qquad \boldsymbol{r}_m = p_m(\boldsymbol{A})\boldsymbol{r}_0,$$

mit

$$q_{m-1} \in \mathscr{P}_{m-1}, \qquad p_m(\lambda) = 1 - \lambda q_{m-1}(\lambda) \in \mathscr{P}_m$$

besitzen, wobei versucht wird, diese Polynome in jedem Schritt möglichst günstig zu wählen.

Entscheidend: in jedem Schritt der Iteration ist nur eine Matrix-Vektor-Multiplikation mit der Koeffizientenmatrix $\boldsymbol{A}$ (bzw. der Iterationsmatrix $\boldsymbol{T}$) erforderlich.

4.2.1 Herleitung des CG-Verfahrens

Gegeben: $\boldsymbol{A} \in \mathbb{R}^{n\times n}$ spd, $\boldsymbol{b} \in \mathbb{R}^n$ (komplexer Fall analog).

Gesucht: Lösung $\boldsymbol{x}^\star$ des LGS

$$\boldsymbol{A}\boldsymbol{x} = \boldsymbol{b}. \tag{4.3}$$

Sei $(\cdot,\cdot)$ ein gegebenes Innenprodukt auf $\mathbb{R}^n$ (mit zugehöriger Norm $\|\cdot\|$).

Wichtige Beobachtung: da $\boldsymbol{A}$ spd[a] ist durch

$$(\boldsymbol{u},\boldsymbol{v})_A := (\boldsymbol{A}\boldsymbol{u},\boldsymbol{v}), \qquad \boldsymbol{u},\boldsymbol{v} \in \mathbb{R}^n,$$

ebenfalls ein Innenprodukt gegeben, das $\boldsymbol{A}$-Innenprodukt oder auch Energie-Innenprodukt (mit zugehöriger Norm $\|\cdot\|_A$).

Da mit $\boldsymbol{A}$ auch $\boldsymbol{A}^{-1}$ spd, gilt dasselbe auch für $(\cdot,\cdot)_{A^{-1}}$.

[a] eigentlich: $\boldsymbol{A}$ selbstadjungiert bez. $(\cdot,\cdot)$

Ausgangspunkt dieser Herleitung des CG-Verfahrens (vgl. Numerik-Vorlesung, Abschnitt über Variationsrechnung):

$$\boxed{\boldsymbol{x}^* \text{ löst (4.3)} \iff \boldsymbol{x}^* \text{ minimiert } J(\boldsymbol{x}) = \tfrac{1}{2}(\boldsymbol{x},\boldsymbol{x})_A - (\boldsymbol{b},\boldsymbol{x}) \text{ über } \mathbb{R}^n.}$$

Minimierung auf Unterräumen: Anstatt auf ganz $\mathbb{R}^n$ minimieren wir J auf einer Folge geschachtelter affiner Unterräume der Gestalt

$$\mathscr{S}_m = \boldsymbol{x}_0 + \mathscr{V}_m, \qquad \mathscr{V}_m = \operatorname{span}\{\boldsymbol{p}_1, \boldsymbol{p}_2, \ldots, \boldsymbol{p}_m\}.$$

Hierbei bilden $\boldsymbol{p}_1, \ldots, \boldsymbol{p}_m$ eine Folge linear unabhängiger Richtungsvektoren.

Als Näherungen $\boldsymbol{x}_m \approx \boldsymbol{x}^*$ erhalten wir die Lösungen der Unterraumaufgaben

$$\boxed{\text{Bestimme } \boldsymbol{x}_m \in \mathscr{S}_m \text{ mit } J(\boldsymbol{x}_m) = \min_{\boldsymbol{x} \in \mathscr{S}_m} J(\boldsymbol{x}).}$$

Wegen

$$J(\boldsymbol{x}) = \tfrac{1}{2}\big(\|\boldsymbol{b}-\boldsymbol{A}\boldsymbol{x}\|_{A^{-1}}^2 - \|\boldsymbol{b}\|_{A^{-1}}^2\big) = \tfrac{1}{2}\|\boldsymbol{x}^\star-\boldsymbol{x}\|_A^2 + J(\boldsymbol{x}^\star)$$

ist die Minimierung von J äquivalent zur Minimierung von $\|\boldsymbol{b}-\boldsymbol{A}\boldsymbol{x}\|_{A^{-1}}$, d.h. gesucht ist ein $\boldsymbol{x} \in \mathscr{S}_m$ welches den Ausdruck

$$\|\boldsymbol{b}-\boldsymbol{A}\boldsymbol{x}\|_{A^{-1}} = \|\boldsymbol{b}-\boldsymbol{A}\boldsymbol{x}_0-\boldsymbol{A}\boldsymbol{p}\|_{A^{-1}} = \|\boldsymbol{r}_0-\boldsymbol{A}\boldsymbol{p}\|_{A^{-1}}$$

minimiert mit einem $\boldsymbol{p} \in \mathscr{V}_m$.

Äquivalent: bestimme Bestapproximation bez. $\|\cdot\|_{A^{-1}}$ an $\boldsymbol{r}_0$ aus dem Unterraum $\boldsymbol{A}\mathscr{V}_m$.

Äquivalent: $\boldsymbol{A}^{-1}$-orthogonale Projektion von $\boldsymbol{r}_0$ nach $\boldsymbol{A}\mathscr{V}_m$

Charakterisierung:

$$\boldsymbol{r}_0-\boldsymbol{A}\boldsymbol{p} \perp_{A^{-1}} \boldsymbol{A}\boldsymbol{p}_j, \qquad j=1,2,\ldots,m,$$

oder

$$(\boldsymbol{p}, \boldsymbol{p}_j)_A = (\boldsymbol{r}_0, \boldsymbol{p}_j), \qquad j = 1, 2, \ldots, m,$$

oder

$$\boldsymbol{r}_m := \boldsymbol{b} - \boldsymbol{A}\boldsymbol{x}_m \perp \mathscr{V}_m.$$

Besitzt $\boldsymbol{p}$ die Koordinatendarstellung

$$\boldsymbol{p} = y_1 \boldsymbol{p}_1 + \cdots + y_m \boldsymbol{p}_m,$$

so erfüllt der Koordinatenvektor $\boldsymbol{y} = [y_1, y_2, \ldots, y_m]^\top \in \mathbb{R}^m$ das LGS

$$[(\boldsymbol{p}_j, \boldsymbol{p}_i)_A]_{i,j=1}^m \boldsymbol{y} = [(\boldsymbol{r}_0, \boldsymbol{p}_i)]_{i=1}^m. \tag{4.4}$$

Bemerkung: Wegen $\|\boldsymbol{b} - \boldsymbol{A}\boldsymbol{x}\|_{A^{-1}} = \|\boldsymbol{A}^{-1}\boldsymbol{b} - \boldsymbol{x}\|_A$ besitzt die Bestapproximation $\boldsymbol{p} \in \mathscr{V}_m$ auch die Eigenschaft, dass die zugehörige Approximation $\boldsymbol{x}_m = \boldsymbol{x}_0 + \boldsymbol{p}$ den Fehler $\boldsymbol{A}^{-1}\boldsymbol{b} - \boldsymbol{x}_m$ bezüglich der $\boldsymbol{A}$-Norm minimiert.

Konjugierte Suchrichtungen

Sind die Richtungsvektoren $\boldsymbol{p}_j$ $\boldsymbol{A}$-konjugiert (ein Synonym für $\boldsymbol{A}$-orthogonal), so vereinfacht sich das LGS (4.4) für die Koordinaten:

- Die Koeffizientenmatrix wird diagonal,
- die Gleichungen sind entkoppelt, d.h. beim Übergang von $\mathscr{V}_m$ nach $\mathscr{V}_{m+1}$ bleiben die Koeffizienten $y_1, \ldots, y_m$ unverändert,
- es gilt

$$y_i = \frac{(\boldsymbol{r}_0, \boldsymbol{p}_i)}{(\boldsymbol{p}_i, \boldsymbol{p}_i)_A}, \qquad i = 1, 2, \ldots, m.$$

Wir erhalten so folgenden Algorithmus:

Algorithmus 2 : Verfahren der konjugierten Richtungen (CD-Verf.)

