Einführung in die Grundlagen der Numerik

Vorlesungsskript WS 2013/14

Mario Bebendorf

Inhaltsverzeichnis

Vorwort

Dieses Skript fasst den Inhalt der von mir im Wintersemester 2013/14 an der Universität Bonn gehaltenen Vorlesung *Einführung in die Grundlagen der Numerik* des dritten Semesters im Bachelorstudiengang Mathematik zusammen. Korrekturvorschläge sind willkommen.

Bonn, 25. März 2014

Einleitung

Die Mathematik stellt eine wichtige Grundlage für viele Anwendungsbereiche des täglichen Lebens dar. Ingenieure, Logistikexperten und Ökonomen profitieren in gleicher Weise von mathematischen Methoden und Modellen. Jedoch kann nur ein Bruchteil der auftretenden Probleme analytisch gelöst werden, der Großteil ist mit Papier und Bleistift nicht zu bewältigen. Als Beispiel kann nicht einmal die Pendelbewegung eines dünnen Stabes geschlossen angegeben werden. Ein weiteres Beispiel ergibt sich aus dem Satz von Abel-Ruffini. Danach gibt es keine geschlossene Berechnungsformel für Eigenprobleme der Dimension $n \geq 5$. Um solch einfache Probleme lösen zu können, sind numerische Methoden unerlässlich. Bei komplexeren Problemstellungen nutzt man zur Umsetzung der entsprechenden numerischen Verfahren den Computer als Hilfsmittel. Der Hörer dieser Einführungsvorlesung lernt grundlegende Konzepte, Algorithmen und Methoden der numerischen Mathematik kennen. Er soll am Ende in der Lage sein, mit Hilfe der erworbenen Kenntnisse selbständig numerische Methoden problemorientiert zu entwickeln, zu analysieren und programmtechnisch umzusetzen.

Literaturangaben:

- P. Deuflhard und A. Hohmann: *Numerische Mathematik*, de Gruyter-Verlag
- P. Deuflhard und F. Bornemann: *Numerische Mathematik II*, de Gruyter-Verlag, 2002.
- M. Hanke-Bourgeois: *Grundlagen der numerischen Mathematik und des wissenschaftlichen Rechnens*, Teubner-Verlag, 2002
- J. Stoer: *Numerische Mathematik I*, Springer-Verlag
- A. Quarteroni, R. Sacco, F. Saleri: *Numerische Mathematik 1 & 2*, Springer-Verlag, 2002

1 Lineare Ausgleichsprobleme

Im Folgenden bezeichne $\mathbb{K}^{m\times n}$, $\mathbb{K} \in \{\mathbb{R}, \mathbb{C}\}$, den Raum der $m \times n$-Matrizen

$$A = \begin{bmatrix} a_{11} & \dots & a_{1n} \\ \vdots & & \vdots \\ a_{m1} & \dots & a_{mn} \end{bmatrix}, \quad a_{ij} \in \mathbb{K}.$$

Dabei verwenden wir die komponentenweise Addition und Multiplikation mit Skalaren. Für $n = 1$ ergibt sich der Raum der Spaltenvektoren $\mathbb{K}^m$.

Beispiel 1.1. Eine Matrix $A \in \mathbb{K}^{m\times n}$ heißt **Diagonalmatrix**, falls $a_{ij} = 0$ für $i \neq j$. Die Matrix $I \in \mathbb{K}^{n\times n}$ mit

$$I_{ij} = \delta_{ij} := \begin{cases} 1, & i = j, \\ 0, & \text{sonst}, \end{cases}$$

wird als **Identität** oder **Einheitsmatrix** bezeichnet. Die Vektoren $e_i \in \mathbb{K}^n$ mit $(e_i)_j = \delta_{ij}$ heißen **kanonische Einheitsvektoren**.

Zwischen Matrizen kann ein Produkt $\cdot : \mathbb{K}^{m\times p} \times \mathbb{K}^{p\times n} \to \mathbb{K}^{m\times n}$ durch

$$(A \cdot B)_{ij} = \sum_{\ell=1}^{p} a_{i\ell} b_{\ell j}, \quad i = 1, \dots, m,\ j = 1, \dots, n,$$

für $A \in \mathbb{K}^{m\times p}$, $B \in \mathbb{K}^{p\times n}$ definiert werden. Es gilt Assoziativität $(A \cdot B) \cdot C = A \cdot (B \cdot C)$. Den Multiplikationspunkt $\cdot$ lassen wir auch weg. Ferner bezeichnet

- $A^T \in \mathbb{K}^{n\times m}$ mit den Einträgen $(A^T)_{ij} = a_{ji}$ die **transponierte Matrix**,
- A^H mit $(A^H)_{ij} = \overline{a}_{ji}$ die **adjungierte Matrix** zu $A \in \mathbb{K}^{m\times n}$.

Gilt $A^T = A$ bzw. $A^H = A$, so heißt A **symmetrisch** bzw. **hermitesch**. Ferner gelten die Rechenregeln

$$(AB)^T = B^T A^T,\ (AB)^H = B^H A^H,\ (A + \lambda B)^T = A^T + \lambda B^T,\ (A + \lambda B)^H = A^H + \overline{\lambda} B^H$$

für $\lambda \in \mathbb{K}$ sowie $(A^T)^T = A = (A^H)^H$.

In diesem Kapitel beschäftigen wir uns mit linearen Gleichungssystemen

$$Ax = b, \tag{1.1}$$

wobei $A \in \mathbb{K}^{m\times n}$ und $b \in \mathbb{K}^m$. Im Fall $m > n$ bezeichnen wir (1.1) als **überbestimmtes Gleichungssystem**, für $m < n$ nennen wir es **unterbestimmt**. Probleme vom Typ (1.1) tauchen oft als Endproblem in vielen Anwendungen auf. (1.1) ist offenbar genau dann lösbar, wenn $b \in \operatorname{Ran} A$. Sei $x_0 \in \mathbb{K}^n$ eine Lösung, dann ist die Lösungsmenge von (1.1) gegeben durch

$$x_0 + \operatorname{Ker} A := \{x_0 + y,\ y \in \operatorname{Ker} A\}.$$

Hier und im Folgenden definieren wir

$$\operatorname{Ker} A = \{x \in \mathbb{K}^n : Ax = 0\} \quad \text{und} \quad \operatorname{Ran} A = \{Ax,\, x \in \mathbb{K}^n\}.$$

Insbesondere ist, falls $\operatorname{rank} A := \dim \operatorname{Ran} A = n$, x_0 die eindeutige Lösung von (1.1). Quadratische Matrizen $A \in \mathbb{K}^{n\times n}$ heißen **invertierbar**, falls $B \in \mathbb{K}^{n\times n}$ existiert mit $AB = BA = I$. Daher wird $B = A^{-1}$ als **Inverse** von A bezeichnet. A ist genau dann invertierbar, wenn $\operatorname{rank} A = n$ ist.

Eine äquivalente Charakterisierung der Lösbarkeit von (1.1) ergibt sich aus dem folgendem Lemma, in dem der **Orthogonalraum**

$$X^\perp = \{y \in \mathbb{K}^n : x^H y = 0 \text{ für alle } x \in X\}$$

von $X \subset \mathbb{K}^n$ verwendet wird. Man beachte, dass $(X^\perp)^\perp = X$, falls X ein Unterraum ist.

Lemma 1.2. *Sei $A \in \mathbb{K}^{m\times n}$. Dann gilt $(\operatorname{Ran} A)^\perp = \operatorname{Ker} A^H$.*

Beweis. Sei $x \in \operatorname{Ker} A^H$, d.h. $A^H x = 0$ und sei $z \in \operatorname{Ran} A$. Dann existiert $y \in \mathbb{K}^n$ mit $z = Ay$. Dann ist $z^H x = (Ay)^H x = y^H (A^H x) = 0$ und somit $x \in (\operatorname{Ran} A)^\perp$.
Mit der Umkehrung des Arguments erhält man die verbleibende Inklusion. □

Satz 1.3. *Das Gleichungssystem (1.1) ist genau dann lösbar, wenn $b^H y = 0$ für alle $y \in \mathbb{K}^m$ mit $A^H y = 0$.*

Beweis. Wir haben bereits bemerkt, dass (1.1) genau dann lösbar ist, wenn $b \in \operatorname{Ran} A$. Nach dem letzten Lemma gilt $\operatorname{Ran} A = ((\operatorname{Ran} A)^\perp)^\perp = (\operatorname{Ker} A^H)^\perp$. □

Ist die Lösbarkeit von (1.1) nicht gegeben, so suchen wir im Folgenden Vektoren $x \in \mathbb{K}^n$, so dass (1.1) bestmöglich gelöst wird, d.h. mit minimaler Norm

$$\|b - Ax\|_2 = \min_{y\in\mathbb{K}^n} \|b - Ay\|_2. \tag{1.2}$$

des **Residuums** $r := b - Ax$. Der Ausdruck $\|r\|_2$ ist ein Maß für den Fehler, und $\|r\|_2 = 0$ impliziert $Ax = b$.

Beispiel 1.4. Beim sog. "curve fitting" sind m Paare $(t_1, b_1), \ldots, (t_m, b_m) \in \mathbb{R}^2$ gegeben. Gesucht ist ein Polynom $p \in \Pi_{n-1}$, $n \le m$, für das der Ausdruck

$$\sum_{i=1}^{m} (b_i - p(t_i))^2$$

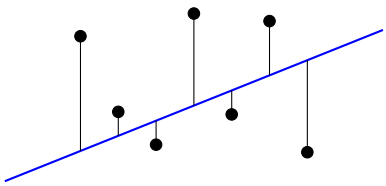
minimal ist, d.h. wir suchen ein Polynom, welches den z.B. aus einer Messung erhaltenen Datensatz (t_i, b_i), $i = 1, \ldots, m$, möglichst gut interpoliert. In der Darstellung

$$p(t) = \sum_{j=1}^{n} x_j t^{j-1}$$

für $p \in \Pi_{n-1}$ mit dem Koeffizientenvektor $x \in \mathbb{R}^n$ gilt es also, die euklidische Norm von

$$\begin{bmatrix} b_1 - p(t_1) \\ \vdots \\ b_m - p(t_m) \end{bmatrix} = \begin{bmatrix} b_1 \\ \vdots \\ b_m \end{bmatrix} - \begin{bmatrix} 1 & t_1 & \cdots & t_1^{n-1} \\ \vdots & & & \vdots \\ 1 & t_m & \cdots & t_m^{n-1} \end{bmatrix} \begin{bmatrix} x_1 \\ \vdots \\ x_n \end{bmatrix} = b - Ax$$

zu minimieren. Hierbei ist üblicherweise $m \gg n$, weil es nicht sinnvoll ist, mit Polynomen hohen Grades zu arbeiten. Im Fall $n = 2$ erhält man die sog. **Regressionsgerade**



Definition 1.5. *Bei gegebenem $A \in \mathbb{K}^{m\times n}$ und $b \in \mathbb{K}^m$ wird das Problem (1.2) als **lineares Ausgleichsproblem** (engl. least squares problem) bezeichnet. Jede Lösung $x \in \mathbb{K}^n$ heißt **Ausgleichslösung** oder **kleinste-Quadrate-Lösung**.*

Orthogonale Projektionen

Das Bestapproximationsproblem (1.2) steht in engem Zusammenhang mit der orthogonalen Projektion. Wir erinnern zunächst an die allgemeine Definition von Skalarprodukten.

Definition 1.6. *Sei V ein Vektorraum über $\mathbb{K}$. Eine Abbildung $(\cdot,\cdot) : V \times V \to \mathbb{K}$ heißt **Skalarprodukt**, falls*

(i) $(u, v) = \overline{(v, u)}$ *für alle* $u, v \in V$, *(Symmetrie)*

(ii) $(\alpha u + \beta v, w) = \alpha(u, w) + \beta(v, w)$ *für alle* $\alpha, \beta \in \mathbb{K}$, $u, v, w \in V$, *(Linearität)*

(iii) $(v, v) > 0$ *für alle* $v \in V \setminus \{0\}$. *(Definitheit)*

*Das Paar $(V, (\cdot,\cdot))$ wird als **Prä-Hilbert-Raum** bezeichnet. Ist V bzgl. der durch das Skalarprodukt induzierten Norm*

$$\|v\| := \sqrt{(v, v)}, \quad v \in V,$$

*vollständig, so heißt $(V, (\cdot,\cdot))$ **Hilbert-Raum**.*

Beispiel 1.7.

(i) Für $V = \mathbb{C}^n$, $\mathbb{K} = \mathbb{C}$ haben wir bereits das Skalarprodukt

$$(u, v) := v^H u = \sum_{i=1}^{n} u_i \overline{v}_i$$

verwendet.

(ii) Sei $L^2[a,b] = \{f : [a,b] \to \mathbb{C} : \int_a^b |f(x)|^2 \,\mathrm{d}x < \infty\}$ die Menge der auf dem Intervall $[a,b]$ quadratisch Lebesgue-integrierbaren Funktionen. Dann definiert

$$(f,g) := \int_a^b f(t)\overline{g(t)}\,\mathrm{d}t$$

ein Skalarprodukt auf $V = L^2[a,b]$.

Sei $\{v_1, v_2, \dots\}$ eine Orthonormalbasis eines Hilbert-Raums V. Dann werden die Projektionen (u, v_i), $i = 1, 2, \dots$, auf die Vektoren der Orthonormalbasis als **Fourier-Koeffizienten** von $u \in V$ bezeichnet.

Satz 1.8. *Sei V ein Hilbert-Raum und $V_n \subset V$ ein endlichdimensionaler Teilraum mit Orthonormalbasis $\{v_1, \dots, v_n\}$. Dann gilt*

(i) Für $u \in V_n$ gilt $u = \sum_{i=1}^n (u, v_i) v_i$.

*(ii) Es gilt die **Parsevalsche Gleichung***

$$\|u\|^2 = \sum_{i=1}^n |(u, v_i)|^2 \quad \text{für alle } u \in V_n.$$

*(iii) Für alle $u \in V$ gilt die **Besselsche Ungleichung***

$$\sum_{i=1}^n |(u, v_i)|^2 \le \|u\|^2.$$

(iv) Die Bestapproximation von $u \in V$ in V_n ist

$$u_n := \sum_{i=1}^n (u, v_i) v_i \in V_n$$

d.h. $\|u - u_n\| \le \|u - v\|$ für alle $v \in V_n$.

Beweis. zu (i): Für $u \in V_n$ gilt $u = \sum_{i=1}^n \alpha_i v_i$ mit geeigneten $\alpha_i \in \mathbb{K}$. Daher folgt

$$(u, v_i) = (\sum_{j=1}^n \alpha_j v_j, v_i) = \sum_{j=1}^n \alpha_j (v_j, v_i) = \alpha_i.$$

zu (ii): Aus (i) folgt

$$\|u\|^2 = (u,u) = (\sum_{i=1}^n \alpha_i v_i, \sum_{j=1}^n \alpha_j v_j) = \sum_{i,j=1}^n \alpha_i \overline{\alpha}_j (v_i, v_j) = \sum_{i=1}^n |\alpha_i|^2 = \sum_{i=1}^n |(u, v_i)|^2.$$

zu (iv): Für beliebiges $v = \sum_{i=1}^n \beta_i v_i \in V_n$ gilt

$$\|u - v\|^2 = \|u\|^2 - 2\,\mathrm{Re} \sum_{i=1}^n \overline{\beta_i}(u, v_i) + \sum_{i=1}^n |\beta_i|^2 = \|u\|^2 - \|u_n\|^2 + \sum_{i=1}^n |(u, v_i) - \beta_i|^2. \quad (1.3)$$

Der letzte Ausdruck wird offenbar genau dann minimal, wenn $\beta_i = (u, v_i)$, $i = 1, \ldots, n$.
zu (iii): Für $u \in V$ und $\beta_i := (u, v_i)$ erhält man analog zu (1.3)

$$0 \leq \|u - \sum_{i=1}^{n} \beta_i v_i\|^2 = \|u\|^2 - 2\,\mathrm{Re} \sum_{i=1}^{n} \overline{\beta}_i \beta_i + \sum_{i=1}^{n} |\beta_i|^2 = \|u\|^2 - \sum_{i=1}^{n} |\beta_i|^2.$$

□

Bemerkung. Satz 1.8 zeigt insbesondere die Bedeutung von Orthonormalbasen in der Numerik. Durch die Fourierkoeffizienten erhält man eine explizite Basisdarstellung und unmittelbar die Bestapproximation. Ferner ist die Bestapproximation von $v \in V$ stabil, weil nach der Besselschen Ungleichung gilt $|(u, v_i)|^2 \leq \sum_{i=1}^{n} |(u, v_i)|^2 \leq \|u\|^2$ und somit die Fourierkoeffizienten (u, v_i), $i = 1, \ldots, n$, der Approximation beschränkt sind.

Für das folgende Lemma überzeugt man sich leicht davon, dass in Prä-Hilbert-Räumen die **Parallelogramm-Gleichung**

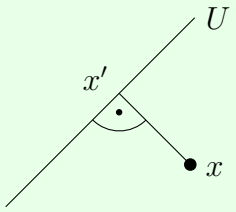
$$2\|u\|^2 + 2\|v\|^2 = \|u + v\|^2 + \|u - v\|^2, \quad u, v \in V, \tag{1.4}$$

gilt.

Lemma 1.9 (Projektionssatz). *Sei V ein Hilbert-Raum und $U \subset V$ eine abgeschlossene und konvexe Menge. Dann existiert zu $x \in V$ das eindeutig bestimmte Element bester Approximation $x' \in U$, d.h. es gilt*

$$\|x - x'\| = \inf_{y \in U} \|x - y\|.$$

Ist $U = u + L$ ein affiner Unterraum mit $u \in V$ und einem linearen Unterraum $L \subset V$, dann erfüllt genau x' die Bedingung $(x - x', z) = 0$ für alle $z \in L$.



Beweis. Sei $d = \inf_{y \in U} \|x - y\|$ und $\{y_n\}_{n \in \mathbb{N}} \subset U$ eine Folge mit $\lim_{n \to \infty} \|x - y_n\| = d$. Dann existiert zu $\varepsilon > 0$ ein $N \in \mathbb{N}$, so dass $\|x - y_n\|^2 \leq d^2 + \varepsilon$ für alle $n \geq N$. Mit der Parallelogramm-Gleichung (1.4) erhalten wir für $m, n \geq N$

$$\|y_m - y_n\|^2 = 2\|x - y_m\|^2 + 2\|x - y_n\|^2 - \|y_m + y_n - 2x\|^2 \leq 4d^2 + 4\varepsilon - 4d^2 = 4\varepsilon,$$

weil aus $\frac{1}{2}(y_m + y_n) \in U$ folgt $\|\frac{1}{2}(y_m + y_n) - x\| \geq d$. Also ist $\{y_n\}_{n \in \mathbb{N}}$ eine Cauchy-Folge, die wegen der Abgeschlossenheit von U gegen ein $x' \in U$ konvergiert. Dies ist ein gesuchtes Minimum. Ist $x'' \in U$ ein weiteres Element bester Approximation, so gilt wie oben

$$\|x' - x''\| = 2\|x - x'\|^2 + 2\|x - x''\|^2 - \|x' + x'' - 2x\|^2 \leq 4d^2 - 4d^2 = 0,$$

woraus die Eindeutigkeit folgt.

Angenommen, es gibt ein $z \in L$ mit $\alpha := (x - x', z) \neq 0$. Betrachte $v := x' + \alpha z/\|z\|^2 \in U$. Dann gilt

$$\begin{aligned}\|x - v\|^2 &= \|x - x' - \frac{\alpha}{\|z\|^2} z\|^2 = \|x - x'\|^2 - 2\,\mathrm{Re}\,\frac{\overline{\alpha}}{\|z\|^2}(x - x', z) + \frac{|\alpha|^2}{\|z\|^2} \\ &= \|x - x'\|^2 - \frac{|\alpha|^2}{\|z\|^2} < \|x - x'\|^2.\end{aligned}$$

Aus diesem Widerspruch folgt $(x - x', z) = 0$ für alle $z \in L$. Ist umgekehrt $(x - x', z) = 0$ für alle $z \in L$, so folgt wegen $x' - y \in L$ für alle $y \in U$

$$\begin{aligned}\|x - y\|^2 &= \|x - x'\|^2 + 2\,\mathrm{Re}\,(x - x', x' - y) + \|x' - y\|^2 \\ &= \|x - x'\|^2 + \|x' - y\|^2 \geq \|x - x'\|^2.\end{aligned}$$

□

Die Abbildung $x \mapsto x'$ definiert einen Projektor. Wir beschränken uns auf den Fall $V = \mathbb{K}^n$.

Definition 1.10. *Ein **Projektor** $P : \mathbb{K}^n \to \mathbb{K}^n$ ist eine lineare Abbildung, die idempotent ist, d.h. es gilt $P^2 = P$.*

Mit P ist auch $I - P$ ein Projektor, und es gilt $P(I - P) = (I - P)P = 0$,

$$\mathrm{Ker}\, P = \mathrm{Ran}\,(I - P) \quad \text{und} \quad \mathrm{Ker}\, P \cap \mathrm{Ran}\, P = \{0\}. \tag{1.5}$$

Wegen $x = Px + (I - P)x$ für alle $x \in \mathbb{K}^n$ gilt daher

$$\mathbb{K}^n = \mathrm{Ker}\, P \oplus \mathrm{Ran}\, P.$$

Definition 1.11. *Sei $U \subset \mathbb{K}^n$ ein Unterraum. Eine Abbildung $P : \mathbb{K}^n \to U$ heißt **orthogonaler Projektor** auf U, falls P ein Projektor ist mit $(x - Px, u) = 0$ für alle $u \in U$ und $x \in \mathbb{K}^n$.*

Für den folgenden Satz benötigen wir das

Lemma 1.12. *Seien $X, Y \in \mathbb{K}^{n\times r}$. Die **Gramsche Matrix** $G \in \mathbb{K}^{r\times r}$ sei durch*

$$g_{ij} = (x_i, y_j), \quad i, j = 1, \ldots, r,$$

definiert. Dabei bezeichnet x_i die i-te Spalte von X und y_j die j-te Spalte von Y. G ist genau dann regulär, wenn $\mathrm{rank}\, X = r$ *und* $\mathrm{span}\, X \cap (\mathrm{span}\, Y)^\perp = \{0\}$.

Beweis. Angenommen, eine nicht-triviale Linearkombination der Zeilen von G verschwindet, d.h. es existieren Koeffizienten α_i, $i = 1, \ldots, r$, so dass mit $x := \sum_{i=1}^r \alpha_i x_i \in \mathrm{span}\, X$ gilt

$$(x, y_j) = \sum_{i=1}^r \alpha_i (x_i, y_j) = 0 \quad \text{für alle } j = 1, \ldots, r.$$

Also ist $x \in (\operatorname{span} Y)^\perp$ und aus $\operatorname{span} X \cap (\operatorname{span} Y)^\perp = \{0\}$ folgt $x = 0$. Dies liefert einen Widerspruch zur linearen Unabhängigkeit der Spalten von X.

Sei umgekehrt G regulär, dann sind die Spalten von X offensichtlich linear unabhängig. Sei $x \in \operatorname{span} X$ mit $(x, y_j) = 0$, $j = 1, \ldots, r$. Aus der Basisdarstellung $x = \sum_{i=1}^r \alpha_i x_i$ mit Koeffizienten α_i, $i = 1, \ldots, r$, erhält man

$$\sum_{i=1}^r \alpha_i g_{ij} = (x, y_j) = 0 \quad \text{für alle } j = 1, \ldots, r,$$

und hieraus wegen der Invertierbarkeit von G, dass $\alpha_i = 0$, $i = 1, \ldots, r$. Also folgt $\operatorname{span} X \cap (\operatorname{span} Y)^\perp = \{0\}$. □

Satz 1.13. *Sei $P : \mathbb{K} \to U$ ein orthogonaler Projektor auf den Unterraum $U \subset \mathbb{K}^n$. Dann gilt*

(i) P ist eindeutig bestimmt;

(ii) Ist $x' \in U$ die Bestapproximation an $x \in \mathbb{K}^n$ aus Lemma 1.9, so gilt $Px = x'$;

(iii) Ist $X = [x_1, \ldots, x_k]$ eine Basis von U, so gilt

$$P = X(X^H X)^{-1} X^H.$$

Insbesondere ist P hermitesch.

Beweis. Wegen

$$\|x - y\|^2 = \|x - Px\|^2 + \|Px - y\|^2 \quad \text{für } y \in U \tag{1.6}$$

folgt $\|x-y\| \geq \|x-Px\|$. Das Minimum wird für $y = Px$ angenommen. Aus der Eindeutigkeit von x' folgen (i) und (ii). Für (iii) erhält man aus Lemma 1.12, dass $X^H X$ regulär ist. Man prüft leicht nach, dass $X(X^H X)^{-1} X^H$ ein orthogonaler Projektor auf U ist, der wegen (i) mit P übereinstimmt. □

Bemerkung.

(a) Es gilt auch die Umkehrung von (iii), d.h. jeder hermitesche Projektor ist ein orthogonaler Projektor. Dies folgt mit (1.5) und Lemma 1.2 aus

$$\operatorname{Ran}(I - P) = \operatorname{Ker} P = \operatorname{Ker} P^H = (\operatorname{Ran} P)^\perp.$$

(b) Sei P ein orthogonaler Projektor. Dann folgt aus (1.6) für $y = 0$, dass $\|Px\| \leq \|x\|$. Ferner folgt

$$\|P\| = \sup_{0 \neq x \in \mathbb{K}^n} \frac{\|Px\|}{\|x\|} = 1,$$

weil das Maximum für die Elemente in $\operatorname{Ran} P$ angenommen wird. P besitzt nur die zwei Eigenwerte 0 und 1. Jeder nicht-triviale Kernvektor ist ein Eigenvektor zum Eigenwert 0, jeder nicht-triviale Bildvektor ein Eigenvektor zum Eigenwert 1.

1.1 Die Normalengleichungen

Im Folgenden werden wir das Problem (1.2) als lineares Gleichungssystem umformulieren.

Satz 1.14. *Alle Lösungen $x \in \mathbb{K}^n$ von (1.2) sind genau die Lösungen der sog.* ***Normalengleichungen***

$$A^H A x = A^H b. \tag{1.7}$$

Die Lösungsmenge L ist nicht leer, und es gilt für $x_1, x_2 \in L,\ \lambda \in \mathbb{K}$, dass

$$(1-\lambda)x_1 + \lambda x_2 \in L, \quad Ax_1 = Ax_2.$$

Insbesondere ist das Residuum eindeutig bestimmt.

Beweis. Nach Lemma 1.9 angewendet auf $U = \operatorname{Ran} A$ gilt

$$\|b - Ax\|_2 = \min_{y \in \mathbb{K}^n} \|b - Ay\|_2 \iff r = b - Ax \perp \operatorname{Ran} A.$$

Nach Lemma 1.2 ist $(\operatorname{Ran} A)^\perp = \operatorname{Ker} A^H$. Also ist (1.2) äquivalent mit

$$A^H r = 0 \iff A^H(b - Ax) = 0 \iff A^H A x = A^H b.$$

Nach Lemma 1.9 existiert eine Bestapproximation Ax an b und ist eindeutig bestimmt. $Ax_1 = Ax_2$ für $x_1, x_2 \in L$ ist wegen der Eindeutigkeit von Ax klar, $(1-\lambda)x_1 + \lambda x_2 \in L$ kann leicht nachgeprüft werden. □

Bemerkung.

(a) Ist $Ax = b$ lösbar, so liefert (1.7) die Lösungsmenge.

(b) Aus $A^H r = 0$ erkennt man, dass das Residuum senkrecht auf den Spalten von A steht. Daraus erklärt sich der Begriff "Normalengleichungen".

(c) (1.7) kann auch aus der Bedingung $\nabla f = 0$ mit $f(x) = \|b - Ax\|_2^2$ hergeleitet werden.

Hinsichtlich der Eindeutigkeit der Lösung von (1.7) gilt

Satz 1.15.

(i) Unter allen Lösungen von (1.7) gibt es genau eine Lösung minimaler euklidischer Norm.

(ii) Die Matrix $A^H A \in \mathbb{K}^{n \times n}$ ist hermitesch und positiv semidefinit. $A^H A$ ist genau dann positiv-definit (und damit (1.7) eindeutig lösbar), falls $\operatorname{rank} A = n$.

Beweis.

(i) Betrachte die Lösungsmenge L von (1.7). Diese ist abgeschlossen und

$$L' := L \cap \{x \in \mathbb{K}^n : \|x\|_2 \leq \|\hat{x}\|_2\}$$

ist kompakt, wobei $\hat{x} \in L$ ein beliebiges aber festes Element bezeichnet. Daher nimmt die stetige Funktion $\|\cdot\|_2$ auf L' ihr Minimum $x^* \in L$ mit $\|x^*\|_2 = \min_{y \in L'} \|y\|_2$ an. Sei $x' \in L$ ein weiteres Element minimaler euklidischer Norm. Dann gilt nach Satz 1.14 $\frac{1}{2}(x^* + x') \in L$ und

$$\|x^*\|_2 \leq \|\frac{1}{2}(x^* + x')\|_2 \leq \frac{1}{2}(\|x^*\|_2 + \|x'\|_2) = \|x^*\|_2.$$

Daher ist $\|\frac{1}{2}(x^* + x')\|_2 = \|x^*\|_2 = \|x'\|_2$, und es folgt

$$\begin{aligned} \|x' - x^*\|_2^2 &= \|x'\|_2^2 - 2\,\mathrm{Re}\,(x^*, x') + \|x^*\|_2^2 \\ &= \|x'\|_2^2 - (\|x' + x^*\|_2^2 - \|x'\|_2^2 - \|x^*\|_2^2) + \|x^*\|_2^2 \\ &= 2(\|x^*\|_2^2 + \|x'\|_2^2) - \|x^* + x'\|_2^2 = 0, \end{aligned}$$

was $x' = x^*$ und somit die Eindeutigkeit von x^* beweist.

(ii) $A^H A$ ist offenbar hermitesch und wegen

$$x^H A^H A x = \|Ax\|_2^2 \geq 0 \quad \text{für alle } x \in \mathbb{K}^n$$

positiv semidefinit. Gleichheit gilt genau für $Ax = 0$. In diesem Fall ist aber $x = 0$ wegen $\operatorname{rank} A = n$.

□

Bemerkung. Nach Satz 1.15 (ii) können im Fall $\operatorname{rank} A = n$ die Normalengleichungen (1.7) mit Hilfe der Cholesky-Zerlegung gelöst werden. Für $m < n$ ist allerdings die Bedingung $\operatorname{rank} A = n$ nicht realisierbar. In diesem Fall betrachtet man $AA^H y = b$ mit $x = A^H y$ und setzt $\operatorname{rank} A = m$ voraus. Dann ist AA^H positiv-definit.

Obwohl die Lösung mittels Cholesky-Zerlegung aus Sicht der Komplexität attraktiv erscheint ($mn^2 + \frac{1}{3}n^3$ Operationen), eignet sich diese für praktische Zwecke nur bedingt, weil $A^H A$ bzw. AA^H deutlich schlechtere Konditionen (die Kondition wird quadriert) als A aufweisen. Im Folgenden werden wir Verfahren herleiten, die ohne die Normalengleichungen auskommen.

1.2 Die QR-Zerlegung

Wir haben im letzten Abschnitt gesehen, dass lineare Ausgleichsprobleme mit Hilfe der Normalengleichungen und im Fall $m \geq n = \operatorname{rank} A$ mit Hilfe der Cholesky-Zerlegung gelöst werden können. Eine numerisch stabile Alternative stellt der folgende Zugang über die *QR-Zerlegung* dar.

Definition 1.16. *Sei $A \in \mathbb{K}^{m \times n}$. Eine Zerlegung der Form $A = QR$ mit unitärem $Q \in \mathbb{K}^{m \times m}$ und einer oberen Dreiecksmatrix $R \in \mathbb{K}^{m \times n}$ heißt* ***QR-Zerlegung*** *von A.*

Bemerkung.

(a) Man beachte, dass es genügt, eine Matrix $\hat{Q} \in \mathbb{K}^{m\times n}$ mit $\hat{Q}^H\hat{Q} = I \in \mathbb{K}^{n\times n}$ und eine obere Dreiecksmatrix $\hat{R} \in \mathbb{K}^{n\times n}$ zu bestimmen. Man spricht von einer **reduzierten QR-Zerlegung** im Gegensatz zur oben definierten **vollen QR-Zerlegung**.

(b) Die QR-Zerlegung kann alternativ zur LR-Zerlegung zur Lösung von linearen Gleichungssystemen verwendet werden. Wegen $Q^{-1} = Q^H$ gilt nämlich

$$Ax = b \iff Rx = Q^H b,$$

was mittels Rückwärtseinsetzen gelöst werden kann.

Sei $m \geq n = \operatorname{rank} A$ und $A = QR$ mit $Q \in \mathbb{K}^{m\times m}$ unitär und $R \in \mathbb{K}^{m\times n}$ in der Darstellung

$$R = \begin{bmatrix} \hat{R} \\ 0 \end{bmatrix} \tag{1.8}$$

mit einer oberen Dreiecksmatrix $\hat{R} \in \mathbb{K}^{n\times n}$. Aus $\operatorname{rank} A = n$ folgt, dass $\hat{R}$ regulär ist. Wegen

$$\|b - Ax\|_2 = \|b - QRx\|_2 = \|Q^H(b - QRx)\|_2 = \|Q^H b - Rx\|_2 = \left\| Q^H b - \begin{bmatrix} \hat{R}x \\ 0 \end{bmatrix} \right\|_2$$

kann die eindeutige Lösung des linearen Ausgleichsproblems (1.2) durch Lösung von

$$\hat{R}x = c \tag{1.9}$$

mittels Rückwärtseinsetzen bestimmt werden. Hierbei ist

$$\begin{bmatrix} c \\ d \end{bmatrix} := Q^H b, \quad c \in \mathbb{K}^n.$$

Dann gilt

$$\min_{y\in\mathbb{K}^n} \|b - Ay\|_2^2 = \min_{y\in\mathbb{K}^n} \left\| \begin{bmatrix} c - \hat{R}y \\ d \end{bmatrix} \right\|_2^2 = \min_{y\in\mathbb{K}^n} \|c - \hat{R}y\|_2^2 + \|d\|_2^2 = \|d\|_2^2.$$

Theorem 1.17. *Jedes $A \in \mathbb{K}^{m\times n}$ mit $m \geq n = \operatorname{rank} A$ besitzt eine eindeutige reduzierte QR-Zerlegung $A = \hat{Q}\hat{R}$ mit $\hat{r}_{ii} > 0$, $i = 1, \ldots, n$.*

Beweis. Die Eindeutigkeit folgt induktiv aus der Gleichung

$$A = \hat{Q}\hat{R} \iff a_k = \sum_{i=1}^{k} \hat{r}_{ik}\hat{q}_i, \quad k = 1, \ldots, n.$$

□

1.2.1 Die QR-Zerlegung nach Householder

Im Folgenden werden wir eine Möglichkeit kennen lernen, eine QR-Zerlegung zu berechnen. Insbesondere folgt die Existenz der QR-Zerlegung.

Definition 1.18. *Jede Matrix $Q \in \mathbb{K}^{n\times n}$ der Form*

$$Q = I - \beta \frac{uu^H}{u^H u}$$

mit einem $u \in \mathbb{K}^n \setminus \{0\}$ und einem $\beta \in \mathbb{K}\setminus\{0\}$, $|\beta|^2 = 2\operatorname{Re}\beta$, wird als ***Householder-Matrix*** *bezeichnet.*

Lemma 1.19. *Sei Q eine Householder-Matrix. Dann gilt*

(i) Q ist unitär;

(ii) Ist $\mathbb{K} = \mathbb{R}$, so gilt $x - Qx \in \operatorname{span}\{u\}$ und $\frac{1}{2}(x + Qx) \in (\operatorname{span}\{u\})^\perp$. Also spiegelt Q den Vektor $x \neq 0$ an der Hyperebene senkrecht zu u. Daher spricht man auch von Householder-Spiegelungen;

(iii) Sei $x \in \mathbb{K}^n$ mit $|x_1| \neq \|x\|_2$ gegeben. Für $u := x - \alpha e_1$, $|\alpha| = \|x\|_2$, und $\beta := 1 + \frac{x^H u}{u^H x}$ gilt $Qx = \alpha e_1$.

Beweis.

(i) Dass Q unitär ist, erkennt man wegen $|\beta|^2 = 2\operatorname{Re}\beta$ aus

$$Q^H Q = \left(I - \beta\frac{uu^H}{u^H u}\right)^H \left(I - \beta\frac{uu^H}{u^H u}\right) = I - \overline{\beta}\frac{uu^H}{u^H u} - \beta\frac{uu^H}{u^H u} + |\beta|^2\frac{u(u^H u)u^H}{(u^H u)^2} = I.$$

(ii) Ist $\mathbb{K} = \mathbb{R}$, so gilt $\beta^2 = 2\beta \Leftrightarrow \beta = 2$. Dann gilt

$$x - Qx = x - \left(x - 2\frac{u(u^T x)}{u^T u}\right) = 2\frac{u^T x}{u^T u}u$$

und

$$u^T(x + Qx) = u^T x + u^T x - 2\frac{(u^T u)u^T x}{u^T u} = 0.$$

(iii) Zunächst überzeugt man sich leicht davon, dass für diese Wahl $|\beta|^2 = 2\operatorname{Re}\beta$ gilt. Wegen $u^H x = (x - \alpha e_1)^H x = \|x\|_2^2 - \overline{\alpha}x_1 \neq 0$ und

$$u^H u = (x - \alpha e_1)^H (x - \alpha e_1) = \|x\|_2^2 - \alpha\overline{x}_1 - \overline{\alpha}x_1 + |\alpha|^2 = 2[\|x\|_2^2 - \operatorname{Re}(\overline{\alpha}x_1)]$$

folgt

$$\begin{aligned} Qx &= x - \beta\frac{u^H x}{u^H u}u = x - \frac{u^H x}{u^H u}u - \frac{x^H u}{u^H u}u = x - 2\,\frac{\operatorname{Re} u^H x}{u^H u}u \\ &= x - \frac{\operatorname{Re}(\|x\|_2^2 - \overline{\alpha}x_1)}{\|x\|_2^2 - \operatorname{Re}(\overline{\alpha}x_1)}u = x - u = \alpha e_1. \end{aligned}$$

□

Bemerkung. Um Auslöschungseffekte zu vermeiden, sollte α in (iii) das entgegengesetzte Vorzeichen der ersten Komponente x_1 von x bekommen. Es sollte also

$$\alpha = -\frac{x_1}{|x_1|}\|x\|_2$$

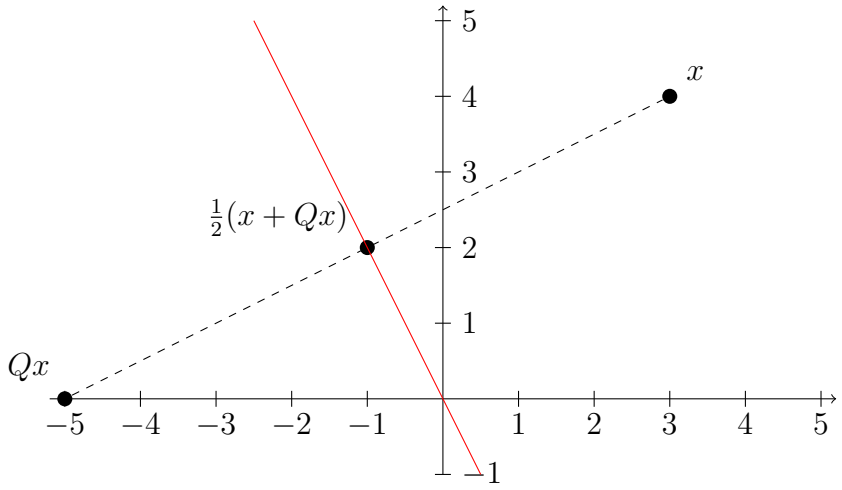
gewählt werden. Außerdem ist für $\mathbb{K} = \mathbb{R}$ natürlich $\beta = 2$.

Beispiel 1.20. Der Punkt $x = [3, 4]^T \in \mathbb{R}^2$ soll durch Anwendung einer Householder-Spiegelung auf die x-Achse transformiert werden. Nach Lemma 1.19 (iii) wird dies durch

$$Q = I - \frac{1}{40}\begin{bmatrix}8\\4\end{bmatrix}\begin{bmatrix}8\\4\end{bmatrix}^T$$

erreicht. Tatsächlich gilt

$$Qx = \begin{bmatrix}3\\4\end{bmatrix} - \frac{1}{40}\begin{bmatrix}8\\4\end{bmatrix}\begin{bmatrix}8\\4\end{bmatrix}^T\begin{bmatrix}3\\4\end{bmatrix} = \begin{bmatrix}3\\4\end{bmatrix} - \begin{bmatrix}8\\4\end{bmatrix} = \begin{bmatrix}-5\\0\end{bmatrix}.$$



Wir werden nun $A \in \mathbb{K}^{m\times n}$, $m \geq n$, mit linear unabhängigen Spalten durch eine Folge von Householder-Spiegelungen in obere Dreiecksform bringen. Wie in Lemma 1.19 (iii) kann die erste Spalte von A durch Multiplikation mit einer Householder-Matrix $Q_1 \in \mathbb{K}^{m\times m}$ von links in ein Vielfaches des ersten kanonischen Einheitsvektors e_1 transformiert werden:

$$A^{(1)} := Q_1 A, \quad A^{(1)}e_1 = \alpha_1 e_1.$$

Sei nun angenommen, dass die Matrix A durch $k-1$ sukzessive Anwendungen von einer Householder-Matrix in folgende partielle obere Dreiecksform gebracht wurde:

$$A^{(k-1)} = Q_{k-1}\cdot\ldots\cdot Q_1 A = \begin{bmatrix} a_{11}^{(1)} & \cdots & \cdots & \cdots & a_{1n}^{(1)} \\ & \ddots & & & \vdots \\ & & a_{kk}^{(k-1)} & \cdots & a_{kn}^{(k-1)} \\ 0 & & \vdots & & \vdots \\ & & a_{mk}^{(k-1)} & \cdots & a_{mn}^{(k-1)} \end{bmatrix}.$$

Im k-ten Schritt soll $A^{(k-1)}$ so transformiert werden, dass $Q_k A^{(k-1)}$ bis zur k-ten Spalte eine obere Dreiecksmatrix ist. Um die ersten $k-1$ Zeilen und Spalten unverändert zu lassen, wählen wir

$$Q_k = \begin{bmatrix} I_{k-1} & 0 \\ 0 & \hat{Q}_k \end{bmatrix},$$

wobei $\hat{Q}_k \in \mathbb{K}^{(m-k+1)\times(m-k+1)}$ eine Householder-Matrix ist, die den Vektor

$$\begin{bmatrix} a_{kk}^{(k-1)} \\ \vdots \\ a_{mk}^{(k-1)} \end{bmatrix}$$

auf ein Vielfaches von $e_1 \in \mathbb{K}^{m-k+1}$ transformiert, falls dies nicht schon der Fall ist. Weil $\hat{Q}_k$ unitär ist, trifft dies auch auf Q_k zu. Nach n Schritten erhält man also die obere Dreiecksmatrix

$$A^{(n)} = Q_n \cdot \ldots \cdot Q_1 A = \begin{bmatrix} a_{11}^{(1)} & \cdots & a_{1n}^{(1)} \\ & \ddots & \vdots \\ & & a_{nn}^{(n)} \\ & & \end{bmatrix} =: R.$$

Setzen wir also $Q := Q_1^H \cdot \ldots \cdot Q_n^H$, so erhalten wir die gewünschte Zerlegung $A = QR$.

Bemerkung.

(a) Für die QR-Zerlegung von $A \in \mathbb{K}^{m\times n}$ mittels Householder-Spiegelungen werden $2mn^2 - \frac{2}{3}n^3$ arithmetische Operationen benötigt. Somit ist dieser Zugang etwa doppelt so teuer wie die Lösung der Normalengleichungen (1.7) mittels Cholesky-Zerlegung. Weil unitäre Matrizen die euklidische Norm eines Vektors erhalten, stimmen die Konditionen von A und $\hat{R}$ aber überein. Dabei verweisen wir auf die Definition der Kondition für rechteckige Matrizen am Ende dieses Kapitels. Anders als bei den Normalengleichungen wird somit bei diesem Zugang über die QR-Zerlegung die Kondition nicht erhöht.

(b) Die Einträge von Q sollten niemals berechnet werden. Effizienter ist es, die Vektoren u zu speichern. Diese können in gerade frei gewordene Spalten der Matrix A abgelegt werden. Auch für die Berechnung der Matrix-Vektor-Multiplikation kann auf die explizite Darstellung der Matrix verzichtet werden. Es gilt nämlich

$$Qx = x - \beta \frac{u^H x}{u^H u} u,$$

d.h. Qx entsteht als Linearkombination der Vektoren x und u, was mit $O(m)$ Operationen im Vergleich zu $O(m \cdot n)$ Operationen bei einträgeweiser Multiplikation durchgeführt werden kann.

1.2.2 Das Gram-Schmidt-Verfahren

Neben der Householder-Transformation kann eine QR-Zerlegung auch mit Hilfe des aus der Linearen Algebra bekannten **Gram-Schmidt-Verfahrens** berechnet werden. Es seien $a_1, \ldots, a_n \in \mathbb{K}^m$ die Spalten von A. Wegen $\operatorname{rank} A = n$ sind diese linear unabhängig. Sei $q_1 = a_1/\|a_1\|_2$ und für $k = 2, \ldots, n$

$$q_k' := a_k - \sum_{\ell=1}^{k-1} (a_k, q_\ell)\, q_\ell, \quad q_k := \frac{q_k'}{\|q_k'\|_2}.$$

Aus der Linearen Algebra ist bekannt, dass $\{q_1, \ldots, q_n\}$ eine Orthonormalbasis des Raums $\operatorname{span}\{a_1, \ldots, a_n\}$ bildet. Setze $r_{\ell k} := (a_k, q_\ell)$, $\ell < k$, und $r_{kk} = \|q_k'\|_2$. Dann gilt

$$a_k = q_k' + \sum_{\ell=1}^{k-1} \underbrace{(a_k, q_\ell)}_{r_{\ell k}}\, q_\ell = \sum_{\ell=1}^{k} r_{\ell k} q_\ell,\ k = 1, \ldots, n \iff A = \hat{Q}\hat{R}$$

mit $\hat{Q} = [q_1, \ldots, q_n]$, $\hat{Q}^H\hat{Q} = I \in \mathbb{K}^{n\times n}$, und der regulären oberen $n \times n$ Dreiecksmatrix

$$\hat{R} := \begin{cases} r_{ij}, & i \le j, \\ 0, & \text{sonst.} \end{cases}$$

Offenbar ist $A = \hat{Q}\hat{R}$ eine reduzierte QR-Zerlegung von A.

Bemerkung. Durch Rundungsfehler kann es beim Gram-Schmidt-Verfahren in der Praxis zum Verlust der Orthogonalität der Vektoren q_k kommen. Interessanterweise ist aber das Produkt $\hat{Q}\hat{R}$ akurat. Eine numerisch stabilere Variante ist das **modifizierte Gram-Schmidt-Verfahren**. Dies erhält man, wenn man berücksichtigt, dass wegen der Orthogonalität der Vektoren q_k gilt

$$(a_k, q_\ell) = (a_k - \sum_{i=1}^{\ell-1} \alpha_{ki} q_i, q_\ell)$$

für alle Skalare $\alpha_{ki} \in \mathbb{K}$.

Algorithmus 1.21 (Modifiziertes Gram-Schmidt-Verfahren).
`for` $k = 1, \ldots, n$
  $q'_k = a_k$;
  `for` $\ell = 1, \ldots, k-1$
    $r_{\ell k} = (q'_k, q_\ell)$;
    $q'_k := q'_k - r_{\ell k} q_\ell$;
  $r_{kk} = \|q'_k\|_2$;
  $q_k = q'_k / r_{kk}$;

1.2.3 Der Businger-Golub-Algorithmus

Bisher haben wir immer vorausgesetzt, dass rank $A = n$ gilt. Wir wollen nun den rangdefizienten Fall, d.h. $r := \operatorname{rank} A < n$ betrachten. Will man die QR-Zerlegung mittels Householder-Transformation in diesem Fall berechnen, so tritt ein Nullvektor in der ersten Spalte der Restmatrix auf. Um dies zu verhindern, sucht man nach der Spalte j^* mit der größten 2-Norm und bringt sie durch Spaltenvertauschen an die aktuelle Stelle:

$$\|a_{j^*}^{(k)}\|_2 = \max_{k \le j \le n} \|a_j^{(k)}\|_2.$$

Statt $A = QR$ erhält man dadurch schließlich

$$AP = QR$$

mit einer Permutationsmatrix $P \in \mathbb{K}^{n\times n}$. Insbesondere sind die Diagonalelemente von R der Größe nach absteigend geordnet und ermöglichen eine Konditionsabschätzung über das Verhältnis von größtem und kleinstem. Dieses Verfahren bezeichnet man als **Businger-Golub-Algorithmus**. Verglichen mit (1.8) hat die Matrix R dann die Gestalt

$$R = \begin{bmatrix} \hat{R}_1 & \hat{R}_2 \\ 0 & 0 \end{bmatrix}$$

mit $\hat{R}_1 \in \mathbb{K}^{r\times r}$ regulär und $\hat{R}_2 \in \mathbb{K}^{r\times(n-r)}$. Entsprechend sei

$$P^{-1}x = \begin{bmatrix} x_1 \\ x_2 \end{bmatrix}, \quad x_1 \in \mathbb{K}^r,\ x_2 \in \mathbb{K}^{n-r},$$

und wie bisher

$$Q^H b = \begin{bmatrix} c \\ d \end{bmatrix}, \quad c \in \mathbb{K}^r,\ d \in \mathbb{K}^{m-r},$$

zerlegt. Wegen

$$\|b - Ax\|_2^2 = \|b - QRP^{-1}x\|_2^2 = \|c - \hat{R}_1 x_1 - \hat{R}_2 x_2\|_2^2 + \|d\|_2^2$$

wird $\|b - Ax\|_2$ genau dann minimal, wenn

$$\hat{R}_1 x_1 + \hat{R}_2 x_2 = c. \tag{1.10}$$

Bemerkung. Für alle $x_2 \in \mathbb{K}^{n-r}$ ist also $P[x_1, x_2]^T$ mit $x_1 = \hat{R}_1^{-1}(c - \hat{R}_2 x_2)$ Lösung des linearen Ausgleichsproblems (1.2). Die Lösungsmenge ist daher $(n-r)$-dimensional. Im Fall $r = n$ erhalten wir wieder $\hat{R}_1 x_1 = c$ wie in (1.9), was eindeutig gelöst werden kann.

Die eindeutig bestimmte Lösung minimaler euklidischer Norm kann im rangdefizienten Fall mit Hilfe des folgenden Lemmas bestimmt werden.

Lemma 1.22. *Sei $r := \operatorname{rank} A < n$, $V := \hat{R}_1^{-1}\hat{R}_2 \in \mathbb{K}^{r\times(n-r)}$ und $u := \hat{R}_1^{-1}c \in \mathbb{K}^r$. Dann ist die Lösung minimaler euklidischer Norm von (1.2) gegeben durch $x = P[x_1, x_2]^T$ mit $x_1 = u - Vx_2 \in \mathbb{K}^r$ und $x_2 \in \mathbb{K}^{n-r}$ Lösung von*

$$(I + V^H V)x_2 = V^H u.$$

Beweis. Nach (1.10) sind alle Lösungen durch $x_1 = u - Vx_2$ beschrieben. Eingesetzt in $\|x\|_2$ erhalten wir

$$\begin{aligned} \|x\|_2^2 &= \|x_1\|_2^2 + \|x_2\|_2^2 = \|u - Vx_2\|_2^2 + \|x_2\|_2^2 \\ &= \|u\|_2^2 - 2\operatorname{Re}(u, Vx_2) + (Vx_2, Vx_2) + (x_2, x_2) \\ &= \|u\|_2^2 + \operatorname{Re}((I + V^H V)x_2 - 2V^H u, x_2) =: \varphi(x_2). \end{aligned}$$

Dann ist $\varphi'(x_2) = 2\operatorname{Re}[(I + V^H V)x_2 - V^H u]$ und $\varphi''(x_2) = 2(I + V^H V)$. Da $I + V^H V$ positiv-definit ist, nimmt φ sein Minimum für x_2 mit $\varphi'(x_2) = 0$ an, d.h.

$$(I + V^H V)x_2 = V^H u.$$

□

1.3 Singulärwertzerlegung und Pseudo-Inverse

Eine weitere Möglichkeit, das Minimierungsproblem (1.2) zu lösen, erhält man mit der *Singulärwertzerlegung*. Die Singulärwertzerlegung ist aber auch im Allgemeinen ein nützliches Mittel, um Matrizen zu analysieren.

In diesem und den folgenden Abschnitten verwenden wir die der euklidischen Vektornorm zugeordnete **Spektralnorm**

$$\|A\|_2 := \sup_{0\neq x\in\mathbb{K}^n} \frac{\|Ax\|_2}{\|x\|_2} = \rho^{1/2}(A^H A)$$

von $A \in \mathbb{K}^{m\times n}$. Dabei bezeichnet $\rho(M)$ den **Spektralradius**, d.h. den betragsmäßig größten Eigenwert, von $M \in \mathbb{K}^{n\times n}$. Ferner verwenden wir die **Frobenius-Norm**

$$\|A\|_F := \left(\sum_{i=1}^{m}\sum_{j=1}^{n} |a_{ij}|^2\right)^{1/2} = \left(\operatorname{trace} A^H A\right)^{1/2}.$$

Hierbei ist die **Spur** von $M \in \mathbb{K}^{n\times n}$ durch $\operatorname{trace} M := \sum_{i=1}^{n} m_{ii}$ definiert. Es gilt

$$\operatorname{trace} M = \sum_{i=1}^{n} \lambda_i(M)$$

mit den Eigenwerten $\lambda_i \in \mathbb{C}$, $i = 1,\dots,n$, von M. Spektral- und Frobenius-Norm sind **unitär invariant**, d.h. für alle unitäre Matrizen $U \in \mathbb{K}^{m\times m}$ und $V \in \mathbb{K}^{n\times n}$ gilt

$$\|UAV\|_2 = \|A\|_2 \quad \text{und} \quad \|UAV\|_F = \|A\|_F.$$

Für spätere Zwecke ist die Abschätzung

$$\rho(A) \le \|A\|, \quad A \in \mathbb{K}^{n\times n}, \tag{1.11}$$

für beliebige Operatornormen $\|\cdot\|$ hilfreich. Um diese zu zeigen, sei (λ, x) ein Eigenpaar. Dann gilt

$$|\lambda|\|x\| = \|\lambda x\| = \|Ax\| \le \|A\|\|x\|.$$

Obwohl die Frobenius-Norm keiner Vektornorm zugeordnet ist, ist sie mit der euklidischen Norm verträglich. Insbesondere wird ersichtlich, dass auch $\rho(A) \le \|A\|_F$ gilt.

Satz 1.23. *Zu $A \in \mathbb{K}^{m\times n}$ existieren unitäre Matrizen $U \in \mathbb{K}^{m\times m}$ und $V \in \mathbb{K}^{n\times n}$, so dass*

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

mit $\Sigma = \operatorname{diag}(\sigma_1,\dots,\sigma_p) \in \mathbb{R}^{m\times n}$, wobei $p = \min\{m,n\}$ und $\sigma_1 \ge \sigma_2 \ge \dots \ge \sigma_p \ge 0$.

Beweis. Es genügt zu zeigen, dass $U \in \mathbb{K}^{m\times m}$ und $V \in \mathbb{K}^{n\times n}$ existieren mit

$$U^H A V = \begin{bmatrix} \sigma & 0 \\ 0 & B \end{bmatrix}; \tag{1.12}$$

per Induktion folgt dann die Behauptung.

Sei $\sigma := \|A\|_2 = \max_{\|x\|_2=1} \|Ax\|_2$. Dabei dürfen wir annehmen, dass $A \neq 0$ und somit $\sigma \neq 0$. Weil das Maximum angenommen wird, gibt es $v \in \mathbb{K}^n$, $\|v\|_2 = 1$, mit $\|A\|_2 = \|Av\|_2$. Definiere

$$u = \frac{Av}{\|Av\|_2}.$$

Dann gilt $Av = \sigma u$. Seien $\hat{U} \in \mathbb{K}^{m\times(m-1)}$ und $\hat{V} \in \mathbb{K}^{n\times(n-1)}$ so gewählt, dass u und v zu unitären Matrizen

$$U = [u, \hat{U}] \in \mathbb{K}^{m\times m} \quad \text{bzw.} \quad V = [v, \hat{V}] \in \mathbb{K}^{n\times n}$$

erweitert werden. Für die Existenz dieser Matrizen verwende z.B. die Householder-Transformation. Dann gilt

$$U^H A V = \begin{bmatrix} u^H \\ \hat{U}^H \end{bmatrix} A [v, \hat{V}] = \begin{bmatrix} u^H A v & u^H A \hat{V} \\ \hat{U}^H A v & \hat{U}^H A \hat{V} \end{bmatrix} = \begin{bmatrix} \sigma & w^H \\ 0 & B \end{bmatrix}$$

mit $w := \hat{V}^H A^H u \in \mathbb{K}^{n-1}$. Da

$$\left\| U^H A V \begin{bmatrix} \sigma \\ w \end{bmatrix} \right\|_2 \geq \sigma^2 + \|w\|_2^2 \quad \text{und} \quad \left\| \begin{bmatrix} \sigma \\ w \end{bmatrix} \right\|_2 = \sqrt{\sigma^2 + \|w\|_2^2},$$

folgt aus der unitären Invarianz der Spektralnorm

$$\sigma^2 + \|w\|_2^2 \leq \frac{\left\| U^H A V \begin{bmatrix} \sigma \\ w \end{bmatrix} \right\|_2^2}{\left\| \begin{bmatrix} \sigma \\ w \end{bmatrix} \right\|_2^2} \leq \|U^H A V\|_2^2 = \|A\|_2^2 = \sigma^2$$

und daher $w = 0$. Hieraus folgt (1.12). □

Definition 1.24. *Eine Zerlegung der Form $A = U\Sigma V^H$ mit $U \in \mathbb{K}^{m\times m}$, $V \in \mathbb{K}^{n\times n}$ unitär und*

$$\Sigma = \operatorname{diag}(\sigma_1, \ldots, \sigma_p) \in \mathbb{R}^{m\times n}, \quad p = \min\{m, n\},$$

mit $\sigma_1 \geq \sigma_2 \geq \ldots \geq \sigma_p \geq 0$ heißt ***Singulärwertzerlegung*** *(engl. singular value decomposition, SVD) von $A \in \mathbb{K}^{m\times n}$. Die Werte σ_i, $i = 1, \ldots, p$, heißen* ***Singulärwerte*** *von A, die Spalten u_i von U bezeichnet man als linke und die Spalten v_i von V als rechte* ***Singulärvektoren****.*

Bemerkung. Die Singulärwerte einer Matrix sind eindeutig bestimmt. Falls σ_i, $i = 1, \ldots, r$, $r := \operatorname{rank} A$, paarweise verschieden sind, so sind die Singulärvektoren u_i, v_i, $i = 1, \ldots, r$, bis auf einen skalaren Faktor, der betragsmäßig 1 ist, eindeutig bestimmt.

Es ergibt sich eine Reihe von Folgerungen.

Satz 1.25. *Sei $A \in \mathbb{K}^{m\times n}$ und $A = U\Sigma V^H$ eine Singulärwertzerlegung.*

(i) Sind u_i, v_i die Spalten von U bzw. V, so gilt

$$Av_i = \sigma_i u_i \quad \text{und} \quad A^H u_i = \sigma_i v_i, \quad i = 1, \ldots, p.$$

(ii) *Es gilt* $AA^H = U\Sigma\Sigma^T U^H$ *und* $A^H A = V\Sigma^T\Sigma V^H$. *Daher sind* u_i, $i = 1, \dots, m$, *die Eigenvektoren von* AA^H *und* v_i, $i = 1, \dots, n$, *die Eigenvektoren von* $A^H A$. *Ferner sind* σ_i^2, $i = 1, \dots, p$, *genau die Eigenwerte (entsprechend ihrer Vielfachheiten) von*

$$\begin{cases} A^H A, & \text{falls } m \geq n, \\ AA^H, & \text{falls } m \leq n. \end{cases}$$

Insbesondere gilt $\|A\|_2 = \sigma_1$ *und* $\|A\|_F^2 = \sum_{i=1}^p \sigma_i^2$.

(iii) *Der Rang* $r = \operatorname{rank} A$ *stimmt mit der Anzahl positiver Singulärwerte überein, und es gilt*

$$\operatorname{Ker} A = \operatorname{span}\{v_{r+1}, \dots, v_n\}, \quad \operatorname{Ran} A = \operatorname{span}\{u_1, \dots, u_r\}$$

und

$$\operatorname{Ker} A^H = \operatorname{span}\{u_{r+1}, \dots, u_m\}, \quad \operatorname{Ran} A^H = \operatorname{span}\{v_1, \dots, v_r\}.$$

Beweis.

(i) folgt direkt aus $A = U\Sigma V^H \iff AV = U\Sigma \iff A^H U = V\Sigma^T$.

(ii) Sei $m \geq n = p$. Wegen $A^H A = V\Sigma^T U^H U\Sigma V^H = V\Sigma^T\Sigma V^H$ stimmen die charakteristischen Polynome von $A^H A$ und $\Sigma^T\Sigma = \operatorname{diag}(\sigma_1^2, \dots, \sigma_p^2) \in \mathbb{R}^{n\times n}$ überein. Ferner gilt

$$\|A\|_2^2 = \rho(A^H A) = \sigma_1^2$$

und

$$\|A\|_F^2 = \operatorname{trace} A^H A = \sum_{i=1}^p \lambda_i(A^H A) = \sum_{i=1}^p \sigma_i^2.$$

Im Fall $p = m \leq n$ besitzen AA^H und $\Sigma\Sigma^T = \operatorname{diag}(\sigma_1^2, \dots, \sigma_p^2) \in \mathbb{R}^{m\times m}$ dasselbe charakteristische Polynom.

(iii) Es gilt $r = \operatorname{rank} A = \operatorname{rank}\Sigma$. Aus

$$A = U\Sigma V^H = \sum_{i=1}^p \sigma_i u_i v_i^H = \sum_{i=1}^r \sigma_i u_i v_i^H$$

erhält man $\operatorname{span}\{v_{r+1}, \dots, v_n\} \subset \operatorname{Ker} A$. Wegen $\dim \operatorname{Ker} A = n - r$ folgt

$$\operatorname{span}\{v_{r+1}, \dots, v_n\} = \operatorname{Ker} A.$$

Außerdem gilt für $x \in \mathbb{K}^n$

$$Ax = \sum_{i=1}^r \sigma_i u_i (v_i^H x)$$

und somit $\operatorname{Ran} A \subset \operatorname{span}\{u_1, \dots, u_r\}$. Der Vergleich der Dimensionen liefert wieder die Behauptung

$$\operatorname{Ran} A = \operatorname{span}\{u_1, \dots, u_r\}.$$

□

Bemerkung.

(a) Frobenius- und Spektralnorm sind Spezialfälle der sog. **Schatten-Normen**

$$\|A\|_p := \left(\sum_{i=1}^{r} \sigma_i^p \right)^{1/p}, \quad p = 1, 2, \ldots, \infty.$$

Durch die Summation der ersten k Singulärwerte erhält man ebenfalls eine Norm, die **Ky Fan-Norm**

$$\|A\|'_k := \sum_{i=1}^{k} \sigma_i, \quad k = 1, \ldots, r.$$

(b) Nach (ii) können die Singulärwerte im Fall $m \geq n$ aus den Eigenwerten von $A^H A$ bestimmt werden. Dies hat aber wie bereits bei den Normalengleichungen eine Quadrierung der Kondition zu Folge. Eine bessere Möglichkeit besteht darin, die Eigenwerte der Matrix

$$A' = \begin{bmatrix} 0 & A \\ A^H & 0 \end{bmatrix} \in \mathbb{K}^{(m+n)\times(m+n)}$$

zu berechnen. Es gilt nämlich

$$A' \begin{bmatrix} u_i \\ \pm v_i \end{bmatrix} = \sigma_i \begin{bmatrix} \pm u_i \\ v_i \end{bmatrix}.$$

Insbesondere setzen sich die Eigenvektoren von A' aus den Singulärvektoren u_i, v_i zusammen.

(c) Nach (iii) gilt

$$A = \sum_{i=1}^{r} \sigma_i u_i v_i^H, \quad r = \operatorname{rank} A. \tag{1.13}$$

Definiert man $\tilde{u}_i := \sigma_i u_i$, so kann also jede Rang-r-Matrix $A \in \mathbb{K}^{m\times n}$ durch $r(m+n)$ Speichereinheiten (statt $m \cdot n$ bei einträgeweiser Speicherung) repräsentiert werden. Außerdem ist die Matrix-Vektor-Multiplikation Ax mit $O(r(m+n))$ Operationen durchführbar, indem man zunächst $\beta_i := v_i^H x$, $i = 1, \ldots, r$, und dann

$$Ax = \sum_{i=1}^{r} \beta_i \tilde{u}_i$$

berechnet. Ist $r(m+n) < m \cdot n$, so kann mit der Darstellung (1.13) eine Steigerung der Effizienz erzielt werden.

Sei nun $A \in \mathbb{K}^{m\times n}$ eine allgemeine Matrix (mit möglicherweise vollem Rang). Um die Vorteile von Niedrigrangmatrizen nutzen zu können, müssen wir herausfinden, wie gut sich A durch eine Matrix von niedrigem Rang bei vorgegebenem Fehler $\varepsilon > 0$ approximieren lässt. Dies führt auf den Begriff des **numerischen Ranges**

$$\operatorname{rank}_\varepsilon A := \min_{B \in \mathbb{K}^{m\times n} : \|A-B\| < \varepsilon} \operatorname{rank} B.$$

Für den Beweis des folgenden Satzes benötigen wir, dass $x \in \operatorname{span}\{v_1, \ldots, v_n\}$ die Basisdarstellung

$$x = \sum_{i=1}^{n} (v_i^H x) v_i$$

mit den Fourierkoeffizienten $v_i^H x$, $i = 1, \dots, n$, besitzt, falls v_i, $i = 1, \dots, n$, orthonormal sind. Insbesondere folgt hieraus, dass

$$\|x\|_2^2 = \sum_{i=1}^{n} |v_i^H x|^2.$$

Satz 1.26. *Sei $A \in \mathbb{K}^{m\times n}$ mit einer Singulärwertzerlegung $A = U\Sigma V^H$ und $\|\cdot\|$ eine unitär invariante Matrixnorm. Die beste Approximation in der Menge der Matrizen vom Rang höchstens $k \le p := \min\{m, n\}$ an A ist*

$$A_k := U\Sigma_k V^H,$$

wobei $\Sigma_k = \operatorname{diag}(\sigma_1, \dots, \sigma_k, 0, \dots, 0)$. Genauer ist

$$\|A - A_k\| = \min_{\operatorname{rank} B \le k} \|A - B\| = \|\Sigma - \Sigma_k\|.$$

Beweis. (nur für Spektralnorm, allgemeines Resultat von L. Mirsky)
Wegen $U^H A_k V = \operatorname{diag}(\sigma_1, \dots, \sigma_k, 0, \dots, 0)$ folgt $\operatorname{rank} A_k \le k$. Weiter gilt

$$U^H(A - A_k)V = \Sigma - \Sigma_k = \operatorname{diag}(0, \dots, 0, \sigma_{k+1}, \dots, \sigma_p)$$

und wegen der unitären Invarianz $\|A - A_k\|_2 = \|\Sigma - \Sigma_k\|_2 = \sigma_{k+1}$.

Es bleibt zu zeigen, dass für jede Matrix B mit $k' := \operatorname{rank} B \le k$ gilt $\|A - B\| \ge \sigma_{k+1}$. Dazu wählen wir eine Basis $\{x_1, \dots, x_{n-k'}\}$ von $\operatorname{Ker} B$. Aus Dimensionsgründen folgt

$$\operatorname{span}\{x_1, \dots, x_{n-k'}\} \cap \operatorname{span}\{v_1, \dots, v_{k+1}\} \neq \{0\}.$$

Sei $z \in \mathbb{K}^n$, $\|z\|_2 = 1$, aus dieser Schnittmenge. Dann gilt $Bz = 0$ und $Az = \sum_{i=1}^{k+1} \sigma_i (v_i^H z) u_i$ und somit

$$\|A - B\|_2^2 \ge \|(A - B)z\|_2^2 = \|Az\|_2^2 = \sum_{i=1}^{k+1} \sigma_i^2 |v_i^H z|^2 \ge \sigma_{k+1}^2 \sum_{i=1}^{k+1} |v_i^H z|^2 = \sigma_{k+1}^2.$$

Hierbei wurde verwendet, dass wegen der Orthonormalität der v_i

$$z = \sum_{i=1}^{k+1} (v_i^H z) v_i$$

und deshalb $1 = \|z\|_2^2 = \sum_{i=1}^{k+1} |v_i^H z|^2$. □

Bemerkung. Im Fall der Frobenius-Norm gilt $\|\Sigma - \Sigma_k\|_F = \sqrt{\sum_{i=k+1}^{p} \sigma_i^2}$, während für die Spektralnorm $\|\Sigma - \Sigma_k\|_2 = \sigma_{k+1}$ ist. Wegen der Fehlerabschätzung $\|A - A_k\| = \|\Sigma - \Sigma_k\|$ kann aus der Bedingung

$$\|\Sigma - \Sigma_k\| < \varepsilon$$

bei vorgegebenem $\varepsilon > 0$ der benötigte minimale Rang k bestimmt werden.

Abbildung 1.1: A_k für ein Bild mit 533×400 Pixeln bei $k = 10, 20, 50, 400$.

Beispiel 1.27. Die Singulärwertzerlegung kann z.B. zur Bildkompression verwendet werden. Sei ein Bild mit $m \times n$ Pixeln gegeben. Man erhält eine Matrix $A \in \mathbb{R}^{m \times n}$, indem die Helligkeit jedes Bildpunktes für einen einzelnen Farbkanal abgelegt wird. Den Rängen $k = 10, 20, 50$ in Abbildung 1.1 entsprechen Kompressionsraten von 4%, 9% und 22%.

Auf Basis der Singulärwertzerlegung führen wir nun die Pseudo-Inverse als Verallgemeinerung der Inversen für nicht quadratische Matrizen ein.

Definition 1.28. *Sei $A = U\Sigma V^H$ eine Singulärwertzerlegung von $A \in \mathbb{K}^{m \times n}$. Dann bezeichnet man die Matrix*

$$A^+ := V\Sigma^+ U^H \in \mathbb{K}^{n \times m}$$

*mit $\Sigma^+ = \operatorname{diag}(1/\sigma_1, \ldots, 1/\sigma_r, 0, \ldots, 0) \in \mathbb{R}^{n \times m}$, $r = \operatorname{rank} A$, als **Pseudo-Inverse** oder **Moore-Penrose-Inverse** von A.*

Bemerkung.

(a) Gilt $\operatorname{rank} A = n$ für $A \in \mathbb{K}^{m \times n}$, so hat man $A^+ = (A^H A)^{-1} A^H$. Dies folgt aus

$$\begin{aligned}(A^H A)^{-1} A^H &= (V\Sigma^H U^H U \Sigma V^H)^{-1} V \Sigma^H U^H \\ &= (V \Sigma^H \Sigma V^H)^{-1} V \Sigma^H U^H \\ &= V(\Sigma^H \Sigma)^{-1} \Sigma^H U^H = V \Sigma^+ U^H = A^+.\end{aligned}$$

Ist zusätzlich A quadratisch, so gilt $(A^H A)^{-1} = A^{-1} A^{-H}$ und somit

$$A^+ = (A^H A)^{-1} A^H = A^{-1} A^{-H} A^H = A^{-1}.$$

(b) Es gilt

$$A^+ = \sum_{i=1}^{r} \frac{1}{\sigma_i} v_i u_i^H$$

und somit $\operatorname{Ker} A^+ = \operatorname{Ker} A^H$ und $\operatorname{Ran} A^+ = \operatorname{Ran} A^H$.

(c) Wegen $(\Sigma^+)^+ = \Sigma$ und $(\Sigma^+)^H = (\Sigma^H)^+$ gilt $(A^+)^+ = A$ und $(A^+)^H = (A^H)^+$. Es gilt aber nicht $(AB)^+ = B^+ A^+$; vgl. Übungen.

Weil die Pseudo-Inverse mit Hilfe der nicht eindeutigen Singulärwertzerlegung definiert wurde, stellt sich die Frage, ob A^+ wohldefiniert ist. Aus dem folgenden Satz ergibt sich außerdem die Bezeichnung "Pseudo-Inverse".

Satz 1.29. *Die Pseudo-Inverse A^+ von $A \in \mathbb{K}^{m\times n}$ erfüllt die sog.* ***Penrose-Bedingungen***

(i) $AA^+A = A$,

(ii) $A^+AA^+ = A^+$,

(iii) $(AA^+)^H = AA^+$,

(iv) $(A^+A)^H = A^+A$.

Diese vier Bedingungen beschreiben A^+ auf eindeutige Weise. Insbesondere ist A^+ wohldefiniert.

Beweis.

(i) Wegen $A^+ = V\Sigma^+U^H$ und $\Sigma\Sigma^+ = \begin{bmatrix} I_r & 0 \\ 0 & 0 \end{bmatrix} \in \mathbb{R}^{m\times m}$ gilt

$$AA^+A = U\Sigma V^H V\Sigma^+U^H U\Sigma V^H = U\Sigma\Sigma^+\Sigma V^H = U\Sigma V^H = A$$

und analog

(ii) $A^+AA^+ = V\Sigma^+U^H U\Sigma V^H V\Sigma^+U^H = V\Sigma^+\Sigma\Sigma^+U^H = V\Sigma^+U^H = A^+$.

(iii) Aus $AA^+ = U\Sigma V^H V\Sigma^+U^H = U\Sigma\Sigma^+U^H$ und $A^+A = V\Sigma^+\Sigma V^H$ erkennt man, dass diese beiden Matrizen hermitesch sind.

Erfüllt auch $B \in \mathbb{K}^{n\times m}$ die Penrose-Bedingungen, so gilt

$$\begin{aligned} A^+ &\overset{(ii)}{=} A^+AA^+ \overset{(i)}{=} A^+ABAA^+ \overset{(iv)}{=} A^H(A^+)^HBAA^+ \overset{(iv)}{=} A^H(A^+)^HA^HB^HA^+ \overset{(i)}{=} A^HB^HA^+ \\ &\overset{(iv)}{=} BAA^+ \overset{(ii)}{=} BABAA^+ \overset{(iii)}{=} BAB(A^+)^HA^H \overset{(iii)}{=} BB^HA^H(A^+)^HA^H \overset{(i)}{=} BB^HA^H \\ &\overset{(iii)}{=} BAB \overset{(ii)}{=} B. \end{aligned}$$

□

Der folgende Satz stellt den Zusammenhang zwischen Pseudo-Inverser und linearen Ausgleichsproblemen her.

Satz 1.30. *Der Vektor $x_{\mathrm{LA}} := A^+b = \sum_{i=1}^r \frac{1}{\sigma_i} v_i(u_i^Hb)$ ist die eindeutig bestimmte Lösung minimaler euklidischer Norm des linearen Ausgleichsproblems (1.2). Für den Residualfehler gilt*

$$\|b - Ax_{\mathrm{LA}}\|_2^2 = \sum_{i=r+1}^{p} |u_i^Hb|^2, \quad p := \min\{m,n\}.$$

Beweis. Wegen Satz 1.29 (ii) ist

$$b - AA^+b \in \operatorname{Ker} A^+ = \operatorname{Ker} A^H = (\operatorname{Ran} A)^\perp,$$

weil $A^+(b - AA^+b) = A^+b - A^+AA^+b = 0$. Also erfüllt x_{LA} die Normalengleichung (1.7)

$$A^H A x_{\mathrm{LA}} = A^H b$$

und ist somit eine Lösung des linearen Ausgleichsproblems (1.2). Ist z eine weitere Lösung der Normalengleichungen, dann gilt nach Satz 1.14

$$w := A^+b - z \in \operatorname{Ker} A.$$

Da $A^+b \in \operatorname{Ran} A^+ = (\operatorname{Ker} A)^\perp$ haben wir $z = A^+b - w$ orthogonal zerlegt, und es folgt

$$\|z\|_2^2 = \|A^+b\|_2^2 + \|w\|_2^2 \geq \|A^+b\|_2^2.$$

Also ist $x_{\mathrm{LA}} = A^+b$ die eindeutig bestimmte Lösung minimaler euklidischer Norm. Ferner gilt

$$\|b - Ax_{\mathrm{LA}}\|_2^2 = \|b - U\Sigma V^H V\Sigma^+ U^H b\|_2^2 = \|U^H b - \Sigma\Sigma^+ U^H b\|_2^2 = \sum_{i=r+1}^{p} |u_i^H b|^2.$$

□

Bemerkung.

(a) Der Weg über Singulärwertzerlegung und Pseudo-Inverse zur Lösung des linearen Ausgleichproblems ist der universellste aber auch der teuerste mit $O(21\,m \cdot n^2)$ Operationen.

(b) Man beachte, dass AA^+ ein orthogonaler Projektor (siehe Definition 1.10) auf $\operatorname{Ran} A$ und A^+A ein orthogonaler Projektor auf $\operatorname{Ran} A^+$ ist.

Mit Hilfe der Pseudo-Inversen läßt sich der Konditionsbegriff auf allgemeine Matrizen erweitern.

Definition 1.31. *Sei $A \in \mathbb{K}^{m\times n}$ und $A^+ \in \mathbb{K}^{n\times m}$ die Pseudo-Inverse. Wir definieren die* ***Konditionszahl*** *als*

$$\operatorname{cond}_{\|\cdot\|} A = \|A\|\|A^+\|.$$

Wegen $A^+ = A^{-1}$ bei regulären Matrizen $A \in \mathbb{K}^{n\times n}$ (siehe die Bemerkung nach Definition 1.28) ist der neue Konditionsbegriff aus Definition 1.31 eine Verallgemeinerung der bisherigen Definition $\operatorname{cond}_{\|\cdot\|} A = \|A\|\|A^{-1}\|$.

Bemerkung. Für $\|\cdot\| = \|\cdot\|_2$ ergibt sich $\|A\|_2 = \sigma_1$, $\|A^+\|_2 = 1/\sigma_r$ und somit

$$\operatorname{cond}_{\|\cdot\|_2} A = \frac{\sigma_1}{\sigma_r}.$$

2 Iterative Lösungsverfahren

In diesem Kapitel wenden wir uns der iterativen Lösung von linearen Gleichungssystemen

$$Ax = b, \quad A \in \mathbb{K}^{n\times n}, \tag{2.1}$$

zu. Diese treten z.B. in Folge der Methode der finiten Differenzen und des Galerkin-Verfahrens zur Lösung von Randwertproblemen partieller Differentialgleichungen auf; mehr dazu in der Vorlesung *Wissenschaftliches Rechnen*. In diesem Fällen ist A großdimensioniert aber schwachbesetzt.

Definition 2.1. *Eine Folge von Matrizen $A_i \in \mathbb{K}^{m_i\times n_i}$, $i \in \mathbb{N}$, wird als* ***schwachbesetzt*** *bezeichnet, falls $\nu \in \mathbb{N}$ existiert, so dass die Anzahl der nicht-verschwindenden Einträge in jeder Zeile oder in jeder Spalte von A_i durch ν beschränkt ist.*

Beispiel 2.2. Wir betrachten das Randwertproblem für u

$$-\Delta u = f \quad \text{in } \Omega := (-1,1)^2, \tag{2.2a}$$
$$u = 0 \quad \text{auf } \partial\Omega. \tag{2.2b}$$

bei gegebenem f. Dabei bezeichnet $\Delta := \partial_x^2 + \partial_y^2$ den Laplace-Operator und $\partial\Omega = \{(x,y) \in \mathbb{R}^2 : \max(|x|,|y|) = 1\}$ den Rand von Ω.

Eine einfache Möglichkeit, das Problem (2.2) numerisch behandelbar zu machen, ist es, die Ableitungen in (2.2) durch Ausdrücke, die nur die Werte der Funktion enthalten, zu ersetzen. Seien $v \in C^1[-1,1]$ und $h > 0$ gegeben. Wir definieren

$$\partial^+ v(x) := \frac{v(x+h) - v(x)}{h} \quad \text{Vorwärtsdifferenz},$$
$$\partial^- v(x) := \frac{v(x) - v(x-h)}{h} \quad \text{Rückwärtsdifferenz}.$$

Die Ausdrücke $\partial^+ v$ und $\partial^- v$ heißen **Differenzenquotienten**. Unter Verwendung der Taylor-Entwicklung um $x \in (-1+h, 1-h)$ erhält man

$$|\partial^+ v(x) - v'(x)| \le ch|v|_{C^2[-1,1]}, \tag{2.3a}$$
$$|\partial^- v(x) - v'(x)| \le ch|v|_{C^2[-1,1]}, \tag{2.3b}$$
$$|\partial^+\partial^- v(x) - v''(x)| \le ch^2|v|_{C^4[-1,1]} \tag{2.3c}$$

mit einer von h und v unabhängigen Konstanten $c > 0$ und $|v|_{C^k[-1,1]} := \sup_{x\in[-1,1]} |v^{(k)}(x)|$. Dabei gehen wir davon aus, dass v den jeweiligen Glattheitsanforderungen genügt.

Um (2.2) zu diskretisieren, verwenden wir das Gitter $(x_i, y_j) \in \overline{\Omega}$, $i,j = 0,\dots,N+1$, mit $x_i := ih - 1$, $y_j := jh - 1$ der Gitterweite $h := 2/(N+1)$. In den inneren Gitterpunkten ersetzen wir den negativen Laplace-Operator $-\Delta = -\partial_x^2 - \partial_y^2$ in (2.2a) durch

$$-\partial_x^+\partial_x^- u - \partial_y^+\partial_y^- u = h^{-2}[4u(x,y) - u(x-h,y) - u(x+h,y) - u(x,y-h) - u(x,y+h)].$$

Denkt man sich die Gitterfunktion $u_{i,j}$ als Approximation an den Wert $u(x_i, y_j)$ der exakten Lösung, so erzeugt diese Diskretisierung die Finite-Differenzen-Gleichung

$$4u_{i,j} - u_{i-1,j} - u_{i+1,j} - u_{i,j-1} - u_{i,j+1} = h^2 f(x_i, y_j), \quad i,j = 1,\dots,N. \tag{2.4}$$

Unter Verwendung der Randbedingung $u_{0,j} = u_{N+1,j} = u_{i,0} = u_{i,N+1} = 0$ und nach Sequenzierung des Indexpaares (i,j) durch $k := i + jN$ erhält man hieraus das lineare Gleichungssystem

$$Au = f, \quad u_k := u_{i,j},\ f_k := h^2 f(x_i, y_j, \quad k = 1,\dots,N^2. \tag{2.5}$$

Die Koeffizientenmatrix $A \in \mathbb{R}^{N^2 \times N^2}$ in (2.5) hat dann die Gestalt

$$A = \begin{bmatrix} T & -I & & \\ -I & T & \ddots & \\ & \ddots & \ddots & -I \\ & & -I & T \end{bmatrix} \quad \text{mit} \quad T = \begin{bmatrix} 4 & -1 & & \\ -1 & 4 & \ddots & \\ & \ddots & \ddots & -1 \\ & & -1 & 4 \end{bmatrix} \in \mathbb{R}^{N \times N}.$$

Diese ist positiv-definit und besitzt obere/untere Bandbreite N. Ferner besitzt A für jedes n höchstens 3 nicht-verschwindende Einträge pro Zeile und ist somit schwachbesetzt.

Ist $A \in \mathbb{K}^{m \times n}$ schwachbesetzt, so ist es unnötig, die Nullen zu speichern. Anstelle des gewöhnlichen spalten- oder zeilenweisen Speicherformats verwendet man die folgenden komprimierenden Formate. Beim **CRS-Format** (engl. compressed row storage) werden die nicht-verschwindenden Einträge (seien mit ihren Spaltenindizes zeilenweise gespeichert. Dabei werden in einem Feld A die Werte, in einem Feld j_A die entsprechenden Spaltenindizes und in i_A die Anfangsindizes der Zeilen in A bzw. j_A beginnend mit 1 abgelegt. Als letzten Index in i_A wird zusätzlich die Länge von A erhöht um Eins gespeichert. Bezeichnet $\mu \in \mathbb{N}$ die Anzahl der nicht-verschwindenden Einträge von $A \in \mathbb{K}^{m \times n}$, so benötigt man statt $m \cdot n$ Speichereinheiten bei gewöhnlicher Speicherung nur $2\mu + m + 1$ Speichereinheiten im CRS-Format.

Beispiel 2.3. Die Matrix

$$A = \begin{bmatrix} 0 & 3 & 4 & 0 & 0 & 7 & 0 \\ 0 & 0 & 0 & 0 & 4 & 0 & 0 \\ 0 & 9 & 2 & -5 & 0 & 0 & 0 \\ 0 & 0 & 3 & 0 & 8 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & -1 & 0 \end{bmatrix}$$

hat im CRS-Format die Gestalt

A	=	3	4	7	4	9	2	-5	3	8	-1
j_A	=	2	3	6	5	2	3	4	3	5	6
i_A	=	1	4	5	8	10	11				

Dieses Format kann nicht nur genutzt werden, um eine schwachbesetzte Matrix effizient zu speichern, man kann auch die Multiplikation von $A \in \mathbb{K}^{m \times n}$ mit einem Vektor $x \in \mathbb{K}^n$ schnell durchführen

```
for i = 1,...,m do
    y_i = 0;
    for k = i_A[i],...,i_A[i+1] - 1 do
        y_i = y_i + A[k] x_{j_A[k]};
```

Offenbar benötigt diese Multiplikation nur etwa doppelt so viele Operationen wie nicht-verschwindende Einträge vorhanden sind.

Natürlich kann A auch spaltenweise komprimiert werden, d.h. man speichert A^T im CRS-Format. Dieses Format wird als **CCS-Format** (engl. compressed column storage) oder **Harwell-Boeing-Format** bezeichnet.

Direkte Lösungsmethoden wie die LR-Zerlegung erhalten zwar die Bandbreite, d.h. die Faktoren L und R besitzen dieselbe Bandbreite wie A, besitzen aber signifikant mehr nicht-verschwindende Einträge als A. Da mit diesem sog. *fill-in*-Effekt ein erhöhter Aufwand verbunden ist, betrachten wir in diesem Kapitel iterative Methoden, bei denen die Matrix nur durch Matrix-Vektor-Multiplikation in die Berechnung eingeht.

2.1 Krylov-Räume

Im Folgenden suchen wir Approximationen der Lösung des linearen Gleichungssystems (2.1) in Räumen, zu deren Konstruktion A nur durch Multiplikationen mit Vektoren eingeht.

Definition 2.4. *Seien $A \in \mathbb{K}^{n\times n}$ und $v \in \mathbb{K}^n$ gegeben. Der Unterrraum*

$$\mathcal{K}_k(A, v) := \operatorname{span}\{v, Av, \ldots, A^{k-1}v\}$$

*des $\mathbb{K}^n$ wird als **Krylov-Raum** der Ordnung k bezeichnet.*

Es ist offensichtlich, dass die Folge der Krylov-Räume geschachtelt ist, d.h.

$$\mathcal{K}_k(A, v) \subset \mathcal{K}_{k+1}(A, v) \quad \text{für alle } v \in \mathbb{K}^n.$$

Ferner wächst die Dimension von $\mathcal{K}_k(A, v)$ offensichtlich um höchstens Eins bei Erhöhung von k. Wir definieren

$$\operatorname{grad}_A(v) := \min\{j : \dim \mathcal{K}_{j+1}(A, v) < j + 1\}$$

als den kleinsten Index, bei dem sich die Dimension nicht erhöht. Wegen $\mathcal{K}_k(A, v) \subset \mathbb{K}^n$ gilt offenbar $\operatorname{grad}_A(v) \leq n$.

Satz 2.5. *Sei $A \in \mathbb{K}^{n\times n}$, $v \in \mathbb{K}^n$ und $g := \operatorname{grad}_A(v)$. Dann gilt*

(i) $A\mathcal{K}_g(A, v) \subset \mathcal{K}_g(A, v)$, d.h. $\mathcal{K}_g(A, v)$ ist A-invariant;

(ii) $\mathcal{K}_k(A, v) = \mathcal{K}_g(A, v)$ für alle $k \geq g$;

(iii) $\dim \mathcal{K}_k(A, v) = \min\{k, g\}$.

Beweis. Sei zunächst $k > g$. Nach Definition von $\operatorname{grad}_A(v)$ existieren Koeffizienten $\beta_\ell \in \mathbb{K}$, $\ell = 0, \ldots, g$, so dass

$$\sum_{\ell=0}^{g} \beta_\ell A^\ell v = 0$$

und nicht alle Koeffizienten verschwinden. Insbesondere ist $\beta_g \neq 0$, weil sonst $\operatorname{grad}_A(v) < g$ wäre. Sei $x = \sum_{\ell=0}^{k-1} \alpha_\ell A^\ell v \in \mathcal{K}_k$. Dann ist

$$x = \sum_{\ell=0}^{k-1} \alpha_\ell A^\ell v - \frac{\alpha_{k-1}}{\beta_g} A^{k-1-g} \sum_{\ell=0}^{g} \beta_\ell A^\ell v \ \in \ \mathcal{K}_{k-1}.$$

In dieser Weise kann fortgefahren werden, bis man $x \in \mathcal{K}_g$ und somit $\mathcal{K}_k = \mathcal{K}_g$ erhält. Insbesondere erhält man $\dim \mathcal{K}_k = \dim \mathcal{K}_g$. Ist $k < g$, so folgt aus der Definition von $\operatorname{grad}_A(v)$, dass $\dim \mathcal{K}_{k+1} = k + 1$. Außerdem gilt offenbar $A\mathcal{K}_g \subset \mathcal{K}_{g+1} = \mathcal{K}_g$. □

Viele auf Krylov-Räumen basierende Methoden verwenden Orthonormalbasen. Wir betrachten zunächst den nicht-hermiteschen Fall $A \in \mathbb{K}^{n\times n}$. Das folgende **Arnoldi-Verfahren** ist identisch mit dem Gram-Schmidt-Verfahren (wir verwenden das modifizierte Verfahren; siehe Algorithmus 1.21) zur Konstruktion einer Orthonormalbasis $\{w_1, \ldots, w_k\}$ des Krylov-Raums $\mathcal{K}_k(A, v)$.

Algorithmus 2.6 (Arnoldi-Verfahren).
`Input:` $A \in \mathbb{K}^{n\times n}$, $v \in \mathbb{K}^n \setminus \{0\}$
`Output:` Orthonormalbasis $\{w_1, \ldots, w_k\}$ von $\mathcal{K}_k(A, v)$ und $\tilde{H}_k \in \mathbb{K}^{(k+1)\times k}$

$w_1 := v/\|v\|_2$;
`for` $j = 1, \ldots, k$
  $q := Aw_j$;
  `for` $i = 1, \ldots, j$
    $h_{ij} := w_i^H q$;
    $q := q - h_{ij} w_i$;
  $h_{j+1,j} = \|q\|_2$;
  `if` $h_{j+1,j} = 0$ `then` STOP;
  $w_{j+1} := q/h_{j+1,j}$;

Bemerkung. Der Algorithmus multipliziert in jedem Schritt den letzten Arnoldi-Vektor w_j mit A und orthogonalisiert den resultierenden Vektor q gegen alle bereits berechneten Vektoren w_i. Die Matrix A geht dabei nur durch die Matrix-Vektor-Multiplikation Aw_j ein. Sei die Anzahl der Operationen für eine solche Multiplikation mit c_{MV} bezeichnet. Dann werden $2k^2n + kc_{\mathrm{MV}}$ Operationen für Algorithmus 2.6 benötigt. Ferner sind $(k+1)n$ Speichereinheiten für die Vektoren w_j, $j = 1, \ldots, k+1$, und $O(k^2)$ Speichereinheiten für die Matrix

$$\tilde{H}_k := \begin{bmatrix} h_{11} & h_{12} & \cdots & h_{1k} \\ h_{21} & h_{22} & \ddots & \vdots \\ 0 & \ddots & \ddots & h_{k-1,k} \\ \vdots & \ddots & \ddots & h_{kk} \\ 0 & \ldots & 0 & h_{k+1,k} \end{bmatrix} \in \mathbb{K}^{(k+1)\times k}$$

nötig.

Im folgenden Satz werden Eigenschaften der durch das Arnoldi-Verfahren generierten Matrizen $W_k := [w_1, \ldots, w_k] \in \mathbb{K}^{n\times k}$ und $\tilde{H}_k$ bewiesen. Neben $\tilde{H}_k$ betrachten wir auch die Hessenberg-Matrix $H_k \in \mathbb{K}^{k\times k}$, die aus $\tilde{H}_k$ durch Streichen der letzten Zeile entsteht.

Definition 2.7. *Eine Matrix $A \in \mathbb{K}^{n\times n}$ heißt* ***Hessenberg-Matrix****, falls $a_{ij} = 0$ für alle $i, j = 1, \ldots, n$ mit $i > j + 1$.*

Hessenberg-Matrizen sind also bis auf eine untere Nebenbendiagonale obere Dreiecksmatrizen.

Satz 2.8. *Das Arnoldi-Verfahren bricht genau im Schritt $j = \operatorname{grad}_A(v)$ ab. Für alle $k \leq \operatorname{grad}_A(v)$ gilt*

(i) Die Spalten von W_k bilden eine Orthonormalbasis von $\mathcal{K}_k(A, v)$;

(ii) Es gilt $AW_k = W_{k+1}\tilde{H}_k$;

(iii) Es gilt $W_k^H AW_k = H_k$.