Gegeben : $\boldsymbol{A}$ spd, $\boldsymbol{b}$

1 Bestimme Startnäherung $\boldsymbol{x}_0$, berechne $\boldsymbol{r}_0 \leftarrow \boldsymbol{b} - \boldsymbol{A}\boldsymbol{x}_0$, $m \leftarrow 0$

2 **while** *nicht konvergiert* **do**

3 $\quad m \leftarrow m + 1$

4 $\quad$ Bestimme nächsten, zu $\boldsymbol{p}_1, \ldots, \boldsymbol{p}_{m-1}$ konjugierten Richtungsvektor $\boldsymbol{p}_m$

5 $\quad \alpha_m \leftarrow \frac{(\boldsymbol{r}_{m-1}, \boldsymbol{p}_m)}{(\boldsymbol{p}_m, \boldsymbol{p}_m)_A}$

6 $\quad \boldsymbol{x}_m \leftarrow \boldsymbol{x}_{m-1} + \alpha_m \boldsymbol{p}_m$

7 $\quad \boldsymbol{r}_m \leftarrow \boldsymbol{r}_{m-1} - \alpha_m \boldsymbol{A}\boldsymbol{p}_m$

Bemerkung: Eine Folge A-orthogonaler Suchrichtungen kann man erzeugen aus einer beliebigen Folge (linear unabhängiger) Vektoren $\widetilde{\boldsymbol{p}}_1, \widetilde{\boldsymbol{p}}_2, \ldots$ mit Hilfe des Gram-Schmidtschen Orthogonalisierungsverfahrens:

$$\boldsymbol{p}_m = \widetilde{\boldsymbol{p}}_m - \sum_{j=1}^{m-1} \frac{(\widetilde{\boldsymbol{p}}_m, \boldsymbol{p}_j)_A}{(\boldsymbol{p}_j, \boldsymbol{p}_j)_A} \boldsymbol{p}_j, \qquad m = 1, 2 \ldots \,. \tag{4.5}$$

Lemma 4.4 *Im m-ten Schritt des CD-Verfahrens gilt:*

$$\boldsymbol{p}_m \perp_A \{\boldsymbol{p}_1, \boldsymbol{p}_2, \ldots, \boldsymbol{p}_{m-1}\}, \tag{4.6}$$

$$\boldsymbol{r}_m \perp \{\boldsymbol{p}_1, \boldsymbol{p}_2, \ldots, \boldsymbol{p}_m\}, \tag{4.7}$$

$$(\boldsymbol{p}_m, \boldsymbol{r}_{m-1}) = (\boldsymbol{p}_m, \boldsymbol{r}_{m-2}) = \cdots = (\boldsymbol{p}_m, \boldsymbol{r}_0). \tag{4.8}$$

Lemma 4.5 *Das CD-Verfahren berechnet die Lösung $\boldsymbol{x}^*$ von (4.3) nach einer endlichen Anzahl $L \leq n$ Schritten.*

Bemerkungen:

1. Die endliche Abbrucheigenschaft geht i.A. verloren bei rundungsbehafteter Rechnung.
2. Obwohl der Fehler die interessierende Größe, ist oft nur das Residuum verfügbar. Der nächste Satz gibt prinzipielle Beziehungen zwischen diesen an.

Satz 4.6 *Sei x eine Approximation an $A^{-1}b$. Dann gelten für den Fehler $e := A^{-1}b - x$ und das zugehörige Residuum $r = b - Ax$*

$$\frac{\|r\|}{\|r\|_A} \leq \frac{\|e\|_A}{\|r\|} \leq \frac{\|e\|}{\|e\|_A}, \tag{4.9}$$

$$\frac{\|r\|^2}{\|r\|_A^2} \leq \frac{\|e\|}{\|r\|} \leq \frac{\|e\|^2}{\|e\|_A^2}. \tag{4.10}$$

Lemma 4.7 *Im m-ten Schritt des CD-Verfahrens wird der Abstand in der A-Norm zur Lösung reduziert gemäß*

$$\|e_{m-1}\|_A^2 - \|e_m\|_A^2 = \frac{(r_{m-1}, p_m)^2}{(p_m, p_m)_A}.$$

Konjugierte Gradienten

$\boldsymbol{A}$-orthogonale Suchrichtungen machen einen Iterationsschritt zwar effizient, müssen aber per se nicht zu schneller Konvergenz führen. (Warum?)

Idee: Wähle in (4.5) den neuen Vektor $\widetilde{\boldsymbol{p}}_{m+1}$ als Richtung stärkster Abnahme von J an der Stelle $\boldsymbol{x}_m$:

$$\widetilde{\boldsymbol{p}}_{m+1} = -\nabla J|_{\boldsymbol{x}=\boldsymbol{x}_m} = \boldsymbol{b} - \boldsymbol{A}\boldsymbol{x}_m = \boldsymbol{r}_m, \qquad m > 1,$$
$$\widetilde{\boldsymbol{p}}_1 = \boldsymbol{p}_1 = \boldsymbol{r}_0.$$

Danach $\boldsymbol{A}$-Orthogonalisierung gegen $\{\boldsymbol{p}_1, \ldots, \boldsymbol{p}_m\}$.

Vorteil: Liefert solange neue Suchrichtungen bis Lösung gefunden. (Warum?)

Aus dieser Wahl folgt sofort

$$\operatorname{span}\{\boldsymbol{r}_0, \boldsymbol{r}_1, \ldots, \boldsymbol{r}_{m-1}\} = \operatorname{span}\{\boldsymbol{p}_1, \boldsymbol{p}_2, \ldots, \boldsymbol{p}_m\}, \quad m = 1, 2, \ldots, L. \quad (4.11)$$

Wir erhalten so den

Algorithmus 3 : CG-Verfahren (vorläufige Version).

Gegeben : $\boldsymbol{A}$ spd, $\boldsymbol{b}$

1 Bestimme Startnäherung $\boldsymbol{x}_0$, berechne $\boldsymbol{r}_0 \leftarrow \boldsymbol{b} - \boldsymbol{A}\boldsymbol{x}_0$

2 Setze $m \leftarrow 0$, $\boldsymbol{p}_1 \leftarrow \boldsymbol{r}_0$

3 **while** *nicht konvergiert* **do**

4 $\quad m \leftarrow m + 1$

5 $\quad \alpha_m \leftarrow \frac{(\boldsymbol{r}_{m-1}, \boldsymbol{p}_m)}{(\boldsymbol{p}_m, \boldsymbol{p}_m)_A}$

6 $\quad \boldsymbol{x}_m \leftarrow \boldsymbol{x}_{m-1} + \alpha_m \boldsymbol{p}_m$

7 $\quad \boldsymbol{r}_m \leftarrow \boldsymbol{r}_{m-1} - \alpha_m \boldsymbol{A}\boldsymbol{p}_m$

8 $\quad \boldsymbol{p}_{m+1} \leftarrow \boldsymbol{r}_m - \sum_{j=1}^{m} \frac{(\boldsymbol{r}_m, \boldsymbol{p}_j)_A}{(\boldsymbol{p}_j, \boldsymbol{p}_j)_A} \boldsymbol{p}_j$

Vereinfachungen:

(1) Solange $m < L$ ist $\alpha_m \neq 0$ und aus $\boldsymbol{r}_m = \boldsymbol{r}_{m-1} - \alpha_m \boldsymbol{A}\boldsymbol{p}_m$ folgt zunächst $\boldsymbol{A}\boldsymbol{p}_m \in \operatorname{span}\{\boldsymbol{r}_{m-1}, \boldsymbol{r}_m\}$ und allgemein

$$\operatorname{span}\{\boldsymbol{A}\boldsymbol{p}_1, \boldsymbol{A}\boldsymbol{p}_2, \ldots, \boldsymbol{A}\boldsymbol{p}_{m-1}\} \subset \operatorname{span}\{\boldsymbol{r}_0, \boldsymbol{r}_1, \ldots, \boldsymbol{r}_{m-1}\} \overset{(4.11)}{=} \operatorname{span}\{\boldsymbol{p}_1, \boldsymbol{p}_2, \ldots, \boldsymbol{p}_m\}.$$