Beweis. Wir zeigen zunächst, dass $w_j = p_j(A)v$ mit einem Polynom $p_j \in \Pi_{j-1}$, dessen führender Koeffizient nicht verschwindet, und dass $\{w_1, \ldots, w_j\}$ orthonormal ist. Für $j = 1$ ist die Behauptung wegen $w_1 = v/\|v\|_2$ wahr. Für den Induktionsschritt beachte man, dass

$$w_{j+1} = \frac{1}{\|q\|_2}\left(Aw_j - \sum_{i=1}^{j} h_{ij}w_i\right) = \frac{1}{\|q\|_2}\left(Ap_j(A)v - \sum_{i=1}^{j} h_{ij}p_i(A)v\right) = p_{j+1}(A)v,$$

wobei

$$p_{j+1}(t) := \frac{1}{\|q\|_2}\left(tp_j(t) - \sum_{i=1}^{j} h_{ij}p_i(t)\right) \in \Pi_j.$$

Also ist $\operatorname{span}\{w_1, \ldots, w_{j+1}\} \subset \mathcal{K}_{j+1}(A, v)$. Im Fall $j < \operatorname{grad}_A(v)$ ist $Aw_j \in \mathcal{K}_{j+1} \setminus \mathcal{K}_j$ und daher $h_{j+1,j} \neq 0$. Die Orthogonalität des Vektorsystems $\{w_1, \ldots, w_{j+1}\}$ schließt man wie beim Gram-Schmidt-Verfahren. Im Fall $j = \operatorname{grad}_A(v)$ ist $Aw_j \in \mathcal{K}_{j+1} = \mathcal{K}_j$ mit der Folge, dass $h_{j+1,j} = 0$ gilt. Insgesamt erhält man für $k \leq \operatorname{grad}_A(v)$, dass $\operatorname{span}\{w_1, \ldots, w_k\} = \mathcal{K}_k$, weil das System $\{w_1, \ldots, w_k\}$ linear unabhängig ist. Wegen

$$Aw_j = \sum_{i=1}^{j+1} h_{ij}w_i = W_{k+1}\tilde{H}_k e_j, \quad j = 1, \ldots, k,$$

gilt $AW_k = W_{k+1}\tilde{H}_k$. Hieraus folgt

$$W_k^H AW_k = W_k^H W_{k+1}\tilde{H}_k = W_k^H [W_k, w_{k+1}] \begin{bmatrix} H_k \\ h_{k+1,k}e_k^T \end{bmatrix} = [I_k, 0] \begin{bmatrix} H_k \\ h_{k+1,k}e_k^T \end{bmatrix} = H_k$$

und somit die Behauptung. □

Bemerkung. Die Matrix $P_k := W_k W_k^H \in \mathbb{K}^{n\times n}$ ist ein orthogonaler Projektor auf

$$\operatorname{Ran} P_k = \operatorname{Ran} W_k = \mathcal{K}_k(A, x_0).$$

Daher ist H_k die Darstellung der Projektion

$$P_k A P_k = W_k W_k^H AW_k W_k^H = W_k H_k W_k^H$$

von A auf $\mathcal{K}_k(A, x_0)$ bzgl. der Basis $\{w_1, \ldots, w_k\}$. Je größer der Unterraum $\operatorname{Ran} P_k$ ist, desto besser approximiert $W_k H_k W_k^H$ die Matrix A.

Bei hermiteschen Matrizen $A \in \mathbb{K}^{n\times n}$ wird das Arnoldi-Verfahren als **Lanczos-Verfahren** bezeichnet. Wegen $W_k^H A W_k = H_k$ ist die Hessenberg-Matrix H_k dann eine hermitesche Tridiagonalmatrix. Wegen $h_{j+1,j} = \|q\|_2 \in \mathbb{R}$ ist sie sogar reell mit positiven Nebendiagonaleinträgen und wird statt mit H_k als

$$T_k := \begin{bmatrix} \alpha_1 & \beta_2 & & & 0 \\ \beta_2 & \ddots & \ddots & & \\ & \ddots & \ddots & \ddots & \\ & & \ddots & \ddots & \beta_k \\ 0 & & & \beta_k & \alpha_k \end{bmatrix} \in \mathbb{R}^{k\times k}$$

bezeichnet. Mit

$$\tilde{T}_k := \begin{bmatrix} T_k \\ \beta_{k+1} e_k^T \end{bmatrix} \in \mathbb{R}^{(k+1)\times k}$$

gilt dann $AW_k = W_{k+1}\tilde{T}_k$. Weil beim Lanczos-Verfahren nur α_j und β_{j+1}, $j = 1, \ldots, k$, berechnet werden müssen, sollte dies auch im Algorithmus Berücksichtigung finden. Wegen $\alpha_j = h_{jj}$ und $\beta_{j+1} = h_{j+1,j}$ folgt $\alpha_j = w_j^H q$, wobei $q := Aw_j - \beta_j w_{j-1}$ und $\beta_{j+1} = \|q - \alpha_j w_j\|_2$. Daher ergibt sich aus Algorithmus 2.6 unter Berücksichtigung der Struktur von T_k.

Algorithmus 2.9 (Lanczos-Verfahren).
`Input:` $A \in \mathbb{K}^{n\times n}$ hermitesch und $v \in \mathbb{K}^n \setminus \{0\}$
`Output:` Orthonormalbasis $\{w_1, \ldots, w_k\}$ von $\mathcal{K}_k(A, v)$ und $\tilde{T}_k \in \mathbb{R}^{(k+1)\times k}$

$w_1 := v/\|v\|_2$; $\beta_1 := 0$;
`for` $j = 1, \ldots, k$
  $q := Aw_j - \beta_j w_{j-1}$;
  $\alpha_j := w_j^H q$;
  $q := q - \alpha_j w_j$;
  $\beta_{j+1} := \|q\|_2$;
  `if` $\beta_{j+1} = 0$ `then` STOP;
  $w_{j+1} := q/\beta_{j+1}$;

2.2 Das Arnoldi-Verfahren bei der Lösung linearer Gleichungssysteme

In diesem Abschnitt wird das Arnoldi-Verfahren verwendet, um lineare Gleichungssysteme (2.1) zu lösen. Sei dazu $x_0 \in \mathbb{K}^n$ ein Startvektor und $W_k = [w_1, \ldots, w_k]$ eine Orthonormalbasis von $\mathcal{K}_k(A, r_0)$ mit $r_0 := b - Ax_0$. H_k bezeichne die zugehörige Hessenberg-Matrix.

Durch die Bedingung

$$r_k := b - Ax_k \perp \mathcal{K}_k(A, r_0), \quad k > 0, \tag{2.6}$$

wird eine Folge $\{x_k\}$ und damit das sog. **FOM-Verfahren** (engl. full orthogonalization method) definiert. Für positiv-definite Matrizen A gilt die folgende Interpretation von (2.6). In diesem Fall wird nämlich durch $(x, y)_A := (Ax, y)$ das **Energieskalarprodukt** und durch

$$\|x\|_A := \sqrt{(x, x)_A}, \quad x \in \mathbb{K}^n,$$

die **Energienorm** definiert.

Lemma 2.10. *Sei A positiv-definit. x_k ist genau dann eine Lösung von (2.6), wenn*

$$\|x - x_k\|_A = \min_{y \in x_0 + \mathcal{K}_k(A, r_0)} \|x - y\|_A. \tag{2.7}$$

Beweis. Nach dem Projektionssatz 1.9 ist $x - x_k$ genau dann bzgl. $\|\cdot\|_A$ minimal, wenn $x - x_k$ bzgl. $(\cdot,\cdot)_A$ orthogonal zu $\mathcal{K}_k(A, r_0)$ ist, d.h.

$$0 = (A(x - x_k), y) = (b - Ax_k, y)$$

für alle $y \in \mathcal{K}_k(A, r_0)$. □

Bemerkung. Wegen (2.7) und $\mathcal{K}_k(A, r_0) \subset \mathcal{K}_{k+1}(A, r_0)$ ist die Konvergenz bzgl. der $\|\cdot\|_A$-Norm monoton, d.h. es gilt $\|x - x_{k+1}\|_A \leq \|x - x_k\|_A$, falls A positiv-definit ist.

Das folgende Lemma päsentiert die explizite Lösung von (2.6) bzw. im positiv-definiten Fall der Minimierungsaufgabe (2.7).

Lemma 2.11. *Sei* $\dim \mathcal{K}_k(A, r_0) = k > 0$ *und H_k aus (2.1) regulär. Dann gilt für den Vektor*

$$x_k := x_0 + \|r_0\|_2 W_k H_k^{-1} e_1,$$

dass

$$x_k \in x_0 + \mathcal{K}_k(A, r_0) \quad \text{und} \quad r_k = b - Ax_k \perp \mathcal{K}_k(A, r_0).$$

Beweis. Weil die Spalten von W_k eine Basis von $\mathcal{K}_k(A, r_0)$ bilden, erhält man sofort, dass $x_k \in x_0 + \mathcal{K}_k(A, r_0)$. Wegen $(\mathcal{K}_k(A, r_0))^\perp = (\operatorname{Ran} W_k)^\perp = \operatorname{Ker} W_k^H$ gilt $b - Ax_k \perp \mathcal{K}_k(A, r_0)$ genau dann, wenn $W_k^H(b - Ax_k) = 0$. Aus

$$W_k^H(b - Ax_k) = W_k^H r_0 - \|r_0\|_2 W_k^H A W_k H_k^{-1} e_1 = W_k^H(r_0 - \|r_0\|_2 W_k e_1) = 0$$

folgt die Behauptung wegen $w_1 = r_0/\|r_0\|_2$. □

Die Matrix $H_k = W_k^H A W_k$ ist beispielsweise regulär, falls A positiv-definit ist. Die Hauptarbeit bei der Bestimmung von x_k liegt in der Lösung des (kleinen) linearen Gleichungssystems

$$H_k y_k = \|r_0\|_2 e_1$$

mit der Hessenberg-Matrix $H_k \in \mathbb{K}^{k \times k}$. Dies kann z.B. mit Hilfe der gaußschen Elimination effizient gelöst werden.

Bemerkung. Der Aufwand verhält sich dann quadratisch in k und es müssen k Arnoldi-Vektoren gespeichert werden, so dass es sinnvoll sein kann, das Verfahren mit der gewonnenen Approximation x_k als neuen Startwert x_0 neu zu beginnen.

Wie wir im folgenden Lemma sehen, kann das Residuum in jedem Schritt auf einfache Weise bestimmt werden.

Lemma 2.12. *Es gilt* $r_k = b - Ax_k = -h_{k+1,k} y_k^T e_k \, w_{k+1}$. *Also ist* $\|r_k\|_2 = h_{k+1,k} |y_k^T e_k|$.

Beweis. Wegen $AW_k = W_{k+1}\tilde{H}_k$ gilt

$$b - Ax_k = b - A(x_0 + W_k y_k) = r_0 - AW_k y_k = \|r_0\|_2 w_1 - W_k H_k y_k - h_{k+1,k} e_k^T y_k \, w_{k+1}.$$

Aus $H_k y_k = \|r_0\|_2 e_1$ folgt die Behauptung. □

Das Arnoldi-Verfahren bricht nur ab, falls $h_{k+1,k} = 0$ ist. Wir sehen also insbesondere, dass das FOM-Verfahren genau dann abbricht, wenn die exakte Lösung erreicht ist. Insbesondere ist das FOM-Verfahren ein endliches Verfahren, weil $\operatorname{grad}_A(r_0) \leq n$.

2.3 Das GMRES-Verfahren

Fordert man anstelle von (2.7), dass $x_k \in x_0 + \mathcal{K}_k(A, r_0)$ die Residual-Norm

$$\|b - Ax_k\|_2 = \min_{y \in x_0 + \mathcal{K}_k(A, r_0)} \|b - Ay\|_2 \tag{2.8}$$

minimiert, so erhält man das **GMRES-Verfahren** (engl. generalized minimal residuals). Aus (2.8) folgt sofort, dass

$$\|r_{k+1}\|_2 \leq \|r_k\|_2.$$

Die Konvergenz ist also monoton.

Bemerkung. Nach dem Projektionssatz 1.9 löst x_k genau dann (2.8), wenn $r_k \perp A\mathcal{K}_k(A, r_0)$.

Praktisches Vorgehen

Wir beschreiben im Folgenden das praktische Vorgehen zur Lösung von (2.8). Sei W_k wie in (2.1) mit der ersten Spalte $r_0/\|r_0\|_2$. Dann ist (2.8) äquivalent zur Minimierung des Ausdrucks

$$\|b - A(x_0 + W_k y)\|_2 = \|r_0 - AW_k y\|_2, \quad y \in \mathbb{K}^k,$$

und es gilt

$$r_0 - AW_k y = W_{k+1}(\|r_0\|_2 e_1 - \tilde{H}_k y).$$

Weil die Spalten von W_{k+1} orthonormal sind, folgt hieraus

$$\|r_0 - AW_k y\|_2 = \left\|\tilde{H}_k y - \|r_0\|_2 e_1\right\|_2.$$

Wegen $AW_k = W_{k+1}\tilde{H}_k$ ist $\operatorname{rank} \tilde{H}_k = k$ und das Minimierungsproblem

$$\text{finde } y_k \in \mathbb{K}^k \text{ mit } \left\|\tilde{H}_k y_k - \|r_0\|_2 e_1\right\|_2 = \min_{y \in \mathbb{K}^k} \left\|\tilde{H}_k y - \|r_0\|_2 e_1\right\|_2$$

ist eindeutig lösbar. Hierbei handelt es sich um ein lineares Ausgleichsproblem; siehe Kapitel 1. Um dies stabil zu lösen, ist die QR-Zerlegung von $\tilde{H}_k \in \mathbb{K}^{(k+1)\times k}$ zu bestimmen. Wegen der Hessenberg-Struktur von $\tilde{H}_k$ ist nur die untere Nebendiagonale zu eliminieren. Die Verwendung einer Folge von Householder-Spiegelungen wäre zu kostspielig. Mit Hilfe von *Givens-Rotationen* $G^{(\ell,\ell+1)}$, $\ell = 1, \ldots, k$, kann dies aber effizient durchgeführt werden.

Definition 2.13. *Jede Matrix $G^{(i,j)} \in \mathbb{K}^{n\times n}$, die sich von der Einheitsmatrix nur in den Positionen (i,i), (i,j), (j,i) und (j,j) durch die Einträge*

$$\begin{bmatrix} c & s \\ -\bar{s} & c \end{bmatrix} \quad \text{mit} \quad c \in \mathbb{R},\ s \in \mathbb{K},\ c^2 + |s|^2 = 1,$$

unterscheidet, wird als ***Givens-Rotation*** *bezeichnet.*

Bemerkung.

(a) Givens-Rotationen sind unitäre Matrizen, weil

$$\begin{bmatrix} c & s \\ -\bar{s} & c \end{bmatrix} \begin{bmatrix} c & -s \\ \bar{s} & c \end{bmatrix} = \begin{bmatrix} c^2 + |s|^2 & 0 \\ 0 & c^2 + |s|^2 \end{bmatrix} = I.$$

(b) Ist $x \in \mathbb{K}^n$, $y := G^{(i,j)}x$, so ist für $k = 1, \ldots, n$

$$y_k = \begin{cases} cx_i + sx_j, & k = i, \\ cx_j - \bar{s}x_i, & k = j, \\ x_k, & \text{sonst.} \end{cases}$$

(c) Sind $c, s \in \mathbb{R}$, dann existiert $\theta \in [0, 2\pi)$ mit $c = \cos\theta$, $s = \sin\theta$. Nach (b) dreht daher $G^{(i,j)}$ den Vektor x in der (x_i, x_j)-Ebene um den Winkel θ im Uhrzeigersinn.

Givens-Rotationen können also verwendet werden, um einen Vektor so zu transformieren, dass eine einzelne Komponente verschwindet. Zu gegebenen $\alpha, \beta \in \mathbb{K}$ sind $c \in \mathbb{R}$ und $s \in \mathbb{K}$ gesucht, so dass

$$\begin{bmatrix} c & s \\ -\bar{s} & c \end{bmatrix} \begin{bmatrix} \alpha \\ \beta \end{bmatrix} = \begin{bmatrix} \gamma \\ 0 \end{bmatrix} \quad \text{und} \quad c^2 + |s|^2 = 1.$$

Die Lösung ergibt sich wie folgt: Ist $\alpha = 0$, so wähle $c = 0$, $s = 1$. Ansonsten setze

$$c = \frac{|\alpha|}{\sqrt{|\alpha|^2 + |\beta|^2}}, \quad s = \frac{\alpha\bar{\beta}}{|\alpha|\sqrt{|\alpha|^2 + |\beta|^2}}. \tag{2.9}$$

Bemerkung. Um einen Vektor $x \in \mathbb{K}^n$ auf ein Vielfaches von e_1 zu transformieren, benötigt man mit Givens-Rotationen etwa doppelt so viele Operationen wie mit einer Householder-Spiegelung. Demnach sind Givens-Rotationen nur geeignet für Vektoren mit vielen Nullen.

Angenommen, eine QR-Zerlegung von $\tilde{H}_k$ ist bereits berechnet, d.h.

$$G^{(k,k+1)} \cdot \ldots \cdot G^{(1,2)} \tilde{H}_k = \begin{bmatrix} R_k \\ 0 \end{bmatrix}$$

mit einer oberen Dreiecksmatrix $R_k \in \mathbb{K}^{k\times k}$, die wegen $\operatorname{rank} \tilde{H}_k = k$ invertierbar ist. Die QR-Zerlegung von $\tilde{H}_{k+1}$ kann unter Verwendung dieser Zerlegung effizient bestimmt werden. Definiert man nämlich

$$\hat{G}^{(\ell,\ell+1)} = \begin{bmatrix} G^{(\ell,\ell+1)} & 0 \\ 0 & 1 \end{bmatrix}, \quad \ell = 1, \ldots, k,$$

so gilt

$$\hat{G}^{(k,k+1)} \cdot \ldots \cdot \hat{G}^{(1,2)} \tilde{H}_{k+1} = \hat{G}^{(k,k+1)} \cdot \ldots \cdot \hat{G}^{(1,2)} \begin{bmatrix} \tilde{H}_k & h_{k+1} \\ 0 & h_{k+2,k+1} \end{bmatrix} = \begin{bmatrix} G^{(k,k+1)} \cdot \ldots \cdot G^{(1,2)} \tilde{H}_k & \tilde{h}_{k+1} \\ 0 & h_{k+2,k+1} \end{bmatrix}$$

mit $\tilde{h}_{k+1} := G^{(k,k+1)} \cdot \ldots \cdot G^{(1,2)} h_{k+1}$. Definiert man $G^{(k+1,k+2)}$ als die Givens-Rotation, die den Eintrag $h_{k+2,k+1}$ zu Null werden lässt, so kann die QR-Zerlegung von $\tilde{H}_{k+1}$ aus der Zerlegung von $\tilde{H}_k$ mit nur einer Givens-Rotation und $O(k)$ Operationen berechnet werden.

Wie von den linearen Ausgleichsproblemen bekannt, kann die Norm des Residuums

$$\|r_k\|_2 = \|\tilde{H}_k y_k - \|r_0\|_2 e_1\|_2$$

aus der rechten Seite nach Anwendung der Givens-Rotationen bestimmt werden. Entsprechend sei

$$\|r_0\|_2 G^{(k,k+1)} \cdot \ldots \cdot G^{(1,2)} e_1 =: \begin{bmatrix} c_k \\ d_k \end{bmatrix} \in \mathbb{K}^{k+1}, \quad d_k \in \mathbb{K}.$$

Dann ist $y_k = R_k^{-1} c_k$; vgl. (1.9). Also minimiert $x_k := x_0 + W_k R_k^{-1} c_k$ den Ausdruck (2.8), und wir haben $\|r_k\|_2 = |d_k|$. Insbesondere erhält man

Satz 2.14. *Sei $A \in \mathbb{K}^{n \times n}$ regulär. Das GMRES-Verfahren bricht mit der exakten Lösung des Gleichungssystems $Ax = b$ nach höchstens n Schritten ab.*

Beweis. Das Verfahren bricht genau dann ab, wenn $h_{k+1,k} = 0$ für ein k. Dann ist $G^{(k,k+1)}$ die Identität und somit $d_k = 0$, weil $G^{(k-1,k)} \cdot \ldots \cdot G^{(1,2)} e_1$ ab der $(k+1)$-ten Komponente verschwindet. □

Bemerkung. Für große k wird GMRES schnell teuer. Insbesondere müssen alle Arnoldi-Vektoren W_k gespeichert werden. In diesem Fall beginnt man das Verfahren nach k Schritten erneut mit der aktuelle Approximation x_k als neuen Startwert x_0. Man bezeichnet dies als **Restarted GMRES**.

Konvergenzanalyse

Bisher haber wir nur den Ablauf des Verfahrens behandelt. Wir wissen zwar, dass GMRES ein endliches Verfahren ist, kennen aber nicht dessen Konvergenzverhalten. Zunächst betrachten wir den Effekt des Restarts.

Satz 2.15. *Sei $A \in \mathbb{K}^{n \times n}$ regulär, und es gelte $x^H Ax \in \mathbb{R}$ für alle $x \in \mathbb{K}^n$. Dann gilt*

$$\|b - Ax_k\|_2 \leq \left(1 - \frac{\mu^2}{\|A\|_2^2}\right)^{1/2} \|b - Ax_0\|_2,$$

wobei $\mu := \lambda_{\min}(A^H + A)/2$.

Beweis. Für beliebiges $\alpha \in \mathbb{K}$ ist $x_0 + \alpha r_0 \in x_0 + \mathcal{K}_k(A, r_0)$. Beim GMRES-Verfahren gilt aber

$$\|r_k\|_2^2 \leq \|b - A(x_0 + \alpha r_0)\|_2^2 = \|r_0 - \alpha A r_0\|_2^2 = \|r_0\|_2^2 - 2\alpha r_0^H A r_0 + \alpha^2 \|A r_0\|_2^2.$$

Speziell für $\alpha = r_0^H A r_0 / \|A r_0\|_2^2$ folgt, dass

$$\|r_k\|_2^2 \leq \|r_0\|_2^2 - \frac{(r_0^H A r_0)^2}{\|A r_0\|_2^2} = \left[1 - \left(\frac{r_0^H A r_0}{\|r_0\|_2^2}\right)^2 \left(\frac{\|r_0\|_2}{\|A r_0\|_2}\right)^2\right] \|r_0\|_2^2 \leq \left[1 - \frac{\mu^2}{\|A\|_2^2}\right] \|r_0\|_2^2,$$

weil nach Voraussetzung $r_0^H A r_0 = r_0^H A^H r_0$ und somit $r_0^H A r_0 = r_0^H (A^H + A) r_0 / 2 \geq \mu \|r_0\|_2^2$. □

Der letzte Satz zeigt, dass GMRES mit Restart für eine positiv-definite Matrix eine Folge von Näherungslösungen liefert, die gegen die Lösung von $Ax = b$ konvergiert. Hierbei beachte man, dass die Abschätzung nicht berücksichtigt, wie groß k ist. Ist die Matrix nicht positiv-definit, so kann der wohl-bekannte **Stagnationseffekt** auftreten, weil in diesem Fall $\mu = 0$ nicht ausgeschlossen werden kann.

Aus Satz 2.14 wissen wir, dass (bei exakter Arithmetik) endlich viele Schritte genügen, um die exakte Lösung von $Ax = b$ zu bestimmen. Weil eine Approximation an x aber in der Regel ausreichend ist, möchte man sich mit wenigen Schritten des GMRES-Verfahrens begnügen. Im Rest des Abschnitts werden wir daher die Genauigkeit der Näherung x_k untersuchen. Für $y \in x_0 + \mathcal{K}_k(A, r_0)$ ist $y = x_0 + q(A) r_0$ mit $q \in \Pi_{k-1}$. Daher gilt

$$b - Ay = b - A(x_0 + p(A) r_0) = (I - A q(A)) r_0 = p(A) r_0$$

mit einem $p \in \tilde{\Pi}_k := \{p \in \Pi_k : p(0) = 1\}$. Auch das Residuum des GMRES-Verfahrens kann daher als Anwendung eines Matrixpolynoms $p_k(A)$ auf r_0 angesehen werden: $r_k = p_k(A) r_0$, $p_k \in \tilde{\Pi}_k$. Daher ist die Minimierung des Residuums (2.8) äquivalent zu

$$\|r_k\|_2 = \min_{p \in \tilde{\Pi}_k} \|p(A) r_0\|_2.$$

Satz 2.16. *Sei $A \in \mathbb{K}^{n \times n}$ diagonalisierbar, d.h. $A = X \Lambda X^{-1}$, $\Lambda = \operatorname{diag}(\lambda_1, \ldots, \lambda_n)$. Dann gilt*

$$\|b - A x_k\|_2 \leq \operatorname{cond}(X) \varepsilon_k \|b - A x_0\|_2,$$

wobei

$$\varepsilon_k := \min_{p \in \tilde{\Pi}_k} \max_{i=1,\ldots,n} |p(\lambda_i)|.$$

Insbesondere ist x_k die exakte Lösung, falls $\lambda_i = 0$, $i > k$.

Beweis. Mit obigen Bezeichnungen folgt für $y \in x_0 + \mathcal{K}_k(A, r_0)$

$$\begin{aligned} \|r_k\|_2 \leq \|b - Ay\|_2 = \|p(A) r_0\|_2 = \|X p(\Lambda) X^{-1} r_0\|_2 \leq \|X\|_2 \|X^{-1}\|_2 \|p(\Lambda)\|_2 \|r_0\|_2 \\ = \operatorname{cond}(X) \|p(\Lambda)\|_2 \|r_0\|_2 = \operatorname{cond}(X) \max_{i=1,\ldots,n} |p(\lambda_i)| \, \|r_0\|_2 \end{aligned}$$

für beliebiges $p \in \tilde{\Pi}_k$. □

2.3.1 Minimax-Eigenschaft von Tschebyscheff-Polynomen

Sei $[a, b]$ ein beliebiges Intervall und $x_0 \notin [a, b]$. In diesem Abschnitt betrachten wir das folgende Minimax-Problem

$$\|p_k\|_{\infty,[a,b]} = \min_{p\in\Pi_k,\, p(x_0)=1} \|p\|_{\infty,[a,b],}, \tag{2.10}$$

um damit das optimale Polynom $p_k \in \Pi_k$ mit $p_k(x_0) = 1$ für eine konkrete Abschätzung von ε_k in Satz 2.16 zu identifizieren. Hierbei bezeichne $\|f\|_{\infty,[a,b],} := \max_{x\in[a,b]} |f(x)|$ die Supremumsnorm von $f \in C[a, b]$.

Für $x \in \mathbb{R}$ definieren wir die **Tschebyscheff-Polynome** $T_n \in \Pi_n$

$$T_0(x) = 1, \quad T_1(x) = x, \quad T_n(x) := 2x\, T_{n-1}(x) - T_{n-2}(x),\ n > 1.$$

Satz 2.17 (Eigenschaften der Tschebyscheff-Polynome).

(i) *Für gerade n ist T_n eine gerade, für ungerade n eine ungerade Funktion;*

(ii) *Der Koeffizient des höchsten Monoms ist 2^{n-1}, $n \geq 1$;*

(iii) *$T_n(1) = 1$, $T_n(-1) = (-1)^n$ und $\|T_n\|_{\infty,[-1,1]} = 1$;*

(iv) *$|T_n(x)|$ nimmt den Wert 1 an den sog. Tschebyscheff-Abszissen*

$$t_k := \cos\left(\frac{k}{n}\pi\right), \quad k = 0, \ldots, n,$$

an, d.h. $|T_n(x)| = 1 \iff x = t_k$ für ein $k = 0, \ldots, n$;

(v) *Die Nullstellen von T_n sind*

$$x_k := \cos\left(\frac{2k-1}{2n}\pi\right), \quad k = 1, \ldots, n;$$

(vi) *Es gilt*

$$T_k(x) = \begin{cases} \cos(k \arccos(x)), & |x| \leq 1, \\ \cosh(k \operatorname{arccosh}(x)), & |x| \geq 1; \end{cases}$$

(vii) *Eine weitere Darstellung ist*

$$T_k(x) = \frac{1}{2}\left(\left(x + \sqrt{x^2-1}\right)^k + \left(x - \sqrt{x^2-1}\right)^k\right) \quad \text{für } x \in \mathbb{R}.$$

Beweis. (i)–(iv) überprüft man leicht. (v) und (vi) beweist man, indem man nachweist, dass die Formeln der Dreitermrekursion (inkl. Startwerten) genügen. □

Der folgende Satz zeigt eine Lösung von (2.10).

Satz 2.18. *Das Polynom*

$$\hat{T}_n(x) := \frac{T_n(t)}{T_n(t_0)} \quad \text{mit } t(x) := 1 - 2\frac{x-a}{b-a},\ t_0 := t(x_0),$$

ist minimal bzgl. $\|\cdot\|_{\infty,[a,b]}$ *unter den Polynomen* $p \in \Pi_n$ *mit* $p(x_0) = 1$.

Beweis. Weil alle Nullstellen von $T_n(t(x))$ in $[a,b]$ liegen, ist $c := |T_n(t_0)| \neq 0$ und $\hat{T}_n \in \Pi_n$ wohldefiniert. Ferner ist $\hat{T}_n(x_0) = 1$ und $\|\hat{T}_n\|_{\infty,[a,b]} = 1/c$.

Angenommen, es gibt ein Polynom $p \in \Pi_n$ mit $p(x_0) = 1$ und $\|p\|_{\infty,[a,b]} < 1/c$. Dann ist x_0 eine Nullstelle von $\hat{T}_n - p$, d.h.

$$\hat{T}_n(x) - p(x) = q(x)(x - x_0)$$

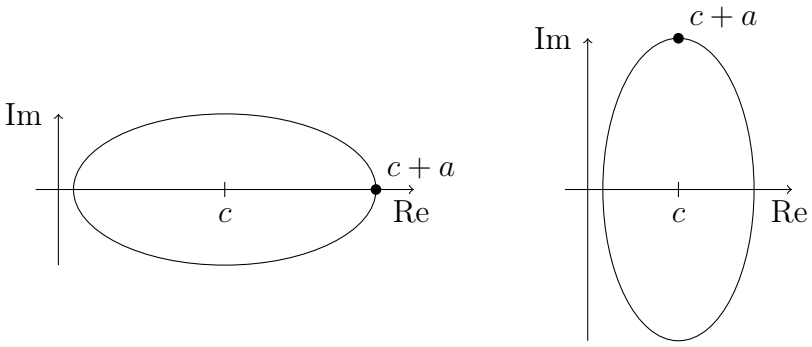
mit einem Polynom $0 \neq q \in \Pi_{n-1}$. An den Tschebyscheff-Abszissen $x_k := x(\cos(\frac{k}{n}\pi))$ mit $x(t) := \frac{1-t}{2}a + \frac{1+t}{2}b$ gilt

$$\begin{aligned} \hat{T}_n(x_{2k}) = \ \ 1/c, &\quad p(x_{2k}) < \ \ 1/c \quad \Rightarrow \quad \hat{T}_n(x_{2k}) - p(x_{2k}) > 0, \\ \hat{T}_n(x_{2k+1}) = -1/c, &\quad p(x_{2k+1}) > -1/c \quad \Rightarrow \quad \hat{T}_n(x_{2k+1}) - p(x_{2k+1}) < 0. \end{aligned}$$

Also hat q in den Punkten x_k wechselndes Vorzeichen für $k = 0, \ldots, n$ und daher mindestens n verschiedene Nullstellen in $[a,b]$. Dies steht im Widerspruch zu $0 \neq q \in \Pi_{n-1}$. □

Fehlerabschätzung für Tschebyscheff-Polynome

Die Größe von ε_k hängt von der Lage der Eigenwerte λ_i, $i = 1, \ldots, n$, ab. Liegen diese in einer entlang der Koordinatenachsen ausgerichteten Ellipse $E(c; d, a)$ mit Zentrum $c \in \mathbb{R}$, Brennpunktabstand d und der Länge der großen Halbachse a,



so kann das Verhalten von ε_k in Abhängigkeit dieser drei Parameter abgeschätzt werden. Dazu setzen wir das Tschebyscheff-Polynom T_k mit Hilfe der Darstellung (siehe Satz 2.17 (v))

$$T_k(x) = \cosh(k \operatorname{arccosh}(x)), \quad |x| \geq 1,$$

auf ganz $\mathbb{C}$ durch

$$T_k(z) = \cosh(k\zeta), \quad z = \cosh(\zeta) = \frac{1}{2}(e^{\zeta} + e^{-\zeta}),$$

fort. Mit der Variablen $w := e^{\zeta}$ erhält man daher

$$T_k(z) = \frac{1}{2}(w^k + w^{-k}), \quad z = \frac{1}{2}(w + w^{-1}).$$

Die Optimalität der Tschebyscheff-Polynome, die in Satz 2.18 für reelle Intervalle gezeigt wurde, gilt für $E(c;d,a)$ nicht. Dennoch kann man die Tschebyscheff-Polynome verwenden, um die folgende praxistaugliche obere Schranke zu zeigen.

Lemma 2.19. *Es gelte $\lambda_i \in E(c;d,a) \subset \mathbb{C} \setminus \{0\}$, $i = 1,\dots,n$. Dann gilt*

$$\varepsilon_k \leq \frac{|T_k(a/d)|}{|T_k(c/d)|} \sim \left(\frac{a+\sqrt{a^2-d^2}}{c+\sqrt{c^2-d^2}}\right)^k.$$

Beweis. Wir definieren

$$\hat{T}_k(z) = \frac{T_k((c-z)/d)}{T_k(c/d)}.$$

Durch Betrachtung des Ausdrucks $w^k + w^{-k}$ für $w = re^{\mathrm{i}\varphi}$ erkennt man, dass $|T_k((c-z)/d)|$ für $z = c+a$ in $E(c;d,a)$ maximal wird. Wegen $\hat{T}_k \in \tilde{\Pi}_k$ folgt daher

$$\varepsilon_k = \min_{p\in\tilde{\Pi}_k} \max_{i=1,\dots,n} |p(\lambda_i)| \leq \min_{p\in\tilde{\Pi}_k} \max_{z\in E(c;d,a)} |p(z)| \leq \max_{z\in E(c;d,a)} |\hat{T}_k(z)| = \frac{|T_k(a/d)|}{|T_k(c/d)|}.$$

Nach Satz 2.17 (vi) erhalten wir für große k

$$\frac{|T_k(a/d)|}{|T_k(c/d)|} \sim \left(\frac{a+\sqrt{a^2-d^2}}{c+\sqrt{c^2-d^2}}\right)^k.$$

□

Man beachte, dass der Quotient

$$\frac{a+\sqrt{a^2-d^2}}{c+\sqrt{c^2-d^2}}$$

in Lemma 2.19 kleiner als 1 ist, weil aus $0 \notin E(c;d,a)$ folgt $a < c$. Allerdings hängt die Größe des letzten Ausdrucks vom Verhältnis der Ausdehnung der Ellipse zu ihrem Abstand zur 0 ab. Die Konvergenz von GMRES wird also von der Lage der Eigenwerte und von der Kondition von X bestimmt. Ist A normal, so ist X unitär und somit $\mathrm{cond}(X) = 1$. Mit einer *Vorkonditionierung* kann man erreichen, dass die Eigenwerte der vorkonditionierten Matrix sich um einen Punkt verschieden vom Ursprung häufen. Dadurch kann eine kleinere Ellipse gewählt werden, die die Eigenwerte umschließt, was zu einer Beschleunigung der Konvergenz führt.

2.3.2 Vorkonditioniertes GMRES

Weil die die Eigenwerte von CA umschliessende Ellipse bei geschickter Wahl der regulären Matrix $C \in \mathbb{K}^{n\times n}$ kleiner sein kann als die Kondition von A, wenden wir das GMRES-Verfahren auf das Gleichungssystem

$$CAx = Cb \quad \Longleftrightarrow \quad Ax = b$$

an. Die Matrix C wird als **linker Vorkonditionierer** bezeichnet. Ebenso kann man **rechte Vorkonditionierer** betrachten:

$$AC\tilde{x} = b,\ x = C\tilde{x} \quad \Longleftrightarrow \quad Ax = b.$$

Die Anwendung eines Vorkonditionierers C auf einen Vektor sollte ähnlich schnell duchführbar sein wie die Multiplikation mit A. Die Konstruktion eines geeigneten Vorkonditionierers ist nicht-trivial und kann in der Regel nur problembezogen geschehen.

Bemerkung. Vom Standpunkt der Konvergenzgeschwindigkeit wäre $C = A^{-1}$ der optimale Vorkonditionierer, weil alle Eigenwerte von $CA = I$ bzw. $AC = I$ Eins sind und somit die Ellipsenparameter $a = d = 0$ gewählt werden können. Dies zeigt, dass die exakte Lösung bereits nach einem Schritt vorliegt. Allerdings ist die Berechnung von $C = A^{-1}$ deutlich teurer als die iterative Lösung und scheidet somit für die Praxis aus.

Das folgende Lemma beschreibt den prinzipiellen Unterschied zwischen rechts- und linksvorkonditioniertem GMRES.

Lemma 2.20. *Die durch das links-vorkonditionierte GMRES-Verfahren berechnete Näherungslösung x_k löst die folgende Aufgabe*

$$\text{minimiere} \quad \|C(b - Ax)\|_2, \quad x \in x_0 + \mathcal{K}_k(CA, Cr_0),$$

die durch das rechts-vorkonditionierte GMRES-Verfahren generierte Näherungslösung x_k die Aufgabe

$$\text{minimiere} \quad \|b - Ax\|_2, \quad x \in x_0 + C\mathcal{K}_k(AC, r_0).$$

Beweis. Die erste Aussage ist klar. Ist C ein rechter Vorkonditionierer, so ist $x_k = C\tilde{x}_k$, wobei $\tilde{x}_k$ die Aufgabe

$$\text{minimiere} \quad \|b - AC\tilde{x}\|_2, \quad \tilde{x} \in \tilde{x}_0 + \mathcal{K}_k(AC, \tilde{r}_0),$$

löst. Hierbei ist $\tilde{r}_0 = b - AC\tilde{x}_0$. Mit der Transformation $x_k = C\tilde{x}_k$ erhält man die Behauptung. □

Die rechte Vorkonditionierung ist demnach für das GMRES-Verfahren natürlicher, weil weiterhin das Residuum bzgl. der euklidischen Norm minimiert wird. Daher betrachten wir im Folgenden ausschließlich rechte Vorkonditionierer.

Der hermitesche Spezialfall des GMRES-Verfahrens wird als **MINRES-Verfahren** bezeichnet. Dieses ist prinzipiell auf indefinite Matrizen anwendbar. Auf die dabei entstehenden Probleme wollen wir hier nicht weiter eingehen.

2.4 Gradientenverfahren

Im Folgenden betrachten wir positiv-definite Matrizen. Bisher war diese Eigenschaft bzgl. des euklidischen Skalarproduktes zu verstehen. Dies verallgemeinern wir nun.

Definition 2.21. *Eine Matrix $A \in \mathbb{K}^{n\times n}$ heißt **positiv-definit** bzgl. eines Skalarproduktes $(\cdot,\cdot)$ auf $\mathbb{K}^n$, falls A **selbstadjungiert** ist, d.h. es gilt $(Ax, y) = (x, Ay)$ für alle $x, y \in \mathbb{K}^n$, und falls $(x, Ax) > 0$ für alle $0 \neq x \in \mathbb{K}^n$.*

Wie in Abschnitt 2.2 kann man sich leicht davon überzeugen, dass $(x,y)_A := (Ax,y)$ für bzgl. $(\cdot,\cdot)$ positiv-definite Matrizen A ein weiteres Skalarprodukt definiert. Die dadurch induzierte Norm $\|x\|_A := \sqrt{(Ax,x)}$ wird als **Energienorm** bezeichnet.

In diesem Abschnitt werden wir einen anderen Zugang zur Lösung großdimensionierter Gleichungssysteme

$$Ax = b \tag{2.11}$$

mit positiv-definiter Matrix $A \in \mathbb{K}^{n\times n}$ und gegebener rechter Seite $b \in \mathbb{K}^n$ kennenlernen. Dazu formulieren wir (2.11) als äquivalentes Minimierungsproblem der Funktion

$$f(y) := \frac{1}{2}(y,y)_A - \operatorname{Re}(y,b).$$

Lemma 2.22. *Die Lösung x von (2.11) ist das eindeutige Minimum von f, und für alle $y \in \mathbb{K}^n$ gilt*

$$f(y) - f(x) = \frac{1}{2}\|y-x\|_A^2.$$

Beweis. Wegen

$$f(y) = \frac{1}{2}(y,y)_A - \operatorname{Re}(y,x)_A = \frac{1}{2}(y-x,y-x)_A - \frac{1}{2}\|x\|_A^2 = \frac{1}{2}\|y-x\|_A^2 - \frac{1}{2}\|x\|_A^2$$

ist f minimal für $y = x$. □

Um das Minimum von f zu finden, verfolgen wir die Strategie, ausgehend von $x_k \in \mathbb{K}^n$ den nächsten Punkt $x_{k+1} \in \mathbb{K}^n$ durch Minimierung von f auf der Geraden durch x_k in einer gegebenen Richtung $p_k \in \mathbb{K}^n$ zu bestimmen (sog. **Liniensuche**), d.h.

$$x_{k+1} = x_k + \alpha_k p_k, \tag{2.12}$$

wobei die Schrittweite α_k so gewählt ist, dass

$$f(x_{k+1}) = f(x) + \frac{1}{2}\|x_{k+1} - x\|_A^2$$

minimal unter allen $x_k + \alpha p_k$, $\alpha \in \mathbb{K}$, ist. Wegen Lemma 1.9 wählen wir x_{k+1} als die bzgl. $(\cdot,\cdot)_A$ orthogonale Projektion von x auf die Gerade $\{x_k + \alpha p_k, \alpha \in \mathbb{K}\}$. Wir erhalten

$$x_{k+1} = x_k + (x - x_k, \frac{p_k}{\|p_k\|_A})_A \frac{p_k}{\|p_k\|_A}$$

und somit

$$\alpha_k = \frac{(x-x_k,p_k)_A}{(p_k,p_k)_A} = \frac{(Ax-Ax_k,p_k)}{(Ap_k,p_k)} = \frac{(r_k,p_k)}{(Ap_k,p_k)}$$

mit dem Residuum $r_k = b - Ax_k$. Obige Wahl von α_k stellt sicher, dass $\{f(x_k)\}_{k\in\mathbb{N}}$ eine monoton fallende Folge ist. Dies folgt schon aus der Konstruktion der x_k, mit (2.12) gilt aber genauer

$$\begin{aligned} f(x_{k+1}) - f(x_k) &= \frac{1}{2}(x_{k+1},x_{k+1})_A - \operatorname{Re}(x_{k+1},x)_A - \frac{1}{2}(x_k,x_k)_A + \operatorname{Re}(x_k,x)_A \\ &= \operatorname{Re}(\alpha_k p_k, x_k)_A + \frac{|\alpha_k|^2}{2}(p_k,p_k)_A - \operatorname{Re}(\alpha_k p_k, x)_A \\ &= \frac{|\alpha_k|^2}{2}(p_k,p_k)_A - \operatorname{Re}\alpha_k(p_k, x-x_k)_A \\ &= -\frac{1}{2}\frac{|(p_k,x-x_k)_A|^2}{\|p_k\|_A^2} \le 0. \end{aligned}$$

Im Folgenden betrachten wir drei Verfahren durch spezielle Wahl der Suchrichtung p_k.

2.4.1 Methode des steilsten Abstiegs

Für dieses Verfahren nehmen wir an, dass $\mathbb{K} = \mathbb{R}$ und $(x, y) := y^T x$ das euklidische Skalarprodukt ist. Sei die Suchrichtung

$$p_k = -\nabla f(x_k) = -Ax_k + b = r_k$$

in Richtung des steilsten Abstieges gewählt. Dann gilt

$$\alpha_k = \frac{\|r_k\|^2}{(Ar_k, r_k)}.$$

Algorithmus 2.23 (Methode des steilsten Abstiegs).
`Input:` $A \in \mathbb{R}^{n\times n}$ positiv-definit, $b, x_0 \in \mathbb{R}^n$ und Fehlertoleranz $\varepsilon > 0$
`Output:` Folge $\{x_k\}_{k\in\mathbb{N}} \subset \mathbb{R}^n$ von Approximationen an die Lösung von $Ax = b$

$r_0 = b - Ax_0$;
$k = 0$;
`do` {
  $\alpha_k = \frac{\|r_k\|^2}{(Ar_k, r_k)}$;
  $x_{k+1} = x_k + \alpha_k r_k$;
  $k = k + 1$;
  $r_k = b - Ax_k$;
} `while` $(\|r_k\| > \varepsilon\|b\|)$;

Satz 2.24. *Ist $A \in \mathbb{R}^{n\times n}$ positiv-definit, so konvergiert die Methode des steilsten Abstiegs für jeden Startwert $x_0 \in \mathbb{R}^n$, d.h. es gilt*

$$\|x_{k+1} - x\|_A \leq \frac{\operatorname{cond}(A) - 1}{\operatorname{cond}(A) + 1} \|x_k - x\|_A,$$

wobei $\operatorname{cond}(A) = \|A\|_2\|A^{-1}\|_2$ die Kondition von A bezeichnet.

Beweis. Sei $T_\alpha := I - \alpha A$, $\alpha \in \mathbb{R}$. Dann gilt wegen $T_\alpha y = y + \alpha(b - Ay) - \alpha b$, $y \in \mathbb{R}^n$,

$$\|x_{k+1} - x\|_A \leq \|x_k + \alpha r_k - x\|_A = \|T_\alpha x_k + \alpha b - T_\alpha x - \alpha b\|_A = \|T_\alpha(x_k - x)\|_A.$$

Sei $v_1, \ldots, v_n \in \mathbb{R}^n$ eine Orthonormalbasis aus Eigenvektoren von A und $\lambda_1 \leq \ldots \leq \lambda_n$ die zugehörigen Eigenwerte. Für $\mathbb{R}^n \ni y = \sum_{i=1}^n \beta_i v_i$ gilt

$$\begin{aligned}\|T_\alpha y\|_A^2 = (T_\alpha y, AT_\alpha y) &= \sum_{i,j=1}^n \beta_i(1 - \alpha\lambda_i)\beta_j\lambda_j(1 - \alpha\lambda_j)\underbrace{(v_i, v_j)}_{\delta_{ij}} \\ &= \sum_{i=1}^n \beta_i^2\lambda_i|1 - \alpha\lambda_i|^2 \leq \underbrace{\max_{i=1,\ldots,n} |1 - \alpha\lambda_i|^2}_{\rho^2(T_\alpha)} \sum_{i=1}^n \beta_i^2\lambda_i = \rho^2(T_\alpha)\,\|y\|_A^2.\end{aligned}$$

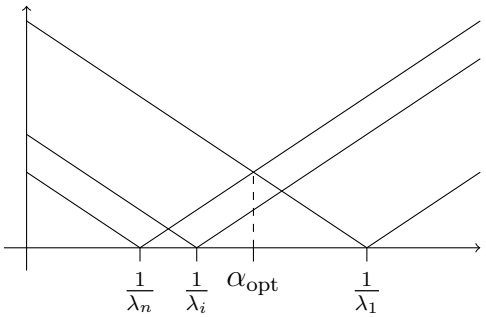
Also folgt $\|x_{k+1} - x\|_A \leq \rho(T_\alpha)\, \|x_k - x\|_A$.

Wir gehen nun der Frage nach, für welches α der Spektralradius von T_α minimal ist. Wegen

$$\rho(T_\alpha) = \max_{i=1,\dots,n} |1 - \alpha\lambda_i| = \max\{|1 - \alpha\lambda_n|,\ |1 - \alpha\lambda_1|\}$$

ist α_{opt} die Schnittstelle der beiden Funktionen

$$g_1(\alpha) := |1 - \alpha\lambda_n| \quad \text{und} \quad g_2(\alpha) := |1 - \alpha\lambda_1|,$$



d.h.

$$-1 + \alpha_{\text{opt}}\lambda_n = 1 - \alpha_{\text{opt}}\lambda_1 \iff \alpha_{\text{opt}} = \frac{2}{\lambda_1 + \lambda_n}.$$

Der Spektralradius für diese Wahl ist

$$\rho(T_{\alpha_{\text{opt}}}) = \frac{\lambda_n - \lambda_1}{\lambda_n + \lambda_1} = \frac{\frac{\lambda_n}{\lambda_1} - 1}{\frac{\lambda_n}{\lambda_1} + 1}.$$

Die Behauptung folgt aus

$$\|A\|_2 = \rho(A) = \lambda_n, \quad \|A^{-1}\|_2 = \rho(A^{-1}) = \frac{1}{\lambda_1}.$$

□

Bemerkung. Wegen

$$\frac{\text{cond}(A) - 1}{\text{cond}(A) + 1} = 1 - \frac{2}{\text{cond}(A) + 1} < 1$$

liegt zwar immer Konvergenz vor, die Konvergenzgeschwindigkeit kann bei großer Kondition aber gering sein.

2.4.2 Methode des steilsten Residuen-Abstiegs

Für die Methode des steilsten Abstiegs ist es wichtig, dass A positiv-definit ist. Betrachtet man beim Abstieg statt der Norm des Fehlers $x_k - x$ die Norm des Residuums $b - Ax_k$, so können allgemeinere Probleme behandelt werden.

Sei $A \in \mathbb{K}^{n\times n}$ regulär. Um in jedem Schritt die Norm des Residuums

$$f(y) := \|b - Ay\|_2^2, \quad y \in \mathbb{K}^n,$$

zu minimieren, kann wegen

$$\|b - Ay\|_2^2 = \|A(y - x)\|_2^2 = (A^H A(y - x), y - x) = \|y - x\|_{A^H A}^2 \tag{2.13}$$

und weil $A^H A$ positiv-definit ist, offenbar die Methode des steilsten Abstiegs auf die Normalengleichungen $A^H Ax = A^H b$ angewendet werden. Entsprechend definiert man hier $v_k := A^H r_k$ mit $r_k = b - Ax_k$ und $w_k := Av_k$. Hieraus erhält man

$$\alpha_k = \frac{\|v_k\|_2^2}{\|w_k\|_2^2}.$$

Pro Iterationsschritt benötigt man daher drei Matrix-Vektor-Multiplikationen. Wegen

$$r_{k+1} = b - Ax_{k+1} = r_k - \alpha_k A v_k$$

kann man das Verfahren aber so umschreiben, dass nur zwei Matrix-Vektor-Multiplikationen benötigt werden:

Algorithmus 2.25 (Methode des steilsten Residuen-Abstiegs).
`Input:` $A \in \mathbb{K}^{n\times n}$ regulär, $b, x_0 \in \mathbb{K}^n$ und Fehlertoleranz $\varepsilon > 0$
`Output:` Folge $\{x_k\}_{k\in\mathbb{N}} \subset \mathbb{K}^n$ von Approximationen an die Lösung von $Ax = b$

$r_0 = b - Ax_0$;
$k = 0$;
`do` {
  $v_k = A^H r_k$;
  $w_k = Av_k$;
  $\alpha_k = \frac{\|v_k\|_2^2}{\|w_k\|_2^2}$;
  $x_{k+1} = x_k + \alpha_k v_k$;
  $r_{k+1} = r_k - \alpha_k w_k$;
  $k = k + 1$;
} `while` $(\|r_k\| > \varepsilon\|b\|)$;

Aus Satz 2.24 erhält man wegen (2.13) die folgende Konvergenzaussage.

Satz 2.26. *Ist $A \in \mathbb{K}^{n\times n}$ regulär, so konvergiert die Methode des steilsten Residuen-Abstiegs für jeden Startwert $x_0 \in \mathbb{R}^n$, d.h. es gilt*

$$\|b - Ax_{k+1}\|_2 \le \frac{\mathrm{cond}^2(A) - 1}{\mathrm{cond}^2(A) + 1} \|b - Ax_k\|_2.$$

Bemerkung. Ziel von Iterationsverfahren ist es, einen möglichst kleinen Fehler $\|x_k - x\|$ zu erzielen. Die Minimierung des Residuums ist nur bedingt von Interesse, weil daraus nicht direkt ein kleiner Fehler $\|x - x_k\|$ folgt. Man hat nämlich im Allgemeinen nur

$$\|x - x_k\| = \|A^{-1}(b - Ax_k)\| \le \|A^{-1}\|\|b - Ax_k\|.$$

Ein kleiner Fehler $\|x_k - x\|$ ergibt sich also aus dem Residuum nur, falls man Stabiltät hat, d.h. falls $\|Ax\| \ge c\|x\|$ mit einer kleinen Konstanten $c > 0$.

2.4.3 Verfahren der konjugierten Gradienten

Das folgende konjugierte Gradienten-Verfahren (engl. conjugate gradients (CG) method) ist wohl das effizienteste bekannte Verfahren zur Lösung linearer Gleichungssysteme $Ax = b$ mit bzgl. eines Skalarproduktes $(\cdot, \cdot)$ positiv-definiter Matrix A. Bei diesem Verfahren sind die Suchrichtungen p_k paarweise *konjugierte* Vektoren.

Definition 2.27. *Zwei Vektoren $x, y \in \mathbb{K}^n$ heißen **konjugiert bzgl. A und** $(\cdot, \cdot)$, falls $(x, y)_A = (Ax, y) = 0$.*

Bemerkung. Sind n Vektoren $v_i \neq 0$, $i = 1, \ldots, n$, paarweise A-konjugiert, so bilden sie eine Basis von $\mathbb{K}^n$. Dies sieht man, weil aus

$$\sum_{i=1}^{n} \beta_i v_i = 0$$

durch Multiplikation mit v_j folgt

$$\sum_{i=1}^{n} \beta_i (v_i, v_j)_A = \beta_j (v_j, v_j)_A$$

und hieraus $\beta_j = 0$, $j = 1, \ldots, n$; vgl. auch Lemma 1.12.

Lemma 2.28. *Seien $p_0, \ldots, p_{n-1}$ paarweise A-konjugierte Vektoren. Dann liefert die durch (2.12) definierte Folge für jedes $x_0 \in \mathbb{K}^n$ nach (höchstens) n Schritten die Lösung $x = A^{-1}b$.*

Beweis. Wegen

$$r_n = b - Ax_n = b - Ax_{n-1} - \alpha_{n-1} Ap_{n-1} = r_{n-1} - \alpha_{n-1} Ap_{n-1} = r_\ell - \sum_{i=\ell}^{n-1} \alpha_i Ap_i$$

für $0 \leq \ell < n$ ergibt sich

$$(r_n, p_\ell) = (r_\ell, p_\ell) - \sum_{i=\ell}^{n-1} \alpha_i (Ap_i, p_\ell) = (r_\ell, p_\ell) - \alpha_\ell (Ap_\ell, p_\ell) = 0.$$

Weil $\{p_0, \ldots, p_{n-1}\}$ eine Basis von $\mathbb{K}^n$ bildet, folgt $r_n = 0$. □

In der Regel ist ein A-konjugiertes System $\{p_0, \ldots, p_{n-1}\}$ von vornherein nicht vorhanden. Es kann aber schrittweise auf Basis des Residuums nach folgender Vorschrift generiert werden:

$$p_0 = r_0, \quad p_{k+1} = r_{k+1} + \gamma_k p_k \quad \text{mit} \quad \gamma_k = -\frac{(Ar_{k+1}, p_k)}{(Ap_k, p_k)}, \ k \geq 0.$$

Lemma 2.29. *Sei $r_j \neq 0$ für $j \leq k$. Dann gilt*

(i) $(r_k, p_j) = 0$ *für alle* $j < k$,

(ii) $(r_k, r_j) = 0$ *für alle* $j < k$,

(iii) die Vektoren $\{p_0, \ldots, p_k\}$ *sind paarweise A-konjugiert.*

Beweis. Wir bemerken zunächst, dass

$$(r_k, p_{k-1}) = (r_{k-1} - \alpha_{k-1} A p_{k-1}, p_{k-1}) = (r_{k-1}, p_{k-1}) - \alpha_{k-1}(A p_{k-1}, p_{k-1}) = 0 \tag{2.14}$$

nach Wahl von α_{k-1}. Wir zeigen die Behauptung per Induktion über k. Für $k = 1$ erhält man (i) und (ii) aus (2.14), (iii) folgt aus

$$(Ap_1, p_0) = (Ar_1, p_0) + \gamma_0 (Ap_0, p_0) = 0.$$

Die Behauptung sei wahr für ein k. Dann erhält man (i) für $j = k$ aus (2.14). Für $0 \leq j < k$ folgt (i) mit der Induktionsannahme aus

$$(r_{k+1}, p_j) = (r_k - \alpha_k A p_k, p_j) = \underbrace{(r_k, p_j)}_{=0} - \alpha_k \underbrace{(Ap_k, p_j)}_{=0} = 0.$$

Wegen $r_j = p_j - \gamma_{j-1} p_{j-1}$, $0 < j < k+1$, erhält man ferner (ii) aus (i). Die A-Konjugiertheit von p_k und p_{k+1} folgt wegen

$$(Ap_{k+1}, p_k) = (Ar_{k+1}, p_k) + \gamma_k (Ap_k, p_k) = 0.$$

Für $0 \leq j < k$ folgt mit der Induktionsannahme

$$(Ap_{k+1}, p_j) = (Ar_{k+1}, p_j) + \gamma_k (Ap_k, p_j) = (Ar_{k+1}, p_j)$$

und wegen (ii)

$$\overline{\alpha}_j (Ar_{k+1}, p_j) = \overline{\alpha}_j (r_{k+1}, Ap_j) = (r_{k+1}, r_j - r_{j+1}) = \underbrace{(r_{k+1}, r_j)}_{=0} - \underbrace{(r_{k+1}, r_{j+1})}_{=0} = 0.$$

Dabei ist $\alpha_0 = \|r_0\|_2^2 / (Ar_0, r_0) \neq 0$ und für $j > 0$ kann α_j nicht verschwinden, weil sonst $(r_j, p_j) = (r_{j+1}, p_j) = 0$ und somit

$$0 = (p_j, r_j) = (r_j + \gamma_{j-1} p_{j-1}, r_j) = (r_j, r_j) + \gamma_{j-1} \underbrace{(p_{j-1}, r_j)}_{=0} = \|r_j\|^2$$

wäre. Dies widerspricht aber der Vorraussetzung. □

Im folgenden Lemma stellen wir eine Beziehung des CG-Verfahren zum Krylov-Raum $\mathcal{K}_k(A, r_0)$ her.

Lemma 2.30. *Sei $r_j \neq 0$ für $j < k$. Dann gilt*

$$x_k \in x_0 + \mathcal{K}_k(A, r_0) \tag{2.15}$$

und

$$\operatorname{span}\{p_0, \ldots, p_{k-1}\} = \operatorname{span}\{r_0, \ldots, r_{k-1}\} = \mathcal{K}_k(A, r_0). \tag{2.16}$$

Beweis. Wir zeigen den Beweis per Induktion über k. Für $k = 1$ sind (2.15) und (2.16) offenbar wahr. Aus $p_k = r_k + \gamma_{k-1} p_{k-1}$ erhält man

$$\operatorname{span}\{p_0, \ldots, p_k\} = \operatorname{span}\{r_0, \ldots, r_k\}.$$

Mit $r_k = r_{k-1} - \alpha_{k-1} A p_{k-1}$ sieht man

$$\operatorname{span}\{r_0, \ldots, r_k\} = \mathcal{K}_{k+1}(A, r_0).$$

(2.15) folgt nun aus $x_{k+1} = x_k + \alpha_k p_k$. □

Bemerkung. Die Wahl der Parameter nach Fletcher-Reeves

$$\alpha_k = \frac{(r_k, r_k)}{(Ap_k, p_k)}, \quad \gamma_k = \frac{(r_{k+1}, r_{k+1})}{(r_k, r_k)}$$

liefert wegen der Orthogonalitätsbeziehung von Lemma 2.29 ein mathematisch äquivalentes Verfahren, das sich in der Praxis allerdings als stabiler und effizienter erweist.

Algorithmus 2.31 (CG-Verfahren).
`Input:` $A \in \mathbb{K}^{n\times n}$ positiv-definit bzgl. $(\cdot,\cdot)$, $b, x_0 \in \mathbb{K}^n$ und Fehlertoleranz $\varepsilon > 0$
`Output:` Folge $\{x_k\}_{k\in\mathbb{N}}$ von Approximationen an die Lösung von $Ax = b$

$p_0 := r_0 = b - Ax_0$;
$k = 0$;
`do` {
  $\alpha_k = \frac{(r_k, r_k)}{(Ap_k, p_k)}$;
  $x_{k+1} = x_k + \alpha_k p_k$;
  $r_{k+1} = r_k - \alpha_k A p_k$;
  $\gamma_k = \frac{(r_{k+1}, r_{k+1})}{(r_k, r_k)}$;
  $p_{k+1} = r_{k+1} + \gamma_k p_k$;
  $k = k + 1$;
} `while` $(\|r_k\| > \varepsilon \|b\|)$;

Die Iterierten x_k des CG-Verfahrens erweisen sich als Bestapproximationen an x im Krylov-Raum $\mathcal{K}_k(A, r_0)$; vgl. Lemma 2.10. Das CG-Verfahren ist also eine andere Realisierung der FOM im positiv-definiten Fall.

Lemma 2.32. *Es gilt*

$$\|x - x_k\|_A = \min_{y \in x_0 + \mathcal{K}_k(A, r_0)} \|x - y\|_A.$$

Insbesondere gilt wegen $\mathcal{K}_k(A, r_0) \subset \mathcal{K}_{k+1}(A, r_0)$, dass $\|x_{k+1} - x\|_A \le \|x_k - x\|_A$. Das CG-Verfahren ist also monoton.

Beweis. Nach Lemma 2.30 wissen wir, dass $x_k \in x_0 + \mathcal{K}_k(A, r_0)$. Für $y \in x_0 + \mathcal{K}_k(A, r_0)$ setze $\delta := x_k - y \in \mathcal{K}_k(A, r_0)$. Dann gilt

$$\|x - y\|_A^2 = \|x - x_k + x_k - y\|_A^2 = \|x - x_k\|_A^2 + \|\delta\|_A^2 + 2\operatorname{Re}(\underbrace{A(x - x_k)}_{r_k}, \delta).$$

Nach Lemma 2.30 ist $\delta \in \mathcal{K}_k(A, r_0) = \operatorname{span}\{p_0, \ldots, p_{k-1}\}$, und Lemma 2.29 impliziert $(r_k, \delta) = 0$. Also wird das Minimum von

$$\|x - y\|_A^2 = \|x - x_k\|_A^2 + \|\delta\|_A^2$$

genau für $\delta = 0 \iff y = x_k$ angenommen. □

Weil normalerweise eine Genauigkeit $\varepsilon > 0$ der Iterierten x_k ausreichend ist, werden in Algorithmus 2.31 oft weniger als n Schritte ausgeführt. Um die Anzahl der Schritte, die für eine vorgegebene Genauigkeit benötigt werden, abzuschätzen, geben wir die folgende Fehlerabschätzung an.

Satz 2.33. *Ist $A \in \mathbb{K}^{n\times n}$ positiv-definit bzgl. $(\cdot,\cdot)$, so konvergiert das CG-Verfahren und es gilt*

$$\|x - x_k\|_A \le 2\left(\frac{\sqrt{\operatorname{cond}(A)} - 1}{\sqrt{\operatorname{cond}(A)} + 1}\right)^k \|x - x_0\|_A, \quad k = 1, \ldots, n-1.$$

Beweis. Nach Lemma 2.30 gilt $x_k = x_0 + p_k(A)r_0$ für ein $p_k \in \Pi_{k-1}$ und somit

$$x - x_k = x - x_0 - p_k(A)A(x - x_0) = q(A)(x - x_0), \quad q(x) := 1 - x\,p_k(x).$$

Es gilt $q \in \tilde{\Pi}_k := \{p \in \Pi_k :\ p(0) = 1\}$. Ferner gilt nach Lemma 2.32

$$\|x - x_k\|_A = \min_{p\in\tilde{\Pi}_k} \|p(A)(x - x_0)\|_A.$$

Sei $v_1, \ldots, v_n$ eine Orthonormalbasis aus Eigenvektoren zu den Eigenwerten $\lambda_1 \le \ldots \le \lambda_n$ von A und $x - x_0 = \sum_{i=1}^n \beta_i v_i$. Dann folgt aus

$$p(A)(x - x_0) = \sum_{i=1}^n \beta_i p(A)v_i = \sum_{i=1}^n \beta_i p(\lambda_i)v_i,$$

dass

$$\begin{aligned}\|p(A)(x - x_0)\|_A^2 &= (p(A)(x - x_0), p(A)(x - x_0))_A = \sum_{i,j=1}^n \beta_i\overline{\alpha}_j p(\lambda_i)\overline{p(\lambda_j)}(v_i, v_j)_A \\ &= \sum_{i=1}^n |\beta_i|^2|p(\lambda_i)|^2 \le \max_{i=1,\ldots,n} |p(\lambda_i)|^2 \sum_{i=1}^n |\beta_i|^2 \\ &= \max_{i=1,\ldots,n} |p(\lambda_i)|^2 \, \|x - x_0\|_A^2\end{aligned}$$

und somit

$$\|x - x_k\|_A \le \min_{p\in\tilde{\Pi}_k} \max_{i=1,\ldots,n} |p(\lambda_i)| \, \|x - x_0\|_A.$$

Daher genügt es, ein Polynom in $\tilde{\Pi}_k$ zu finden, das die gewünschte Abschätzung liefert. Ist $\lambda_n = \lambda_1$, so existiert $p \in \tilde{\Pi}_k$ mit $p(\lambda_1) = 0$, was $\|x - x_k\|_A = 0$ zeigt. Im Fall $\lambda_n > \lambda_1$ verwenden wir nach Satz 2.18 das Tschebyscheff-Polynom

$$\hat{T}_k(x) = \frac{T_k(t(x))}{T_k(t_0)} \in \tilde{\Pi}_k, \quad t(x) = 1 - 2\frac{x - \lambda_1}{\lambda_n - \lambda_1}, \quad t_0 := t(0) = \frac{\lambda_n + \lambda_1}{\lambda_n - \lambda_1} > 1. \tag{2.17}$$

Nach Satz 2.17 (vi) gilt

$$T_k(t_0) \geq \frac{1}{2}\left(t_0 + \sqrt{t_0^2 - 1}\right)^k = \frac{1}{2}\left(\frac{\sqrt{\frac{\lambda_n}{\lambda_1}} + 1}{\sqrt{\frac{\lambda_n}{\lambda_1}} - 1}\right)^k$$

und nach Satz 2.17 (ii)

$$\max_{i=1,\dots,n} |\hat{T}_k(\lambda_i)| \leq \frac{1}{T_k(t_0)} \leq 2\left(\frac{\sqrt{\operatorname{cond}(A)} - 1}{\sqrt{\operatorname{cond}(A)} + 1}\right)^k.$$

□

Bemerkung. Will man mit Hilfe des CG-Verfahrens die exakte Lösung berechnen, so müssen höchstens n Schritte durchgeführt werden, von denen jeder nur eine Multiplikation von A mit einem Vektor benötigt. Ist A schwachbesetzt, so genügen $O(n)$ Operationen pro Iterationsschritt. Ist ferner eine Genauigkeit $\varepsilon > 0$ der Approximation an die Lösung ausreichend, so gilt nach dem letzten Satz $\|x - x_k\|_A \leq 2\eta^k \|x - x_0\|_A$ mit einem $\eta < 1$. Wegen $\log 1/x \geq 1 - x$ für alle $0 < x < 1$ gilt für

$$k \geq \frac{\sqrt{\operatorname{cond}(A)} + 1}{2} |\log \varepsilon/2| = \frac{1}{1-\eta} |\log \varepsilon/2| \geq \frac{|\log \varepsilon/2|}{\log 1/\eta} = \log_\eta(\varepsilon/2),$$

dass $2\gamma^k < \varepsilon$. Also werden $k \sim \sqrt{\operatorname{cond}(A)}\,|\log \varepsilon|$ Schritte benötigt, um einen relativen Fehler ε für x_k bei $x_0 = 0$ zu garantieren. In diesem Fall ist die Gesamtkomplexität von der Ordnung $n\sqrt{\operatorname{cond}(A)}\,|\log \varepsilon|$.

Wir wollen die Konvergenz des in Algorithmus 2.31 vorgestellten Verfahrens der konjugierten Gradienten genauer untersuchen. Nach Satz 2.33 spielt für die Konvergenzgeschwindigkeit des CG-Verfahrens das Verhältnis von größtem zu kleinstem Eigenwert von A eine entscheidende Rolle. Man beobachtet jedoch, dass wenige kleine Eigenwerte λ_i', $i = 1, \dots, p$, und wenige große Eigenwerte λ_j'', $j = 1, \dots, q$, (sog. "outliers") die Konvergenzgeschwindigkeit nicht beeinflussen. Um diesen Effekt zu analysieren, schreiben wir das Spektrum von A in der Form

$$\sigma(A) = \left(\bigcup_{i=1}^{p} \lambda_i'\right) \cup M \cup \left(\bigcup_{j=1}^{q} \lambda_j''\right), \quad M \subset [a, b],$$

mit $0 < \lambda_i' < a$, $i = 1, \dots, p$, und $b < \lambda_j''$, $j = 1, \dots, q$. Die folgende Fehlerabschätzung zeigt, dass $p + q$ zusätzliche Schritte nötig sind, um die p kleinsten und q größten Eigenwerte abzuarbeiten. Danach konvergiert das CG-Verfahren, als wäre das Spektrum in $[a, b]$ enthalten, d.h. anstelle von $\operatorname{cond}(A) = \lambda_{\max}/\lambda_{\min}$ wird die Konvergenzrate durch das kleinere Verhältnis b/a bestimmt.

Satz 2.34. *Sei $A \in \mathbb{K}^{n\times n}$ positiv-definit bzgl. $(\cdot,\cdot)$. Für den Fehler des CG-Verfahrens gilt*

$$\|x - x_k\|_A \leq 2(\operatorname{cond}(A) + 1)^p \left(\frac{\sqrt{b/a} - 1}{\sqrt{b/a} + 1}\right)^{k-p-q} \|x - x_0\|_A, \quad k = p + q, \dots, n - 1.$$

Beweis. Anstelle von (2.17) wählen wir

$$\hat{T}_k(x) = \frac{T_k(t(x))}{T_k(t_0)} \prod_{i=1}^{p} \left(1 - \frac{x}{\lambda_i'}\right) \prod_{j=1}^{q} \left(1 - \frac{x}{\lambda_j''}\right) \in \tilde{\Pi}_{k+p+q}$$

mit

$$t(x) = 2\frac{x-a}{b-a} - 1 \quad \text{und} \quad t_0 := t(0) = \frac{b+a}{b-a} > 1.$$

Weil für $\lambda \in \{\lambda_1', \ldots, \lambda_p', \lambda_1'', \ldots, \lambda_q''\}$ gilt

$$\prod_{i=1}^{p} \left(1 - \frac{\lambda}{\lambda_i'}\right) \prod_{j=1}^{q} \left(1 - \frac{\lambda}{\lambda_j''}\right) = 0$$

und für $x \in [a,b]$

$$\left|1 - \frac{x}{\lambda_i'}\right| \leq \operatorname{cond}(A) + 1 \quad \text{und} \quad \left|1 - \frac{x}{\lambda_j''}\right| \leq 1,$$

folgt

$$\begin{aligned} \max_{\lambda \in \sigma(A)} |\hat{T}_k(\lambda)| \leq \max_{x \in [a,b]} |\hat{T}_k(x)| &\leq (\operatorname{cond}(A)+1)^p \max_{x \in [a,b]} \frac{|T_k(t(x))|}{|T_k(t_0)|} \\ &\leq 2(\operatorname{cond}(A)+1)^p \left(\frac{\sqrt{b/a}-1}{\sqrt{b/a}+1}\right)^k. \end{aligned}$$

Daher gilt

$$\|x_{k+p+q} - x\|_A \leq \min_{\xi \in \tilde{\Pi}_{k+p+q}} \max_{\lambda \in \sigma(A)} |\xi(\lambda)| \leq \max_{\lambda \in \sigma(A)} |\hat{T}_k(\lambda)| \leq 2(\operatorname{cond}(A)+1)^p \left(\frac{\sqrt{b/a}-1}{\sqrt{b/a}+1}\right)^k$$

und somit die Behauptung. □

2.4.4 Das vorkonditionierte CG-Verfahren (PCG)

Seien A und C positiv-definit bzgl. des Skalarproduktes $(\cdot,\cdot)$. Wir wollen wieder vorkonditionierte Systeme mit Koeffizientenmatrix CA bzw. AC betrachten. Um das CG-Verfahren anwenden zu können, müssen diese Matrizen aber positiv-definit sein. Im Allgemeinen werden aber weder AC noch CA selbstadjungiert bzgl. $(\cdot,\cdot)$ sein. Allerdings ist die Matrix CA positiv-definit bzgl. $(x,y)_{C^{-1}} := (C^{-1}x, y)$, weil für $x, y \in \mathbb{K}^n$ gilt

$$(CAx, y)_{C^{-1}} = (Ax, y) = (x, Ay) = (CC^{-1}x, Ay) = (C^{-1}x, CAy) = (x, CAy)_{C^{-1}}.$$

Insbesondere gilt $(CAx, x)_{C^{-1}} = (Ax, x) > 0$ für $x \neq 0$. Anstelle von linken Vorkonditionieren kann man auch rechte Vorkonditionierer betrachten. Dann ist AC positiv-definit bzgl. $(\cdot,\cdot)_C$, weil für $x, y \in \mathbb{K}^n$ gilt

$$(ACx, y)_C = (CACx, y) = (ACx, Cy) = (Cx, ACy) = (x, ACy)_C,$$

und es ist $(ACx, x)_C = (ACx, Cx) > 0$, $x \neq 0$. Die **symmetrische Vorkonditionierung** $C^{1/2}AC^{1/2}$ erhält die Positivität von A, bedarf aber der Berechnung von $C^{1/2}$.

Wir werden uns beim **vorkonditionierten Gradienten-Verfahren** (PCG) auf rechte Vorkonditionierer konzentrieren, d.h. wir wenden das CG-Verfahren bzgl. des Skalarproduktes $(\cdot,\cdot)_C$ auf die Systemmatrix AC an. Die entsprechenden Variablen kennzeichnen wir jeweils durch $\tilde{\cdot}$. Der folgende Algorithmus ergibt sich aus dem CG-Verfahren durch Ersetzen der Variablen $\tilde{p}_k$ und $\tilde{x}_k$ durch $p_k := C\tilde{p}_k$ und $x_k := C\tilde{x}_k$.

Algorithmus 2.35 (PCG-Verfahren).
`Input:` A, C positiv-definit bzgl. $(\cdot,\cdot)$, $b, x_0 \in \mathbb{K}^n$ und Fehlertoleranz $\varepsilon > 0$
`Output:` Folge $\{x_k\}_{k\in\mathbb{N}}$ von Approximationen an die Lösung von $Ax = b$

$r_0 = b - Ax_0$, $p_0 = v_0 = Cr_0$ und $\rho_0 = (v_0, r_0)$;
$k = 0$;
`do` {
  $w_k = Ap_k$;
  $x_{k+1} = x_k + \alpha_k p_k$, wobei $\alpha_k = \frac{\rho_k}{(p_k, w_k)}$;
  $r_{k+1} = r_k - \alpha_k w_k$;
  $v_{k+1} = Cr_{k+1}$;
  $\rho_{k+1} = (v_{k+1}, r_{k+1})$;
  $p_{k+1} = v_{k+1} + \gamma_k p_k$, wobei $\gamma_k := \frac{\rho_{k+1}}{\rho_k}$;
  $k = k + 1$;
} `while` $(\|r_k\| > \varepsilon\|b\|)$;

Die durch $(\cdot,\cdot)_C$ induzierte Energie-Norm stimmt mit $\|\cdot\|_{CAC}$ überein. Berücksichtigt man die Umbenennung der Variablen, so erhält man aus Satz 2.33 wieder die Abschätzungen bzgl. derselben Norm $\|\cdot\|_A$. Auch Satz 2.34 gilt entsprechend mit den Eigenwerten von AC.

Satz 2.36. *Für den Fehler des vorkonditionieren konjugierten Gradienten-Verfahrens gilt*

$$\|x_k - x\|_A \le 2\left(\frac{\sqrt{\kappa}-1}{\sqrt{\kappa}+1}\right)^k \|x_0 - x\|_A, \quad k = 0, \dots, n-1,$$

wobei $\kappa := \lambda_{\max}(AC)/\lambda_{\min}(AC)$.

Man beachte, dass in Satz 2.33 die Konditionszahl durch $\kappa = \lambda_{\max}(AC)/\lambda_{\min}(AC)$ zu ersetzen ist. Diese beiden Ausdrücke stimmen nicht überein, weil AC im Allgemeinen nicht hermitesch ist. Das folgende Lemma stellt dabei sicher, dass die Eigenwerte von AC bzw. CA reell sind.

Lemma 2.37. *Seien $A \in \mathbb{K}^{n\times n}$ positiv-definit und $B \in \mathbb{K}^{n\times n}$ hermitesch. Dann sind die Eigenwerte von AB und BA reell.*

Beweis. Sei $BAx = \lambda x$ mit $x \neq 0$. Multiplikation mit $x^H A$ von links ergibt

$$x^H ABAx = \lambda x^H Ax \quad \Longleftrightarrow \quad x^H ABAx = \overline{\lambda} x^H Ax.$$

Hieraus folgt $(\lambda - \overline{\lambda})x^H Ax = 0$. Wegen $x \neq 0$ ist $x^H Ax \neq 0$. Also ist λ reell. Wegen $\sigma(BA) \setminus \{0\} = \sigma(AB) \setminus \{0\}$ sind auch die Eigenwerte von BA reell. □

Gelingt es mit Hilfe eines passenden Vorkonditionierers, κ unabhängig von n zu beschränken, so existiert nach dem letzten Satz eine von n unabhängige Konstante $0 < \eta < 1$, so dass

$$\|x - x_k\|_A \le 2\eta^k \|x - x_0\|_A.$$

Eine vorgegebene relative Genauigkeit $\varepsilon > 0$ wird also nach $k \sim |\log \varepsilon|$ Schritten erreicht. Wenn man berücksichtigt, dass ε üblicherweise von n abhängt, sind dies $O(\log n)$ Schritte.

Bemerkung. Die nicht-hermitesche Matrix $A \in \mathbb{K}^{m\times n}$ habe vollen Rang. Wir betrachten das lineare Gleichungssystem $Ax = b$ bei gegebener rechter Seite $b \in \operatorname{Im} A$. Eines der einfachsten Verfahren zur Lösung eines solchen Systems, das **CGN-Verfahren**, erhält man durch Anwendung des CG-Verfahrens auf die Normalengleichungen

$$A^H Ax = A^H b.$$

Die positiv-definite Matrix $A^H A$ wird dabei nicht explizit berechnet, sondern geht in das Verfahren nur durch zwei Matrix-Vektor-Multiplikationen $A^H Av = A^H(Av)$ ein.

Wir wissen bereits, dass $\|z - x\|_{A^H A}$ durch $z = x_k$ minimiert wird. Wegen

$$\|Az - b\|_2^2 = \|A(z - x)\|_2^2 = (A^H A(z - x), z - x) = \|z - x\|_{A^H A}^2$$

minimiert x_k daher wie beim GMRES-Verfahren auch das Residuum. CGN und GMRES sind trotz dieser gemeinsamen Eigenschaft wegen der unterschiedlichen Definition der Krylov-Räume jedoch nicht äquivalent. Die Konvergenz von GMRES hängt von der Verteilung der Eigenwerte von A ab, während CGN von den Eigenwerten von $A^H A$ also den Singulärwerten von A bestimmt ist. Daher eignet sich das Verfahren besonders, wenn die Singulärwerte statt der Eigenwerte geclustert sind. Ferner gilt

$$\|r_k\|_2 \leq 2\left(\frac{\operatorname{cond}(A) - 1}{\operatorname{cond}(A) + 1}\right)^k \|r_0\|_2.$$

Im Vergleich zum CG-Verfahren ist hier also die Kondition quadriert.

2.5 Vorkonditionierer

Die Konvergenzgeschwindigkeit der in diesem Kapitel vorgestellten Iterationsverfahren zur Lösung von Gleichungssystemen $Ax = b$ ist durch die das Spektrum von A bestimmt. Wir hatten bereits erwähnt, dass mit Hilfe von Vorkonditionierern die Konvergenzeigenschaften verbessert werden können.

Die Kondition einer Matrix kann schon für kleine Dimensionen sehr groß sein. In diesem Fall ist man an einem Vorkonditionierer für eine feste Matrixdimension interessiert. Üblicherweise steigt die Konditionszahl aber erst für wachsende Problemgrößen. In diesem Fall hängt die Konvergenzgeschwindigkeit von n ab. Daher ist es unerlässlich, die Spektraleigenschaften von Folgen von Matrizen zu untersuchen.

Definition 2.38. *Zwei Folgen $\{A_n\}_{n\in\mathbb{N}}, \{C_n\}_{n\in\mathbb{N}}$ positiv-definiter Matrizen $A_n, C_n \in \mathbb{K}^{n\times n}$ heißen **spektraläquivalent**, falls für jedes n Konstanten $\gamma_n, \Gamma_n > 0$ mit $\Gamma_n/\gamma_n < c$ existieren, so dass*

$$\gamma_n(C_n x, x) \leq (A_n x, x) \leq \Gamma_n(C_n x, x) \quad \text{für alle } x \in \mathbb{K}^n$$

mit einer von n unabhängigen Konstanten $c > 0$.

Nach dem folgenden Lemma ist die Konvergenzrate des PCG-Verfahrens genau dann unabhängig von n, wenn Systemmatrix und Vorkonditionierer spektraläquivalent sind.

Lemma 2.39. *$\{A_n\}_{n\in\mathbb{N}}$ und $\{C_n\}_{n\in\mathbb{N}}$ sind genau dann spektraläquivalent, wenn*

$$\frac{\lambda_{\max}\left(A_n C_n^{-1}\right)}{\lambda_{\min}\left(A_n C_n^{-1}\right)}$$

unabhängig von n beschränkt ist.

Beweis. Übungsaufgabe □

Wir wollen einige Vorkonditionierungstechniken, die im Zusammenhang mit Differentialgleichungen verwendet werden, vorstellen.

2.5.1 ILU-Vorkonditionierer

Sei $A \in \mathbb{K}^{n\times n}$ eine schwachbesetzte Matrix. Wir wissen bereits, dass eine LR-Zerlegung von A im Allgemeinen vollbesetzt ist. Diesen Effekt umgeht man, indem Einträge außerhalb eines vorgegebenen **Besetzungsmusters** $P \subset I \times I$ mit $(i,i) \in P$ für alle $i \in I := \{1,\dots,n\}$ künstlich auf Null gesetzt werden (sog. "Dropping"). Für P kann beispielsweise das Besetzungsmuster von A

$$P_A := \{(i,j) : a_{ij} \neq 0\}$$

verwendet werden. Unter einer **unvollständigen LR-Zerlegung** (engl. incomplete LU factorization) versteht man eine Zerlegung

$$A = LR + E$$

mit (normierter) unterer und oberer Dreiecksmatrix L bzw. R, so dass

$$\ell_{ij} = 0 = r_{ij}, \quad (i,j) \notin P, \quad \text{und} \quad e_{ij} = 0, \quad (i,j) \in P.$$

Die Einträge des Fehlers $E = A - LR$ verschwinden also auf dem Besetzungsmuster P und die Einträge von L und R an den übrigen Positionen. Abhängig von der Wahl des Besetzungsmusters P kann ein kleiner oder ein großer Fehler E erwartet werden. Ist $P = I\times I$, so reproduzieren wir die gewöhnliche LR-Zerlegung und es ist $E = 0$. Umgekehrt kann auch bei vorgegebener Genauigkeit das Besetzungsmuster adaptiert werden.

Algorithmus 2.40 (ILU-Zerlegung).

```
for k = 1,...,n-1
   for i = k+1,...,n
      if (i,k) ∈ P {
         a_ik := a_ik / a_kk;
         for j = k+1,...,n
            if (i,j) ∈ P then a_ij := a_ij - a_ik a_kj;
      }
```

Als Ergebnis des Algorithmus definieren wir L und R durch

$$\ell_{ij} = \begin{cases} a_{ij}, & (i,j) \in P \text{ und } i > j, \\ 1, & i = j, \\ 0, & \text{sonst}, \end{cases} \quad \text{und} \quad r_{ij} = \begin{cases} a_{ij}, & (i,j) \in P \text{ und } j \geq i, \\ 0, & \text{sonst}. \end{cases} \tag{2.18}$$

Bemerkung. Die Berechnung der ILU-Zerlegung benötigt $O(n)$ Operationen und $O(n)$ Speicher. Zur Vorkonditionierung von A kann somit $C := (LR)^{-1} = R^{-1}L^{-1}$ als Vorkonditionierer verwendet werden. Dabei ist zu berücksichtigen, dass die Inversen nicht explizit berechnet werden müssen. Es genügt, sie durch Vorwärts- bzw. Rückwärtssubstitution anzuwenden.

Satz 2.41. *Sei $P \supset P_A$. Das obige Verfahren breche nicht vorzeitig ab. Dann sind die Faktoren L und R regulär und $e_{ij} = 0$ für $(i,j) \in P$.*

Beweis. Im k-ten Schritt des obigen Algorithmus gilt

$$A^{(k)} = L_k A^{(k-1)} - E^{(k)} \tag{2.19}$$

mit $A^{(0)} = A$ und der Gauß-Matrix $L_k := I - \ell_k e_k^T$,

$$\ell_k := \frac{1}{a_{kk}^{(k-1)}}[\underbrace{0,\ldots,0}_{k}, a_{k+1,k}^{(k-1)}, \ldots, a_{nk}^{(k-1)}]^T.$$

Die Matrix $E^{(k)}$ enthält die im k-ten Schritt künstlich zu Null gesetzten Einträge. Für die in (2.18) definierten Matrizen gilt $L = (L_{n-1} \cdot \ldots \cdot L_1)^{-1}$ und $R = A^{(n-1)}$. Also erhält man aus (2.19)

$$R = L^{-1}A - F, \quad F := L_{n-1} \cdot \ldots \cdot L_2 E^{(1)} + \ldots + L_{n-1}E^{(n-2)} + E^{(n-1)}.$$

Weil im k-ten Schritt nur unterhalb der k-ten Zeile von $A^{(k-1)}$ Einträge zu Null gesetzt werden, gilt $e_i^T E^{(k)} = 0$, $i = 1,\ldots,k$. Hieraus folgt sofort

$$L_{n-1} \cdot \ldots \cdot L_{k+1} E^{(k)} = L_{n-1} \cdot \ldots \cdot L_1 E^{(k)}$$

mit der Konsequenz, dass $F = L_{n-1} \cdot \ldots \cdot L_1 E$, wobei $E := E^{(1)} + E^{(2)} + \ldots + E^{(n-1)}$. Wir erhalten also die gewünschte Zerlegung

$$A = LR + E,$$

bei der E an den Positionen aus P verschwindet. □

Bemerkung. Der letzte Beweis zeigt, dass in E die während des Algorithmus zu Null gesetzten Einträge stehen.

Für sog. *M-Matrizen* kann garantiert werden, dass obiger Algorithmus nicht vorzeitig abbricht. Wir verwenden die Notation $A \geq 0 \iff a_{ij} \geq 0$, $i,j = 1,\ldots,n$.

Definition 2.42. *Eine invertierbare Matrix $A \in \mathbb{R}^{n\times n}$ heißt **M-Matrix**, falls*

(i) $a_{ij} \leq 0$ *für alle* $i \neq j$,

(ii) $A^{-1} \geq 0$.

Bemerkung.

(i) Man kann zeigen, dass die Finite-Differenzen-Diskretisierungen des negativen Laplace-Operators aus Beispiel 2.2 M-Matrizen sind.

(ii) Mit der zweiten Bedingung $A^{-1} \geq 0$ folgt aus $Ax \leq Ay$, dass $x \leq y$.

(iii) Sei $a_i := Ae_i$ die i-te Spalte von A. Wäre $a_i \leq 0$, so hätte man $e_i \leq 0$ nach (ii). Aus diesem Widerspruch erhält man, dass die Diagonaleinträge a_{ii} von M-Matrizen positiv sind.

Weil der Eintrag a_{11} einer M-Matrix A positiv ist, kann der erste Eliminationsschritt ohne Pivotisierung durchgeführt werden. Dann hat die erste Gauß-Matrix die Gestalt

$$L_1 = I - \ell_1 e_1^T, \quad \ell_1 := \frac{1}{a_{11}} \begin{bmatrix} 0 \\ a_{21} \\ \vdots \\ a_{n1} \end{bmatrix}.$$

Lemma 2.43. *Sei A eine M-Matrix. Dann sind sowohl $L_1 A$ als auch die Matrix, die aus $L_1 A$ durch Löschen der ersten Spalte und ersten Zeile entsteht, M-Matrizen.*

Beweis. Weil L_1 und A regulär sind, ist auch $B := L_1 A$ regulär. Die erste Zeile von B und A stimmen überein. Die Außerdiagonaleinträge der ersten Spalte von B verschwinden. Für $i, j = 2, \ldots, n$, $i \neq j$, ist

$$b_{ij} = a_{ij} - \frac{a_{i1} a_{1j}}{a_{11}} \leq 0.$$

Schließlich ist

$$B^{-1} = A^{-1} L_1^{-1} = A^{-1}(I + \ell_1 e_1^T)$$

und daher

$$B^{-1} e_1 = \frac{1}{a_{11}} A^{-1} \begin{bmatrix} a_{11} \\ \vdots \\ a_{n1} \end{bmatrix} = \frac{1}{a_{11}} e_1 \geq 0$$

und $B^{-1} e_j = A^{-1} e_j \geq 0$ für $j \geq 2$. Also ist $B^{-1} \geq 0$ und B somit eine M-Matrix.

Mit der Zerlegung

$$B = \begin{bmatrix} a_{11} & a^T \\ 0 & \hat{B} \end{bmatrix}, \quad a := \begin{bmatrix} a_{12} \\ \vdots \\ a_{1n} \end{bmatrix},$$

sieht man, dass mit B auch $\hat{B}$ regulär ist. Es bleibt also zu zeigen, dass $\hat{B}^{-1} \geq 0$. Dies folgt aber aus

$$0 \leq B^{-1} = \begin{bmatrix} 1/a_{11} & -a^T (\hat{A}^{(1)})^{-1} / a_{11} \\ 0 & \hat{B}^{-1} \end{bmatrix}.$$

□

Das folgende Lemma analysiert den Einfluss des Droppings auf die M-Matrix-Eigenschaft. Jeder zu Null gesetzte Eintrag ist nicht-positiv, d.h. man erhält

$$A^{(1)} = L_1 A - E \geq L_1 A$$

mit $e_{ii} = 0$ und $e_{ij} \leq 0$, $i \neq j$. Wir geben die Aussage ohne Beweis an.

Lemma 2.44. *Sei A eine M-Matrix, und $B \geq A$ erfülle $b_{ij} \leq 0$, $i \neq j$. Dann ist auch B eine M-Matrix und $B^{-1} \leq A^{-1}$.*

Der nächste Schritt kann nun durchgeführt werden, bis die unvollständige LR-Zerlegung von A berechnet ist. Zusammenfassend erhalten wir in Ergänzung zu Satz 2.41:

Satz 2.45. *Sei A eine M-Matrix. Dann bricht Algorithmus 2.40 nicht vorzeitig ab. Die berechneten Faktoren L und R sind regulär, $E = A - LR \leq 0$ und $(LR)^{-1} \geq 0$.*

Beweis. Für M-Matrizen ist jedes $E^{(k)}$ in (2.19) nicht-positiv. Daher gilt auch für die Summe $E \leq 0$. Wegen

$$LR = A - E \geq A$$

ist nach Lemma 2.44 auch LR eine M-Matrix und daher $(LR)^{-1} \geq 0$. □

Natürlich kann analog zur LR-Zerlegung auch eine unvollständige Choleky-Zerlegung (IC) bestimmt werden. Dann gilt $A = LL^T + E$, und neben der linken- bzw. rechten Vorkonditionierung mit $C := (LL^T)^{-1} = L^{-T}L^{-1}$ kann auch das CG-Verfahren auf die positiv-definite Matrix $L^{-1}AL^{-T}$ (symmetrische Vorkonditionierung) angewendet werden.

2.5.2 Approximative-Inverse-Vorkonditionierer

Ähnlich wie bei der LR-Zerlegung ist auch die Inverse einer schwachbesetzten Matrix im Allgemeinen vollbesetzt. Ziel der **Approximative-Inverse-Vorkonditionierer** ist es, $C \in \mathbb{K}^{n\times n}$ mit Besetzungsmuster P, d.h. $c_{ij} = 0$ für $(i,j) \notin P$, so zu finden, dass das Funktional

$$f(C) := \|I - AC\|_F^2$$

minimiert wird.

Bei Abstiegsverfahren (vgl. Abschnitt 2.4) wird ausgehend von C ein neues C' in Richtung $G \in \mathbb{K}^{n\times n}$ gesucht:

$$C' = C + \alpha G.$$

Der Parameter α wird als Minimum des Funktionals f bestimmt. Wir wissen bereits, dass die Minimierung des Residuums $R := I - AC$ äquivalent mit der Bedingung $R - \alpha AG \perp AG$ ist. Hieraus folgt

$$\alpha = \frac{(R, AG)}{\|AG\|_F^2},$$

wobei

$$(A, B) := \operatorname{trace} B^H A = \sum_{i,j=1}^{n} a_{ij}\overline{b_{ij}}$$

das die Frobenius-Norm induzierende Skalarprodukt auf $\mathbb{K}^{n\times n}$ bezeichnet. Durch diesen Update füllt die Matrix C auf, so dass Einträge außerhalb des Besetzungsmusters P verworfen werden müssen. Dann gilt aber nicht mehr, dass $f(C') \leq f(C)$ ist.

Eine mögliche Wahl für die Richtung ist das Residuum $G = R$. Eine andere Möglichkeit ist der steilste Abstieg, d.h. die Richtung des negativen Gradienten $-\nabla f = 2A^H R$.

Wegen

$$f(C) = \sum_{j=1}^{n} \|e_j - Ac_j\|_2^2,$$

wobei c_j die j-te Spalte von C bezeichnet, kann man neben der globalen Minimierung auch jede der Funktionen

$$f_j(c) := \|e_j - Ac\|_2^2, \quad j = 1, \ldots, n,$$

einzeln minimieren. Dieser Zugang eignet sich insbesondere für eine Parallelisierung.