Da aber $\boldsymbol{r}_m \perp \operatorname{span}\{\boldsymbol{p}_1, \ldots, \boldsymbol{p}_m\}$ nach (4.7) folgt hieraus $\boldsymbol{r}_m \perp_A \operatorname{span}\{\boldsymbol{p}_1, \ldots, \boldsymbol{p}_{m-1}\}$ sodass Zeile 8 sich vereinfacht zu

$$\boldsymbol{p}_{m+1} \leftarrow \boldsymbol{r}_m - \frac{(\boldsymbol{r}_m, \boldsymbol{p}_m)_A}{(\boldsymbol{p}_m, \boldsymbol{p}_m)_A}\boldsymbol{p}_m =: \boldsymbol{r}_m + \beta_{m+1}\boldsymbol{p}_m. \tag{4.12}$$

(2) Aus $\boldsymbol{p}_m = \boldsymbol{r}_{m-1} + \beta_m \boldsymbol{p}_{m-1}$ folgt wegen $\boldsymbol{r}_{m-1} \perp \boldsymbol{p}_{m-1}$ für Zeile 5

$$\alpha_m = \frac{(\boldsymbol{r}_{m-1}, \boldsymbol{p}_m)}{(\boldsymbol{p}_m, \boldsymbol{p}_m)_A} = \frac{(\boldsymbol{r}_{m-1}, \boldsymbol{r}_{m-1})}{(\boldsymbol{p}_m, \boldsymbol{p}_m)_A}. \tag{4.13}$$

(3) Aus $\boldsymbol{r}_m = \boldsymbol{r}_{m-1} - \alpha_m \boldsymbol{A}\boldsymbol{p}_m$ folgt

$$(\boldsymbol{r}_m, \boldsymbol{r}_m) = -\alpha_m(\boldsymbol{r}_m, \boldsymbol{A}\boldsymbol{p}_m) = -\alpha_m(\boldsymbol{r}_m, \boldsymbol{p}_m)_A$$

und daher wegen (4.13) für (4.12)

$$\beta_{m+1} = -\frac{(\boldsymbol{r}_m, \boldsymbol{p}_m)_A}{(\boldsymbol{p}_m, \boldsymbol{p}_m)_A} = \frac{(\boldsymbol{r}_m, \boldsymbol{r}_m)}{\alpha_m(\boldsymbol{p}_m, \boldsymbol{p}_m)_A} = \frac{(\boldsymbol{r}_m, \boldsymbol{r}_m)}{(\boldsymbol{r}_{m-1}, \boldsymbol{r}_{m-1})}.$$

Mit diesen Vereinfachungen erhalten wir die endgültige Formulierung des CG-Verfahrens:

Algorithmus 4 : CG-Verfahren.

Gegeben : $\boldsymbol{A}$ spd, $\boldsymbol{b}$

1 Bestimme Startnäherung $\boldsymbol{x}_0$, berechne $\boldsymbol{r}_0 \leftarrow \boldsymbol{b} - \boldsymbol{A}\boldsymbol{x}_0$

2 Setze $m \leftarrow 0$, $\boldsymbol{p}_1 \leftarrow \boldsymbol{r}_0$

3 **while** *nicht konvergiert* **do**

4 $\quad m \leftarrow m + 1$

5 $\quad \alpha_m \leftarrow \frac{(\boldsymbol{r}_{m-1}, \boldsymbol{r}_{m-1})}{(\boldsymbol{p}_m, \boldsymbol{p}_m)_A}$

6 $\quad \boldsymbol{x}_m \leftarrow \boldsymbol{x}_{m-1} + \alpha_m \boldsymbol{p}_m$

7 $\quad \boldsymbol{r}_m \leftarrow \boldsymbol{r}_{m-1} - \alpha_m \boldsymbol{A}\boldsymbol{p}_m$

8 $\quad \beta_{m+1} \leftarrow \frac{(\boldsymbol{r}_m, \boldsymbol{r}_m)}{(\boldsymbol{r}_{m-1}, \boldsymbol{r}_{m-1})}$

9 $\quad \boldsymbol{p}_{m+1} \leftarrow \boldsymbol{r}_m + \beta_{m+1} \boldsymbol{p}_m$

4.2.2 Eigenschaften des CG-Verfahrens

Aufwand: Operationen pro Schritt: 1 Matrix-Vektor Produkt,
2 Innenprodukte,
3 Vektoraufdatierungen.

Speicher: 4 Vektoren.

Lemma 4.8 *Für die Suchrichtungen des CG-Verfahrens gilt*

$$\boldsymbol{p}_m = \|\boldsymbol{r}_{m-1}\|^2 \sum_{j=0}^{m-1} \frac{\boldsymbol{r}_j}{\|\boldsymbol{r}_j\|^2}, \qquad m = 1, 2, \ldots, L. \tag{4.14}$$

Satz 4.9 *Beim CG-Verfahren gelten die Beziehungen*

$$(\boldsymbol{r}_m, \boldsymbol{p}_j) = 0, \qquad 1 \le j \le m, \qquad 1 \le m \le L-1, \tag{4.15}$$

$$(\boldsymbol{r}_m, \boldsymbol{r}_j) = 0, \qquad m \ne j, \qquad 0 \le m, j \le L-1, \tag{4.16}$$

$$(\boldsymbol{r}_m, \boldsymbol{p}_j) = \|\boldsymbol{r}_{j-1}\|^2, \qquad m+1 \le j \le L, \qquad 0 \le m \le L-1, \tag{4.17}$$

$$(\boldsymbol{p}_m, \boldsymbol{p}_j)_A = 0, \qquad m \ne j, \qquad 1 \le m, j \le L, \tag{4.18}$$

$$(\boldsymbol{p}_m, \boldsymbol{r}_{m-1})_A = \|\boldsymbol{p}_m\|_A^2, \qquad 1 \le m \le L, \tag{4.19}$$

$$(\boldsymbol{p}_m, \boldsymbol{r}_j)_A = 0, \qquad m+1 \le j \le L-1, \qquad m = 1, 2 \ldots, L, \tag{4.20}$$

Lemma 4.10 *Für die Residuen und Richtungsvektoren des CG-Verfahrens gelten*

$$(\boldsymbol{p}_m, \boldsymbol{p}_j) = \frac{\|\boldsymbol{r}_{j-1}\|^2 \|\boldsymbol{p}_m\|^2}{\|\boldsymbol{r}_{m-1}\|^2}, \qquad j \ge m, \tag{4.21}$$

$$\|\boldsymbol{p}_m\|^2 = \|\boldsymbol{r}_{m-1}\|^2 + \beta_m^2 \|\boldsymbol{p}_{m-1}\|^2 = \|\boldsymbol{r}_{m-1}\|^4 \sum_{j=0}^{m-1} \frac{1}{\|\boldsymbol{r}_j\|^2}. \tag{4.22}$$

Lemma 4.11 *Im m-ten Schritt des CG-Verfahrens gilt*

$$\|\boldsymbol{e}_{m-1}\|_A^2 - \|\boldsymbol{e}_m\|_A^2 = \alpha_m \|\boldsymbol{r}_{m-1}\|^2 = \frac{\|\boldsymbol{p}_m\|_A^2}{\|\boldsymbol{p}_m\|^2} \|\boldsymbol{x}_m - \boldsymbol{x}_{m-1}\|^2. \tag{4.23}$$

Für $j < m$ gilt

$$\|\boldsymbol{e}_j\|_A^2 - \|\boldsymbol{e}_m\|_A^2 = \alpha_{j+1} \|\boldsymbol{r}_j\|^2 + \cdots + \alpha_m \|\boldsymbol{r}_{m-1}\|^2. \tag{4.24}$$

Die m-te Iterierte besitzt die Darstellung

$$\boldsymbol{x}_m = \boldsymbol{x}_0 + \sum_{j=1}^{m} \alpha_j \boldsymbol{p}_j = \boldsymbol{x}_0 + \sum_{j=1}^{m} \frac{\|\boldsymbol{e}_j\|_A^2 - \|\boldsymbol{e}_m\|_A^2}{\|\boldsymbol{r}_{j-1}\|^2} \boldsymbol{r}_{j-1}. \tag{4.25}$$

Satz 4.12 *Im m-ten Schritt des CG-Verfahrens gilt*

$$\|\boldsymbol{e}_{m-1}\|^2 - \|\boldsymbol{e}_m\|^2 = \frac{\|\boldsymbol{p}_m\|^2}{\|\boldsymbol{p}_m\|_A^2} \left[\|\boldsymbol{e}_m\|_A^2 + \|\boldsymbol{e}_{m-1}\|_A^2 \right]. \tag{4.26}$$

Insbesondere ist der CG-Fehler auch bezüglich der Norm $\|\cdot\|$ monoton fallend.