3 Numerische Behandlung von Eigenwertproblemen

Wir betrachten quadratische Matrizen $A \in \mathbb{C}^{n\times n}$. Das algebraische Polynom

$$\chi_A(\lambda) := \det(A - \lambda I)$$

über $\mathbb{C}$ vom Grad n wird als **charakteristische Polynom** von A bezeichnet. Jede Nullstelle $\lambda \in \mathbb{C}$ von χ_A wird als **Eigenwert** von A bezeichnet. Die **algebraische Vielfachheit** eines Eigenwertes λ ist die Vielfachheit der Nullstelle λ des charakteristischen Polynoms. Die Menge aller Eigenwerte von A heißt **Spektrum** $\sigma(A) \subset \mathbb{C}$. Jeder Vektor $x \in \mathbb{C}^n \setminus \{0\}$ mit

$$Ax = \lambda x \tag{3.1}$$

heißt **Eigenvektor**. Die Dimension des durch die Eigenvektoren zum Eigenwert λ aufgespannten Raums wird als **geometrische Vielfachheit** von λ bezeichnet.

Nach dem Abel-Ruffini-Theorem können die Nullstellen eines Polynoms im Fall $n > 4$ nicht durch eine endliche Folge von arithmetischen Operationen und Wurzeln berechnet werden. Daher werden wir in diesem Kapitel ausschließlich iterative Verfahren zur numerischen Lösung von (3.1) behandeln. Die wahre Schwierigkeit besteht aber in der effizienten Behandlung großdimensionierter Eigenwertprobleme.

Bemerkung. Die Bedeutung der numerischen Berechnung von Eigenwerten im Zusammenhang mit Eigenschwingungen von Gebäuden kann man am Beispiel der Tacoma Narrows Brücke sehen. Interessant ist auch die ARD-Sendung zu Brücken.

3.1 Theoretische Grundlagen

Das folgende Resultat ist die Grundlage für viele Algorithmen zur numerischen Berechnung von Eigenwerten.

Satz 3.1. *Sind $A, B \in \mathbb{C}^{n\times n}$ **ähnlich**, d.h. existiert $T \in \mathbb{C}^{n\times n}$ mit $B = T^{-1}AT$, so stimmen die Eigenwerte von A und B überein.*

Beweis. Für alle $\lambda \in \mathbb{C}$ gilt

$$\begin{aligned}\chi_B(\lambda) &= \det(T^{-1}AT - \lambda I) = \det(T^{-1}(A - \lambda I)T) = \det(T^{-1})\det(A - \lambda I)\det(T)\\ &= \frac{\det T}{\det T}\det(A - \lambda I) = \chi_A(\lambda).\end{aligned}$$

□

Die Idee vieler numerischer Verfahren ist es, eine Folge von Matrizen $\{T_n\}_{n\in\mathbb{N}}$ zu konstruieren, so dass $\{T_n^{-1}AT_n\}_{n\in\mathbb{N}}$ gegen eine Matrix D konvergiert, deren Eigenwerte leicht zu bestimmen sind. Wie wir später sehen werden, hängen die Eigenwerte stetig von der Matrix ab. Daher stimmen die Eigenwerte von D und A überein.

Der folgende Satz besagt, dass jede Matrix unitär ähnlich zu einer (komplexen) oberen Dreiecksmatrix ist.

Satz 3.2 (Schur). *Zu $A \in \mathbb{C}^{n\times n}$ existiert $U \in \mathbb{C}^{n\times n}$ unitär, so dass U^HAU eine obere Dreiecksmatrix ist.*

Beweis. Für $n = 1$ ist die Aussage trivial. Wir nehmen an, sie sei für $n-1$ gezeigt. Es existiert mindestens ein Eigenpaar (λ, x) von A. Außerdem gibt es nach Householder eine unitäre Matrix $Q \in \mathbb{C}^{n\times n}$ mit $Qx = \alpha e_1$, $\alpha = \|x\|_2 \neq 0$. Dann gilt

$$\lambda\alpha e_1 = \lambda Qx = QAx = QAQ^HQx = \alpha(QAQ^H)e_1$$

und somit $QAQ^He_1 = \lambda e_1$, d.h.

$$QAQ^H = \begin{bmatrix}\lambda & w^T\\ 0 & B\end{bmatrix}.$$

Die $(n-1)\times(n-1)$-Matrix B hat nach Induktionsvorraussetzung die Zerlegung $B = PSP^H$ mit oberer Dreiecksmatrix S und P unitär. Wir definieren

$$\hat{P} = \begin{bmatrix}1 & 0\\ 0 & P\end{bmatrix}.$$

Dann ist auch $\hat{P}$ unitär, und es gilt

$$A = Q^H\hat{P}\begin{bmatrix}\lambda & w^TP\\ 0 & S\end{bmatrix}(Q^H\hat{P})^H.$$

Weil mit Q^H und $\hat{P}$ auch das Produkt $Q^H\hat{P}$ unitär ist, folgt die Behauptung. □

Die Zerlegung $A = URU^H$ mit U unitär und R obere Dreiecksmatrix heißt **Schur-Zerlegung**. Wegen der Ähnlichkeit von A und R sind die Diagonaleinträge von R die Eigenwerte von A. Eine ähnliche Zerlegung ist die aus der Linearen Algebra bekannte *Jordansche Normalenform*. Diese verwendet aber keine unitäre Ähnlichkeit und ist deshalb für numerische Zwecke schwierig zu behandeln.

Ein reelles Analogon ($\mathbb{C}$ wird durch $\mathbb{R}$ ersetzt und unitär durch orthogonal) zu letztem Satz kann nicht existieren, weil reelle Matrizen auch komplexe Eigenwerte besitzen können. Diese treten jedoch in komplex-konjugierten Paaren auf, weil mit $A \in \mathbb{R}^{n\times n}$ aus $Ax = \lambda x$ folgt $A\overline{x} = \overline{Ax} = \overline{\lambda x} = \overline{\lambda}\overline{x}$. Daher kann man folgendes Resultat zeigen.

Satz 3.3. *Sei $A \in \mathbb{R}^{n\times n}$. Dann existiert ein $Q \in \mathbb{R}^{n\times n}$ orthogonal, so dass Q^TAQ eine reelle obere Blockdreiecksmatrix mit entweder 1×1 oder 2×2-Blöcken auf der Blockdiagonalen ist. Sind die Eigenwerte ausschließlich reell, ist Q^TAQ eine obere Dreiecksmatrix.*

Im Folgenden untersuchen wir Matrizen, für die R in der Schur-Zerlegung als diagonal angenommen werden darf.

Definition 3.4. *Eine Matrix $A \in \mathbb{C}^{n\times n}$ heißt* ***diagonalisierbar****, falls sie zu einer Diagonalmatrix D ähnlich ist. Ist D zusätzlich reell, so wird A als* ***reell diagonalisierbar*** *bezeichnet. Man nennt A* ***unitär diagonalisierbar****, falls A mit einer unitären Ähnlichkeitstransformation diagonalisierbar ist.*

Satz 3.5. *Eine Matrix $A \in \mathbb{C}^{n\times n}$ ist genau dann* ***normal****, d.h. es gilt $AA^H = A^HA$, wenn A unitär diagonalisierbar ist.*

Für den Beweis von Satz 3.5 benötigen wir die beiden folgenden Lemmata.

Lemma 3.6. *Jede normale Dreiecksmatrix ist diagonal.*

Beweis. Sei A eine obere Dreiecksmatrix und normal. Der Vergleich der Diagonalelemente von A^HA und AA^H ergibt

$$|a_{11}|^2 = (A^HA)_{11} = (AA^H)_{11} = \sum_{j=1}^{n} |a_{1j}|^2.$$

Daher verschwindet die erste Zeile von A bis auf das Diagonalelement. Das gleiche Argument wende man nun sukzessive auf die übrigen Zeilen an. □

Lemma 3.7. *Sei $U \in \mathbb{C}^{n\times n}$ unitär. Die Matrix $A \in \mathbb{C}^{n\times n}$ ist genau dann normal, wenn U^HAU normal ist.*

Beweis. Es gelten

$$(U^HAU)^H(U^HAU) = U^HA^HUU^HAU = U^HA^HAU \text{ und } U^HAA^HU = (U^HAU)(U^HAU)^H.$$

□

Beweis von Satz 3.5. Ist A unitär diagonalsierbar, so existiert U unitär und D diagonal mit $A = UDU^H$. Dann gilt

$$A^HA = UD^HU^HUDU^H = UD^HDU^H = UDD^HU^H = UDU^HUD^HU^H = AA^H.$$

Ist umgekehrt A normal, so gilt nach der komplexen Schur-Zerlegung $U^HAU = R$ mit einer oberen Dreiecksmatrix R und einer unitären Matrix U. R ist nach Lemma 3.7 normal und nach Lemma 3.6 diagonal. □

Man beachte, dass die Eigenvektoren und die Eigenwerte in Satz 3.5 im Allgemeinen komplex sind, selbst wenn $A \in \mathbb{R}^{n\times n}$ gilt.

Beispiel 3.8. Die reelle Matrix

$$A = \begin{bmatrix} 0 & 1 \\ -1 & 0 \end{bmatrix}$$

ist normal und besitzt die **Spektralzerlegung** $A = UDU^H$ mit

$$U = \frac{1}{\sqrt{2}} \begin{bmatrix} 1 & 1 \\ \mathrm{i} & -\mathrm{i} \end{bmatrix} \quad \text{und} \quad D = \begin{bmatrix} \mathrm{i} & \\ & -\mathrm{i} \end{bmatrix}.$$

Hermitesche Matrizen sind offenbar normal. Die einschränkende Bedingung ist im folgenden Satz formuliert.

Satz 3.9. *Eine Matrix $A \in \mathbb{C}^{n\times n}$ ist genau dann normal und hat reelle Eigenwerte, wenn A hermitesch ist.*

Beweis. Sei A hermitesch. Dann ist A normal und nach Satz 3.5 gilt $A = UDU^H$ mit unitärem U und diagonalem D. Aus $A^H = A$ folgt $D^H = D$. Also ist D reell. Die Umkehrung ergibt sich analog. □

Das folgende Resultat aus der Linearen Algebra haben wir schon mehrfach benutzt.

Satz 3.10 (Spektralsatz für hermitesche Matrizen). *Ist $A \in \mathbb{K}^{n\times n}$ hermitesch, so existieren $D \in \mathbb{R}^{n\times n}$ diagonal und $U \in \mathbb{K}^{n\times n}$ unitär, so dass $A = UDU^H$.*

Beweis. Sei zunächst $A \in \mathbb{C}^{n\times n}$. Aus Satz 3.9 folgt, dass genau die hermiteschen Matrizen unitär reell diagonalisierbar sind.

Ist $A \in \mathbb{R}^{n\times n}$ symmetrisch, so sind wie eben gesehen alle Eigenwerte reell. Nach Satz 3.3 gilt $A = QRQ^T$ mit einer orthogonalen Matrix $Q \in \mathbb{R}^{n\times n}$ und einer oberen Dreiecksmatrix R. Weil A symmetrisch ist, gilt $R^T = R$. Also ist R eine Diagonalmatrix. □

Variationelle Charakterisierung von Eigenwerten

Definition 3.11. *Sei $A \in \mathbb{C}^{n\times n}$ und $x \in \mathbb{C}^n \setminus \{0\}$. Bei gegebenem Skalarprodukt $(\cdot,\cdot)$ wird der Ausdruck*

$$\mu_A(x) := \frac{(Ax, x)}{(x, x)}$$

*als **Rayleigh-Quotient** bezeichnet. Die Menge aller Rayleigh-Quotienten*

$$W(A) := \{\mu_A(x),\, x \in \mathbb{C}^n \setminus \{0\}\}$$

*heißt **Wertebereich** von A.*

Im Folgenden betrachten wir den Fall $(x, y) := y^H x$ des euklidischen Skalarproduktes.

Bemerkung.

(a) Sei A hermitesch. Dann gilt wegen $x^H Ax = (x^H A^H x)^T = \overline{x^H Ax}$, $x \in \mathbb{C}^n$, für den Wertebereich $W(A) \subset \mathbb{R}$. In den Übungsaufgaben zeigen wir, dass aus $x^H Ax \in \mathbb{R}$ für alle $x \in \mathbb{C}^n$ folgt, dass A hermitesch ist.

(b) Sei $A \in \mathbb{K}^{n\times n}$ hermitesch. Definiert man $f : \mathbb{R} \to \mathbb{R}$ bei festem $x \in \mathbb{K}^n \setminus \{0\}$ durch

$$f(t) = \frac{1}{2}\|Ax - tx\|^2 = \frac{t^2}{2}\|x\|^2 - t\,(Ax, x) + \frac{1}{2}\|Ax\|^2,$$

so nimmt f wegen $f'(t) = t\|x\|^2 - (Ax, x)$ im Wert des Rayleigh-Quotienten $t = \mu_A(x)$ sein Minimum an. Ist daher x eine Näherung an einen Eigenvektor, so kann man erwarten, dass $\mu_A(x)$ eine gute Approximation des zugehörigen Eigenwerts ist.

(c) Nach einem Resultat von Hausdorff ist $W(A)$ eine konvexe Menge. Ferner gilt für jedes Eigenpaar (λ, x) von A

$$\mu_A(x) = \frac{(Ax, x)}{(x, x)} = \frac{\lambda(x, x)}{(x, x)} = \lambda.$$

Daher ist $\sigma(A) \subset W(A)$ und somit ist auch die konvexe Hülle $\operatorname{conv}\sigma(A)$ mit

$$\operatorname{conv}(x_1, \ldots, x_k) := \left\{ \sum_{i=1}^{k} \alpha_i x_i,\ 0 \le \alpha_i \le 1,\ \sum_{i=1}^{k} \alpha_i = 1 \right\}$$

in $W(A)$ enthalten.

Satz 3.12. *Für normale Matrizen $A \in \mathbb{C}^{n\times n}$ gilt* $\operatorname{conv}\sigma(A) = W(A)$.

Beweis. Weil A normal ist, existiert nach Satz 3.5 eine Orthonormalbasis $\{u_1, \ldots, u_n\} \subset \mathbb{C}^n$ von Eigenvektoren von A. Für $x \in \mathbb{C}^n$ in der Darstellung

$$x = \sum_{i=1}^{n} \alpha_i u_i$$

gilt dann

$$\mu_A(x) = \frac{(Ax, x)}{(x, x)} = \frac{\sum_{i,j=1}^{n} \overline{\alpha}_i \alpha_j (Au_j, u_i)}{\sum_{i=1}^{n} |\alpha_i|^2} = \frac{\sum_{i,j=1}^{n} \overline{\alpha}_i \alpha_j \lambda_j \delta_{ij}}{\sum_{i=1}^{n} |\alpha_i|^2} = \frac{\sum_{i=1}^{n} \lambda_i |\alpha_i|^2}{\sum_{i=1}^{n} |\alpha_i|^2} = \sum_{i=1}^{n} \beta_i \lambda_i,$$

wobei

$$0 \le \beta_i := \frac{|\alpha_i|^2}{\sum_{i=1}^{n} |\alpha_i|^2} \le 1 \quad \text{und} \quad \sum_{i=1}^{n} \beta_i = \frac{\sum_{i=1}^{n} |\alpha_i|^2}{\sum_{i=1}^{n} |\alpha_i|^2} = 1.$$

□

Im Folgenden sei $A \in \mathbb{K}^{n\times n}$ hermitesch. Die der Größe nach geordneten reellen Eigenwerte von A bezeichnen wir mit

$$\lambda_1 \ge \ldots \ge \lambda_n. \tag{3.2}$$

Diese werden nun durch Optimierungseigenschaften charakterisiert.

Satz 3.13 (Rayleigh). *Sei $A \in \mathbb{K}^{n\times n}$ hermitesch und $\{u_1, \ldots, u_n\} \subset \mathbb{K}^n$ eine zugehörige Orthonormalbasis aus Eigenvektoren zu den Eigenwerten aus (3.2). Mit den linearen Unterräumen $E_0 := \{0\}$ und $E_j := \operatorname{span}\{u_1, \ldots, u_j\}$, $j = 1, \ldots, n$, gilt*

$$\lambda_j = \max_{x \in E_{j-1}^{\perp} \setminus \{0\}} \mu_A(x) = \min_{x \in E_j \setminus \{0\}} \mu_A(x), \quad j = 1, \ldots, n.$$

Beweis. Seien $j \in \{1, \ldots, n\}$ und $x \in E_{j-1}^{\perp} \setminus \{0\}$ beliebig mit

$$x = \sum_{\ell=1}^{n} \alpha_\ell u_\ell.$$

Dann ist

$$\alpha_k = (x, u_k) = 0 \quad \text{für } k = 1, \dots, j-1.$$

Wegen $\lambda_k \leq \lambda_j$ für $k = j, \dots, n$ ist

$$\mu_A(x) = \frac{(Ax, x)}{(x, x)} = \frac{\sum_{k,\ell=j}^{n} \lambda_k \alpha_k \overline{\alpha_\ell}(u_k, u_\ell)}{\sum_{k=j}^{n} |\alpha_k|^2} = \frac{\sum_{k=j}^{n} \lambda_k |\alpha_k|^2}{\sum_{k=j}^{n} |\alpha_k|^2} \leq \lambda_j$$

und somit

$$\sup_{x \in E_{j-1}^{\perp} \setminus \{0\}} \frac{(Ax, x)}{(x, x)} \leq \lambda_j.$$

Andererseits erhält man für $u_j \in E_{j-1}^{\perp} \setminus \{0\}$

$$\mu_A(u_j) = \frac{(Au_j, u_j)}{(u_j, u_j)} = \frac{\lambda_j (u_j, u_j)}{(u_j, u_j)} = \lambda_j.$$

Daher wird das Maximum angenommen. Die Charakterisierung über das Minimum beweist man analog. □

Bemerkung. Aus Satz 3.13 folgt insbesondere für hermitesche $A \in \mathbb{K}^{n \times n}$

$$\lambda_1 = \max_{x \in \mathbb{K}^n \setminus \{0\}} \frac{(Ax, x)}{(x, x)} \quad \text{und} \quad \lambda_n = \min_{x \in \mathbb{K}^n \setminus \{0\}} \frac{(Ax, x)}{(x, x)}.$$

Also gilt

$$\lambda_n \|x\|^2 \leq (Ax, x) \leq \lambda_1 \|x\|^2 \quad \text{für alle } x \in \mathbb{K}^n \setminus \{0\}.$$

Satz 3.14 (Courant-Fischer). *Sei $A \in \mathbb{K}^{n \times n}$ hermitesch. Für $j = 1, \dots, n$ sei*

$$\mathcal{U}_j := \{U \subset \mathbb{K}^n : U \text{ ist Unterraum der Dimension } j\}.$$

Dann gilt für die Eigenwerten aus (3.2)

$$\lambda_j = \min_{U \in \mathcal{U}_{n+1-j}} \max_{x \in U \setminus \{0\}} \mu_A(x) = \max_{U \in \mathcal{U}_j} \min_{x \in U \setminus \{0\}} \mu_A(x), \quad j = 1, \dots, n.$$

Beweis. Sei $\{u_1, \dots, u_n\}$ eine Orthonormalbasis aus Eigenvektoren zu den Eigenwerten (3.2). Für $j \in \{1, \dots, n\}$ definiere $E_j = \operatorname{span}\{u_1, \dots, u_j\}$ und wähle $U \in \mathcal{U}_{n+1-j}$. Wegen

$$\dim(E_j \cap U) = \dim E_j + \dim U - \dim(E_j + U) \geq j + n + 1 - j - n = 1$$

existiert ein $0 \neq x \in E_j \cap U$. Als Element von E_j lässt sich x eindeutig in der Form

$$x = \sum_{i=1}^{j} \alpha_i u_i$$

darstellen. Dann ist

$$\mu_A(x) = \frac{(Ax, x)}{(x, x)} = \frac{\sum_{i=1}^{j} \lambda_i |\alpha_i|^2}{\sum_{i=1}^{j} |\alpha_i|^2} \geq \lambda_j$$

und daher

$$\min_{U \in \mathcal{U}_{n+1-j}} \max_{x \in U \setminus \{0\}} \mu_A(x) \geq \lambda_j.$$

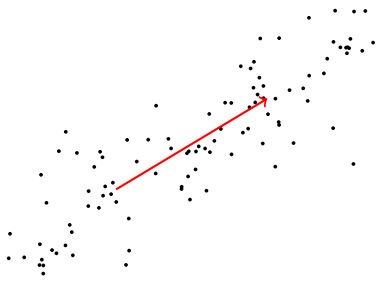
Wählt man andererseits den $(n+1-j)$-dimensionalen Unterraum $U = E_{j-1}^{\perp}$, so ist nach Satz 3.13

$$\max_{x \in E_{j-1}^{\perp} \setminus \{0\}} \mu_A(x) = \lambda_j.$$

und das Minimum wird angenommen. Die andere Charakterisierung beweist man analog. □

Bemerkung. Wir haben den Satz von Courant-Fischer für den Spezialfall des euklidischen Skalarprodukts und hermitescher Matrix bewiesen. Er gilt aber auch für allgemeine Skalarprodukte $(\cdot,\cdot)$, falls A bzgl. $(\cdot,\cdot)$ selbstadjungiert ist.

Beispiel 3.15 (Principal component analysis (PCA)). Seien $p_i \in \mathbb{R}^d$, $i = 1, \ldots, n$, und $m := \frac{1}{n}\sum_{i=1}^n p_i$ ihr Schwerpunkt. Wir suchen die **Hauptrichtung** der Vektoren p_i,



d.h. einen Vektor $v \in \mathbb{R}^d$, $\|v\|_2 = 1$, in dessen Richtung die Varianz

$$\sum_{i=1}^n |v^T(p_i - m)|^2 = \max_{\|x\|_2=1} \sum_{i=1}^n |x^T(p_i - m)|^2$$

am größten ist. Dieser Vektor v ist der Eigenvektor zum größten Eigenwert der sog. **Kovarianz-Matrix**

$$C := \sum_{i=1}^n (p_i - m)(p_i - m)^T \in \mathbb{R}^{d \times d}.$$

Es gilt nämlich für $x \in \mathbb{R}^d$

$$x^T C x = \sum_{i=1}^n |x^T(p_i - m)|^2,$$

und nach Satz 3.13 ist

$$\lambda_1(C) = \max_{x \in \mathbb{R}^d \setminus \{0\}} \frac{x^T C x}{x^T x} = \max_{\|x\|_2=1} \sum_{i=1}^n |x^T(p_i - m)|^2.$$

Die Hauptrichtung kann also durch Lösen eines im Vergleich zur Anzahl der Punkte n kleinen $d \times d$ Eigenwertproblems berechnet werden.

3.1.1 Störungsanalyse von Eigenwerten

Bei numerischen Verfahren ist der Einfluss von Rundungsfehlern ein generelles Problem. Es ist bekannt, dass die Nullstellen stetig von den Koeffizienten eines Polynoms abhängen. Daher hängen zwar die Eigenwerte stetig von den Matrixeinträgen ab, wir wollen aber im Folgenden die Kondition des Eigenwertproblems untersuchen. Dazu betrachten wir die sog. **Begleitmatrix**

$$A := \begin{bmatrix} 0 & & & & -a_0 \\ 1 & \ddots & & & \vdots \\ & \ddots & \ddots & & \vdots \\ & & \ddots & 0 & -a_{n-2} \\ & & & 1 & -a_{n-1} \end{bmatrix} \in \mathbb{C}^{n\times n}$$

zum Polynom $p(t) := a_0 + a_1 t + a_2 t^2 + \ldots + a_{n-1}t^{n-1} + t^n \in \Pi_n$. Durch Entwicklung nach der letzten Spalte von A sieht man

$$\det(A - tI) = (-1)^n p(t).$$

Die Nullstellen von p stimmen also mit den Eigenwerten der Begleitsmatrix überein.

Das Polynom $q_\varepsilon(t) := (t-a)^n + \varepsilon$ für $\varepsilon > 0$ und $a \neq 0$ besitzt die Nullstellen

$$\lambda_k = a - \varepsilon^{1/n} e^{2\pi i k/n}, \quad k = 0, \ldots, n-1.$$

Vergleicht man q_0 und q_ε bzw. deren Begleitmatrizen A_0 und A_ε, so gilt

$$\Delta A := A_\varepsilon - A_0 = \begin{bmatrix} 0 & \cdots & 0 & \varepsilon \\ \vdots & & \vdots & 0 \\ \vdots & & \vdots & \vdots \\ 0 & \cdots & 0 & 0 \end{bmatrix}.$$

Die Eigenwerte von A_0 und A_ε unterscheiden sich aber um $|\Delta\lambda| = \varepsilon^{1/n}$. Daher gilt

$$\frac{|\Delta\lambda|}{|\lambda|} = \frac{\varepsilon^{1/n}}{|a|} = \underbrace{\frac{\|A\|\varepsilon^{1/n}}{|a|\varepsilon}}_{\to\infty \text{ für } \varepsilon\to 0} \frac{\|\Delta A\|}{\|A\|}.$$

Die Kondition des Eigenwertproblems kann also ohne weitere Vorraussetzungen an die Matrix beliebig groß werden. Der folgende Satz zeigt zumindest die stetige Abhängigkeit der Eigenwerte von den Einträgen.

Satz 3.16 (Bauer-Fike). *Sei $A \in \mathbb{C}^{n\times n}$ eine diagonalisierbare Matrix mit $A = T^{-1}DT$, $D = \operatorname{diag}(\lambda_1, \ldots, \lambda_n)$. Ferner sei $A + E \in \mathbb{C}^{n\times n}$ eine Störung von A und λ ein Eigenwert von $A + E$. Dann gilt bzgl. der 1, 2 und ∞-Norm*

$$\min_{j=1,\ldots,n} |\lambda - \lambda_j| \leq \|TET^{-1}\| \leq \|E\| \operatorname{cond}_{\|\cdot\|}(T).$$

Beweis. Ist λ ein Eigenwert von A, so ist die Behauptung trivial. Sei $\lambda \neq \lambda_j$, $j = 1, \ldots, n$, und $x \neq 0$ ein zugehöriger Eigenvektor von $A + E$. Dann gilt

$$(A + E)x = \lambda x \quad \Rightarrow \quad Ex = (\lambda I - A)x = T^{-1}(\lambda I - D)Tx.$$

Hieraus folgt $Tx = (\lambda I - D)^{-1}(TET^{-1})Tx$ und somit

$$\|Tx\| \leq \|(\lambda I - D)^{-1}\|\|TET^{-1}\|\|Tx\| \leq \max_{j=1,\ldots,n} |\lambda - \lambda_j|^{-1}\|TET^{-1}\|\|Tx\|.$$

Division durch $\|Tx\| \neq 0$ liefert die Behauptung. □

Nach dem Satz von Bauer-Fike bestimmt die Kondition der Matrix der Eigenvektoren die Störanfälligkeit der Eigenwerte. Bei normalen Matrizen ist T nach Satz 3.10 unitär. In diesem Fall gilt $\text{cond}_{\|\cdot\|_2}(T) = 1$. Das symmetrische Eigenwertproblem ist also gut konditioniert, das unsymmetrische in der Regel schlecht.

Bei hermiteschen Matrizen kann man aus einer Näherung für ein Eigenpaar eine A-posteriori-Fehlerabschätzung gewinnen.

Satz 3.17. *Sei $A \in \mathbb{C}^{n\times n}$ hermitesch. Seien $\lambda \in \mathbb{R}$ und $x \in \mathbb{C}^n \setminus \{0\}$ Näherungen an ein Eigenpaar. Dann gilt*

$$\min_{j=1,\ldots,n} |\lambda - \lambda_j| \leq \frac{\|Ax - \lambda x\|_2}{\|x\|_2}.$$

Beweis. Für $B := A + \|x\|_2^{-2}(\lambda x - Ax)x^H$ gilt $Bx = \lambda x$. Also besitzt B den Eigenwert λ und $\|A-B\|_2^2 = \rho((A-B)^H(A-B)) = \|Ax-\lambda x\|_2^2/\|x\|_2^2$. Die Behauptung folgt aus Satz 3.16. □

Im Fall hermitescher Matrizen kann man eine stärkere Aussage als Satz 3.16 zeigen.

Satz 3.18 (Weyl). *Seien $A, B \in \mathbb{C}^{n\times n}$ hermitesch mit Eigenwerten $\lambda_j(A)$ und $\lambda_j(B)$, $j = 1, \ldots, n$, die jeweils (3.2) genügen. Dann gilt*

$$|\lambda_j(A) - \lambda_j(B)| \leq \|A - B\|_2, \quad j = 1, \ldots, n.$$

Beweis. Mit A, B ist auch $A - B$ hermitesch. Für $x \in \mathbb{C}^n \setminus \{0\}$ ist daher nach Satz 3.13

$$\frac{((A - B)x, x)}{(x, x)} \leq \rho(A - B) = \|A - B\|_2.$$

Also gilt

$$\frac{(Ax, x)}{(x, x)} \leq \frac{(Bx, x)}{(x, x)} + \|A - B\|_2.$$

Sei $j \in \{1, \ldots, n\}$ und $\mathcal{U}_j = \{U \subset \mathbb{K}^n : U \text{ ist Unterraum der Dimension } j\}$. Dann gilt

$$\max_{U\in\mathcal{U}_j} \min_{x\in U\setminus\{0\}} \frac{(Ax, x)}{(x, x)} \leq \max_{U\in\mathcal{U}_j} \min_{x\in U\setminus\{0\}} \frac{(Bx, x)}{(x, x)} + \|A - B\|_2,$$

und nach Satz 3.14 folgt

$$\lambda_j(A) \leq \lambda_j(B) + \|A - B\|_2, \quad j = 1, \ldots, n.$$

Durch Vertauschen der Rollen von A und B folgt die Behauptung. □

Weil die Singulärwerte einer Matrix $A \in \mathbb{C}^{m\times n}$ die Eigenwerte der hermiteschen Matrix

$$A' = \begin{bmatrix} 0 & A \\ A^H & 0 \end{bmatrix}$$

sind (vgl. die Bemerkung nach Satz 1.25), erhalten wir insbesondere

Korollar 3.19. *Sind $A, B \in \mathbb{C}^{m\times n}$, so gilt für die absteigend geordneten Singulärwerte von A und B*

$$|\sigma_j(A) - \sigma_j(B)| \le \|A - B\|_2, \quad j = 1, \dots, \min\{m, n\}.$$

Beweis. Mit Satz 1.25 (ii) gilt

$$|\sigma_j(A) - \sigma_j(B)| = |\lambda_{2j}(A') - \lambda_{2j}(B')| \le \rho(A' - B') = \sigma_1(A - B) = \|A - B\|_2.$$

□

3.1.2 Lokalisierung von Eigenwerten

Da wir ausschließlich iterative Verfahren zur Lösung des Eigenwertproblems verwenden werden, benötigen wir eine gute Approximation des Eigenwerts als Startwert. Hierfür sind die folgenden Abschätzungen hilfreich, die die Lage der Eigenwerte auf Basis der Matrixeinträge einschränken.

Durch Anwendung des Beweises von Satz 3.18 auf $B := \operatorname{diag}(A)$ folgt sofort

Korollar 3.20. *Sei $A \in \mathbb{C}^{n\times n}$ hermitesch und $\{a'_{11}, \dots, a'_{nn}\}$ eine Permutation der reellen Diagonalelemente von A mit $a'_{11} \ge \dots \ge a'_{nn}$. Für die Eigenwerte $\lambda_1 \ge \dots \ge \lambda_n$ von A gelten die Abschätzungen*

$$|\lambda_i - a'_{ii}| \le \max_{j=1,\dots,n} \sum_{\substack{k=1 \\ k\neq j}}^{n} |a_{jk}| \quad \text{und} \quad |\lambda_i - a'_{ii}| \le \left(\sum_{\substack{j,k=1 \\ k\neq j}}^{n} |a_{jk}|^2 \right)^{1/2}$$

für $i = 1, \dots, n$.

Beweis. Die Behauptung folgt aus (1.11). □

Die Radien der folgenden Kreise sind kleiner. Allerdings ist nicht klar, in welchem Kreis ein Eigenwert liegt.

Satz 3.21 (Gerschgorin). *Sei $A \in \mathbb{C}^{n\times n}$. Für $i = 1, \dots, n$ definiere die sog.* ***Gerschgorin-Kreise***

$$G_i := \{z \in \mathbb{C} : |z - a_{ii}| \le r_i\}, \quad r_i := \sum_{\substack{j=1 \\ j\neq i}}^{n} |a_{ij}|.$$

Dann gilt $\sigma(A) \subset \bigcup_{i=1}^{n} G_i$. Hat die Vereinigung $\hat{G}_m$ von $m < n$ Kreisen einen leeren Durchschnitt mit den restlichen $n - m$ Kreisen, so enthält $\hat{G}_m$ genau m Eigenwerte (jeder entsprechend seiner algebraischen Vielfachheit gezählt) von A.

Beweis. Sei (λ, x) ein Eigenpaar von A. Dann gilt

$$(\lambda - a_{ii})x_i = \sum_{\substack{j=1 \\ j\neq i}}^{n} a_{ij}x_j, \quad i = 1, \ldots, n.$$

Für $i \in \{1, \ldots, n\}$ mit $|x_i| = \max_{j=1,\ldots,n} |x_j| \neq 0$ folgt hieraus

$$|\lambda - a_{ii}| \leq \left| \sum_{\substack{j=1 \\ j\neq i}}^{n} \frac{a_{ij}x_j}{x_i} \right| \leq \sum_{\substack{j=1 \\ j\neq i}}^{n} |a_{ij}| = r_i.$$

Also gilt $\lambda \in G_i \subset \bigcup_{j=1}^{n} G_j$. Für den zweiten Teil der Behauptung setzen wir $D = \operatorname{diag}(A)$ und betrachten

$$B(t) := D + t(A - D), \quad 0 \leq t \leq 1,$$

deren Gerschgorin-Kreise durch

$$G_i(t) = \{z \in \mathbb{C} : |z - a_{ii}| \leq t r_i\}, \quad i = 1, \ldots, n,$$

gegeben sind. Die Eigenwerte von $B(t)$ hängen stetig von t ab. Die Eigenwerte von $B(0) = D$ sind genau die Mittelpunkte der Kreise $G_i(t)$, $i = 1, \ldots, n$. Wir wenden den ersten Teil auf $B(t)$ an und lassen t von 0 nach 1 laufen. Dabei blähen sich die Gerschgorin-Kreise $G_i(t)$ bei festem Mittelpunkt immer mehr auf. Die Anzahl der Eigenwerte in einem Kreis $G_i(t)$ kann sich dabei erst ändern, wenn dieser einen anderen Kreis trifft. □

Da die Eigenwerte von A und A^H übereinstimmen, gilt Satz 3.21 auch mit

$$G'_j = \left\{ z \in \mathbb{C} : |z - a_{jj}| \leq \sum_{\substack{i=1 \\ i\neq j}}^{n} |a_{ij}| \right\}, \quad j = 1, \ldots, n,$$

anstelle von G_i, $i = 1, \ldots, n$. Folglich haben wir auch

$$\sigma(A) \subset \left(\bigcup_{i=1}^{n} G_i \right) \cap \left(\bigcup_{j=1}^{n} G'_j \right).$$

Eine weitere Aussage zur Lage der Eigenwerte basiert auf der Zerlegung

$$A = A_0 + \mathrm{i}A_1$$

mit den hermiteschen Matrizen

$$A_0 := \frac{1}{2}(A + A^H) \quad \text{und} \quad A_1 := \frac{1}{2\mathrm{i}}(A - A^H).$$

Nach Satz 3.12 gilt

$$W(A_i) = [\lambda_n(A_i), \lambda_1(A_i)] \subset \mathbb{R}, \quad i = 0, 1. \tag{3.3}$$

Satz 3.22 (Bendixson). *Sei $A \in \mathbb{C}^{n\times n}$. Dann liegt das Spektrum von A im Rechteck*

$$\sigma(A) \subset W(A_0) + \mathrm{i}\, W(A_1).$$

Beweis. Für $x \in \mathbb{C}^n \setminus \{0\}$ gilt

$$\frac{(Ax,x)}{(x,x)} = \frac{([A_0 + \mathrm{i}A_1]x,x)}{(x,x)} = \frac{(A_0x,x)}{(x,x)} + \mathrm{i}\frac{(A_1x,x)}{(x,x)} \in W(A_0) + \mathrm{i}\,W(A_1).$$

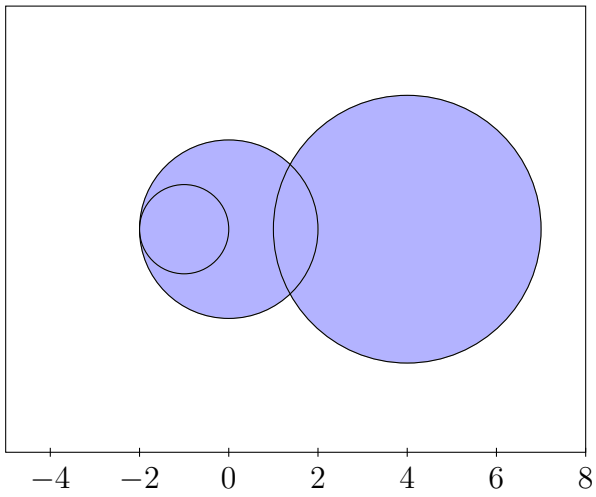
Also gilt $\sigma(A) \subset W(A) \subset W(A_0) + \mathrm{i}\,W(A_1)$. □

Beispiel 3.23. Betrachte

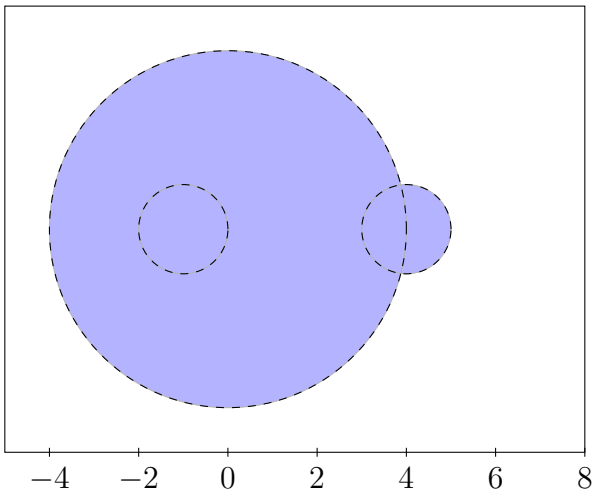
$$A = \begin{bmatrix} 4 & 0 & 3 \\ 0 & -1 & 1 \\ 1 & 1 & 0 \end{bmatrix}.$$

Nach dem Satz von Gerschgorin ergeben sich folgende Einschließungen.

Gerschgorin-Kreise für A



Gerschgorin-Kreise für A^H



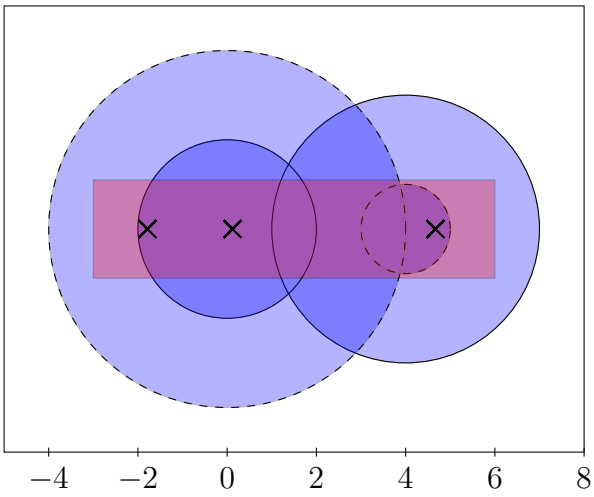
Für den Satz von Bendixson berechnen wir

$$A_0 = \frac{1}{2}(A + A^H) = \begin{bmatrix} 4 & 0 & 2 \\ 0 & -1 & 1 \\ 2 & 1 & 0 \end{bmatrix}, \quad A_1 = \frac{1}{2\mathrm{i}}(A - A^H) = \begin{bmatrix} 0 & 0 & -\mathrm{i} \\ 0 & 0 & 0 \\ \mathrm{i} & 0 & 0 \end{bmatrix}.$$

Um $W(A_0)$ und $W(A_1)$ abzuschätzen, schließen wir wegen (3.3) die Spektren von A_0 und A_1 wieder mit dem Satz von Gerschgorin ein. Dies führt auf das Rechteck

$$[-3,6] + \mathrm{i}\,[-1,1].$$

Einschließungen und tatsächliche Eigenwerte



Das Spektrum von A ist tatsächlich

$$\sigma(A) = \{-1.7878, 0.1198, 4.6679\}.$$

Wir zeigen noch eine weitere Lokalisationseigenschaft, die sog. **Verschachtelung** (engl. interlacing) auf.

Satz 3.24. *Sei $A \in \mathbb{K}^{n\times n}$ hermitesch und $P \in \mathbb{K}^{n\times m}$, $m \le n$, mit $P^H P = I \in \mathbb{R}^{m\times m}$. Dann gilt für die jeweils absteigend geordneten Eigenwerte von A und $B := P^H AP \in \mathbb{K}^{m\times m}$*

$$\lambda_j(A) \ge \lambda_j(B) \ge \lambda_{j+n-m}(A), \quad j = 1, \dots, m.$$

Beweis. Sei u_j ein Eigenvektor von B zu $\lambda_j(B)$ und $E_j := \operatorname{span}\{u_1, \dots, u_j\} \subset \mathbb{K}^m$. Nach Satz 3.13 gilt mit $U_j := \{Px,\, x \in E_{j-1}^{\perp}\} \subset \mathbb{K}^n$

$$\lambda_j(B) = \max_{x\in E_{j-1}^{\perp}\setminus\{0\}} \frac{(Bx,x)}{(x,x)} = \max_{x\in E_{j-1}^{\perp}\setminus\{0\}} \frac{(APx,Px)}{(Px,Px)} = \max_{y\in U_j\setminus\{0\}} \frac{y^H Ay}{y^H y}.$$

Wegen $\dim U_j = \dim E_{j-1}^{\perp} = m - j + 1$ folgt nach Satz 3.14

$$\max_{y\in U_j\setminus\{0\}} \frac{(Ay,y)}{(y,y)} \ge \min_{V\subset\mathbb{K}^n:\dim V = m-j+1} \max_{y\in V\setminus\{0\}} \frac{(Ay,y)}{(y,y)} = \lambda_{j+n-m}(A).$$

Also gilt

$$\lambda_{j+n-m}(A) \le \lambda_j(B), \quad j = 1, \dots, m. \tag{3.4}$$

Für die andere Abschätzung wenden wir (3.4) auf $-A$ statt auf A an. Dann gilt wegen $\lambda_j(A) = -\lambda_{n+1-j}(-A)$ und $\lambda_j(B) = -\lambda_{m+1-j}(-B)$

$$\lambda_j(B) = -\lambda_{m+1-j}(-B) \overset{(3.4)}{\le} -\lambda_{n+1-j}(-A) = \lambda_j(A).$$

□

Bemerkung. Aus Satz 3.24 ergibt sich insbesondere für die Wahl

$$P = [e_1, \dots, e_{n-1}] \in \mathbb{R}^{n\times(n-1)}$$

und die Matrix

$$\hat{A} = \begin{bmatrix} A & b \\ b^T & c \end{bmatrix} \in \mathbb{K}^{n\times n} \quad \text{mit } A \in \mathbb{K}^{(n-1)\times(n-1)} \text{ hermitesch,}$$

dass

$$\lambda_j(\hat{A}) \ge \lambda_j(A) \ge \lambda_{j+1}(\hat{A}), \quad j = 1, \dots, n-1. \tag{3.5}$$

Sind $b, c = 0$, so besitzt $\hat{A}$ nach dem Satz von Rayleigh sogar dieselben Eigenwerte wie A ergänzt um 0.

Korollar 3.25. *Seien $A \in \mathbb{C}^{n\times n}$ hermitesch, $x \in \mathbb{C}^n \setminus \{0\}$ und $B := A + \alpha xx^H$, $\alpha \neq 0$. Dann gilt für die absteigend geordneten Eigenwerte von A und B*

$$\lambda_1(B) \geq \lambda_1(A) \geq \lambda_2(B) \geq \ldots \geq \lambda_{n-1}(A) \geq \lambda_n(B) \geq \lambda_n(A), \quad \text{falls } \alpha > 0,$$

und

$$\lambda_1(A) \geq \lambda_1(B) \geq \lambda_2(A) \geq \ldots \geq \lambda_{n-1}(B) \geq \lambda_n(A) \geq \lambda_n(B), \quad \text{falls } \alpha < 0.$$

Beweis. Sei $\alpha > 0$. Da $B - A = \alpha xx^H$ positiv-semidefinit ist, folgt aus dem Satz von Rayleigh, dass $\lambda_j(B) \geq \lambda_j(A)$, $j = 1, \ldots, n$. Durch $\{v_1, \ldots, v_{n-1}\}$ sei eine Orthonormalbasis von $(\operatorname{span}\{x\})^\perp$ gegeben. Definiere

$$P = [v_1, \ldots, v_{n-1}] \in \mathbb{C}^{n\times(n-1)}.$$

Dann ist $P^H P = I \in \mathbb{R}^{(n-1)\times(n-1)}$ und

$$C := P^H BP = P^H(A + \alpha xx^H)P = P^H AP + \alpha\|P^H x\|_2^2 = P^H AP.$$

Nach Satz 3.24 gilt

$$\lambda_j(A) \geq \lambda_j(C) \geq \lambda_{j+1}(A) \quad \text{und} \quad \lambda_j(B) \geq \lambda_j(C) \geq \lambda_{j+1}(B), \quad j = 1, \ldots, n-1.$$

Für $j = 2, \ldots, n$ ist somit

$$\lambda_j(A) \leq \lambda_j(B) \leq \lambda_{j-1}(C) \leq \lambda_{j-1}(A) \leq \lambda_{j-1}(B).$$

Der Beweis der Aussage für $\alpha < 0$ verläuft analog. □

Wegen der Verwandtschaft der Singulärwerte mit den Eigenwerten erhält man folgendes Resultat.

Korollar 3.26. *Sei $\hat{A} = [A, a]$ mit $A \in \mathbb{C}^{m\times(n-1)}$ und $a \in \mathbb{C}^m$. Dann gilt für die Singulärwerte $\sigma_j(A)$, $\sigma_j(\hat{A})$ von A bzw. $\hat{A}$*

$$\sigma_1(\hat{A}) \geq \sigma_1(A) \geq \sigma_2(\hat{A}) \geq \ldots \geq \sigma_{n-1}(A) \geq \sigma_n(\hat{A}), \quad \text{falls } m \geq n,$$

und

$$\sigma_1(\hat{A}) \geq \sigma_1(A) \geq \sigma_2(\hat{A}) \geq \ldots \geq \sigma_{m-1}(A) \geq \sigma_m(\hat{A}) \geq \sigma_m(A), \quad \text{falls } m < n.$$

Ein analoges Resultat bei Hinzunahme einer Zeile anstelle einer Spalte erhält man durch Transposition.