CG als Krylov-Unterraumverfahren

Für $\boldsymbol{A} \in \mathbb{R}^{n\times n}$ und $\boldsymbol{v} \in \mathbb{R}^n$ heißt der Unterraum

$$\mathscr{K}_m(\boldsymbol{A}, \boldsymbol{v}) := \operatorname{span}\{\boldsymbol{v}, \boldsymbol{A}\boldsymbol{v}, \ldots, \boldsymbol{A}^{m-1}\boldsymbol{v}\} \subset \mathbb{R}^n$$

der m-te Krylov-Unterraum von $\boldsymbol{A}$ und $\boldsymbol{v}$.

Es ist

$$\mathscr{K}_m(\boldsymbol{A}, \boldsymbol{v}) = \{p(\boldsymbol{A})\boldsymbol{v} : p \in \mathscr{P}_{m-1}\}.$$

Proposition 4.13

(a) Die m-te CG-Iterierte liegt im affinen Unterraum $\boldsymbol{x}_0 + \mathscr{K}_m(\boldsymbol{A}, \boldsymbol{r}_0)$.

(b) Für den Abbruchindex L des CG-Verfahrens gilt

$$\begin{aligned} L &= \min\big\{m : \mathscr{K}_m(\boldsymbol{A}, \boldsymbol{r}_0) = \mathscr{K}_{m+1}(\boldsymbol{A}, \boldsymbol{r}_0)\big\} \\ &= \min\big\{\deg q : q \text{ monisch und } q(\boldsymbol{A})\boldsymbol{r}_0 = \boldsymbol{0}\big\}. \end{aligned}$$

4.2.3 Konvergenz des CG-Verfahrens

Die Rolle der Konditionszahl

- Bei einer $n \times n$-Matrix $\boldsymbol{A}$ liefert das CG -Verfahren die Lösung von $\boldsymbol{A}\boldsymbol{x} = \boldsymbol{b}$ nach $L \leq n$ Schritten (in exakter Arithmetik). In diesem Sinne „konvergiert“ das CG-Verfahren immer gegen $\boldsymbol{x}^\star = \boldsymbol{A}^{-1}\boldsymbol{b}$. Gesucht sind hier obere Schranken für den Fehler $\boldsymbol{x}^\star - \boldsymbol{x}_m$ ($m < L$).
- Von entscheidender Bedeutung ist die Optimalitätseigenschaft

$$\|\boldsymbol{x}^\star - \boldsymbol{x}_m\|_A = \min_{p \in \mathscr{P}_m,\, p(0)=1} \|p(\boldsymbol{A})(\boldsymbol{x}^\star - \boldsymbol{x}_0)\|_A.$$

- Sei $\Omega \subset (0, \infty)$ eine kompakte Menge, welche die Eigenwerte von $\boldsymbol{A}$ enthält. Für jedes $p_m \in \mathscr{P}_m$ mit $p_m(0) = 1$ gilt dann

$$\frac{\|\boldsymbol{x}^\star - \boldsymbol{x}_m\|_A}{\|\boldsymbol{x}^\star - \boldsymbol{x}_0\|_A} \leq \|p_m\|_{\infty,\Omega} := \max_{\lambda \in \Omega} |p_m(\lambda)|.$$

- Alle echten Fehlerschranken für das CG-Verfahren basieren auf obiger (trivialer) Beobachtung: ausgehend von einer kompakten Obermenge $\Omega \supset \Lambda(\boldsymbol{A})$ (typischerweise $\Omega = [\lambda_{\min}(\boldsymbol{A}), \lambda_{\max}(\boldsymbol{A})]$) versuchen wir, Polynome p_m zu konstruieren mit $p_m(0) = 1$, die auf Ω ein möglichst kleines Maximum haben. Dies führt auf natürliche Weise zu einer Aufgabe der polynomialen Bestapproximation:
  Bestimme $p_m \in \mathscr{P}_m$, $p_m(0) = 1$, sodass

$$\|p_m\|_{\infty,\Omega} = \min_{\substack{p \in \mathscr{P}_m, \\ p(0)=1}} \|p\|_{\infty,\Omega}.$$

- Für $\Omega = \Lambda(\boldsymbol{A})$ ist die Abschätzung

$$\frac{\|\boldsymbol{x}^\star - \boldsymbol{x}_m\|_A}{\|\boldsymbol{x}^\star - \boldsymbol{x}_0\|_A} \leq \min_{\substack{p \in \mathscr{P}_m, \\ p(0)=1}} \max_{\lambda \in \Lambda(\boldsymbol{A})} |p(\lambda)|$$

  in folgendem Sinne scharf: Für jedes m existiert ein Startvektor $\boldsymbol{x}_0 = \boldsymbol{x}_0(m)$, sodass im m-ten Schritt Gleichheit gilt.

- Die Aufgabe, $p_m \in \mathscr{P}_m$, $p_m(0) = 1$, so zu bestimmen, dass

$$\|p_m\|_{\infty,\Omega} = \min_{\substack{p\in\mathscr{P}_m,\\ p(0)=1}} \|p\|_{\infty,\Omega}$$

  ist lösbar für jedes $m \in \mathbb{N}_0$ und jede kompakte Teilmenge Ω der komplexen Ebene. Die Lösung ist eindeutig, sofern Ω aus mindestens m Punkten besteht. Eine geschlossene Darstellung dieser Lösung existiert jedoch nur in Spezialfällen, etwa wenn Ω ein Intervall ist.