Beweis. Zunächst beobachtet man, dass

$$\hat{A}^H\hat{A} = \begin{bmatrix} A^HA & A^Ha \\ a^HA & a^Ha \end{bmatrix}.$$

Da die Quadrate der Singulärwerte von A mit den Eigenwerten von A^HA übereinstimmen, folgt die Behauptung aus (3.5). □

3.2 Einfache Vektoriteration, inverse Iteration und Rayleigh-Quotienten-Verfahren

In diesem Abschnitt werden wir einfache Verfahren zur Lösung des Eigenwertproblems untersuchen. Dazu sei $A \in \mathbb{C}^{n\times n}$ diagonalisierbar, d.h. es existiert $V = [v_1, \ldots, v_n]$ mit $\|v_i\|_2 = 1$ und $\Lambda = \operatorname{diag}(\lambda_1, \ldots, \lambda_n)$, $|\lambda_1| \geq \ldots \geq |\lambda_n|$ mit

$$A = V\Lambda V^{-1}.$$

Einfache Vektoriteration

Vom Konzept her sehr einfach ist die **Vektoriteration nach von Mises** (engl. power method) zur Berechnung eines dominanten Eigenpaares, d.h. es gelte

$$|\lambda_1| > |\lambda_2| \geq \ldots \geq |\lambda_n|.$$

Gegeben sei $x_0 \in \mathbb{C}^n$, $\|x_0\|_2 = 1$. Es wird vorausgesetzt, dass der zu v_1 gehörende Koeffizient α_1 in der Darstellung

$$x_0 = \sum_{i=1}^{n} \alpha_i v_i \tag{3.6}$$

nicht verschwindet, d.h. $\alpha_1 \neq 0$. Wendet man die Matrix A wiederholt auf x_0 an, so dominiert wegen

$$A^k x_0 = \sum_{i=1}^{n} \alpha_i A^k v_i = \sum_{i=1}^{n} \alpha_i \lambda_i^k v_i \approx \alpha_1 \lambda_1^k v_1 \tag{3.7}$$

das erste Eigenpaar. Um die damit einhergehende sukzessive Vergrößerung von $A^k x_0$ zu vermeiden, normiert man die Approximation in jedem Schritt.

Algorithmus 3.27 (Vektoriteration nach von Mises).
`Input:` $A \in \mathbb{C}^{n\times n}$, Startvektor $x_0 \in \mathbb{C}^n$ mit $\alpha_1 \neq 0$ in (3.6) und Fehlertoleranz $\varepsilon > 0$.
`Output:` Approximation an den zum dominanten Eigenwert gehörenden Eigenvektor v_1.

$k = 0$;
`do` {
  $\tilde{x}_{k+1} = Ax_k$;
  sei σ_k das Vorzeichen von $\tilde{x}_{k+1}^H x_k$;
  $x_{k+1} = \sigma_k \frac{\tilde{x}_{k+1}}{\|\tilde{x}_{k+1}\|_2}$;
  $k = k + 1$;
} `while` $(\|x_k - x_{k-1}\|_2 > \varepsilon)$;

Satz 3.28. *Sei $\alpha_1 \neq 0$ in (3.6). Dann gilt mit $q := |\lambda_2|/|\lambda_1| < 1$*

(i) $\|\tilde{x}_k\|_2 = |\lambda_1| + O(q^k)$

(ii) $\|x_k - \beta_k v_1\|_2 = O(q^k)$ *mit* $|\beta_k| = 1$

für $k \to \infty$.

Beweis. Aus (3.7) folgt

$$A^k x_0 = \lambda_1^k \alpha_1 (v_1 + w_k), \quad w_k := \sum_{i=2}^{n} \left(\frac{\lambda_i}{\lambda_1}\right)^k \frac{\alpha_i}{\alpha_1} v_i$$

mit

$$\|w_k\|_2 \leq q^k \sum_{i=2}^{n} \left|\frac{\alpha_i}{\alpha_1}\right| =: c\, q^k.$$

Also ist

$$x_k = \frac{A^k x_0}{\|A^k x_0\|_2} = \underbrace{\frac{\lambda_1^k \alpha_1}{|\lambda_1^k||\alpha_1|}}_{=:\beta_k} \frac{v_1 + w_k}{\|v_1 + w_k\|_2}$$

und somit

$$\begin{aligned}\|x_k - \beta_k v_1\|_2 &= \left\| \beta_k \left(\frac{v_1 + w_k}{\|v_1 + w_k\|_2} - v_1 \right) \right\|_2 = \frac{1}{\|v_1 + w_k\|_2} \big\| v_1 + w_k - v_1 \|v_1 + w_k\|_2 \big\|_2 \\ &\leq \frac{1}{\|v_1 + w_k\|_2} \left(\big| 1 - \|v_1 + w_k\|_2 \big| \|v_1\|_2 + \|w_k\|_2 \right).\end{aligned}$$

Die Behauptung folgt aus

$$\big| \|v_1 + w_k\|_2 - 1 \big| = \big| \|v_1 + w_k\|_2 - \|v_1\|_2 \big| \leq \|w_k\|_2 \leq c\, q^k$$

und

$$\tilde{x}_{k+1} = A x_k = \lambda_1 \beta_k v_1 + O(q^k) \quad \text{für } k \to \infty.$$

□

Bemerkung.

(a) Die Voraussetzung $\alpha_1 \neq 0$ kann nicht überprüft werden. Im Laufe der Iteration wird jedoch durch Rundungsfehler in der Regel eine Komponente von x_k in Richtung v_1 entstehen.

(b) Ist $A \in \mathbb{R}^{n \times n}$ hermitesch, so können die Vektoren $x_k \in \mathbb{R}^n$ gewählt werden.

(c) Weil die Eigenvektoren nur bis auf Skalare bestimmt sind, ist die Konvergenz in Richtung von v_1 ausreichend.

(d) Sind die m-größten Eigenpaare zu berechnen, so lässt sich die einfache Vektoriteration leicht zur sog. **Orthogonalen Iteration** verallgemeinern, indem man A simultan auf m-Vektoren (in Form einer m-spaltigen Matrix Q_k) anwendet und diese nach jedem Schritt reorthonormalisiert, d.h. man berechnet eine QR-Zerlegung

$$Q_{k+1} R_{k+1} = A Q_k.$$

Die Diagonaleinträge von R_k konvergieren dann gegen die Eigenwerte (mehr dazu in Abschnitt 3.3). Ist A hermitesch, so konvergiert ferner Q_k gegen die den m-größten Eigenwerten zugeordneten Eigenvektoren.

Beispiel 3.29. Wir betrachten nochmals Beispiel 3.15. Zur Berechnung der Hauptrichtung einer Punktmenge ist die einfache Vektoriteration das Verfahren der Wahl, weil es dann besonders schnell konvergiert, wenn die Varianzen in den Richtungen stark differieren. Umgekehrt muss das Verfahren bei etwa gleichen Varianzen nicht schnell konvergieren, weil alle Richtungen gleichberechtigt sind.

Nach der Bemerkung zu Definition 3.11 ist $\mu_A(x_k)$ die beste Approximation an λ_1, falls A hermitesch ist. Der folgende Satz zeigt, dass $\mu_A(x_k)$ in diesem Fall eine deutlich bessere Approximation liefert als $\|\tilde{x}_k\|_2$, welches auch nur den Betrag von λ_1 approximiert.

Satz 3.30. *Sei A hermitesch. Dann gilt mit $q := |\lambda_2|/|\lambda_1| < 1$*

$$|\mu_A(x_k) - \lambda_1| = O(q^{2k}).$$

Beweis. in den Übungsaufgaben. □

Inverse Iteration

Die einfache Vektoriteration erlaubt nur die Berechnung eines dominanten Eigenpaares. Oft sind aber gerade die betragsmäßig kleinsten Eigenwerte oder Eigenwerte in der Mitte des Spektrums von besonderem Interesse.

Beispiel 3.31 (Partitionierung von Graphen). Sei $G = (V, E)$ ein einfacher, ungerichteter Graph mit n Knoten und $A_G \in \mathbb{R}^{n\times n}$ dessen Adjazenzmatrix. Die Matrix

$$L_G := \operatorname{diag}(\deg v_1, \ldots, \deg v_n) - A_G$$

wird als Laplace-Matrix von G bezeichnet. L_G ist positiv-semidefinit und $\sum_{j=1}^{n}(L_G)_{ij} = 0$, $i = 1, \ldots, n$.

Sei $V = V_1 \cup V_2$, $|V_1| = |V_2|$, eine Partition der Menge der n Knoten V und $x \in \mathbb{R}^n$ der Vektor mit den Komponenten

$$x_v := \begin{cases} 1, & \text{falls } v \in V_1 \\ -1, & \text{falls } v \in V_2. \end{cases}$$

Dann gilt (siehe M. Fiedler, 1975) für die Anzahl der Kanten C zwischen V_1 und V_2

$$|C| = \frac{1}{4} x^T L_G x. \tag{3.8}$$

Sucht man also eine Partitionierung mit möglichst kleinem Kantenschitt, so ist der Ausdruck (3.8) unter der Nebenbedingung

$$x^T x = n,\ x^T \mathbf{1} = 0,\ x_v \in \{-1, 1\}, \quad \mathbf{1} := \begin{bmatrix} 1 \\ \vdots \\ 1 \end{bmatrix}.$$

zu minimieren. Hierbei handelt es sich um diskretes, NP-vollständiges Optimierungsproblem. Ein Ausweg besteht in der Betrachtung des kontinuierlichen Optimierungsproblems

$$f(z) := z^T L_G z \to \min$$

mit der Nebenbedingung $z^T z = n$, $z \in \mathbb{R}^n$ und $z^T \mathbf{1} = 0$. Wegen $L_G \mathbf{1} = 0$ ist $\lambda_n = 0$ der kleinste Eigenwert von L_G mit Eigenvektor $u_n := \mathbf{1}$. Ferner existieren weitere $n-1$ Eigenvektoren, die $\{u_1, \ldots, u_n\}$ zur Orthonormalbasis von $\mathbb{R}^n$ werden lassen. Sei $z \in \mathbb{R}^n$ in dieser Basis dargestellt, d.h.

$$z = \sum_{i=1}^{n} \alpha_i u_i.$$

Wegen $0 = z^T u_n = \alpha_n$ verschwindet α_n. Ferner folgt aus $n = z^T z = \sum_{\ell=1}^{n-1} \alpha_\ell^2$, dass

$$\frac{z^T L_G z}{z^T z} = \frac{1}{n} \sum_{i=1}^{n-1} \lambda_i \alpha_i^2 \geq \frac{\lambda_{n-1}}{n} \sum_{i=1}^{n-1} \alpha_i^2 = \lambda_{n-1}.$$

Die Partitionierung erfolgt deshalb unter Verwendung des zu λ_{n-1} gehörenden Eigenvektors u_{n-1} (sog. Fiedler-Vektor).

Betrachtet man A^{-1} anstelle von A, so fällt auf, dass λ_n^{-1} der dominante Eigenwert von A^{-1} ist, falls $0 \neq |\lambda_n| < |\lambda_{n-1}|$. Durch Übergang zur Inversen kann also auch der betragskleinste Eigenwert von A berechnet werden. Diese Vorgehensweise bezeichnet man als **Inverse Iteration** nach Wieland. Um innere Eigenwerte mit Hilfe der Vektoriteration zu bestimmen, sei $\mu \in \mathbb{C}$ eine Zahl, die in der Nähe eines isolierten Eigenwerts λ_i liegt, d.h.

$$|\lambda_i - \mu| < |\lambda_j - \mu|, \quad j \neq i.$$

Dann ist $(\lambda_i - \mu)^{-1}$ der dominante Eigenwert von $(A - \mu I)^{-1}$ und die einfache Vektoriteration für die Matrix $(A - \mu I)^{-1}$, die sog. **Inverse Iteration mit Shift**, konvergiert gegen das Eigenpaar $((\lambda_i - \mu)^{-1}, v_i)$ von A. Der Konvergenzfaktor ist dabei $q := \max_{j \neq i} |\lambda_i - \mu| / |\lambda_j - \mu|$.

Algorithmus 3.32 (Inverse Iteration).
`Input:` $x_0 \in \mathbb{C}^n$ mit $\alpha_i \neq 0$ in (3.6), μ Approximation an λ_i und Toleranz $\varepsilon > 0$.
`Output:` Approximation an den zum Eigenwert λ_i gehörenden Eigenvektor v_i.

$k = 0$;
`do` {
  löse $(A - \mu I)\tilde{x}_{k+1} = x_k$;
  sei σ_k das Vorzeichen von $\tilde{x}_{k+1}^H x_k$;
  $x_{k+1} = \sigma_k \frac{\tilde{x}_{k+1}}{\|\tilde{x}_{k+1}\|_2}$;
  $k = k + 1$;
} `while` $(\|x_k - x_{k-1}\|_2 > \varepsilon)$;

Man beachte, dass in jedem Schritt der Iteration ein lineares Gleichungssystem (mit derselben Matrix) gelöst werden muss. Hierfür bietet sich eine vorgeschaltete LR-Zerlegung von $A - \mu I$ an.

Rayleigh-Quotienten-Verfahren

Die Konvergenz der Inversen Iteration kann durch Adaption von μ beschleunigt werden. Ist A hermitesch, so wissen wir aus Satz 3.30, dass der Rayleigh-Quotient von x_k schnell gegen den größten Eigenwert konvergiert. Das folgende **Rayleigh-Quotienten-Verfahren** ist die Kombination aus Inverser Iteration und der Approximation des Eigenwerts mit Hilfe des Rayleigh-Quotienten.

Algorithmus 3.33 (Rayleigh-Quotienten-Verfahren).
`Input:` $A \in \mathbb{C}^{n\times n}$, Näherung x_0, $\|x_0\|_2 = 1$, an einen Eigenvektor und Toleranz $\varepsilon > 0$.
`Output:` Approximation an nächstgelegenes Eigenpaar.

$k = 0$;
`do` {
  $\mu_{k+1} = x_k^H A x_k$;
  `if` $A - \mu_{k+1} I$ singulär `then`
    bestimme $\tilde{x}_{k+1}$ aus $(A - \mu_{k+1} I)\tilde{x}_{k+1} = 0$;
    $x_{k+1} = \frac{\tilde{x}_{k+1}}{\|\tilde{x}_{k+1}\|_2}$;
    `break`;
  `else`
    löse $(A - \mu_{k+1} I)\tilde{x}_{k+1} = x_k$;
    $x_{k+1} = \frac{\tilde{x}_{k+1}}{\|\tilde{x}_{k+1}\|_2}$;
  $k = k + 1$;
} `while` ($\|\tilde{x}_k\|_2$ nicht zu groß);

Man beachte, dass hier in jedem Schritt ein lineares Gleichungssystem mit einer neuen Koeffizientenmatrix gelöst werden muss. Durch die Adaption von μ konvergiert das Verfahren aber superlinear.

Satz 3.34. *Sei A hermitesch und (μ_0, x_0) eine ausreichend gute Approximation an ein Eigenpaar (λ, v) von A. Dann konvergiert $\{\mu_k\}$ aus Algorithmus 3.33 kubisch gegen λ, d.h. es existiert ein $c > 0$ mit*

$$|\mu_{k+1} - \lambda| \le c|\mu_k - \lambda|^3, \quad k = 0, 1, 2, \ldots$$

Beweis. Es bezeichne $\hat{\lambda}$ den Eigenwert von A, der den kleinsten positiven Abstand zu λ hat. Wir zerlegen

$$x_{k-1} = y_{k-1} + z_{k-1} \quad \text{mit} \quad A y_{k-1} = \lambda y_{k-1}, \quad z_{k-1} \perp y_{k-1}, \tag{3.9}$$

mit dem Anteil y_{k-1} im Eigenraum zu λ und dem Anteil z_{k-1} in den anderen Eigenräumen. Wir nehmen zunächst an, dass

$$\|z_{k-1}\|_2 \le \frac{1}{2}\|x_{k-1}\|_2 = \frac{1}{2} \quad \text{und} \quad |\mu_k - \lambda| \le \frac{1}{3}|\hat{\lambda} - \lambda| = \frac{1}{3} \min_{\tilde{\lambda}\in\sigma(A)\setminus\{\lambda\}} |\tilde{\lambda} - \lambda|. \tag{3.10}$$

In diesem Fall ist für $\tilde{\lambda} \in \sigma(A) \setminus \{\lambda\}$

$$|\tilde{\lambda} - \mu_k| \ge |\tilde{\lambda} - \lambda| - |\mu_k - \lambda| \ge \frac{2}{3}|\tilde{\lambda} - \lambda| \ge \frac{2}{3}|\hat{\lambda} - \lambda|. \tag{3.11}$$

Es gilt

$$|\mu_{k+1} - \lambda| = |x_k^H A x_k - \lambda| = |x_k^H (A - \lambda I) x_k| = \frac{|\tilde{x}_k^H (A - \lambda I)\tilde{x}_k|}{\|\tilde{x}_k\|_2^2}, \tag{3.12}$$

und Einsetzen von $\tilde{x}_k = (A - \mu_k I)^{-1} x_{k-1}$ ergibt mit $B := (A - \mu_k I)^{-H}(A - \mu_k I)^{-1} = B^H$

$$|\mu_{k+1} - \lambda| = \left| \frac{x_{k-1}^H B (A - \lambda I) x_{k-1}}{x_{k-1}^H B x_{k-1}} \right|. \tag{3.13}$$

Wegen (3.9) gilt auch $By_{k-1} = |\mu_k - \lambda|^{-2} y_{k-1} \perp z_{k-1}$ und daher

$$\begin{aligned} x_{k-1}^H B x_{k-1} &= |\mu_k - \lambda|^{-2} \|y_{k-1}\|_2^2 + z_{k-1}^H B z_{k-1} \geq |\mu_k - \lambda|^{-2} \|y_{k-1}\|_2^2 \\ &\geq |\mu_k - \lambda|^{-2} (\|x_{k-1}\|_2 - \|z_{k-1}\|_2)^2 \overset{(3.10)}{\geq} \frac{1}{4} |\mu_k - \lambda|^{-2}. \end{aligned}$$

Andererseits ist wegen $(A - \lambda I) y_{k-1} = 0$ und der Orthogonalität

$$|x_{k-1}^H B (A - \lambda I) x_{k-1}| = |x_{k-1}^H B (A - \lambda I) z_{k-1}| = |z_{k-1}^H B (A - \lambda I) z_{k-1}|.$$

Mit $A = V^H \Lambda V$ und der Diagonalmatrix $\Gamma \in \mathbb{C}^{n \times n}$ mit $\Gamma^H \Gamma = \Lambda - \lambda I$ erhält man

$$\begin{aligned} |z_{k-1}^H B (A - \lambda I) z_{k-1}| &= |(\Gamma V z_{k-1})^H (\Lambda - \mu_k I)^{-H} (\Lambda - \mu_k I)^{-1} \Gamma V z_{k-1}| \\ &\leq \max_{\tilde{\lambda} \in \sigma(A) \setminus \{\lambda\}} |\tilde{\lambda} - \mu_k|^{-2} |z_{k-1}^H (A - \lambda I) z_{k-1}| \\ &\overset{(3.11)}{\leq} \frac{9}{4} |\hat{\lambda} - \lambda|^{-2} |x_{k-1}^H (A - \lambda I) x_{k-1}| \\ &\overset{(3.12)}{=} \frac{9}{4} |\hat{\lambda} - \lambda|^{-2} |\mu_k - \lambda|. \end{aligned}$$

Aus (3.13) ergibt sich daher insgesamt

$$|\mu_{k+1} - \lambda| \leq \frac{9}{4} \frac{|\mu_k - \lambda|}{|\hat{\lambda} - \lambda|^2} 4 |\mu_k - \lambda|^2 = 9 |\hat{\lambda} - \lambda|^{-2} |\mu_k - \lambda|^3.$$

Wir müssen noch zeigen, dass (3.10) gilt. Dazu sei x_0 so gewählt, dass diese Annahme für $k = 1$ wahr ist. Angenommen, sie gilt für ein $k \in \mathbb{N}$. Dann ist

$$|\mu_{k+1} - \lambda| \leq 9 |\hat{\lambda} - \lambda|^{-2} \left(\frac{|\hat{\lambda} - \lambda|}{3} \right)^3 \leq \frac{1}{3} |\hat{\lambda} - \lambda|.$$

Ferner gilt

$$\tilde{x}_k = (A - \mu_k I)^{-1} x_{k-1} = \frac{1}{\lambda - \mu_k} y_{k-1} + (A - \mu_k I)^{-1} z_{k-1}.$$

Sei $\tilde{z}_k = (A - \mu_k I)^{-1} z_{k-1}$. Dann ist wegen

$$\|\tilde{z}_k\|_2 = \|(A - \mu_k I)^{-1} z_{k-1}\|_2 \overset{(3.11)}{\leq} \frac{3}{2} |\hat{\lambda} - \lambda|^{-1} \|z_{k-1}\|_2 \overset{(3.10)}{\leq} \frac{1}{2} |\mu_k - \lambda|^{-1} \|z_{k-1}\|_2$$

der Anteil von $z_k = \tilde{z}_k / \|\tilde{x}_k\|_2$ an x_k sogar noch kleiner als der von z_{k-1} an x_{k-1}. Es gilt nämlich wegen $\|\tilde{x}_k\|_2 \geq |\mu_k - \lambda|^{-1}$, dass

$$\|z_k\|_2 = \frac{\|\tilde{z}_k\|_2}{\|\tilde{x}_k\|_2} \leq |\mu_k - \lambda| \, \|\tilde{z}_k\|_2 \leq \frac{1}{2} \|z_{k-1}\|_2 \leq \frac{1}{4}.$$

□

Bemerkung. Algorithmus 3.33 kann auch bei nicht-hermiteschen Matrizen verwendet werden. Man kann zeigen, dass die Konvergenz dann lokal quadratisch ist.

3.3 Das QR-Verfahren

In diesem Abschnitt werden wir ein Verfahren vorstellen, das alle Eigenpaare einer Matrix $A \in \mathbb{C}^{n \times n}$ berechnet.

Algorithmus 3.35 (QR-Verfahren).
Setze $k = 0$ und $A_0 = A$;
`do` {
  berechne QR-Zerlegung von $A_k = Q_k R_k$;
  setze $A_{k+1} = R_k Q_k$;
  $k = k + 1$;
} `while` (A_k keine obere Dreiecksmatrix);

Lemma 3.36. *Die Matrizen aus Algorithmus 3.35 besitzen die Eigenschaften*

(i) $A_{k+1} = Q_k^H A_k Q_k$, $k \geq 0$;

(ii) $A = X_k A_k X_k^H$ *mit* $X_k := Q_0 Q_1 \dots Q_{k-1}$, $k \geq 1$;

(iii) $A^k = X_k U_k$ *mit* $U_k := R_{k-1} \dots R_1 R_0$, $k \geq 1$.

Beweis.

(i) Nach Konstruktion gilt $A_{k+1} = R_k Q_k = Q_k^H (Q_k R_k) Q_k = Q_k^H A_k Q_k$.

(ii) folgt sofort aus (i) wegen $A_0 = A$.

(iii) Für $k = 1$ ist die Behauptung gerade durch die erste QR-Zerlegung $A = Q_0 R_0$ gegeben. Der Induktionsschritt $k \to k + 1$ folgt aus

$$Q_k R_k = A_k \overset{\text{(ii)}}{=} X_k^H A X_k$$

und der Induktionsannahme $X_{k+1} U_{k+1} = X_k (Q_k R_k) U_k = A X_k U_k = A^{k+1}$.

□

Bemerkung.

(a) Wegen Lemma 3.36 (i) sind alle Matrizen A_k ähnlich zueinander und besitzen daher dieselben Eigenwerte. Weil sie sogar unitär ähnlich sind, ist der QR-Algorithmus stabil.

(b) Anstelle der QR-Zerlegung kann man auch die LR-Zerlegung verwenden:

$$\text{berechne } A_k = L_k R_k \text{ und setze } A_{k+1} := R_k L_k.$$

Das ist das **Rutishauser-Verfahren**, welches allerdings instabil ist.

Wegen Lemma 3.36 (iii) hat man

$$A^k e_1 = X_k U_k e_1 = (U_k)_{11} X_k e_1 = (U_k)_{11} x_1^{(k)}.$$

Weil $A^k e_1$ (bis auf Normierung) als Ergebnis von k Schritten der einfachen Vektoriteration zum Startwert $x_0 = e_1$ aufgefasst werden kann, konvergiert die erste Spalte $x_1^{(k)}$ von X_k der Richtung nach gegen den Eigenvektor zum betragsgrößten Eigenwert λ_1. Daher gilt

$$A_k e_1 = X_k^H A X_k e_1 = X_k^H A x_1^{(k)} \approx \lambda_1 X_k^H x_1^{(k)} = \lambda_1 e_1,$$

d.h. A_k hat näherungsweise die Gestalt

$$A_k \approx \begin{bmatrix} \lambda_1 & * & \dots & * \\ 0 & \vdots & & \vdots \\ \vdots & \vdots & & \vdots \\ 0 & * & \dots & * \end{bmatrix}.$$

Lemma 3.36 (iii) zeigt auch, dass das QR-Verfahren als Verallgemeinerung der Orthogonalen Iteration (siehe Seite 72) interpretiert werden kann.

Ist A invertierbar, so gilt

$$(U_k)_{nn} e_n^H A^{-k} = e_n^H U_k A^{-k} = e_n^H X_k^H = (x_n^{(k)})^H.$$

Daher ist $x_n^{(k)}$ das Resultat von k Schritten der Inversen Iteration für A^H, und es gilt

$$A_k^H e_n = X_k^H A^H X_k e_n = X_k^H A^H x_n^{(k)} \approx \overline{\lambda_n} X_k^H x_n^{(k)} = \overline{\lambda_n} e_n.$$

Zusammen mit der ersten Beobachtung ergibt sich

$$A_k \approx \begin{bmatrix} \lambda_1 & * & \dots & * \\ 0 & \vdots & & \vdots \\ \vdots & * & \dots & * \\ 0 & \dots & 0 & \lambda_n \end{bmatrix}.$$

Als Verallgemeinerung der letzten Beobachtung werden wir im Folgenden zeigen, dass der untere linke $(n-p) \times p$-Block $A_{21}^{(k)}$ von

$$A_k = \begin{bmatrix} A_{11}^{(k)} & A_{12}^{(k)} \\ A_{21}^{(k)} & A_{22}^{(k)} \end{bmatrix}, \quad A_{11}^{(k)} \in \mathbb{C}^{p \times p}, \tag{3.14}$$

gegen Null konvergiert, falls

$$|\lambda_1| \geq |\lambda_2| \geq \dots \geq |\lambda_p| > |\lambda_{p+1}| \geq \dots \geq |\lambda_n|. \tag{3.15}$$

In jedem Schritt verringert sich dabei die Größe der Einträge um den Faktor $|\lambda_{p+1}|/|\lambda_p|$. Hieraus folgt sofort, dass bei paarweise betragsverschiedenen Eigenwerten, d.h. p durchläuft hier die Menge $\{1, \dots, n-1\}$, das QR-Verfahren gegen eine obere Dreiecksmatrix konvergiert. Für den Konvergenzbeweis nehmen wir im Folgenden an, dass eine reguläre Matrix $V \in \mathbb{C}^{n \times n}$ existiert, so dass

(i) die p-te Hauptabschnittsmatrix von V^{-1} regulär ist;

(ii) es gilt

$$A = V \Lambda V^{-1} \quad \text{mit} \quad \Lambda = \begin{bmatrix} \Lambda_1 & 0 \\ 0 & \Lambda_2 \end{bmatrix}, \quad \Lambda_1 \in \mathbb{C}^{p \times p}, \tag{3.16}$$

und $\sigma(\Lambda_1) = \{\lambda_1, \dots, \lambda_p\}$, $\sigma(\Lambda_2) = \{\lambda_{p+1}, \dots, \lambda_n\}$.

Bedingung (i) ist eine Verallgemeinerung der Voraussetzung bei der Vektoriteration, dass der Startvektor nicht eine Linearkombination der übrigen Eigenvektoren ist.

Lemma 3.37. *Die p-te Hauptabschnittsmatrix von V^{-1} ist genau dann regulär, wenn*

$$\operatorname{span}\{e_1,\dots,e_p\}\cap\operatorname{span}\{v_{p+1},\dots,v_n\}=\{0\}.$$

Daher besitzen die Spalten $v_1,\dots,v_p$ von V Komponenten in Richtung der ersten p kanonischen Vektoren.

Beweis. Der Vektor $x\in\mathbb{C}^n$ gehört genau dann zu $\operatorname{span}\{e_1,\dots,e_p\}\cap\operatorname{span}\{v_{p+1},\dots,v_n\}$, d.h.

$$x=\sum_{j=1}^{p}\alpha_j e_j=\sum_{j=p+1}^{n}\beta_j v_j$$

mit Koeffizenten $\alpha_j,\beta_j\in\mathbb{C}$, wenn

$$V^{-1}x=\sum_{j=1}^{p}\alpha_j V^{-1}e_j=\sum_{j=p+1}^{n}\beta_j e_j.$$

Ein solches $x\neq 0$ gibt es genau dann, wenn die p-te Hauptabschnittsmatrix von V^{-1} verschwindet. □

Als Nächstes analysieren wir einen Schritt des QR-Verfahrens.

Lemma 3.38. *Die Matrix A_0 erfülle (3.16). Weiter sei*

$$f(A_0)=QR,\quad A_1=Q^HA_0Q$$

mit Q unitär, R obere Dreiecksmatrix und einem Polynom f. Sind die Matrizen A_0 und A_1 wie in (3.14) zerlegt und sind $F_1:=f(\Lambda_1)$ und $F_2:=f(\Lambda_2)$ regulär, so gilt

$$\|A_{21}^{(1)}\|_2\le c_1(1+c_2\phi)^2\phi\|A_{21}^{(0)}\|_2,$$

wobei $\phi:=\|F_2\|_2\|F_1^{-1}\|_2$ und die Konstanten $c_1,c_2>0$ nur von p und V abhängen.

Beweis. Wegen (3.16) lässt V^{-1} die folgende Block-LR-Zerlegung zu

$$V^{-1}=LU\quad\text{mit}\quad L:=\begin{bmatrix}I_p & 0\\ L_{21} & I_{n-p}\end{bmatrix}\quad\text{und}\quad U:=\begin{bmatrix}U_{11} & U_{12}\\ 0 & U_{22}\end{bmatrix},\quad U_{11}\in\mathbb{C}^{p\times p}.$$

Die Matrizen L und U sind regulär und ihre Inversen besitzen dieselbe Struktur

$$L^{-1}=\begin{bmatrix}I_p & 0\\ -L_{21} & I_{n-p}\end{bmatrix}\quad\text{und}\quad U^{-1}=\begin{bmatrix}U_{11}^{-1} & -U_{11}^{-1}U_{12}U_{22}^{-1}\\ 0 & U_{22}^{-1}\end{bmatrix}.$$

Mit der Block-Diagonalmatrix $F:=f(\Lambda)$ sieht man wegen

$$Q=f(A_0)R^{-1}=VFV^{-1}R^{-1}=VFLUR^{-1}$$

leicht, dass

$$\begin{aligned}A_1=Q^{-1}A_0Q&=(RU^{-1}L^{-1}F^{-1}V^{-1})(V\Lambda V^{-1})(VFLUR^{-1})=RU^{-1}L^{-1}\Lambda LUR^{-1}\\&=RU^{-1}F^{-1}(FL^{-1}\Lambda LF^{-1})FUR^{-1}.\end{aligned}$$

Weil sowohl $Q = V(FLF^{-1})(FUR^{-1})$ als auch Q^{-1} unitär sind, folgt

$$\|FUR^{-1}\|_2 \le \|V^{-1}\|_2 \|FL^{-1}F^{-1}\|_2 \quad \text{und} \quad \|RU^{-1}F^{-1}\|_2 \le \|V\|_2 \|FLF^{-1}\|_2.$$

Wegen

$$FLF^{-1} = \begin{bmatrix} I_p & 0 \\ F_2 L_{21} F_1^{-1} & I_{n-p} \end{bmatrix} = I + \begin{bmatrix} 0 & 0 \\ F_2 L_{21} F_1^{-1} & 0 \end{bmatrix}$$

erhält man

$$\|FLF^{-1}\|_2 = 1 + \|F_2 L_{21} F_1^{-1}\|_2 \le 1 + \|L_{21}\|_2 \phi \quad \text{und} \quad \|FL^{-1}F^{-1}\| \le 1 + \|L_{21}\|_2 \phi$$

auf analoge Weise. Wir betrachten nun die Block-Zerlegung

$$F(L^{-1}\Lambda L)F^{-1} = \begin{bmatrix} F_1 \Lambda_1 F_1^{-1} & 0 \\ F_2 (L^{-1}\Lambda L)_{21} F_1^{-1} & F_2 \Lambda_2 F_2^{-1} \end{bmatrix}.$$

Aus $L^{-1}\Lambda L = U A_0 U^{-1}$ folgern wir $(L^{-1}\Lambda L)_{21} = U_{22} A_{21}^{(0)} U_{11}^{-1}$, woraus man

$$\|(L^{-1}\Lambda L)_{21}\|_2 \le \|U_{22}\|_2 \|U_{11}^{-1}\|_2 \|A_{21}^{(0)}\|_2$$

erhält. Weil $RU^{-1}F^{-1}$ und FUR^{-1} obere Block-Dreiecksmatrizen sind, folgt schließlich

$$\|A_{21}^{(1)}\|_2 = \left\| RU^{-1}F^{-1} \begin{bmatrix} 0 & 0 \\ F_2 (L^{-1}\Lambda L)_{21} F_1^{-1} & 0 \end{bmatrix} FUR^{-1} \right\|_2 \le c_1 (1 + c_2\phi)^2 \phi \|A_{21}^{(0)}\|_2,$$

wobei $c_1 := \|V\|_2 \|V^{-1}\|_2 \|U_{22}\|_2 \|U_{11}^{-1}\|_2$ und $c_2 = \|L_{21}\|_2$. □

Satz 3.39. *Für $A \in \mathbb{C}^{n\times n}$ regulär gelte (3.15) und (3.16). Dann erzeugt das QR-Verfahren Matrizen A_k, so dass in der Darstellung (3.14) gilt*

$$\|A_{21}^{(k)}\|_2 \le c(q)\, q^k, \quad k = 1, 2, \ldots$$

für alle $q \in \mathbb{R}$ mit $|\lambda_{p+1}|/|\lambda_p| < q < 1$.

Beweis. Nach Lemma 3.36 gilt mit $f(x) := x^k$

$$f(A) = X_k U_k, \quad A_k = X_k^H A X_k.$$

Daher können k Schritte des QR-Verfahrens als ein Schritt für die Matrix A^k interpretiert werden. Nach Lemma 3.38 hat man

$$\|A_{21}^{(k)}\|_2 \le c \|\Lambda_2^k\|_2 \|\Lambda_1^{-k}\|_2.$$

Die Behauptung erhält man aus der Beobachtung, dass für beliebiges $\delta > 0$ und genügend große k gilt

$$\|\Lambda_2^k\|_2 \le (|\lambda_{p+1}| + \delta)^k, \quad \|\Lambda_1^{-k}\|_2 \le (|\lambda_p|^{-1} + \delta)^k.$$

□

Korollar 3.40. *Unter den Voraussetzungen von Satz 3.39 und bei diagonaler Matrix Λ gilt*

$$\|A_{21}^{(k)}\|_2 \leq c \left(\frac{|\lambda_{p+1}|}{|\lambda_p|} \right)^k, \quad k = 1, 2, \ldots$$

Korollar 3.41. *Sei $A \in \mathbb{C}^{n\times n}$ diagonalisierbar, d.h. $A = V\Lambda V^{-1}$, $\Lambda = \operatorname{diag}(\lambda_1, \ldots, \lambda_n)$ mit $|\lambda_1| > |\lambda_2| > \ldots > |\lambda_n| > 0$. Sind alle Hauptabschnittsmatrizen von V^{-1} regulär, so konvergiert das QR-Verfahren gegen eine obere Dreiecksmatrix. Die Diagonaleinträge $(A_k)_{ii}$ konvergieren mindestens linear gegen die Eigenwerte λ_i, $i = 1, \ldots, n$.*

Bemerkung. Die Folge $\{A_k\}$ konvergiert nach Lemma 3.36 im Fall betragsmäßig verschiedener Eigenwerte also gegen eine Schur-Zerlegung von A. Treten Eigenwerte gleichen Betrags auf, z.B. konjugierte Eigenwerte reeller Matrizen, so konvergiert $\{A_k\}$ gegen eine obere Blockdreiecksmatrix. Die Größe der Diagonalblöcke entspricht nach Satz 3.39 der Anzahl der Eigenwerte gleichen Betrags.

Konvergenzbeschleunigung durch Shifts

Die Konvergenzgeschwindigkeit des QR-Verfahrens hängt von der Größe der Quotienten $|\lambda_{p+1}|/|\lambda_p|$ ab. Ist $|\lambda_{p+1}| \approx |\lambda_p|$, so konvergiert das Verfahren nur langsam. Die Konvergenz kann durch Shifts $A \mapsto A - \mu I$ beschleunigt werden, indem die beiden Eigenwerte näher an die Null geschoben werden, um $|\lambda_{p+1} - \mu|/|\lambda_p - \mu|$ zu verkleinern. Im folgenden Algorithmus sei $\{f_k\}$ eine Folge von Polynomen, d.h. wir betrachten nicht nur einfache Shifts sondern sog. **Multishifts**.

Algorithmus 3.42 (QR-Verfahren mit Multishift).
Setze $k = 0$ und $A_0 = A$;
`do` {
  berechne QR-Zerlegung von $f_k(A_k) = Q_k R_k$;
  setze $A_{k+1} = Q_k^H A_k Q_k$;
  $k = k + 1$;
} `while` ($\|A_{21}^{(k)}\|_2$ zu groß);

Als Verallgemeinerung von Lemma 3.36 erhält man

Lemma 3.43. *Für $k \geq 1$ gilt*

(i) $A = X_k A_k X_k^H$ *mit* $X_k := Q_0 \ldots Q_{k-1}$;

(ii) $\prod_{i=0}^{k-1} f_i(A) = X_k U_k$ *mit* $U_k := R_{k-1} \ldots R_0$.

Beweis. Die erste Aussage ist offensichtlich. Um die zweite Aussage induktiv zu beweisen, erkennen wir, dass $k = 1$ genau der ersten QR-Zerlegung entspricht. Daher nehmen wir an, dass sie für ein $k \in \mathbb{N}$ gilt. Dann folgt aus der Induktionsvorraussetzung

$$\begin{aligned} X_{k+1}U_{k+1} &= X_k(Q_kR_k)U_k = X_kf_k(A_k)U_k = f_k(X_kA_kX_k^H)X_kU_k \overset{\text{(i)}}{=} f_k(A)X_kU_k \\ &= f_k(A)\prod_{i=0}^{k-1} f_i(A) = \prod_{i=0}^{k} f_i(A). \end{aligned}$$

□

Bemerkung. Die Konvergenzanalyse des QR-Verfahrens kann leicht auf das QR-Verfahren mit Shift übertragen werden, weil Lemma 3.38 bereits allgemeine Polynome berücksichtigt.

Im folgenden Satz werden wir die Konvergenz für einen speziellen Multishift, dem sog. **Rayleigh-Shift**

$$f_k := \chi_{A_{22}^{(k)}},$$

untersuchen. Dabei bezeichnet $\chi_{A_{22}^{(k)}}$ das charakteristische Polynom des rechten unteren Blocks $A_{22}^{(k)}$ von A_k in der Darstellung (3.14).

Satz 3.44. *Sei A diagonalisierbar, d.h. es gilt $\Lambda_1 = \operatorname{diag}(\lambda_1, \dots, \lambda_p)$, $\Lambda_2 = \operatorname{diag}(\lambda_{p+1}, \dots, \lambda_n)$ in (3.16) mit $|\lambda_{p+1}| < |\lambda_p|$. Angenommen, das QR-Verfahren mit Rayleigh-Shift $f_k := \chi_{A_{22}^{(k)}}$ konvergiert, d.h.*

$$\varepsilon_k := \|A_{21}^{(k)}\|_2 \to 0,$$

dann ist die Konvergenz quadratisch, d.h. es existieren $\delta, c > 0$, so dass für $\varepsilon_k \le \delta$ folgt

$$\varepsilon_{k+1} \le c\,\varepsilon_k^2.$$

Beweis. Betrachte einen Schritt von Algorithmus 3.42

$$f_k(A_k) = Q_kR_k \quad \text{und} \quad A_{k+1} = Q_k^HA_kQ_k.$$

Aus Lemma 3.38 folgt

$$\varepsilon_{k+1} \le c\,\varepsilon_k\alpha_k\beta_k, \quad \alpha_k := \|f_k(\Lambda_2)\|_2, \quad \beta_k := \|(f_k(\Lambda_1))^{-1}\|_2.$$

Seien $s_1, \dots, s_{n-p}$ die Eigenwerte von $A_{22}^{(k)}$. Mit

$$E := \begin{bmatrix} 0 & 0 \\ -A_{21}^{(k)} & 0 \end{bmatrix}$$

gilt nach dem Satz von Bauer-Fike (Satz 3.16) angewendet auf A_k und $A_k + E$

$$\min_{j=1,\dots,n} |\lambda_j - \lambda_i(A_k + E)| \le \operatorname{cond}(V)\,\varepsilon_k, \quad i = 1, \dots, n.$$

Weil die Eigenwerte von $A_k + E$ durch die Eigenwerte von $A_{11}^{(k)}$ ergänzt um die Eigenwerte von $A_{22}^{(k)}$ gegeben sind, folgt für genügend kleine ε_k nach dem Satz von Gerschgorin

$$\min_{i=1,\dots,n-p} |\lambda_{p+j} - s_i| \le \operatorname{cond}(V)\,\varepsilon_k, \quad j = 1, \dots, n-p.$$

Daher gilt

$$|f_k(\lambda_{p+j})| = \left|\prod_{i=1}^{n-p}(\lambda_{p+j} - s_i)\right| \leq c_1\varepsilon_k, \quad j = 1, \ldots, n-p,$$

und

$$|f_k(\lambda_j)| = \left|\prod_{i=1}^{n-p}(\lambda_j - s_i)\right| \geq c_2 > 0, \quad j = 1, \ldots, p,$$

wobei $c_1, c_2 > 0$ nicht von k abhängen. Hieraus folgt $\alpha_k \leq c_1\varepsilon_k$ und $\beta_k \leq c_2^{-1}$. □

Bemerkung. Ist A hermitesch, so kann sogar kubische Konvergenz nachgewiesen werden.

Gilt $|\lambda_n| < |\lambda_{n-1}|$, so konvergiert nach Satz 3.44 das QR-Verfahren mit Shiftpolynom $f_k(x) = x - a_{n,n}^{(k)}$ quadratisch. Als noch besser hat sich die Strategie herausgestellt, bei der der rechte untere 2×2-Block

$$B := \begin{bmatrix} a_{n-1,n-1}^{(k)} & a_{n-1,n}^{(k)} \\ a_{n,n-1}^{(k)} & a_{n,n}^{(k)} \end{bmatrix}$$

von A_k zur Bestimmung des Shifts herangezogen wird. Dazu wählt man $f_k(x) = x - \mu_k$, wobei μ_k derjenige Eigenwert von B ist, der am nächsten bei $a_{n,n}^{(k)}$ liegt. In beiden Fällen konvergiert $\{A_k\}$ schnell gegen

$$\lim_{k\to\infty} A_k = \left[\begin{array}{ccc|c} & & & * \\ & C & & \vdots \\ & & & * \\ \hline 0 & \ldots & 0 & \lambda_n \end{array}\right].$$

Anschließend kann man mit der Anwendung des QR-Verfahrens auf die kleinere Matrix C fortfahren ($\rightarrow$ sog. **Deflation**).

Praktische Realisierung des QR-Verfahrens

Weil QR-Zerlegung und Matrix-Multiplikation von allgemeinen Matrizen $O(n^3)$ Operationen benötigen, ist das QR-Verfahren sehr aufwändig. Der Aufwand kann erheblich reduziert werden, wenn die Matrix A vorab durch unitäre Ähnlichkeitstransformationen in Hessenberg-Form gebracht wird.

Matrizen $A \in \mathbb{C}^{n\times n}$ können beispielsweise mit Hilfe der Householder-Transformation (siehe Abschnitt 1.2.1) auf Hessenberg-Form gebracht werden. Anders als bei der QR-Zerlegung müssen hier Transformationen allerdings von links und rechts auf A angewendet werden. Wir stellen das Vorgehen schematisch dar ($+/*$ veränderter bzw. unveränderter Eintrag):

$$A = \begin{bmatrix} * & * & * & * \\ * & * & * & * \\ * & * & * & * \\ * & * & * & * \end{bmatrix} \xrightarrow{Q_1\cdot} \begin{bmatrix} * & * & * & * \\ + & + & + & + \\ & + & + & + \\ & + & + & + \end{bmatrix} \xrightarrow{\cdot Q_1^H} \begin{bmatrix} * & + & + & + \\ * & + & + & + \\ & + & + & + \\ & + & + & + \end{bmatrix}$$

$$\xrightarrow{Q_2\cdot} \begin{bmatrix} * & * & * & * \\ * & * & * & * \\ & + & + & + \\ & & + & + \end{bmatrix} \xrightarrow{\cdot Q_2^H} \begin{bmatrix} * & * & + & + \\ * & * & + & + \\ & * & + & + \\ & & + & + \end{bmatrix}.$$

Die Reduktion auf Hessenberg-Form benötigt $O(n^3)$ Operationen. Ist A hermitesch, so ist die resultierende Matrix eine hermitesche Hessenberg-Matrix und somit tridiagonal.

Bemerkung. Man beachte, dass wegen der symmetrischen Anwendung der Householder-Transformation von links und von rechts hierdurch keine obere Dreiecksstruktur erzielt werden kann. Die Multiplikation von rechts zerstört diese sofort wieder:

$$A = \begin{bmatrix} * & * & * & * \\ * & * & * & * \\ * & * & * & * \\ * & * & * & * \end{bmatrix} \xrightarrow{\hat{Q}_1 \cdot} \begin{bmatrix} + & + & + & + \\ & + & + & + \\ & + & + & + \\ & + & + & + \end{bmatrix} \xrightarrow{\cdot \hat{Q}_1^H} \begin{bmatrix} + & + & + & + \\ + & + & + & + \\ + & + & + & + \\ + & + & + & + \end{bmatrix}.$$

Wäre es im Allgemeinen möglich, mittels Ähnlichkeitstransformationen eine obere Dreiecksstruktur zu gewinnen, so hätten wir die Eigenwerte durch eine endliche Anzahl arithmetischer Operationen gefunden, was einen Widerspruch zum Abel-Ruffini-Theorem darstellt.

QR-Verfahren für Hessenberg-Matrizen

Die Effizienz des QR-Verfahrens wird dadurch gesteigert, dass die in jedem Schritt

$$f(H_0) = QR, \quad H_1 := Q^H H_0 Q$$

des QR-Verfahrens die Hessenberg-Struktur ausgenutzt werden kann. Für Hessenberg-Matrizen H_0 kann jeder QR-Schritt mit $O(n^2)$ Operationen realisiert werden. Wie wir sehen werden, ist ferner H_1 wieder eine Hessenberg-Matrix.