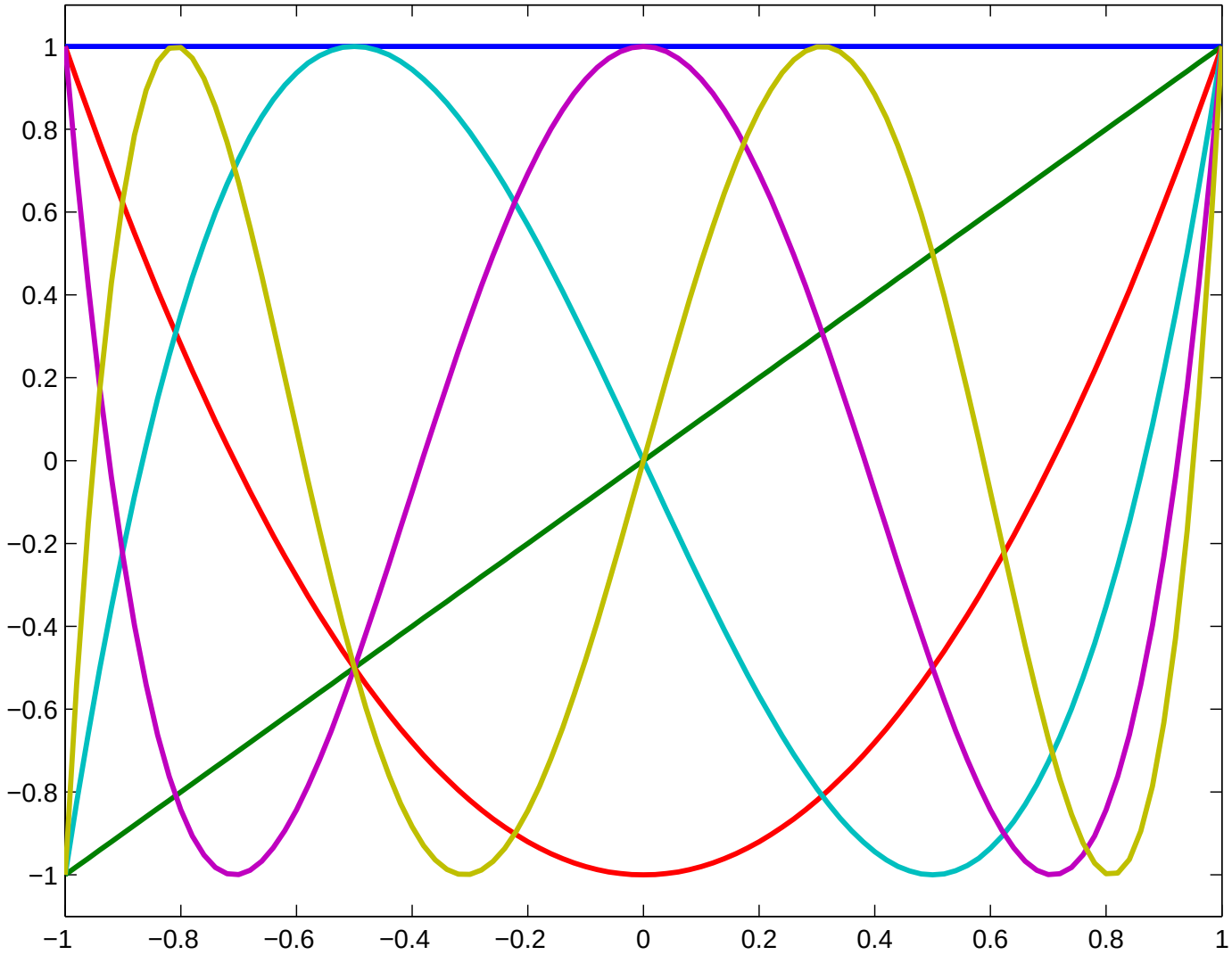
- Die Chebyshev-Polynome sind definiert durch

$$T_m(\xi) := \cos(m \arccos \xi) \in \mathscr{P}_m \quad (\xi \in [-1,1]\,,\ m = 0,1,\ldots) \tag{4.27}$$

- Sie genügen der dreistufigen Rekursion

$$T_m(\xi) = 2\xi\, T_{m-1}(\xi) - T_{m-2}(\xi) \qquad (m = 2,3,\ldots)$$

  mit $T_0(\xi) = 1$ and $T_1(\xi) = \xi$. Hieran erkennt man, dass T_m tatsächlich ein Polynom vom Grad (exakt) m ist.

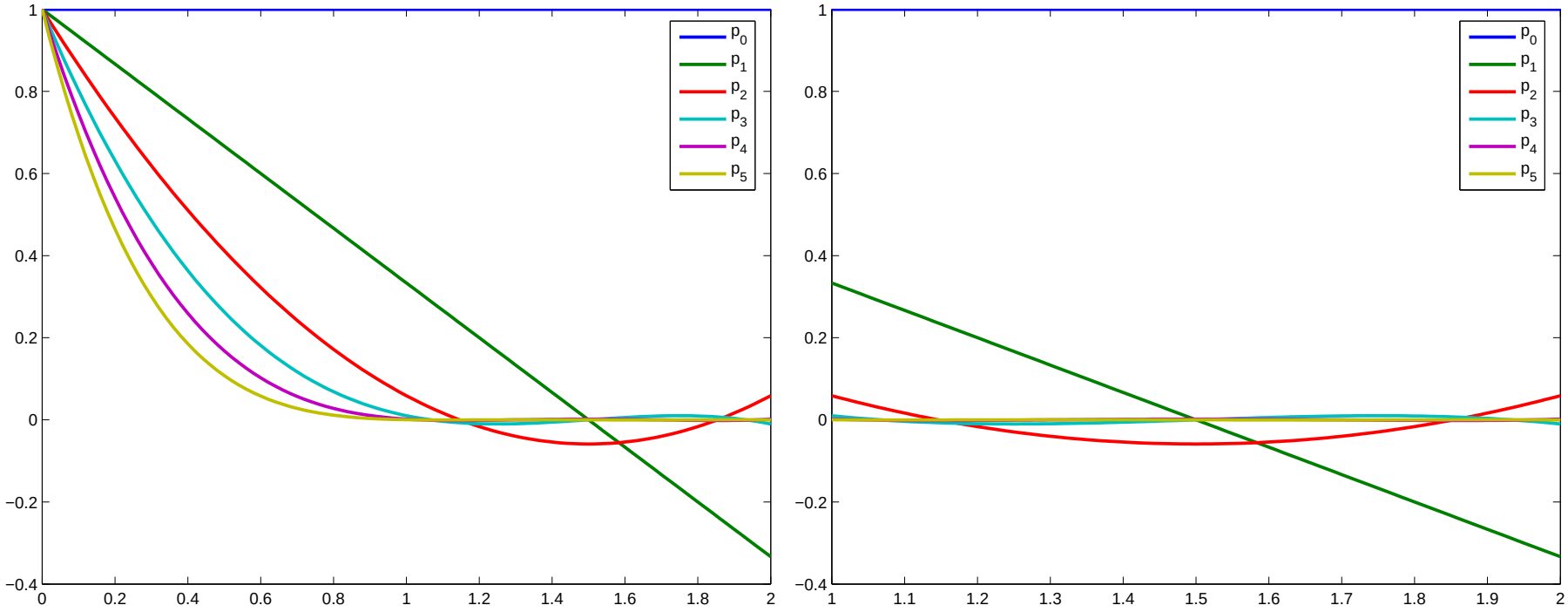


Graphen von $T_0, T_1, \ldots T_5$ auf $[-1, 1]$.

- T_m besitzt die m einfachen Nullstellen, $\xi_k^{(m)} = \cos((2k+1)\pi/(2m))$, $k = 0, \ldots, m-1$, welche sämtlich in $(-1, 1)$ liegen.
- Es ist $|T_m(\xi)| \leq 1$ für alle $\xi \in [-1, 1]$. Gleichheit gilt für $\eta_k^{(m)} := \cos(k\pi/m)$, $k = 0, \ldots, m$, genauer gesagt: $T_m(\eta_k^{(m)}) = (-1)^k$.
- Als nächstes transformieren wir die Polynome T_m auf ein beliebiges reelles Intervall $[\delta - \gamma, \delta + \gamma] \subset (0, +\infty)$, d.h., $0 < \gamma < \delta$.
  $\lambda \mapsto \ell(\lambda) := (\lambda - \delta)/\gamma$ ist eine Bijektion von $[\delta - \gamma, \delta + \gamma]$ auf $[-1, 1]$.
  Die Umkehrabbildung ist $\xi \mapsto \ell^{-1}(\xi) = \delta + \xi\gamma$.
  Wegen $-\delta/\gamma < -1$ gilt $T_m(-\delta/\gamma) \neq 0$ und

$$p_m(\lambda) := \frac{T_m(\ell(\lambda))}{T_m(\ell(0))} = \frac{T_m((\lambda - \delta)/\gamma)}{T_m(-\delta/\gamma)} \qquad (m = 0, 1, \ldots) \qquad (4.28)$$

  ist somit eine Folge von Residualpolynomen, d.h., $p_m(0) = 1$.



Graphen von $p_0, p_1, \ldots p_5$ auf $[0, 2]$ bzw. auf $[\delta - \gamma, \delta + \gamma] = [1, 2]$.