Für die Berechnung der QR-Zerlegung von $f(H_0)$ gehen wir davon aus, dass f ein Polynom ersten Grades ist. Dies ist deshalb möglich, weil sich jeder Multishift als Folge von einfachen Shifts darstellen lässt. Dann ist auch $f(H_0)$ eine Hessenberg-Matrix.

Beispiel 3.45. Betrachte den Multishift $f(x) = (x - \mu_2)(x - \mu_1)$. Sei $\hat{Q}\hat{R} = H_0 - \mu_1 I$ und $\bar{Q}\bar{R} = \hat{H}_0 - \mu_2 I$ mit $\hat{H}_0 := \hat{Q}^H H_0 \hat{Q}$. Wegen

$$f(H_0) = (H_0 - \mu_2 I)(H_0 - \mu_1 I) = (H_0 - \mu_2 I)\hat{Q}\hat{R} = \hat{Q}(\hat{H}_0 - \mu_2 I)\hat{R} = (\hat{Q}\bar{Q})(\bar{R}\hat{R})$$

lässt sich der Multishift durch zwei einfache Shifts darstellen.

Wir haben aber im Zusammenhang mit dem GMRES-Verfahren in Abschnitt 2.3 die Givens-Transformation kennengelernt, die Einträge selektiv zu Null transformieren kann. Dann eliminert $G^{(n,n-1)} \cdot \ldots \cdot G^{(2,1)}$ die Einträge unterhalb der Diagonalen von $f(H_0)$. Wir veranschaulichen dies wieder für $n = 4$:

$$f(H_0) = \begin{bmatrix} * & * & * & * \\ * & * & * & * \\ & * & * & * \\ & & * & * \end{bmatrix} \xrightarrow{G^{(2,1)} \cdot} \begin{bmatrix} + & + & + & + \\ & + & + & + \\ & * & * & * \\ & & * & * \end{bmatrix} \xrightarrow{G^{(3,2)} \cdot} \begin{bmatrix} * & * & * & * \\ & + & + & + \\ & & + & + \\ & & * & * \end{bmatrix} \xrightarrow{G^{(4,3)} \cdot} \underbrace{\begin{bmatrix} * & * & * & * \\ & * & * & * \\ & & + & + \\ & & & + \end{bmatrix}}_{=:R}$$

Die Anzahl der Operationen zur Berechnung der QR-Zerlegung

$$f(H_0) = QR, \quad Q := (G^{(n,n-1)} \cdot \ldots \cdot G^{(2,1)})^H,$$

ist von der Ordnung n^2.

Wir zeigen noch, dass $H_1 = Q^H H_0 Q$ wieder eine Hessenberg-Matrix ist. Dies folgt aus der Eigenschaft von Matrizen mit m unteren Nebendiagonalen, dass die Summe solcher Matrizen m Nebendiagonalen und das Produkt $2m$ Nebendiagonalen besitzt. Ist nämlich H_0 eine Hessenberg-Matrix, so besitzt $Q = f(H_0)R^{-1}$ im Allgemeinen g Nebendiagonalen, wobei g den Grad des Polynoms f bezeichnet. Wegen

$$f(H_1) = Q^H f(H_0) Q = RQ$$

hat $f(H_1)$ höchstens g Nebendiagonalen. Daher kann H_1 höchstens eine Nebendiagonale besitzen. Die Berechnung von H_1 benötigt ebenfalls nur $O(n^2)$ Operationen.

Bemerkung. Wir haben bereits bemerkt, dass H_0 tridiagonal ist, falls A hermitesch ist. Eine QR-Zerlegung einer Tridiagonalmatrix kann mittels Givens-Rotationen mit $O(n)$ statt $O(n^2)$ Operationen durchgeführt werden. Weil die Eigenwerte von hermiteschen Matrizen reell sind, kann das QR-Verfahren ferner in reeller Arithmetik durchgeführt werden.

Zur Berechnung der Eigenvektoren können wir die Transformationsmatrizen Q entweder in der Schur-Zerlegung $A = QRQ^H$ akkumulieren oder sie per Inverser Iteration bestimmen. Letztere Vorgehensweise hat den Vorteil, dass die Eigenvektoren selektiv berechnet werden können und die Transformationsmatrizen nicht gespeichert werden müssen. Man sollte allerdings die Transformation P auf Hessenberg-Form $A = PHP^H$ speichern und die Eigenvektoren von H bestimmen. Die auftretenden Inversen $(H-\mu I)^{-1}$ können sehr viel effizienter berechnet werden (z.B. mittels QR-Zerlegung nach Givens) als $(A-\mu I)^{-1}$. Die resultierenden Eigenvektoren müssen dann noch mit P transformiert werden. Aus der Schur-Zerlegung $A = QRQ^H$ können die Eigenvektoren von A dadurch bestimmt werden, dass zunächst $(R-\lambda_i I)v_i = 0$ mittels Rückwärtssubstitution gelöst und dann v_i mit Q multipliziert wird.

Berechnung der Singulärwertzerlegung

Nach Satz 1.25 könnte man zur Berechnung der Singulärwerte von $A \in \mathbb{C}^{m\times n}$, $m \geq n$, die Eigenwerte von $A^H A$ mit Hilfe des QR-Verfahrens bestimmen. Dies ist wegen der hohen Kosten beim Aufstellen von $A^H A$ und der damit verbundenen Quadrierung der Konditionszahl in der Regel nicht empfehlenswert.

Beispiel 3.46. Sei $\varepsilon > 0$ so gewählt, dass $1 + \varepsilon \doteq 1$ (in Gleitkommaarithmetik). Für

$$A = \begin{bmatrix} 1 & 1 \\ 0 & \sqrt{\varepsilon} \\ \sqrt{\varepsilon} & 0 \end{bmatrix} \quad \text{ist} \quad A^H A = \begin{bmatrix} 1+\varepsilon & 1 \\ 1 & 1+\varepsilon \end{bmatrix} \doteq \begin{bmatrix} 1 & 1 \\ 1 & 1 \end{bmatrix}.$$

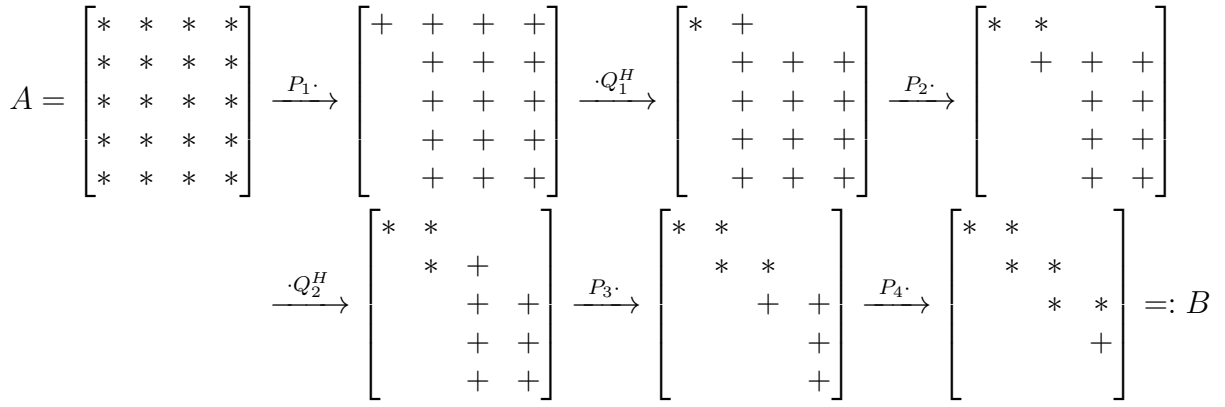
Die Eigenwerte von $A^H A$ wären $\sigma_1^2 = 2$, $\sigma_2^2 = 0$. Tatsächlich hat A den Rang 2 (auch in Gleitkommaarithmetik) und die Singulärwerte sind $\sqrt{\varepsilon}$, $\sqrt{2+\varepsilon}$.

Vorteilhafter ist es, die Singulärwerte von A direkt aus den Einträgen von A zu bestimmen. In Analogie zur Invarianz von Eigenwerten unter Ähnlichkeitstransformation werden die Singulärwerte durch Multiplikationen von A mit zwei unitären Matrizen P, Q von links und rechts nicht verändert. Denn ist $A = U\Sigma V^H$ eine Singulärwertzerlegung, so ist auch

$$PAQ^H = (PU)\Sigma(QV)^H$$

eine Singulärwertzerlegung von A. Daher kann A durch sukzessive Anwendung zweier unabhängiger Householdertransformationen auf Bidiagonalgestalt gebracht werden. Wir zeigen

die Vorgehensweise wieder schematisch:



Die Anzahl der Operationen beträgt $O(n^2 m)$. Die Singulärwerte der Bidiagonalmatrix B können dadurch bestimmt werden, dass das QR-Verfahren auf die Tridiagonalmatrix $B^H B$ angewendet wird. Aus Genauigkeitsgründen ist es allerdings vorteilhafter, B direkt zu betrachten. Dafür verweisen wir allerdings auf Kapitel 8.6 in Golub & Van Loan, *Matrix Computations*, 1996.

3.4 Das Lanczos-Verfahren

Das QR-Verfahren berechnet alle Eigenwerte einer Matrix und benötigt dafür einen kubischen Aufwand. In vielen Anwendungen (z.B. gewöhnliche und partielle Differentialgleichungen) ist man einerseits an nur wenigen Eigenwerten (meistens die kleinsten oder größten) interessiert und andererseits sind die auftretenden Matrizen großdimensioniert. Weil solche Matrizen aber oft schwachbesetzt sind, ist man an Verfahren interessiert, bei denen A nur durch Multiplikation mit Vektoren eingeht. Ein Beispiel dafür ist die in Abschnitt 3.2 vorgestellte Vektoriteration nach von Mises. Damit konnte jedoch ausschließlich der Eigenvektor zu einem dominanten Eigenwert berechnet werden. Mit dem im Folgenden vorgestellten *Lanczos-Verfahren* können auch weitere Eigenwerte berechnet werden. Zudem erhalten wir eine bessere Approximation an den größten Eigenwert als mit Hilfe der Vektoriteration.

Sei $A \in \mathbb{K}^{n\times n}$ hermitesch mit Eigenwerten $\lambda_1 \geq \ldots \geq \lambda_n$ und zugehörigen orthonormalen Eigenvektoren $v_1, \ldots, v_n$. Dann sind die extremalen Eigenwerte genau die Extremalwerte des Rayleigh-Quotienten

$$\lambda_1 = \max_{x\in\mathbb{K}^n\setminus\{0\}} \frac{x^H A x}{x^H x} \quad \text{und} \quad \lambda_n = \min_{x\in\mathbb{K}^n\setminus\{0\}} \frac{x^H A x}{x^H x}.$$

Anstelle die Maximierung bzw. Minimierung des Rayleigh-Quotienten über den gesamten $\mathbb{K}^n$ werden wir diese nur über dem **Krylov-Raum** (siehe Abschnitt 2.1)

$$\mathcal{K}_k(A, x_0) := \operatorname{span}\{x_0, Ax_0, A^2x_0, \ldots, A^{k-1}x_0\}, \quad x_0 \neq 0,$$

durchführen. Dazu wählen wir eine Orthonormalbasis $\{w_1, \ldots, w_k\}$ von $\mathcal{K}_k(A, x_0)$ und setzen

$$W_k = [w_1, \ldots, w_k] \in \mathbb{K}^{n\times k}.$$

Die Projektion

$$A_k := W_k^H A W_k \in \mathbb{R}^{k\times k}$$

von A auf den Krylov-Raum $\mathcal{K}_k(A, x_0)$ ist eine hermitesche, reelle Tridiagonalmatrix mit positiven Nebendiagonaleinträgen. Mit

$$\tilde{A}_k := \begin{bmatrix} A_k \\ \beta_{k+1} e_k^T \end{bmatrix} \in \mathbb{R}^{(k+1)\times k}$$

gilt dann $AW_k = W_{k+1}\tilde{A}_k$.

Für die Eigenwerte $\theta_1^{(k)} \geq \ldots \geq \theta_k^{(k)}$ von A_k gilt nach Satz 3.13

$$\theta_1^{(k)} = \max_{y\in\mathbb{K}^k\setminus\{0\}} \frac{y^H A_k y}{y^H y} = \max_{y\in\mathbb{K}^k\setminus\{0\}} \frac{(W_k y)^H A (W_k y)}{(W_k y)^H (W_k y)} = \max_{x\in\mathcal{K}_k(A,x_0)\setminus\{0\}} \frac{x^H A x}{x^H x} \leq \lambda_1$$

und analog

$$\theta_k^{(k)} = \min_{y\in\mathbb{K}^k\setminus\{0\}} \frac{y^H A_k y}{y^H y} \geq \lambda_n.$$

Bemerkung. Nach der Bemerkung zu Satz 2.8 wissen wir bereits, dass $P_k := W_k W_k^H$ ein orthogonaler Projektor auf

$$\operatorname{Ran} P_k = \operatorname{Ran} W_k = \mathcal{K}_k(A, x_0)$$

und A_k die Darstellung der Projektion

$$P_k A P_k = W_k W_k^H A W_k W_k^H = W_k A_k W_k^H$$

von A auf $\mathcal{K}_k(A, x_0)$ bzgl. der Basis $\{w_1, \ldots, w_k\}$ ist. Je größer der Unterraum $\operatorname{Ran} P_k$ ist, desto besser approximiert $W_k A_k W_k^H$ die Matrix A und damit $\theta_j^{(k)}$ die Eigenwerte λ_j.

Diese Idee der Projektionsverfahren oder **Ritz-Verfahren** ist nicht auf Krylov-Räume beschränkt. Wie wir im Abschnitt 2.1 gesehen haben, erlauben Krylov-Räume allerdings eine einfache Berechnung der Orthonormalbasis W_k. Weil die Eigenwerte $\theta_j^{(k)}$ durch Projektion entstehen, bezeichnet man sie auch als **Ritz-Werte**. Diese können mit Hilfe des QR-Verfahrens angewendet auf die (kleine) Tridiagonalmatrix $A_k \in \mathbb{R}^{k\times k}$ effizient bestimmt werden.

Der folgende Satz beschreibt den Approximationsfehler $\lambda_1 - \theta_1^{(k)}$.

Satz 3.47. *Es bezeichnen $\theta_1^{(k)} \geq \ldots \geq \theta_k^{(k)}$ die Ritz-Werte von A. Dann gilt*

$$\lambda_1 \geq \theta_1^{(k)} \geq \lambda_1 - \frac{(\lambda_1 - \lambda_n)\tan^2(\phi_1)}{T_{k-1}^2(2\mu - 1)}$$

mit dem Tschebyscheff-Polynom T_{k-1}, $\mu := (\lambda_1 - \lambda_n)/(\lambda_2 - \lambda_n)$ und $\cos(\phi_1) = |v_1^H x_0|/\|x_0\|_2$.

Beweis. O.B.d.A. sei $\|x_0\|_2 = 1$. Wegen $\mathcal{K}_k(A, x_0) = \operatorname{span}\{p(A)x_0,\, p \in \Pi_{k-1}\}$ gilt

$$\theta_1^{(k)} = \max_{x\in\mathcal{K}_k(A,x_0)\setminus\{0\}} \frac{x^H A x}{x^H x} = \max_{p\in\Pi_{k-1},\, p(A)x_0\neq 0} \frac{(p(A)x_0)^H A (p(A)x_0)}{(p(A)x_0)^H (p(A)x_0)}.$$

Mit der Darstellung von x_0 in der Orthonormalbasis $\{v_1, \ldots, v_n\}$

$$x_0 = \sum_{i=1}^n \alpha_i v_i, \quad \alpha_i := v_i^H x_0,$$

folgt

$$(p(A)x_0)^H A(p(A)x_0) = \sum_{i=1}^{n} |\alpha_i|^2 p^2(\lambda_i)\lambda_i \quad \text{und} \quad (p(A)x_0)^H (p(A)x_0) = \sum_{i=1}^{n} |\alpha_i|^2 p^2(\lambda_i).$$

Somit gilt

$$\begin{aligned}\frac{(p(A)x_0)^H A(p(A)x_0)}{(p(A)x_0)^H (p(A)x_0)} &= \lambda_1 - \frac{\sum_{i=2}^{n} |\alpha_i|^2 p^2(\lambda_i)(\lambda_1 - \lambda_i)}{\sum_{i=1}^{n} |\alpha_i|^2 p^2(\lambda_i)} \\ &\geq \lambda_1 - (\lambda_1 - \lambda_n)\frac{\sum_{i=2}^{n} |\alpha_i|^2 p^2(\lambda_i)}{|\alpha_1|^2 p^2(\lambda_1) + \sum_{i=2}^{n} |\alpha_i|^2 p^2(\lambda_i)}.\end{aligned}$$

Für eine möglichst scharfe Abschätzung müssen wir ein Polynom $p \in \Pi_{k-1}$ einsetzen, das innerhalb des Intervalls $[\lambda_n, \lambda_2]$ möglichst klein ist. Wir wählen das transformierte Tschebyscheff-Polynom (siehe Abschnitt 2.3.1)

$$p(x) := T_{k-1}\left(2\frac{x - \lambda_n}{\lambda_2 - \lambda_n} - 1\right).$$

Dieses erfüllt $|p(x)| \leq 1$ für $x \in [\lambda_n, \lambda_2]$. Wegen $\sum_{i=1}^{n} |\alpha_i|^2 = \|x_0\|_2^2 = 1$ gilt dann

$$\theta_1^{(k)} \geq \lambda_1 - (\lambda_1 - \lambda_n)\frac{1 - |\alpha_1|^2}{|\alpha_1|^2 p^2(\lambda_1)} = \lambda_1 - (\lambda_1 - \lambda_n)\frac{1 - |\alpha_1|^2}{|\alpha_1|^2 T_{k-1}^2(2\mu - 1)}.$$

Die Behauptung folgt aus

$$\frac{1 - |\alpha_1|^2}{|\alpha_1|^2} = \frac{1 - \cos^2(\phi_1)}{\cos^2(\phi_1)} = \frac{\sin^2(\phi_1)}{\cos^2(\phi_1)} = \tan^2(\phi_1).$$

□

Wendet man Satz 3.47 auf $-A$ an, so erhält man

Korollar 3.48. *Unter den Voraussetzungen von Satz 3.47 gilt*

$$\lambda_n \leq \theta_k^{(k)} \leq \lambda_n + \frac{(\lambda_1 - \lambda_n)\tan^2(\phi_n)}{T_{k-1}^2(2\mu' - 1)}$$

mit $\mu' := (\lambda_1 - \lambda_n)/(\lambda_1 - \lambda_{n-1})$ *und* $\cos(\phi_n) = |v_n^H x_0|/\|x_0\|_2$.

Bemerkung.

(a) Weil der Krylov-Raum $\mathcal{K}_k(A, x_0)$ den Vektor $A^{k-1}x_0$ enthält, ist durch den betragsgrößten Eigenwert von $A_k = W_k^H A W_k$ eine bessere Näherung an den betragsgrößten Eigenwert von A gegeben als durch den Rayleigh-Quotienten zu $A^{k-1}x_0/\|A^{k-1}x_0\|_2$ der Vektoriteration nach von Mises.

(b) Auch die übrigen Ritz-Werte können als Näherung an die Eigenwerte von A herangezogen werden. Nach Satz 3.24 gilt für $j = 1, \ldots, k$

$$\lambda_j \geq \theta_j^{(k+1)} \geq \theta_j^{(k)} \quad \text{und} \quad \theta_{k-j+1}^{(k)} \geq \theta_{k-j+2}^{(k+1)} \geq \lambda_{n-j+1}.$$

Daher wächst der j-größte Ritz-Wert mit wachsendem k monoton von unten gegen den j-größten Eigenwert von A und der j-kleinste Ritz-Wert fällt monoton von oben gegen den j-kleinsten Eigenwert von A.

(c) Ist u_j ein Eigenvektor von A_k zum Eigenwert $\theta_j^{(k)}$, so kann gezeigt werden, dass $W_k u_j$ eine Näherung an den Eigenvektor v_j zum Eigenwert λ_j ist.

Die folgende a-posteriori Abschätzung ist hilfreich, um die Genauigkeit der Approximation $\theta_j^{(k)}$ zu überprüfen.

Satz 3.49. *Sei $A \in \mathbb{K}^{n\times n}$ hermitesch und (θ, u), $\|u\|_2 = 1$, ein Eigenpaar von $A_k = W_k^H A W_k$. Dann gilt $\|AW_k u - \theta W_k u\|_2 = \beta_{k+1}|e_k^T u|$ und*

$$\min_{j=1,\dots,n} |\theta - \lambda_j| \le \beta_{k+1}|e_k^T u|.$$

Beweis. Wegen $AW_k = W_{k+1}\tilde{A}_k = W_k A_k + \beta_{k+1} w_{k+1} e_k^T$ folgt für $x := W_k u$

$$Ax - \theta x = (AW_k - W_k A_k)u = \beta_{k+1} w_{k+1} e_k^T u.$$

Wegen $\beta_{k+1} \ge 0$ ergibt dies $\|Ax - \theta x\|_2 = \beta_{k+1}|e_k^T u|$, und mit $\|x\|_2 = \|u\|_2 = 1$ folgt die Behauptung aus Satz 3.17. □

Wir haben bereits in Satz 2.8 gesehen, dass das Arnoldi- und somit auch das Lanczos-Verfahren genau im Schritt $k = \operatorname{grad}_A(x_0)$ mit $\beta_{k+1} = 0$ abbricht. Dann ist $\mathcal{K}_k(A, x_0)$ ein A-invarianter Unterraum. Daher gilt mit dem orthogonalen Projektor $P_k = W_k W_k^H$

$$W_k A_k = P_k A W_k = A W_k,$$

und die k Eigenwerte von A_k sind auch Eigenwerte von A. Dies erkennt man auch an Satz 3.49, weil die rechte Seite der Abschätzung für $\beta_{k+1} = 0$ verschwindet. Um weitere Eigenwerte von A zu bestimmen, muss das Lanczos-Verfahren mit einem anderen Vektor $x_0' \notin \mathcal{K}_k(A, x_0)$ neu gestartet werden.

Bemerkung. Durch Rundungsfehler kann es beim Lanczos-Verfahren passieren, dass nur die ersten Vektoren w_j orthogonal sind. Im Gegensatz zum Arnoldi-Verfahren werden diese nämlich nicht explizit orthogonalisiert, sondern die Orthogonalität entsteht aus mathematischen Eigenschaften, die in der Praxis aber nicht exakt vorliegen. Um diesem Effekt entgegenzutreten, werden Reorthogonalisierungen durchgeführt; siehe B. N. Parlett: *The Symmetric Eigenvalue Problem*, Prentice Hall, Englewood Cliffs, 1980. Die Orthogonalität geht immer dann verloren, wenn die Ritz-Werte gegen die Eigenwerte von A konvergieren.

3.5 Weitere Verfahren für tridiagonale Eigenwertprobleme

In diesem Abschnitt betrachten wir symmetrische Tridiagonalmatrizen

$$A = \begin{bmatrix} \alpha_1 & \beta_2 & & \\ \beta_2 & \ddots & \ddots & \\ & \ddots & \ddots & \beta_n \\ & & \beta_n & \alpha_n \end{bmatrix} \in \mathbb{R}^{n\times n}. \tag{3.17}$$

Diese resultieren z.B. aus dem Lanczos-Verfahren. Das zugehörige Eigenwertproblem kann beispielsweise mit Hilfe des QR-Verfahrens gelöst werden. Hier wollen wir zwei weitere Verfahren vorstellen. Wir dürfen annehmen, dass A **irreduzibel** ist, d.h. $\beta_i \neq 0$, $i = 2, \dots, n$. Sonst zerfällt A in irreduzible Untermatrizen, die voneinander getrennt betrachtet werden können.

Bisektionsverfahren

Das folgende Bisektionsverfahren ist das Mittel der Wahl, falls nur ein Teil des Spektrums benötigt wird. Dabei wird die Interlacing-Eigenschaft (Bemerkung nach Satz 3.24) genutzt, um die reelle Achse effizient nach Nullstellen des charakteristischen Polynoms abzusuchen. Im Folgenden bezeichne $A_k \in \mathbb{R}^{k\times k}$ die k-te Hauptabschnittsmatrix von A.

Beispiel 3.50. Wir betrachten die 4×4-Tridiagonalmatrix

$$A = \begin{bmatrix} 1 & 1 & & \\ 1 & 0 & 1 & \\ & 1 & 2 & 1 \\ & & 1 & -1 \end{bmatrix}.$$

Wegen

$$\det A_1 = 1, \quad \det A_2 = -1, \quad \det A_3 = -3, \quad \det A_4 = 4$$

und $\det A = \prod_{i=1}^{n} \lambda_i$ hat A_1 keinen und A_2 genau einen negativen Eigenwert. Nach der Interlacing-Eigenschaft (siehe die Bemerkung nach Satz 3.24) besitzt A_3 einen und A_4 zwei negative Eigenwerte.

Ist allgemeiner $A \in \mathbb{R}^{n\times n}$ eine symmetrische Tridiagonalmatrix, so betrachte die Anzahl der Vorzeichenwechsel $\nu(A)$ der Folge

$$1, \quad \det A_1, \quad \det A_2, \quad \dots, \quad \det A_n, \tag{3.18}$$

die auch als **Sturmsche Kette** bezeichnet wird. Verschwindet das nächste Folgenglied, so liegt kein Vorzeichenwechsel vor. Verschwindet das k-te Folgenglied, so liegt ein Vorzeichenwechsel vor, falls $(k-1)$-tes und $(k+1)$-tes Folgenglied nicht verschwinden und unterschiedliche Vorzeichen haben.

Wir werden in Satz 3.53 sehen, dass die Anzahl negativer Eigenwerte von A mit $\nu(A)$ übereinstimmt. Durch Verschiebung des Spektrums von A um $\alpha \in \mathbb{R}$ kann man bestimmen, wie viele Eigenwerte kleiner als α sind. Die Anzahl der Eigenwerte im Intervall $[a, b)$ ist also durch $\nu(A - bI) - \nu(A - aI)$ gegeben. Diese Information kann genutzt werden, um festzustellen, ob in durch rekursive Teilung erzeugten Intervalle Eigenwerte enthalten sind oder nicht.

Um die Berechnung der Sturmschen Kette effizient durchzuführen, kann die folgende Rekursionsformel verwendet werden.

Lemma 3.51. *Sei A wie in (3.17) eine symmetrische Tridiagonalmatrix. Dann gilt für $k \geq 2$*

$$\det A_k = \alpha_k \det A_{k-1} - \beta_k^2 \det A_{k-2}.$$

Beweis. Die Behauptung folgt aus dem Laplaceschen Entwicklungssatz durch Entwicklung nach der letzten Spalte. □

Im folgenden Lemma beweisen wir eine strikte Interlacing-Eigenschaft.

Lemma 3.52. *Sei $A \in \mathbb{R}^{n\times n}$ eine irreduzibele, symmetrische Tridiagonalmatrix. Dann gilt für die Eigenwerte $\lambda_j^{(k)},\ j = 1,\dots,k$, von A_k*

$$\lambda_1^{(k)} > \lambda_1^{(k-1)} > \lambda_2^{(k)} > \ldots > \lambda_{k-1}^{(k-1)} > \lambda_k^{(k)}, \quad k = 2,\dots,n.$$

Beweis. Nach der Interlacing-Eigenschaft (3.5) gilt

$$\lambda_1^{(k)} \geq \lambda_1^{(k-1)} \geq \lambda_2^{(k)} \geq \ldots \geq \lambda_{k-1}^{(k-1)} \geq \lambda_k^{(k)}, \quad k = 2,\dots,n.$$

Wir zeigen per Induktion, dass alle Eigenwerte von A_{k-1} und A_k verschieden sind. Aus Lemma 3.51 folgt, dass für das charakteristische Polynom p_k von A_k gilt

$$p_k(\lambda) = (\alpha_k - \lambda)p_{k-1}(\lambda) - \beta_k^2\, p_{k-2}(\lambda),$$

wobei $p_{-1}(\lambda) := 0$ und $p_0(\lambda) := 1$. Für $k = 2$ ist $p_2(\lambda) = (\alpha_2 - \lambda)p_1(\lambda) - \beta_2^2$. Wegen $\beta_2 \neq 0$ können p_1 und p_2 keine gemeinsamen Nullstellen besitzen. Seien alle Eigenwerte von A_{k-1} und A_k verschieden. Angenommen, es existiert ein $\lambda \in \mathbb{R}$ mit $p_{k+1}(\lambda) = 0 = p_k(\lambda)$. Wegen

$$0 = p_{k+1}(\lambda) = (\alpha_{k+1} - \lambda)p_k(\lambda) - \beta_{k+1}^2\, p_{k-1}(\lambda)$$

und $\beta_{k+1} \neq 0$ folgt, dass $p_{k-1}(\lambda) = 0$. Dann wäre λ aber ein gemeinsamer Eigenwert von A_{k-1} und A_k. Aus diesem Widerspruch folgt die Behauptung. □

Satz 3.53. *Sei $A \in \mathbb{R}^{n\times n}$ eine irreduzibele, symmetrische Tridiagonalmatrix. Dann stimmt die Anzahl $\nu(A)$ der Vorzeichenwechsel der Sturmschen Kette mit der Anzahl negativer Eigenwerte von A überein.*

Beweis. Wir zeigen die Behauptung per Induktion über n. Für $n = 1$ ist die Behauptung wahr. Angenommen, sie gilt für ein $n \in \mathbb{N}$. Wegen Lemma 3.52 ist

$$\lambda_j^{(n+1)} > \lambda_j^{(n)} \geq 0, \quad j = 1,\dots,n-\nu, \quad \text{und} \quad \lambda_{j+1}^{(n+1)} < \lambda_j^{(n)} < 0, \quad j = n-\nu+1,\dots,n.$$

Es bleiben nur zwei Fälle für $\lambda_{n-\nu+1}^{(n+1)}$:

$$\begin{aligned} &\text{(a)} \quad \lambda_{n-\nu+2}^{(n+1)} < \lambda_{n-\nu+1}^{(n)} < 0 \leq \lambda_{n-\nu+1}^{(n+1)} < \lambda_{n-\nu}^{(n)}, \\ &\text{(b)} \quad \lambda_{n-\nu+1}^{(n)} < \lambda_{n-\nu+1}^{(n+1)} < 0 \leq \lambda_{n-\nu}^{(n)} < \lambda_{n-\nu}^{(n+1)}. \end{aligned}$$

Im Fall (a) stimmt die Anzahl der negativen Eigenwerte von A_{n+1} mit der von A_n überein, welche nach Induktionsvoraussetzung ν ist. Auf der anderen Seite ist entweder $\det A_{n+1} = 0$ oder

$$\operatorname{sign}\det A_{n+1} = \operatorname{sign}\prod_{j=1}^{n+1}\lambda_j^{(n+1)} = \operatorname{sign}\prod_{j=n-\nu+1}^{n}\lambda_{j+1}^{(n+1)} = \operatorname{sign}\prod_{j=n-\nu+1}^{n}\lambda_j^{(n)} = \operatorname{sign}\det A_n.$$

In beiden Fällen liegt nach unserer Definition kein Vorzeichenwechsel vor.

Wir betrachten nun Fall (b). In diesem Fall besitzt A_{n+1} einen negativen Eigenwert mehr als A_n, und es gilt daher $\operatorname{sign}\det A_{n+1} = (-1)^{\nu+1}$. Im Fall $\lambda_{n-\nu}^{(n)} \neq 0$ gilt $\operatorname{sign}\det A_n = (-1)^{\nu}$. Ist jedoch $\lambda_{n-\nu}^{(n)} = 0$, so folgt nach der Interlacing-Eigenschaft $\lambda_{n-\nu-1}^{(n-1)} > \lambda_{n-\nu}^{(n)} = 0 > \lambda_{n-\nu}^{(n-1)}$. Daher ist $\det A_n = 0$ und $\operatorname{sign}\det A_{n-1} = (-1)^{\nu}$. Also hat die Sturmsche Kette vom n-ten zum $(n+1)$-ten Folgenglied einen weiteren Vorzeichenwechsel. □

Wir wollen diese Eigenschaft nun nutzen, um Eigenwerte durch rekursive Zweiteilung eines vorgegebenes Intervall mit Genauigkeit $\varepsilon > 0$ zu bestimmen. Die Rekursion stoppt dabei in solchen Intervallen, die für die Lokalisierung der Eigenwerte keine Bedeutung besitzen, also beispielsweise solche, die keine Eigenwerte enthalten. Unter Verwendung vom Lemma 3.51 kann ein Eigenwert auf diese Weise mit Genauigkeit ε in $O(n|\log\varepsilon|)$ Operationen lokalisiert werden. Wenn nur wenige Eigenwerte (z.B. der kleinste oder der größte) benötigt werden, ist dies eine Verbesserung gegenüber den $O(n^2)$ Operationen des QR-Verfahrens. Außerdem eignet sich dieses Vorgehen zur mathematischen Berechnung verschiedener Eigenwerte auf Parallelrechnern. Die zugehörigen Eigenvektoren können mittels Inverser Iteration bestimmt werden.

Algorithmus von Cuppen

Sei $A \in \mathbb{R}^{n\times n}$, $n \geq 2$, eine irreduzible, symmetrische Tridiagonalmatrix wie in (3.17). Wir geben im Folgenden nur die Grundidee eines divide-and-conquer-Verfahrens an. Auf Details bei der Implemtierung verzichten wir. Es sei jedoch bemerkt, dass diese essentiell sind, wenn man einen stabilen Algorithmus erhalten möchte. Die Idee des Verfahrens besteht darin, A wie folgt zu zerlegen

$$A = \left[\begin{array}{cc|cc} & A_1 & & \\ & & \beta_{m+1} & \\ \hline & \beta_{m+1} & & \\ & & & A_2 \end{array}\right] = \left[\begin{array}{cc|cc} & \hat{A}_1 & & \\ & & & \\ \hline & & & \\ & & & \hat{A}_2 \end{array}\right] + \left[\begin{array}{cc|cc} & & \beta_{m+1} & \beta_{m+1} \\ & & & \\ \hline & & \beta_{m+1} & \beta_{m+1} \\ & & & \end{array}\right]$$

mit $A_1 \in \mathbb{R}^{m\times m}$ und $A_2 \in \mathbb{R}^{(n-m)\times(n-m)}$ und $1 \leq m < n$, $m \approx \frac{n}{2}$. Auf die Weise kann also eine Tridiagonalmatrix als Summe einer 2×2-Blockdiagonal-Matrix mit tridiagonalen Blöcken und einer Rang-1-Korrektur dargestellt werden.

Angenommen, die Eigenwerte und Eigenvektoren von $\hat{A}_1 = Q_1 D_1 Q_1^T$ und $\hat{A}_2 = Q_2 D_2 Q_2^T$ sind bekannt. Dann gilt

$$A = \begin{bmatrix} Q_1 & \\ & Q_2 \end{bmatrix} \left(\begin{bmatrix} D_1 & \\ & D_2 \end{bmatrix} + \beta_{m+1} \begin{bmatrix} q_1 \\ q_2 \end{bmatrix} \begin{bmatrix} q_1 \\ q_2 \end{bmatrix}^T \right) \begin{bmatrix} Q_1 & \\ & Q_2 \end{bmatrix}^T,$$

wobei q_1^T und q_2^T die letzte Zeile von Q_1 bzw. die erste Zeile von Q_2 bezeichnen. Bei der Berechnung der Eigenwerte von $\hat{A}_1$ und $\hat{A}_2$ wendet man dieselbe Idee an, bis man bei einer Menge von 1×1 Eigenwertproblemen endet. Wir können uns daher auf die Bestimmung der Eigenwerte von Diagonal-plus-Rang-1-Matrizen $D \pm ww^T$ zurückziehen. Dabei dürfen wir annehmen, dass $w_j \neq 0$ für alle j, weil das Problem sonst reduzierbar ist.

Lemma 3.54. *Sei $D \in \mathbb{R}^{n\times n}$ eine Diagonalmatrix mit paarweise verschiedenen Einträgen und $w \in \mathbb{R}^n$, $w_i \neq 0$, $i = 1, \ldots, n$. Die Eigenwerte von $D \pm ww^T$ erfüllen die sog.* ***Säkulargleichung*** *$f(x) = 0$ mit*

$$f(x) := 1 \pm \sum_{i=1}^{n} \frac{w_i^2}{d_i - x}. \tag{3.19}$$

Beweis. Ist nämlich $(D \pm ww^T)q = \lambda q$ für ein $q \neq 0$, so gilt

$$\begin{aligned} & (D - \lambda I)q \pm w(w^T q) = 0 \\ \Leftrightarrow \quad & q \pm (D - \lambda I)^{-1} w(w^T q) = 0 \\ \Leftrightarrow \quad & w^T q \pm w^T (D - \lambda I)^{-1} w(w^T q) = 0 \\ \Leftrightarrow \quad & f(\lambda) w^T q = 0. \end{aligned}$$

Angenommen, es gilt $w^T q = 0$, so folgt $Dq = \lambda q$. Weil die Einträge von D paarweise verschieden sind, kann dann q nur genau einen nicht-verschwindenden Eintrag $q_j \neq 0$, $j \in \{1, \ldots, n\}$, besitzen. Dann ist aber $0 = w^T q = w_j q_j$, woraus $w_j = 0$ im Widerspruch zur Voraussetzung folgt. Daher ist $f(\lambda) = 0$. Die Umkehrung gilt, weil f genau n Nullstellen besitzt. □

In jedem Rekursionschritt des divide-and-conquer Algorithmus können die Nullstellen von (3.19) mit Hilfe des Newton-Verfahrens in $O(n^2)$ Operationen bestimmt werden. Zur Bestimmung aller Eigenwerte einer symmetrischen $n \times n$-Tridiagonalmatrix mit Hilfe des beschriebenen divide-and-conquer-Algorithmus sind also

$$n^2 + 2\left(\frac{n}{2}\right)^2 + 4\left(\frac{n}{4}\right)^2 + \ldots + n\left(\frac{n}{n}\right)^2 \leq n^2 \sum_{\ell=0}^{\infty} 2^{-\ell} = 2n^2$$

Operationen nötig. Hinsichtlich der Anzahl der Operationen verhalten sich dieser nach Cuppen benannte Algorithmus und das QR-Verfahren bei der Berechnung der Eigenwerte etwa gleich. Die Eigenvektoren können aus dem Ergebnis des Cuppen-Algorithmus aber deutlich effizienter bestimmt werden; vgl. Trefethen & Bau: *Numerical Linear Algebra*, SIAM.

3.6 Das Jacobi-Verfahren

Einer der ältesten Algorithmen zur Berechnung der Eigenwerte einer Matrix ist das **Jacobi-Verfahren**, das von Jacobi 1845 vorgestellt wurde. Es eignet sich besonders für die Parallelisierung. Das Verfahren transformiert die symmetrische Matrix $A \in \mathbb{R}^{n\times n}$ durch eine Folge orthogonaler Transformationen auf approximative Diagonalgestalt, d.h. nach jeder Transformation verringert sich die Größe der Außerdiagonaleinträge.

Wir betrachten zunächst reelle symmetrische 2×2-Matrizen. Diese können wie folgt diagonalisiert werden:

$$J^T \begin{bmatrix} a & d \\ d & b \end{bmatrix} J = \begin{bmatrix} \tilde{a} & 0 \\ 0 & \tilde{b} \end{bmatrix}$$

mit der Givens-Rotation

$$J = \begin{bmatrix} c & s \\ -s & c \end{bmatrix},$$

wobei $c = (1+t^2)^{-1/2}$, $s = tc$ und

$$t = \tau \pm \sqrt{1+\tau^2} \tag{3.20}$$

mit $\tau = (a-b)/2d$. Die orthogonale Matrix J wird auch als **Jacobi-Rotation** bezeichnet.

Im allgemeinen Fall $n \in \mathbb{N}$ wird die Matrix J zu einer Matrix $J_{k\ell} \in \mathbb{R}^{n\times n}$ vergrößert, wobei sich $J_{k\ell}$ nur in den Einträgen (k,k), (k,ℓ), (ℓ,k) und (ℓ,ℓ) von der Identität unterscheidet. In diesen vier Einträgen stehen die Einträge von J. Wir zeigen, dass der Ausdruck

$$\operatorname{off}(A) := \|A\|_F^2 - \sum_{i=1}^{n} a_{ii}^2$$

als Maß für die Größe des Außerdiagonalanteils reduziert wird.

Lemma 3.55. *Es gilt* $\operatorname{off}(J_{k\ell}^T A J_{k\ell}) = \operatorname{off}(A) - 2a_{k\ell}^2$.

Beweis. Wir setzen $B = J_{k\ell}^T A J_{k\ell}$. Dann gilt $b_{kk}^2 + b_{\ell\ell}^2 = a_{kk}^2 + 2a_{k\ell}^2 + a_{\ell\ell}^2$. Weil sich die Diagonalen von A und B nur an der k-ten und der ℓ-ten Position unterscheiden, gilt

$$\begin{aligned}
\operatorname{off}(B) &= \|B\|_F^2 - \sum_{i=1}^{n} b_{ii}^2 = \|A\|_F^2 - b_{kk}^2 - b_{\ell\ell}^2 - \sum_{i\notin\{k,\ell\}} b_{ii}^2 \\
&= \|A\|_F^2 - b_{kk}^2 - b_{\ell\ell}^2 - \sum_{i\notin\{k,\ell\}} a_{ii}^2 = \|A\|_F^2 - 2a_{k\ell}^2 - \sum_{i=1}^{n} a_{ii}^2 \\
&= \operatorname{off}(A) - 2a_{k\ell}^2.
\end{aligned}$$

□

Für den Parameter t stehen nach (3.20) zwei Möglichkeiten zur Verfügung. Um die Transformation auszuwählen, die die kleinere Differenz $\|A - J_{k\ell}^T A J_{k\ell}\|_F$ hervorruft, sollte man nach folgendem Lemma t als die kleinere der beiden Möglichkeiten wählen. Auf diese Weise ist z.B. sichergestellt, dass $J_{k\ell}$ die Matrixeinträge innerhalb des Außerdiagonalanteils nicht verschiebt.

Lemma 3.56. *Sei $A \in \mathbb{R}^{n\times n}$. Dann gilt*

$$\|A - J_{k\ell}^T A J_{k\ell}\|_F^2 = \frac{2}{c^2} a_{k\ell}^2 + 4(1-c) \sum_{i\notin\{k,\ell\}} a_{ik}^2 + a_{i\ell}^2.$$

Beweis. Siehe Übungsaufgaben. □

Im Folgenden beschreiben wir zwei mögliche Vorgehensweisen bei der Wahl der Indizes k und ℓ.

Maximale Außerdiagonaleinträge

In jedem Transformationsschritt wählt man das Paar (k, ℓ) so, dass $a_{k\ell}$ der betragsgrößte Außerdiagonaleintrag ist. Mit dieser Wahl erhält man folgendes Reduktionsverhalten.

Lemma 3.57. *Sei $a_{k\ell}$ ein betragsgrößte Außerdiagonaleintrag. Dann gilt*

$$\operatorname{off}(J_{k\ell}^T A J_{k\ell}) \leq \left(1 - \frac{2}{n(n-1)}\right) \operatorname{off}(A).$$

Beweis. Wegen

$$\operatorname{off}(A) = 2\sum_{i<j} a_{ij}^2 \leq 2\sum_{i<j} a_{k\ell}^2 = n(n-1)a_{k\ell}^2$$

erhält man nach Lemma 3.55

$$\operatorname{off}(J_{k\ell}^T A J_{k\ell}) = \operatorname{off}(A) - 2a_{k\ell}^2 \leq \operatorname{off}(A) - \frac{2}{n(n-1)}\operatorname{off}(A) = \left(1 - \frac{2}{n(n-1)}\right) \operatorname{off}(A).$$

□

Nach $O(n^2)$ Anwendungen von Jacobi-Rotationen, von denen jede $O(n)$ Operationen benötigt, verringert sich $\operatorname{off}(A)$ wegen

$$\left(1 - \frac{2}{n(n-1)}\right)^{n(n-1)/2} \leq \frac{1}{e}$$

um einen konstanten Faktor. Daher wird nach $O(n^2|\log\varepsilon|)$ Schritten ein relativer Fehler $\varepsilon > 0$ erreicht. Tatsächlich ist die Konvergenzgeschwindigkeit aber quadratisch, so dass nur $O(n^2|\log|\log\varepsilon||)$ Schritte benötigt werden.

Zyklisches Vorgehen

Um die n^2 Operationen für die Suche des maximalen Eintrages zu vermeiden, ist die folgenden Vorgehensweise sinnvoll. Man eleminiert zeilenweise die $n(n-1)/2$ Einträge oberhalb der Diagonalen. Hierbei beachte man, dass der im letzten Schritt eleminierte Eintrag im nachfolgenden Schritt im Allgemeinen aufgefüllt wird. Auch für dieses Vorgehen kann man wieder obiges Konvergenzverhalten zeigen: nach einem Durchlauf durch alle $n(n-1)/2$ Außerdiagonaleeinträge hat sich $\operatorname{off}(A)$ um einen konstanten Faktor verkleinert.

Das Konvergenzverhalten der Einträge im Jacobi-Verfahren ist im Allgemeinen sogar besser als beim QR-Verfahren. Allerdings sind QR-Verfahren und Cuppen-Algorithmus deutlich schneller.

Bemerkung. Falls die Indexpaare (k, ℓ) und (k', ℓ') die Bedingung $\{k, \ell\} \cap \{k', \ell'\} = \emptyset$ erfüllen, können die Jacobi-Rotationen dieser Einträge unabhängig voneinander durchgeführt werden. Daher eignet sich das Jacobi-Verfahren zur Parallelisierung.

4 Numerische Integration

Ziel dieses Kapitels ist die numerische Approximation von Integralen

$$I(f) := \int_a^b f(x)\,\mathrm{d}x,$$

die nicht in geschlossener Form durch Angabe einer Stammfunktionen integriert werden können.

Definition 4.1. *Eine Abbildung $Q_n : C[a,b] \to \mathbb{R}$ der Form*

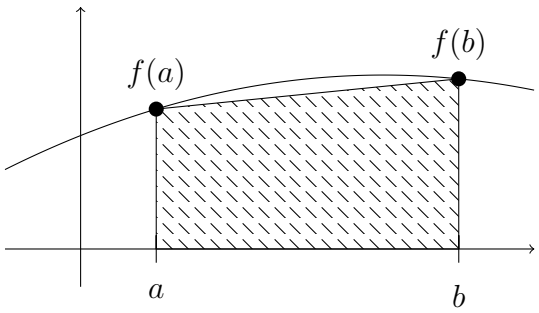
$$Q_n(f) = \sum_{j=0}^{n} w_j f(x_j)$$

mit Stützstellen $a \le x_0 < x_1 < \ldots < x_n \le b$ und Gewichten $w_0, \ldots, w_n \in \mathbb{R}$ heißt ***Quadraturformel****.*

Beispiel 4.2.