Lemma 4.14 *Sei* $\Omega := [\delta - \gamma,\, \delta + \gamma] \subset (0, \infty)$. *Für die Polynome* p_m *aus* (4.28) *gilt*

$$\|p_m\|_{\infty,\Omega} = \min_{\substack{p \in \mathscr{P}_m, \\ p(0)=1}} \|p\|_{\infty,\Omega}.$$

- Es verbleibt die Bestimmung von $T_m(-\delta/\gamma)$. Hierzu setzen wir

$$\theta := \frac{\delta - \sqrt{\delta^2 - \gamma^2}}{\gamma}.$$

- Es ist $0 < \theta < 1$ und

$$\max_{\lambda \in \Omega} |p_m(\lambda)| = \frac{1}{T_m(-\delta/\gamma)} = \frac{2\,\theta^m}{1 + \theta^{2m}} \leq 2\,\theta^m \quad (m = 0, 1, \ldots).$$

Satz 4.15 (Fehlerschranke für CG) *Für die Näherungen des CG-Verfahrens gilt die Fehlerabschätzung*

$$\frac{\|\boldsymbol{x}^\star - \boldsymbol{x}_m\|_A}{\|\boldsymbol{x}^\star - \boldsymbol{x}_0\|_A} \leq \frac{1}{T_m\left(\frac{\kappa+1}{\kappa-1}\right)} = \frac{2\left(\frac{\sqrt{\kappa}-1}{\sqrt{\kappa}+1}\right)^m}{1+\left(\frac{\sqrt{\kappa}-1}{\sqrt{\kappa}+1}\right)^{2m}} \leq 2\left(\frac{\sqrt{\kappa}-1}{\sqrt{\kappa}+1}\right)^m.$$

Hierbei bezeichnet $\kappa = \mathrm{cond}_2(\boldsymbol{A}) = \lambda_{\max}(\boldsymbol{A})/\lambda_{\min}(\boldsymbol{A})$ *die Konditionszahl von* $\boldsymbol{A}$ *bezüglich der Euklid-Norm.*

Diese Schranke ist (nahezu) scharf in folgendem Sinne: es gibt lineare Gleichungssysteme $\boldsymbol{A}\boldsymbol{x} = \boldsymbol{b}$ (mit $\boldsymbol{A} \in \mathbb{R}^{n\times n}$) sowie $\mathrm{cond}_2(\boldsymbol{A}) = \kappa$ und Startnäherungen $\boldsymbol{x}_0$ sodass

$$\frac{\|\boldsymbol{x}^\star - \boldsymbol{x}_{n-1}\|_A}{\|\boldsymbol{x}^\star - \boldsymbol{x}_0\|_A} = \frac{1}{T_{n-1}((\kappa+1)/(\kappa-1))}$$

(Setze $\boldsymbol{A} = \mathrm{diag}(\widetilde{\eta}_1, \ldots, \widetilde{\eta}_n)$ mit $\widetilde{\eta}_j$ die Extremalstellen des transformierten Chebyshev-Polynoms p_{n-1}.)

Abbruchbedingung

Typische Abbruchbedingung: iteriere bis $\|\boldsymbol{e}_m\|_A / \|\boldsymbol{e}_0\|_A \leq \epsilon$ mit vorgegebener Toleranz $\epsilon > 0$. Ausgehend von Satz 4.15 ist dies spätestens der Fall, wenn

$$\epsilon \geq 2 \left(\frac{\sqrt{\kappa} - 1}{\sqrt{\kappa} + 1} \right)^m = 2 \left(1 - \frac{2/\sqrt{\kappa}}{1 + 1/\sqrt{\kappa}} \right)^m .$$

Logarithmieren und die Approximation $\log(1 + x) \approx x$ für x klein führt auf die Bedingung

$$\boxed{m \geq \frac{1}{2} \left| \log \frac{\epsilon}{2} \right| \sqrt{\kappa}}.$$

Beispiel: Poisson-Gleichung, uniformes Gitter, bilineare FE. Hier gilt $\kappa(\boldsymbol{A}) = O(h^{-2})$. Dies bedeutet: bei Halbierung der Gitterweite verdoppelt sich die Anzahl Iterationen.

Beispiel 1: die symmetrische 500×500 Matrix $\boldsymbol{A} = \boldsymbol{A}(\tau)$ definiert durch

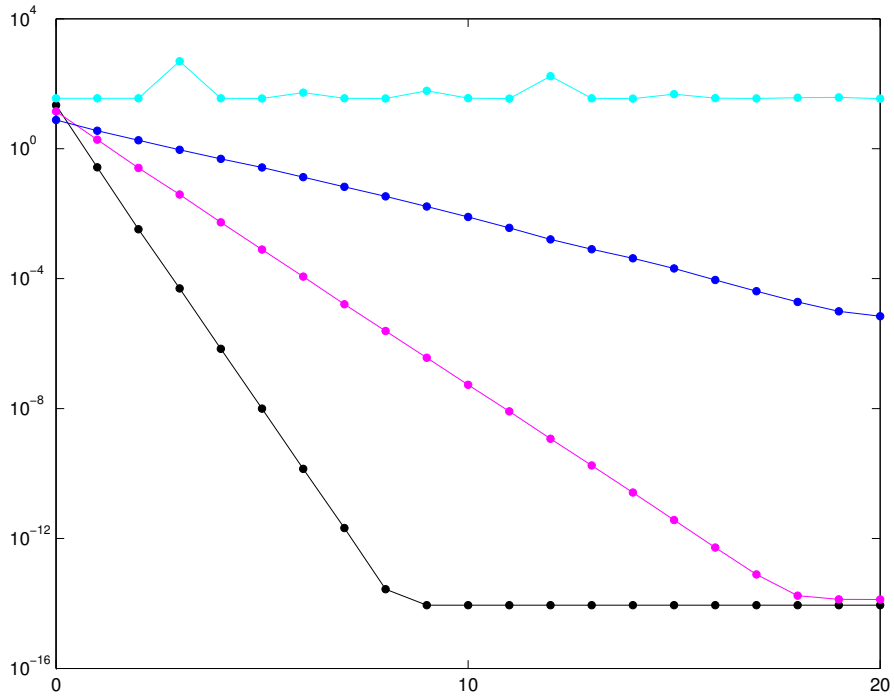
$$a_{i,j} = \begin{cases} 1 & \text{falls } i = j, \\ X_{i,j} & \text{falls } i \neq j \text{ und } X_{i,j} \leq \tau, \\ 0 & \text{sonst.} \end{cases}$$

wobei $X_{i,j}$ auf $[0, 1]$ gleichverteilte Zufallsvariable bezeichnet.

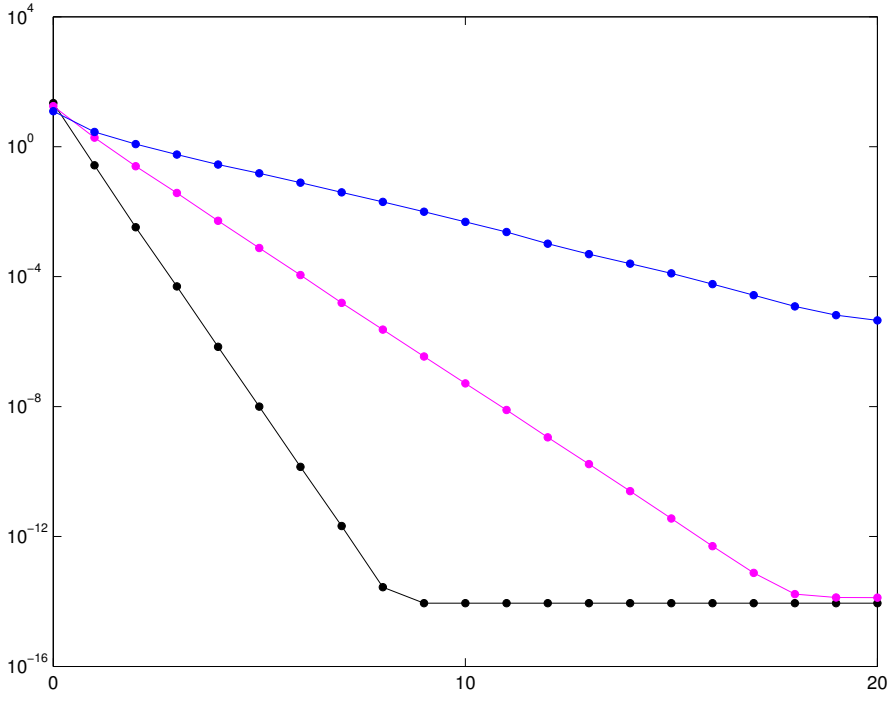
Man beachte: mit wachsendem τ nimmt die Anzahl von Null verschiedener Einträge in $\boldsymbol{A}$ ebenso zu wie die Konditionszahl. Für $\tau = 0.2$ ist $\lambda_{\min}(\boldsymbol{A}) < 0$, $\boldsymbol{A}$ also nicht mehr positiv definit. Wir erhalten so:

τ		$\mathtt{nnz}(\boldsymbol{A})$	$\lambda_{\min}(\boldsymbol{A})$	$\lambda_{\max}(\boldsymbol{A})$	$\mathrm{cond}_2(\boldsymbol{A})$	θ
0.01	(schwarz)	3016	0.9713	1.0325	1.0630	0.0153
0.05	(magenta)	12940	0.7150	1.6479	2.3049	0.2058
0.1	(blau)	25248	0.2097	3.5457	16.9059	0.6087
0.2	(cyan)	50510	−1.1677	11.1262	—	—

Wir wählen $\boldsymbol{b} = [1, 1, \ldots, 1]^\top$, $\boldsymbol{x}_0 = \boldsymbol{0}$ und führen 20 Schritte des CG-Verfahrens durch.



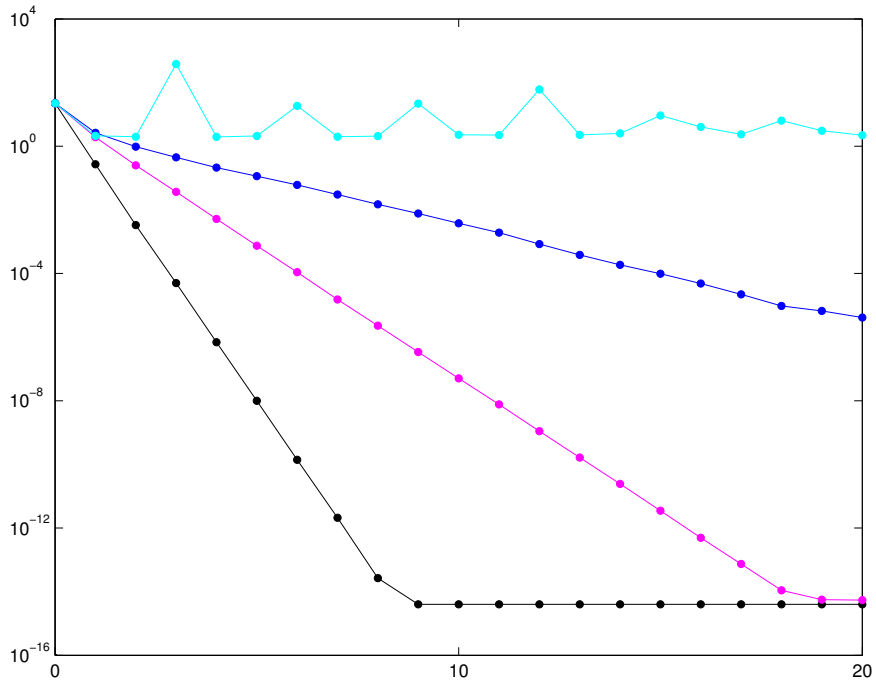
$\|\boldsymbol{x}_* - \boldsymbol{x}_m\|_2$



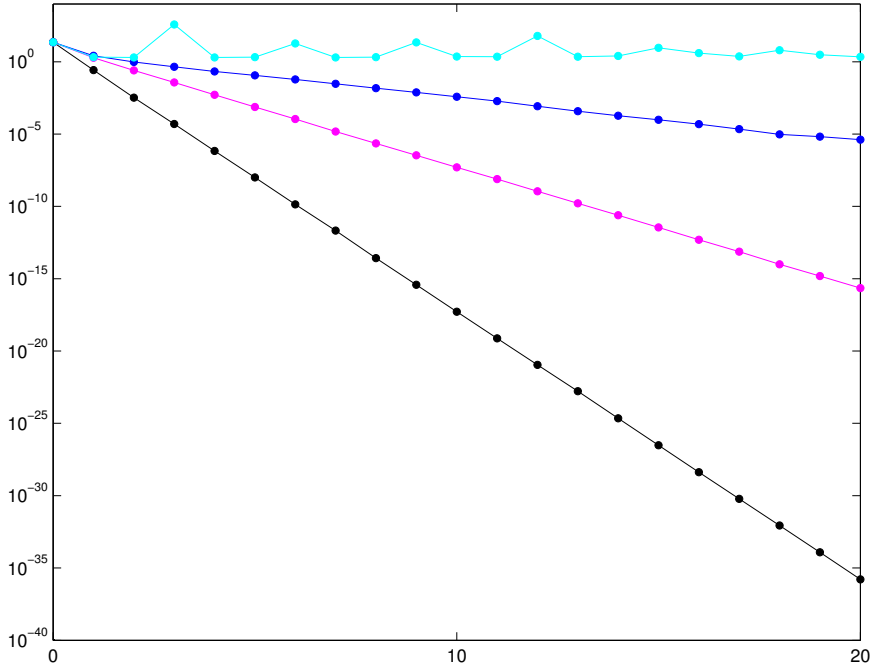
$\|\boldsymbol{x}_* - \boldsymbol{x}_m\|_A$

Beobachtungen:

- Divergenz im Fall $\tau = 0.2$.
- Im Fall $\tau = 0.1$ wird der Fehler nach 20 Schritten nur auf 10^{-5} reduziert, bei den übrigen Beispielen mit kleineren Konditionszahlen beobachten wir rasche Konvergenz.
- Für $\tau = 0.01$ erreicht der Fehler nach 9 Schritten den Bereich der Maschinengenauigkeit. In diesem Fall löst das CG-Verfahren $\boldsymbol{A}\boldsymbol{x} = \boldsymbol{b}$ in etwa 6×10^4 Flops und ist damit um den Faktor 700 schneller als die Cholesky-Zerlegung.



$\|\boldsymbol{r}_m\|_2$, mit $\boldsymbol{r}_m = \boldsymbol{b} - A\boldsymbol{x}_m$



$\|\boldsymbol{r}_m\|_2$, mit $\boldsymbol{r}_m = \boldsymbol{r}_{m-1} - \alpha_m A\boldsymbol{p}_m$

In der Praxis ist der Fehler nicht verfügbar und Abbruchkriterien basieren oft auf dem Residuum r_m. In exakter Arithmetik ist es einerlei, ob wir $\boldsymbol{r}_m = \boldsymbol{b} - \boldsymbol{A}\boldsymbol{x}_m$ oder die rekursive Formel aus dem Algorithmus auswerten, wobei die erste Variante eine weitere Matrix-Vekto- Multiplikation erfordert.

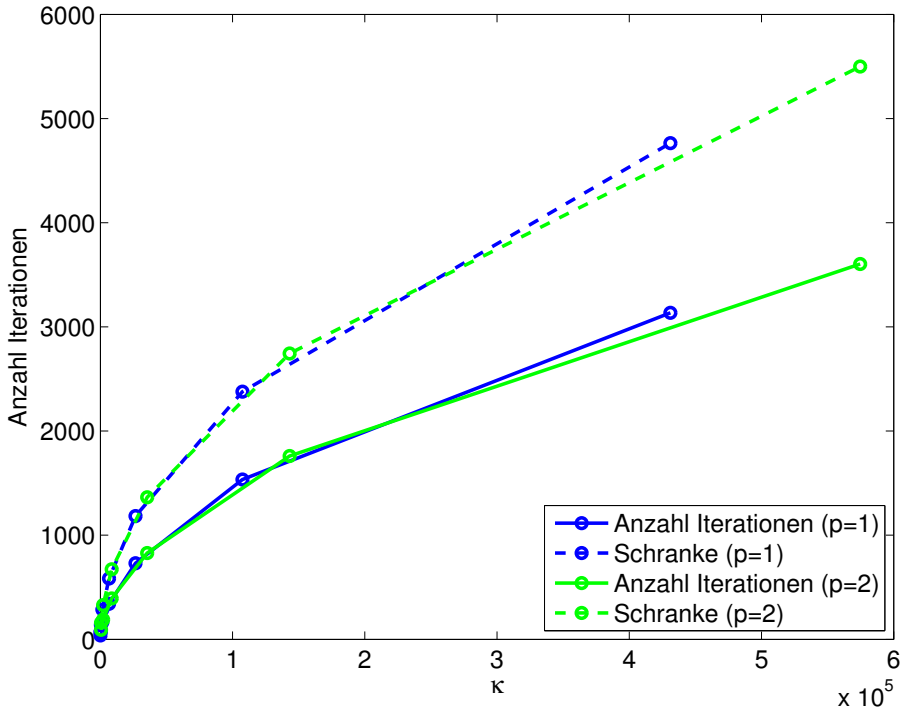
In Gleitpunktrechnung kann der Unterschied allerdings dramatisch sein.

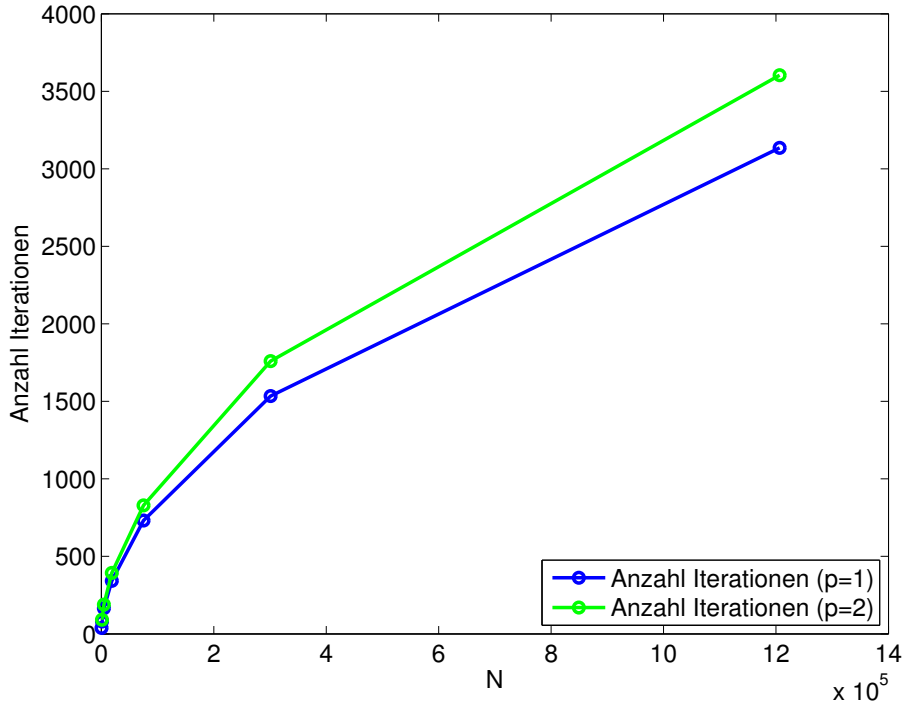
Die obere Schranke aus Satz 4.15 ist in vielerlei Hinsicht unbefriedigend:

- Sie spiegelt nicht den Einfluß von $\boldsymbol{r}_0$ (also $\boldsymbol{b}$ und $\boldsymbol{x}_0$) wider.
- Sie hängt nur von den Extremaleigenwerten ab; die Verteilung der Eigenwerte innerhalb des Intervalls $[\lambda_{\min}(\boldsymbol{A}), \lambda_{\max}(\boldsymbol{A})]$ bleibt unberücksichtigt.
- In vielen für die Praxis relevanten Fällen ist die schranke aus Satz 4.15 viel zu pessimistisch.

In vielen Fällen wird bei CG ein „superlineares“ Konvergenzverhalten beobachtet. Dies ist erst vor kurzem nathematisch zufriedenstellend erklärt worden.

Beispiel 2: Wir betrachten die FE-Diskretisierung des ersten Referenzbeispiels aus Abschnitt 3.2 (Poisson-Gleichung auf Einheitsquadrat, homogene Dirichlet-Randbedingung, Quellterm $\equiv 1$) mit linearen und quadratischen Dreieckelementen.


Anzahl Iterationen
κ
x 10^5
Anzahl Iterationen (p=1)
Schranke (p=1)
Anzahl Iterationen (p=2)
Schranke (p=2)


Anzahl Iterationen
N
x 10^5
Anzahl Iterationen (p=1)
Anzahl Iterationen (p=2)

4.2.4 Vorkonditionierung

Idee: Falls CG für gegebenes LGS zu langsam konvergiert, löse stattdessen ein äquivalentes, für das CG schneller konvergiert.

Ein solches ist gegeben durch

$$M^{-1}Ax = M^{-1}b, \qquad M \text{ geeignete spd Matrix.}$$

Die resultierenden Modifikationen des Algorithmus erhöhen den Aufwand um

- Die Lösung eines LGS mit der Matrix M in jedem Schritt,
- Die Speicherung eines weiteren Vektors z_m, des „vorkonditionierten Residuums“.

Algorithmus 5 : Vorkonditioniertes CG-Verfahren.

Gegeben : A spd, b, M spd

1 Bestimme Startnäherung x_0, berechne $r_0 \leftarrow b - Ax_0$

2 Löse $Mz_0 = r_0$

3 Setze $m \leftarrow 0$, $p_1 \leftarrow z_0$

4 **while** *nicht konvergiert* **do**

5 $\quad m \leftarrow m + 1$

6 $\quad \alpha_m \leftarrow \frac{(r_{m-1}, z_{m-1})}{(p_m, p_m)_A}$

7 $\quad x_m \leftarrow x_{m-1} + \alpha_m p_m$

8 $\quad r_m \leftarrow r_{m-1} - \alpha_m A p_m$

9 $\quad$ Löse $Mz_m = r_m$

10 $\quad \beta_{m+1} \leftarrow \frac{(r_m, z_m)}{(r_{m-1}, z_{m-1})}$

11 $\quad p_{m+1} \leftarrow z_m + \beta_{m+1} p_m$

Obwohl unverzichtbar in der Praxis ändert die Berücksichtigung von Vorkonditionierung wenig für die Konvergenzanalyse: es müssen lediglich folgende Ersetzungen vorgenommen werden:

$$\begin{aligned} \boldsymbol{A} &\leftarrow \boldsymbol{M}^{-1}\boldsymbol{A}, \\ \boldsymbol{b} &\leftarrow \boldsymbol{M}^{-1}\boldsymbol{b}, \\ (\boldsymbol{x},\boldsymbol{y}) &\leftarrow (\boldsymbol{x},\boldsymbol{y})_M := (\boldsymbol{M}\boldsymbol{x},\boldsymbol{y}). \end{aligned}$$

Beachte: die $\boldsymbol{M}^{-1}\boldsymbol{A}$-Norm bezüglich $(\cdot,\cdot)_M$ ist nichts als die $\boldsymbol{A}$-Norm bezüglich $(\cdot,\cdot)$. Das vorkonditionierte CG-Verfahren minimiert also weiterhin die $\boldsymbol{A}$-Norm der Fehlers.

Mit unvollständiger Cholesky-Zerlegung als Vorkonditionierer ergeben sich in Beispiel 2 für $p = 1$ folgende Iterationszahlen.