(a) **Mittelpunkt-Regel** $Q_0(f) = (b-a) f(\frac{a+b}{2})$

(b) **Trapez-Regel** $Q_1(f) = \frac{b-a}{2}(f(a) + f(b))$



(c) **Simpson-Regel** $Q_2(f) = \frac{b-a}{6}(f(a) + 4f(\frac{a+b}{2}) + f(b))$

Bemerkung. Der Operator $Q_n : C[a,b] \to \mathbb{R}$ ist linear. Der Ausdruck

$$c_Q := \sum_{j=0}^{n} |w_j| \tag{4.1}$$

beschreibt die Stabilität der Quadraturformel, weil für $f, g \in C[a,b]$ gilt

$$|Q_n(f) - Q_n(g)| = |Q_n(f-g)| \le c_Q \max_{j=0,\ldots,n} |f(x_j) - g(x_j)|.$$

Um Auslöschung zu vermeiden, sind insbesondere Quadraturformeln mit positiven Gewichten w_j zu bevorzugen.

Als Maß für die Qualität einer Quadraturformel verwendet man den folgenden Begriff.

Definition 4.3. *Eine Quadraturformel Q_n hat* ***Exaktheitsgrad*** *k, falls*

$$Q_n(p) = I(p) \tag{4.2}$$

für alle $p \in \Pi_k$ gilt.

Bemerkung.

(a) Zur Bestimmung des Exaktheitsgrades genügt es, die Bedingung (4.2) für eine Basis von Π_k zu überprüfen, weil sowohl I als auch Q_n lineare Abbildungen sind.

(b) Weil $1 \in \Pi_0$, gilt

$$\sum_{j=0}^{n} w_j = \int_a^b 1 \,\mathrm{d}x = b - a.$$

Beispiel 4.4. Die Mittelpunkt-Regel und Trapez-Regel haben Exaktheitsgrad $k = 1$, weil

$$Q_0(1) = Q_1(1) = b - a = I(1) \quad \text{und} \quad Q_0(x) = Q_1(x) = \frac{b-a}{2}(a+b) = \frac{1}{2}(b^2 - a^2) = I(x).$$

Die Simpson-Regel hat Exaktheitsgrad $k = 2$, weil

$$Q_2(1) = b - a = I(1), \quad Q_2(x) = \frac{b-a}{6}(a + 2(a+b) + b) = \frac{1}{2}(b^2 - a^2) = I(x)$$

und

$$Q_2(x^2) = \frac{b-a}{6}(a^2 + (a+b)^2 + b^2) = \frac{b-a}{3}(a^2 + ab + b^2) = \frac{1}{3}(b^3 - a^3) = I(x^2).$$

Ein hoher Exaktheitsgrad k führt auf eine hohe Genauigkeit der Quadraturformel.

Satz 4.5. *Sei Q_n eine Quadraturformel mit Exaktheitsgrad k. Für $f \in C[a,b]$ gilt*

$$|I(f) - Q_n(f)| \leq (b - a + c_Q) \inf_{p \in \Pi_k} \|f - p\|_\infty.$$

mit c_Q aus (4.1). Im Fall nicht-negativer Gewichte w_j gilt

$$|I(f) - Q_n(f)| \leq 2(b-a) \inf_{p \in \Pi_k} \|f - p\|_\infty.$$

Beweis. Weil Q_n Exaktheitsgrad k besitzt, gilt $I(p) = Q_n(p)$ für alle $p \in \Pi_k$. Der erste Teil der Behauptung folgt aus

$$\begin{aligned} |I(f) - Q_n(f)| &= |I(f-p) - Q_n(f-p)| \leq |I(f-p)| + \sum_{j=0}^{n} |w_j||f(x_j) - p(x_j)| \\ &\leq (b - a + c_Q)\|f - p\|_\infty \end{aligned}$$

für alle $p \in \Pi_k$. Im Fall nicht-negativer Gewichte gilt entsprechend der Bemerkung nach Definition 4.3, dass $c_Q = b - a$. □

Bemerkung. Ist $f \in C^{k+1}[a,b]$, so kann die rechte Seite der Abschätzungen im vorangehenden Satz abgeschätzt werden durch

$$\inf_{p\in\Pi_k} \|f-p\|_\infty \le \frac{(b-a)^{k+1}}{2^{2k+1}} \frac{\|f^{(k+1)}\|_\infty}{(k+1)!}.$$

Um dies zu sehen, seien x_j, $j=0,\dots,k$, die Nullstellen des transformierten Tschebyscheff-Polynoms $T_{k+1}((2x-b-a)/(b-a))$ und $p \in \Pi_k$ das Interpolationspolynom von f zu den Stützstellen x_j. Dann gilt für den Interpolationsfehler

$$\|f-p\|_\infty \le \frac{\|f^{(k+1)}\|_\infty}{(k+1)!} \|\omega_{k+1}\|_\infty$$

mit dem Stützstellenpolynom $\omega_{k+1}(x) := \prod_{i=0}^{k}(x-x_i)$. Man überzeugt sich leicht davon, dass das transformierte Tschebyscheff-Polynom $T_{k+1}((2x-b-a)/(b-a))$ den führenden Koeffizienten $\left(\frac{2}{b-a}\right)^{k+1} 2^k$ besitzt. Daher gilt

$$\omega_{k+1}(x) = \prod_{i=0}^{k}(x-x_i) = \left(\frac{b-a}{2}\right)^{k+1} 2^{-k}\, T_{k+1}\left(\frac{2x-b-a}{b-a}\right)$$

und somit

$$\|\omega_{k+1}\|_\infty \le \frac{(b-a)^{k+1}}{2^{2k+1}}.$$

In diesem Kapitel werden weitere Verfahren zur numerischen Integration behandelt. Im Vergleich zur AlMa II, wo im Wesentlichen Interpolationsquadraturformeln behandelt wurden, werden wir hier Formeln mit maximalem Exaktheitsgrad, Formeln für singuläre und für hochdimensionale Kernfunktionen betrachten.

4.1 Gaußsche Quadraturformel

In diesem Abschnitt beantworten wir die Frage, wie die Stützstellen x_j, $j=0,\dots,n$, gewählt werden müssen, damit Q_n einen möglichst hohen Exaktheitsgrad k besitzt. Aus der AlMa II ist bekannt, dass bei äquidistanten Stützstellen (sog. Newton-Côtes-Formeln) der Exaktheitsgrad n ist bzw. $n+1$, falls n gerade ist. Das folgende Lemma gibt eine obere Schranke für k.

Lemma 4.6. *Der Exaktheitsgrad k von Q_n ist höchstens $2n+1$.*

Beweis. Angenommen, Q_n wäre exakt für Π_k mit einem $k \ge 2n+2$. Dann ergäbe sich für $p(x) := \prod_{j=0}^{n}(x-x_j)^2 \in \Pi_{2n+2}$ der Widerspruch

$$0 < \int_a^b p(x)\,\mathrm{d}x = Q_n(p) = 0.$$

□

Im Folgenden konstruieren wir Quadraturformeln, die Exaktheitsgrad $2n+1$ besitzen, indem wir die Stützstellen x_j als Nullstellen von Orthogonalpolynomen wählen. Grundlage dafür ist der folgende Satz. Man beachte, dass wir die Definition 4.3 des Exaktheitsgrades um eine Gewichtsfunktion w verallgemeinern. Der Nutzen von w wird sich im Zusammenhang mit singulären Integralen zeigen.

Satz 4.7. *Sei $w : [a,b] \to \mathbb{R}$ eine nicht-negative Gewichtsfunktion mit $0 < \int_a^b w(x)\,\mathrm{d}x < \infty$. Sei Q_n eine Quadraturformel zu den Stützstellen $a \le x_0 < \ldots < x_n \le b$. Dann sind folgende Aussagen äquivalent*

(i) Q_n ist exakt für Π_{2n+1}, d.h. $\int_a^b p(x)w(x)\,\mathrm{d}x = Q_n(p)$ für alle $p \in \Pi_{2n+1}$.

(ii) Q_n ist exakt für Π_n, und für $q_n(x) := \prod_{j=0}^n (x - x_j) \in \Pi_{n+1}$ gilt

$$\int_a^b p(x)q_n(x)w(x)\,\mathrm{d}x = 0 \quad \text{für alle } p \in \Pi_n. \tag{4.3}$$

Beweis. Sei Q_n exakt für Π_{2n+1}. Für $p \in \Pi_n$ ist $pq_n \in \Pi_{2n+1}$ und somit

$$\int_a^b p(x)q_n(x)w(x)\,\mathrm{d}x = Q_n(pq_n) = \sum_{j=0}^n w_j p(x_j)q_n(x_j) = 0.$$

Sei umgekehrt Q_n exakt auf Π_n und $\int_a^b p(x)q_n(x)w(x)\,\mathrm{d}x = 0$ für alle $p \in \Pi_n$. Ist $p \in \Pi_{2n+1}$, so bestimme $r_1, r_2 \in \Pi_n$ per Polynomdivision, so dass $p = r_1 q_n + r_2$. Dann gilt

$$\begin{aligned}\int_a^b p(x)w(x)\,\mathrm{d}x &= \int_a^b r_1(x)q_n(x)w(x)\,\mathrm{d}x + \int_a^b r_2(x)w(x)\,\mathrm{d}x = \int_a^b r_2(x)w(x)\,\mathrm{d}x \\ &= Q_n(r_2) = Q_n(r_1q_n) + Q_n(r_2) = Q_n(r_1q_n + r_2) = Q_n(p).\end{aligned}$$

□

Bemerkung. Die Bedingung (4.3) bedeutet also, dass q_n orthogonal zu Π_n bzgl. des Skalarproduktes

$$(f,g)_w := \int_a^b f(x)g(x)w(x)\,\mathrm{d}x \tag{4.4}$$

ist. Ist w nämlich eine nicht-negative Funktion, die nur auf einer Nullmenge verschwindet und für die gilt $0 < \int_a^b w(x)\,\mathrm{d}x < \infty$, so definiert der Ausdruck (4.4) ein Skalarprodukt (vgl. Definition 1.6) auf

$$L^2_w[a,b] := \{f : [a,b] \to \mathbb{R} : \int_a^b |f(x)|^2 w(x)\,\mathrm{d}x < \infty\}.$$

Daher sind die Stützstellen x_j als Nullstellen von Orthogonalpolynomen zu wählen.

4.1.1 Orthogonalpolynome

Orthogonalpolynome können natürlich mit Hilfe des Gram-Schmidtschen Orthogonalisierungsverfahrens generiert werden. Der folgende Satz zeigt, dass Orthogonalpolynome aber auch einer einfachen Dreitermrekursion genügen.

Satz 4.8. *Zum Skalarprodukt $(\cdot,\cdot)_w$ existiert eine eindeutig bestimmte Folge von Polynomen $u_n \in \Pi_n$, $n = 0, \ldots, \infty$, mit führenden Koeffizienten $\gamma_n > 0$ und*

$$(u_i, u_j)_w = \delta_{ij}. \tag{4.5}$$

Die Folge $\{u_n\}$ genügt der Dreitermrekursion

$$a_{n+1}u_{n+1}(x) = (x - b_n)u_n(x) - a_n u_{n-1}(x), \quad n \geq 0, \tag{4.6}$$

wobei

$$a_n = \frac{\gamma_{n-1}}{\gamma_n} > 0, \quad b_n = (xu_n, u_n)_w \quad \text{und} \quad u_{-1} = 0,\ u_0 = (1,1)_w^{-1/2}.$$

Beweis. Wir zeigen die Aussage induktiv bzgl. n. Sei $\{u_0, \ldots, u_n\}$ eine Orthonormalbasis von Π_n, die die gewünschten Eigenschaften erfüllt. Für

$$p(x) := (x - b_n)u_n(x) - a_n u_{n-1}(x)$$

gilt

$$(p, u_n)_w = (xu_n, u_n)_w - b_n(u_n, u_n)_w - a_n(u_{n-1}, u_n)_w = (xu_n, u_n)_w - b_n = 0$$

und für jedes $m < n$

$$\begin{aligned}(p, u_m)_w &= (xu_n, u_m)_w - b_n(u_n, u_m)_w - a_n(u_{n-1}, u_m)_w = (u_n, xu_m)_w - a_n\delta_{n-1,m} \\ &= (u_n, a_{m+1}u_{m+1} + b_m u_m + a_m u_{m-1})_w - a_n\delta_{n-1,m} = a_{m+1}\delta_{n,m+1} - a_n\delta_{n-1,m} = 0.\end{aligned}$$

Daher ist p für alle $0 \leq m \leq n$ orthogonal zu u_m. Nach Induktionsvoraussetzung ist $\gamma_n > 0$ der führende Koeffizient von p. Sei

$$u_{n+1} := \frac{p}{\|p\|_w}.$$

Der führende Koeffizient γ_{n+1} von u_{n+1} ist positiv, und durch Vergleich der führenden Koeffizienten von u_{n+1} und p sieht man, dass $a_{n+1}u_{n+1} = p$ mit $a_{n+1} = \gamma_n/\gamma_{n+1}$. Ferner erfüllt u_{n+1} alle gewünschten Eigenschaften.

Ist $\hat{u}_{n+1} \in \Pi_{n+1}$ ein zweites Polynom mit $(\hat{u}_{n+1}, u_j)_w = 0$, $j = 0, \ldots, n$, und positivem führendem Koeffizienten. Dann gilt nach Satz 1.8

$$\hat{u}_{n+1} = \sum_{j=0}^{n+1} (\hat{u}_{n+1}, u_j)_w u_j = (\hat{u}_{n+1}, u_{n+1})_w u_{n+1} =: c\, u_{n+1}.$$

Daher ist $\hat{u}_{n+1}$ ein Vielfaches von u_{n+1}. Wegen $\|\hat{u}_{n+1}\|_w^2 = c^2\|u_{n+1}\|_w^2$ ergibt sich $c^2 = 1$. Weil $c = -1$ auf einen negativen führenden Koeffizienten führt, folgt schließlich $c = 1$ und somit die Eindeutigkeit. □

Beispiel 4.9. Die bekanntesten Orthogonalpolynome sind

(a) Tschebyscheff-Polynome: $w(x) = (1-x)^{-1/2}$, $(a,b) = (-1,1)$;

(b) Legendre-Polynome: $w(x) = 1$, $(a,b) = (-1,1)$;

(c) Jacobi-Polynome: $w(x) = (1-x)^{\alpha}(1+x)^{\beta}$ für $\alpha, \beta > -1$, $(a,b) = (-1,1)$;

(d) Hermite-Polynome: $w(x) = e^{-x^2}$, $(a,b) = (-\infty, \infty)$;

(e) Laguerre-Polynome: $w(x) = x^{\alpha}e^{-x}$ für $\alpha > -1$, $(a,b) = (0,\infty)$.

Mit Hilfe der Dreitermrekursion (4.6) lassen sich beliebige Polynome $p \in \Pi_n$ in der Darstellung

$$p(x) = \sum_{i=0}^{n} \alpha_i u_i(x) \tag{4.7}$$

effizient auswerten. Wegen

$$\begin{aligned}
p(x) &= \underbrace{\alpha_n}_{=:\beta_n} u_n(x) + \sum_{i=0}^{n-1} \alpha_i u_i(x) \\
&= \underbrace{\left[\frac{\beta_n}{a_n}(x - b_{n-1}) + \alpha_{n-1}\right]}_{=:\beta_{n-1}} u_{n-1}(x) + (\alpha_{n-2} - \beta_n \frac{a_{n-1}}{a_n}) u_{n-2} + \sum_{i=0}^{n-3} \alpha_i u_i(x) \\
&= \underbrace{\left[\frac{\beta_{n-1}}{a_{n-1}}(x - b_{n-2}) + \alpha_{n-2} - \beta_n \frac{a_{n-1}}{a_n}\right]}_{=:\beta_{n-2}} u_{n-2}(x) + (\alpha_{n-3} - \beta_{n-1} \frac{a_{n-2}}{a_{n-1}}) u_{n-3}(x) + \sum_{i=0}^{n-4} \alpha_i u_i(x) \\
&= \ldots = \beta_1 u_1(x) + \left(\alpha_0 - \beta_2 \frac{a_1}{a_2}\right) u_0(x) = \left(\frac{\beta_1}{a_1}(x - b_0) + \alpha_0 - \beta_2 \frac{a_1}{a_2}\right) (1,1)_w^{-1/2}
\end{aligned}$$

erhält man den folgenden Algorithmus, der p aus (4.7) in einem Punkt $x \in \mathbb{R}$ mit $O(n)$ Operationen auswertet.

Algorithmus 4.10 (Clenshaw-Algorithmus).
`Input:` Fourier-Koeffizienten α_i, $i = 0, \ldots, n$, und $x \in \mathbb{R}$
`Output:` Wert des Polynoms $p(x)$

Setze $\beta_n = \alpha_n$, $\beta_{n-1} = \frac{\beta_n}{a_n}(x - b_{n-1}) + \alpha_{n-1}$;
`for` $i = n-2, n-3, \ldots, 1$
    setze $\beta_i = \frac{\beta_{i+1}}{a_{i+1}}(x - b_i) + \alpha_i - \beta_{i+2}\frac{a_{i+1}}{a_{i+2}}$;
setze $p(x) = \left(\frac{\beta_1}{a_1}(x - b_0) + \alpha_0 - \beta_2 \frac{a_1}{a_2}\right)(1,1)_w^{-1/2}$;

Satz 4.11 (Nullstellen von Orthogonalpolynomen). *Die bzgl. $(\cdot,\cdot)_w$ orthogonalen Polynome u_n besitzen ausschließlich einfache Nullstellen, die alle in (a,b) liegen.*

Beweis. Definiere die Menge

$$N_n := \{\lambda \in (a,b) : \lambda \text{ ist Nullstelle ungerader Vielfachheit von } u_n\}$$

und setze

$$q(x) = \begin{cases} 1, & \text{falls } N_n = \emptyset, \\ \prod_{i=1}^{m}(x-\lambda_i), & \text{falls } N_n = \{\lambda_1, \dots, \lambda_m\}. \end{cases}$$

Dann ist $q_n \in \Pi_m$. Ferner ist $u_n q$ ist reell und hat in (a,b) keinen Vorzeichenwechsel. Daher folgt

$$(u_n, q)_w = \int_a^b u_n(x) q(x) w(x)\,\mathrm{d}x \neq 0.$$

Für $m < n$ widerspricht dies $u_n \perp \Pi_{n-1}$. Daher gilt $m = n$ und aus

$$\{\lambda_1, \dots, \lambda_n\} = N_n \subset (a,b)$$

folgt die Behauptung. □

Weil die Nullstellen von u_{n+1} paarweise verschieden sind, können sie als Stützstellen einer Interpolationsquadraturformel verwendet werden. Diese ist nach Satz 4.7 vom Grad $2n+1$. Wir fassen die Resultate dieses Abschnitts zusammen:

Satz 4.12. *Es existiert genau eine Quadraturformel Q_n der Ordnung $2n+1$. Ihre Stützstellen $a \le x_0 < \ldots < x_n \le b$ sind die Nullstellen des Orthogonalpolynoms $u_{n+1} \in \Pi_{n+1}$ mit führendem Koeffizienten γ_{n+1}, und für die Gewichte gilt*

$$w_j = \int_a^b L_j^2(x) w(x)\,\mathrm{d}x > 0, \quad L_j(x) = \prod_{\substack{i=0 \\ i\neq j}}^{n} \frac{x - x_i}{x_j - x_i}.$$

Dabei ist $L_1(x) = 1$ im Fall $n = 0$. Für $f \in C^{2n+2}[a,b]$ gilt ferner

$$\int_a^b f(x) w(x)\,\mathrm{d}x - Q_n(f) = \gamma_{n+1}^{-2} \frac{f^{(2n+2)}(\xi)}{(2n+2)!}$$

mit einem $\xi \in (a,b)$.

Beweis. Wegen $L_j^2 \in \Pi_{2n}$ gilt

$$0 < \int_a^b L_j^2(x) w(x)\,\mathrm{d}x = Q_n(L_j^2) = \sum_{k=0}^{n} w_k L_j^2(x_k) = w_j.$$

Für die Fehlerabschätzung sei $h \in \Pi_{2n+1}$ das Polynom, welches die Hermitesche Interpolationsaufgabe

$$h(x_i) = f(x_i), \quad h'(x_i) = f'(x_i), \quad i = 0, \dots, n,$$

löst. Dann gilt nach Satz 10.12 (AlMa II) die Restglieddarstellung

$$f(x) - h(x) = \delta[x_0, \dots, x_n, x] \prod_{i=0}^{n} (x - x_i)^2 = \delta[x_0, \dots, x_n, x] \frac{u_{n+1}^2(x)}{\gamma_{n+1}^2}.$$

Hieraus folgt wegen $Q_n(h) = \int_a^b h(x)w(x)\,\mathrm{d}x$ nach Satz 10.14 (AlMa II)

$$\begin{aligned}\int_a^b f(x)w(x)\,\mathrm{d}x - Q_n(f) &= \int_a^b [f(x)-h(x)]w(x)\,\mathrm{d}x - Q_n(f-h)\\ &= \gamma_{n+1}^{-2}\int_a^b \delta[x_0,\dots,x_n,x]\,\underbrace{u_{n+1}^2(x)w(x)\,\mathrm{d}x}_{\geq 0} - \sum_{j=0}^{n} w_j\,\underbrace{[f(x_j)-h(x_j)]}_{=0}\\ &= \gamma_{n+1}^{-2}\frac{f^{(2n+2)}(\xi)}{(2n+2)!}\|u_{n+1}\|_w^2.\end{aligned}$$

Die Behauptung folgt wegen $\|u_{n+1}\|_w = 1$. □

Bemerkung. Die nach Satz 4.12 eindeutig bestimmte Quadraturformel wird als **Gaußsche Formel** bezeichnet.

Beispiel 4.13. Die folgende Tabelle enthält die Stützstellen und Gewichte im Fall $w(x) = 1$ und $(a,b) = (-1,1)$ für $0 \leq n \leq 4$.

n	x_j	w_j
0	0	2
1	$-\frac{1}{\sqrt{3}},\ \frac{1}{\sqrt{3}}$	1, 1
2	$-\frac{\sqrt{15}}{5},\ 0,\ \frac{\sqrt{15}}{5}$	$\frac{5}{9},\ \frac{8}{9},\ \frac{5}{9}$
3	$\pm\frac{1}{35}\sqrt{525-70\sqrt{30}}$	$\frac{1}{36}(18+\sqrt{30})$
	$\pm\frac{1}{35}\sqrt{525+70\sqrt{30}}$	$\frac{1}{36}(18-\sqrt{30})$
4	0	$\frac{128}{225}$
	$\pm\frac{1}{21}\sqrt{245-14\sqrt{70}}$	$\frac{1}{900}(322+13\sqrt{70})$
	$\pm\frac{1}{21}\sqrt{245+14\sqrt{70}}$	$\frac{1}{900}(322-13\sqrt{70})$

Satz 4.14 (Konvergenz der Gauß-Quadratur). *Sei Q_n die Gauß-Formel aus Satz 4.12. Dann konvergiert die Folge $\{Q_n(f)\}_{n\in\mathbb{N}}$ gegen $\int_a^b f(x)w(x)\,\mathrm{d}x$ für jedes $f \in C[a,b]$.*

Beweis. Für die Gewichte $w_j > 0$ der Gauß-Formel gilt

$$\int_a^b w(x)\,\mathrm{d}x = Q_n(1) = \sum_{j=0}^{n} w_j =: c.$$

Sei $\varepsilon > 0$ beliebig vorgegeben. Nach dem Weierstraßschen Approximationssatz (siehe Übungen zur AlMa II) existiert ein $p_\varepsilon \in \Pi_m$ (m hinreichend groß) mit

$$\max_{a\leq x\leq b} |f(x) - p_\varepsilon(x)| \leq \frac{\varepsilon}{2c}.$$

Für $2n+1 \geq m$ ist $Q_n(p_\varepsilon) = \int_a^b p_\varepsilon(x)w(x)\,\mathrm{d}x$. Für solche n ist daher

$$\left|\int_a^b f(x)w(x)\,\mathrm{d}x - Q_n(f)\right| \leq \underbrace{\left|\int_a^b (f(x)-p_\varepsilon(x))w(x)\,\mathrm{d}x\right|}_{\leq \frac{\varepsilon}{2c}\cdot c} + \underbrace{|Q_n(f-p_\varepsilon)|}_{\leq \frac{\varepsilon}{2c}\cdot c} \leq \varepsilon.$$

Weil $\varepsilon > 0$ beliebig gewählt war, konvergiert $\{Q_n(f)\}_{n\in\mathbb{N}}$ gegen $\int_a^b f(x)w(x)\,\mathrm{d}x$. □

Bemerkung. Mit Hilfe der Fehlerabschätzung aus Satz 4.12 kann ebenfalls Konvergenz gezeigt werden. Allerdings sind bei Satz 4.14 die Glattheitsvoraussetzungen an f deutlich schwächer. Man beachte ferner, dass der vorangehende Beweis für alle Quadraturformeln mit positiven Gewichten gültig bleibt.

4.1.2 Berechnung der Stützstellen und Gewichte

Wir wollen abschließend ein Verfahren angeben, mit dem die Stützstellen und Gewichte der Gauß-Formel bestimmt werden können. Dazu sei

$$T_{n+1} = \begin{bmatrix} b_0 & a_1 & & \\ a_1 & \ddots & \ddots & \\ & \ddots & \ddots & a_n \\ & & a_n & b_n \end{bmatrix} \in \mathbb{R}^{(n+1)\times(n+1)}$$

mit den Koeffizienten der Rekursionsformel (4.6). Die Stützstellen x_j und die Gewichte w_j der Gauß-Formel Q_n ergeben sich aus den Eigenwerten und Eigenvektoren von T_{n+1}.

Lemma 4.15. *Die Eigenwerte von T_{n+1} stimmen mit den Nullstellen von u_{n+1} überein. Ist λ ein Eigenwert von T_{n+1}, so ist $z := [u_0(\lambda), \ldots, u_n(\lambda)]^T$ ein zugehörige Eigenvektor.*

Beweis. Sei λ eine Nullstelle von u_{n+1}. Dann folgt aus der Rekursionsformel (4.6), dass für $k = 1, \ldots, n-1$

$$\begin{aligned}(T_{n+1}z)_{k+1} &= a_k z_k + b_k z_{k+1} + a_{k+1} z_{k+2} = a_k u_{k-1}(\lambda) + b_k u_k(\lambda) + a_{k+1} u_{k+1}(\lambda) \\ &= \lambda u_k(\lambda) = \lambda z_{k+1}.\end{aligned}$$

Wegen $u_{-1}(\lambda) = 0 = u_{n+1}(\lambda)$ gilt $(T_{n+1}z)_k = \lambda z_k$ auch für $k = 1, n+1$. Weil u_{n+1} paarweise veschiedene Nullstellen besitzt, ergeben sich auf diese Weise alle Eigenwerte von T_{n+1}. □

Bemerkung. Weil T_{n+1} eine irreduzible, symmetrische Tridiagonalmatrix ist, liegt zwischen zwei aufeinanderfolgenden Nullstellen von u_{n+1} genau eine Nullstelle von u_n; siehe die Interlacing-Eigenschaft nach Satz 3.24.

Satz 4.16. *Sei Q_n die Gaußsche Quadraturformel. Die Stützstellen x_j sind die Eigenwerte von T_{n+1}. Ist $v_j \in \mathbb{R}^{n+1}$, $\|v_j\| = 1$, ein normierter Eigenvektor zum Eigenwert x_j von T_{n+1}, so gilt für das Gewicht w_j von Q_n*

$$w_j = (1,1)_w (v_j)_1^2, \quad j = 0, \ldots, n,$$

mit der ersten Komponente $(v_j)_1$ von v_j.

Beweis. Der erste Teil der Aussage folgt aus Lemma 4.15. Für den zweiten Teil betrachte die Lagrange-Polynome $L_j \in \Pi_n$, $j = 0, \dots, n$. Dann gilt $u_i L_j \in \Pi_{2n}$, $i = 0, \dots, n$. Wegen $L_j = \sum_{i=0}^n (L_j, u_i)_w u_i$ mit

$$(L_j, u_i)_w = \int_a^b u_i(x) L_j(x) w(x)\,\mathrm{d}x = Q_n(u_i L_j) = \sum_{k=0}^n w_k u_i(x_k) L_j(x_k) = w_j u_i(x_j)$$

folgt

$$\|L_j\|_w^2 = \sum_{i=0}^n |(L_j, u_i)_w|^2 = w_j^2 \sum_{i=0}^n |u_i(x_j)|^2.$$

Auf der anderen Seite gilt nach Satz 4.12, dass $w_j = \|L_j\|_w^2 > 0$. Hieraus erhalten wir

$$w_j = \left(\sum_{i=0}^n |u_i(x_j)|^2 \right)^{-1}. \tag{4.8}$$

Nach Lemma 4.15 gilt für die erste Komponente von v_j

$$(v_j)_1^2 = \frac{u_0^2(x_j)}{\sum_{i=0}^n |u_i(x_j)|^2} = (1,1)_w^{-1} w_j.$$

□

Beispiel 4.17. Wir betrachten das Intervall $(a, b) = (-1, 1)$ und die Gewichtsfunktion

$$w(x) = (1 - x^2)^{-1/2}.$$

Mit der Substitution $x = \cos\varphi$ erhält man

$$(f, g)_w = \int_{-1}^1 f(x) g(x) w(x)\,\mathrm{d}x = \int_0^\pi f(\cos\varphi) g(\cos\varphi)\,\mathrm{d}\varphi = (\tilde{f}, \tilde{g})_{L^2(0,\pi)}.$$

Man prüft leicht nach, dass die Funktionen $\tilde{T}_k(\varphi) := \cos(k\varphi)$, $k = 0, 1, 2, \dots$, orthogonal bzgl. $(\cdot,\cdot)_{L^2(0,\pi)}$ sind. Also sind die Tschebyscheff-Polynome $T_k(x) = \cos(k \arccos x) \in \Pi_k$ orthogonal bzgl. $(\cdot,\cdot)_w$. Die Dreitermrekursionsformel für T_k haben wir bereits in Abschnitt 2.3.1 kennengelernt. Durch Normierung erhält man

$$2^{-n} T_{n+1}(x) = \prod_{j=0}^n (x - x_j)$$

mit den Tschebyscheff-Knoten

$$x_j = \cos\frac{2j+1}{2n+2}\pi, \quad j = 0, \dots, n.$$

Normierung ergibt $u_0(x) = 1/\sqrt{\pi}$ und $u_k(x) = \sqrt{2/\pi}\, T_k(x)$, $k > 0$. Für die Gewichte ergibt sich aus (4.8) für $j = 0, \dots, n$

$$\begin{aligned} w_j^{-1} &= \sum_{k=0}^n |u_k(x_j)|^2 = \frac{1}{\pi} + \frac{2}{\pi} \sum_{k=1}^n \left[\cos k \frac{2j+1}{2n+2}\pi \right]^2 = \frac{1}{\pi} + \frac{1}{\pi} \sum_{k=1}^n \left[1 + \cos k \frac{2j+1}{n+1}\pi \right] \\ &= \frac{n+1}{\pi} + \frac{1}{\pi} \sum_{k=1}^n \left[\cos k \frac{2j+1}{n+1}\pi \right]. \end{aligned}$$

Die Summe verschwindet, weil sich der Summand $k = \ell$ mit dem Summanden $k = n + 1 - \ell$ aufhebt. Ist n ungerade, so verschwindet der Summand $k = (n+1)/2$ ebenfalls. Wir erhalten also $w_j = \pi/(n+1)$, $j = 0, \dots, n$.

Das folgende Beispiel zeigt den Nutzen der Gewichtsfunktion w.

Beispiel 4.18. Wir betrachten die Funktion

$$g(x) := x^{-\alpha} f(x)$$

mit $\alpha \in (0,1)$ auf $(0,1]$. Dabei sei $f : [0,1] \to \mathbb{R}$ analytisch. Da g in $x = 0$ singulär ist, kann die Gaußsche Quadratur nicht direkt auf g angewendet werden (die Fehlerabschätzung von Satz 4.12 benötigt $g \in C^{2n+2}(0,1]$). Setzen wir jedoch $w(x) = x^{-\alpha}$, so gilt $w(x) \geq 0$ auf $(0,1]$ und

$$0 < \int_0^1 w(x)\,\mathrm{d}x = \int_0^1 x^{-\alpha}\,\mathrm{d}x = \frac{1}{1-\alpha} x^{1-\alpha}\Big|_0^1 = \frac{1}{1-\alpha} < \infty.$$

Die Integration von g kann also durch eine Gauß-Quadratur von f mit Gewicht $w(x) = x^{-\alpha}$ approximiert werden.

4.2 hp-Quadratur

In folgendem Abschnitt stellen wir ein anderes Prinzip vor, singulären Funktionen mit Quadraturformeln ohne Verwendung der Gewichtsfunktion zu behandeln. Wie in Beispiel 4.18 betrachten wir $g(x) = x^{-\alpha} f(x)$, $\alpha \in (0,1)$, auf $(0,1]$.

Lemma 4.19. *Es existieren Konstanten $R > 0$ und $c > 0$, die nur von f abhängen, so dass*

$$\left|(x^{-\alpha} f(x))^{(n)}\right| \leq \frac{c\,n!}{R^n x^{\alpha+n}}$$

für alle $x \in (0,1]$ und $n \in \mathbb{N}$.

Beweis. Weil f analytisch in $[0,1]$ ist, kann f in jedem Punkt $x_0 \in [0,1]$ in eine Taylor-Reihe

$$f(x) = \sum_{k=0}^{\infty} \frac{f^{(k)}(x_0)}{k!}(x - x_0)^k, \quad |x - x_0| < r(x_0),$$

mit Konvergenzradius

$$r(x) = \left(\limsup_{k\to\infty} \sqrt[k]{\frac{f^{(k)}(x)}{k!}} \right)^{-1} > 0$$

entwickelt werden. Daher existiert $c > 0$ mit

$$\|f^{(k)}\|_\infty \leq c\frac{k!}{R^k}, \quad R := \min_{x\in[0,1]} r(x) > 0.$$

Ferner gilt

$$\left|\left(x^{-\alpha}\right)^{(k)}\right| = \alpha(\alpha+1)\cdot\ldots\cdot(\alpha+k-1)x^{-(\alpha+k)} \leq k!\,x^{-(\alpha+k)}.$$

Mit der Leibniz-Regel

$$(uv)^{(n)} = \sum_{k=0}^{n} \binom{n}{k} u^{(k)} v^{(n-k)}$$

folgt dann

$$\left|\left(x^{-\alpha}f(x)\right)^{(n)}\right| \le c\sum_{k=0}^{n}\binom{n}{k}\frac{k!(n-k)!}{R^k x^{\alpha+n-k}} = \frac{c\,n!}{R^n x^{\alpha+n}}\sum_{k=0}^{n} R^{n-k}x^k.$$

Aus

$$\sum_{k=0}^{n} R^{n-k}x^k \approx R^n\int_0^n \left(\frac{x}{R}\right)^t \mathrm{d}t = \frac{x^n - R^n}{\log x - \log R} \le c_R$$

erhält man die Behauptung. □

Sei Q_n die Gauß-Legendre-Quadraturformel (d.h. $w = 1$) auf $[a, b]$, $0 < a < b \le 1$. Dann gilt nach Satz 4.12

$$\left|\int_a^b g(x)\,\mathrm{d}x - Q_n(g)\right| \le \gamma_{n+1}^{-2}\frac{\|g^{(2n+2)}\|_{[a,b]}}{(2n+2)!}.$$

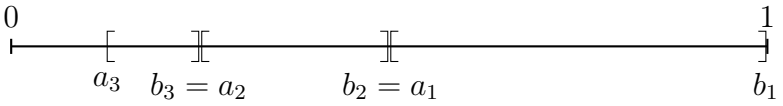
Wegen

$$\gamma_{n+1}^{-2} = \int_a^b \frac{u_{n+1}^2(x)}{\gamma_{n+1}^2}\,\mathrm{d}x = \int_a^b \prod_{i=0}^{n}(x-x_i)^2\,\mathrm{d}x \le (b-a)^{2n+3}$$

erhält man nach Lemma 4.19

$$\left|\int_a^b g(x)\,\mathrm{d}x - Q_n(g)\right| \le c\frac{b-a}{a^\alpha}\left(\frac{b-a}{Ra}\right)^{2n+2}.$$

Eine Erhöhung von n verbessert also dann den Fehler, wenn $b - a < Ra$. Daher muss das Intervall $(0, 1]$ in Teilintervalle $[a, b]$ unterteilt werden, in denen diese Bedingung gilt.



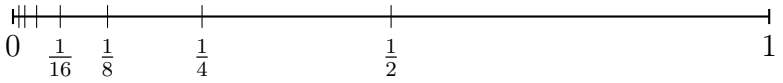
Wir setzen für $R \le \rho < R + 1$

$$b_1 = 1, \quad a_1 = \frac{1}{\rho}, \quad b_2 = a_1, \quad a_2 = \frac{a_1}{\rho}$$

und allgemein

$$b_k = a_{k-1}, \quad a_k = \frac{a_{k-1}}{\rho} = \rho^{-k}.$$

Dann ist $b_i - a_i = (1 - 1/\rho)a_{i-1} = (1 - 1/\rho)\rho^{-(i-1)} = (\rho - 1)\rho^{-i} < Ra_i$. Beispielsweise für $\rho = 2$ erhalten wir folgende Zerlegung von $(0, 1]$:



Das Intervall $(0, a_k]$ können wir wegen

$$\int_0^{a_k} x^{-\alpha}f(x)\,\mathrm{d}x \le \|f\|_{[a,b]}\frac{a_k^{1-\alpha}}{1-\alpha} \xrightarrow{k\to\infty} 0$$

für genügend große k vernachlässigen. Zusammengefasst gilt

Lemma 4.20. *Sei $b-a < Ra$ und $n := \left\lceil \frac{\log(\varepsilon a^\alpha)}{2\log\left(\frac{b-a}{Ra}\right)} - 1 \right\rceil$. So gilt*

$$\left| \int_a^b g(x)\,\mathrm{d}x - Q_n(g) \right| < c(b-a)\varepsilon. \tag{4.9}$$

Ferner gilt für $a \le B(\varepsilon) := ((1-\alpha)\varepsilon)^{1/(1-\alpha)}$, dass

$$\left| \int_0^a g(x)\,\mathrm{d}x \right| \le \tilde{c}\varepsilon. \tag{4.10}$$

Weil $\int_{a_i}^{b_i} g(x)\,\mathrm{d}x$ durch die Gauss-Legendre Quadraturformel $Q_{[a_i,b_i]}(g)$ auf $[a_i, b_i]$ mit Genauigkeit $O(\varepsilon)$ approximiert wird, definieren wir

$$Q_{(0,1]}(g) := \sum_{i=1}^{k} Q_{[a_i,b_i]}(g).$$

Die Bedingung $a_k \le B(\varepsilon)$ für k ist äquivalent mit $\rho^{-k} \le B(\varepsilon)$. Daher gilt für

$$k \ge \frac{\log B(\varepsilon)}{\log \frac{1}{\rho}} \sim \frac{1}{1-\alpha} |\log \varepsilon|$$

nach (4.9) und (4.10) die gewünschte Fehlerabschätzung

$$\begin{aligned} \left| \int_0^1 g(x)\,\mathrm{d}x - Q_{(0,1]}(g) \right| &\le \left| \int_0^{a_k} g(x)\,\mathrm{d}x \right| + \sum_{i=1}^{k} \left| \int_{a_i}^{b_i} g(x)\,\mathrm{d}x - Q_{[a_i,b_i]}(b) \right| \\ &\le \tilde{c}\varepsilon + c(\rho-1)\varepsilon \sum_{i=1}^{k} \rho^{-i} \le (\tilde{c}+c)\varepsilon. \end{aligned}$$

Weil $(0,1]$ in $k \sim |\log \varepsilon|$ Intervalle zerlegt wird, für jedes von denen eine Quadraturformel mit

$$n_i \sim |\log(\varepsilon \rho^{-\alpha i})| = |\log \varepsilon| + \alpha i \left| \log \frac{1}{\rho} \right|, \quad i = 1, \dots, k,$$

Stützstellen verwendet wird, genügen

$$\sum_{i=0}^{k} n_i \sim k|\log \varepsilon| + \alpha \left| \log \frac{1}{\rho} \right| \sum_{i=1}^{k} i \sim k|\log \varepsilon| + k^2 \sim |\log \varepsilon|^2$$

Auswertungen von g. Daher kann trotz der Singularität eine exponentielle Konvergenz der Quadraturformel garantiert werden. Dies wird durch passende Verkleinerung der Intervallbreite $h = b_i - a_i$ und gleichzeitiger Erhöhung des Grades der Quadraturformel $p = n_i$ erreicht. Von diesem Prinzip ist der Name **hp-Quadratur** abgeleitet.

4.3 Hierarchische Quadratur

Für $n \in \mathbb{N}$ betrachte den Raum der stückweise linearen Funktionen

$$V_n := \{f \in C[0,1] : f|_{[k/n,(k+1)/n]} \in \Pi_1 \text{ für } k = 0, 1, \ldots, n-1\}.$$

Diese Funktionen sind also zwischen den Gitterpunkten $x_i = i/n$, $i = 0, \ldots, n$, stückweise linear. Die Punkte sind äquidistant mit Gitterweite $1/n$.

In folgendem Lemma beschreiben wir eine Basis von V_n, die sich durch Verschiebung und Stauchung der sog. **Hutfunktion**

$$\varphi(x) := \begin{cases} 1 - |x|, & x \in [-1, 1], \\ 0, & \text{sonst}, \end{cases}$$

darstellen lässt.

Lemma 4.21. *Die Funktionen*

$$\varphi_i^{(n)}(x) := \varphi(i - nx)|_{[0,1]}, \quad 0 \le i \le n,$$

bilden die sog. ***nodale Basis*** *des Raums V_n.*

Eine Funktion $f \in C[0,1]$ kann mit Hilfe der nodalen Basis von V_n approximiert werden:

$$f \approx f_n := \sum_{i=0}^{n} \alpha_i \varphi_i^{(n)} \in V_n \quad \text{mit } \alpha_i = f(x_i).$$

Daher gilt

$$I(f) = \int_0^1 f(x)\,\mathrm{d}x \approx Q_n(f) := I(f_n) = \sum_{i=0}^{n} \alpha_i \int_0^1 \varphi_i^{(n)}(x)\,\mathrm{d}x.$$

Wegen

$$\int_0^1 \varphi_i^{(n)}(x)\,\mathrm{d}x = \begin{cases} 1/n, & 0 < i < n, \\ 1/(2n), & \text{sonst}, \end{cases}$$

stimmt Q_n mit der Trapezsumme

$$T_n(f) = \frac{1}{2n}\left(f(0) + 2\sum_{i=1}^{n-1} f(x_i) + f(1)\right)$$

überein.

Man beachte, dass die nodalen Basisfunktionen $\varphi_i^{(n)}$, $i = 0, \ldots, n$, überlappende Träger besitzen. Sei $n = 2^L$ mit $L \in \mathbb{N}$. Eine weitere Basis, die **hierarchische Basis**, setzt sich aus nicht-überlappenden Basisfunktionen zu verschiedenen Gitterweiten $h_\ell := 2^{-\ell}$, $\ell = 0, \ldots, L$, zusammen.

Lemma 4.22. *Sei*

$$\nabla_\ell = \begin{cases} \{0,1\}, & \ell = 0, \\ \{1,3,5,\ldots,2^\ell - 1\}, & \ell > 0. \end{cases}$$

Dann bildet

$$\{\varphi_{\ell i} := \varphi_i^{(2^\ell)},\ i \in \nabla_\ell,\ 0 \le \ell \le L\}$$

die hierarchische Basis von V_{2^L}.

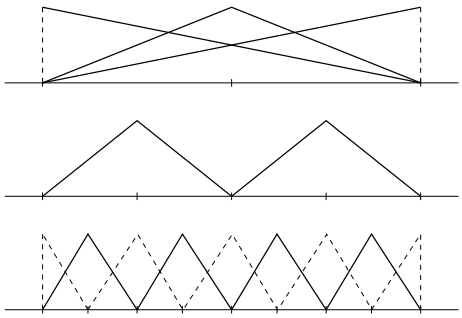


Abbildung 4.1: Nodale und hierarchische Basis

Man überzeugt sich leicht davon, dass folgende Beziehung zwischen den Basisfunktionen der Level ℓ und $\ell+1$ gilt:

$$\varphi_{\ell i} = \frac{1}{2}\varphi_{\ell+1,2i+1} + \varphi_{\ell+1,2i} + \frac{1}{2}\varphi_{\ell+1,2i-1}, \quad i = 1,\ldots,2^\ell - 1,$$

und am Rand

$$\varphi_{\ell,0} = \frac{1}{2}\varphi_{\ell+1,1} + \varphi_{\ell+1,0}, \quad \varphi_{\ell,2^\ell} = \frac{1}{2}\varphi_{\ell+1,2^{\ell+1}-1} + \varphi_{\ell+1,2^{2\ell}}.$$

In der hierarchischen Basis besitzt f_{2^L} daher die Darstellung

$$f_{2^L} = \sum_{\ell=0}^{L} \sum_{i \in \nabla_\ell} \beta_{\ell i}\, \varphi_{\ell i} = f_{2^{L-1}} + \sum_{i \in \nabla_L} \beta_{Li}\, \varphi_{Li}$$

mit

$$\beta_{\ell i} = \begin{cases} f(i), & \ell = 0, \\ f(ih_\ell) - \frac{1}{2}\left[f((i-1)h_\ell) + f((i+1)h_\ell)\right], & \ell > 0. \end{cases}$$

Die Koeffizienten werden als **hierarchische Überschüsse** bezeichnet. Das folgende Lemma stellt den Bezug zwischen den hierarchischen Überschüssen und dem zentralen Differenzenquotienten

$$Z_h(f)[x] := \frac{f(x-h) - 2f(x) + f(x+h)}{h^2}, \quad h > 0,$$

her.

Lemma 4.23. *Für $f \in C^2[0,1]$ gilt $|\beta_{\ell i}| \leq \frac{1}{2} h_\ell^2 \|f''\|_{[x_{i-1}, x_{i+1}]}$ für $i \in \nabla_\ell$ und $\ell > 0$.*

Beweis. Sei $h > 0$, so dass $x - h \geq 0$ und $x + h \leq 1$. Wegen des Hauptsatzes der Differential- und Integralrechnung gilt

$$f(x \pm h) = f(x) \pm \int_0^h f'(x \pm t) \, \mathrm{d}t.$$

Addition beider Gleichungen und erneute Anwendung des Hauptsatzes liefert

$$\begin{aligned} h^2 Z_h(f)[x] &= \int_0^h f'(x+s) - f'(x-s) \, \mathrm{d}s \\ &= \int_0^h [f'(x+s) - f'(x) - [f'(x-s) - f'(x)] \, \mathrm{d}s \\ &= \int_0^h \int_0^s f''(x+t) - f''(x-t) \, \mathrm{d}t \, \mathrm{d}s \\ &= \int_0^h (h-t)[f''(x+t) - f''(x-t)] \, \mathrm{d}t. \end{aligned}$$

Die Behauptung folgt aus

$$\beta_{\ell i} = -\frac{1}{2} h_\ell^2 Z_{h_\ell}(f)[i h_\ell].$$

□

Daher fallen die hierarchischen Überschüsse bei genügender Glattheit von f mit wachsendem ℓ schnell ab. Sind die Überschüsse groß, so ist dies umgekehrt ein Indikator dafür, dass die Funktion an dieser Stelle nicht glatt ist.

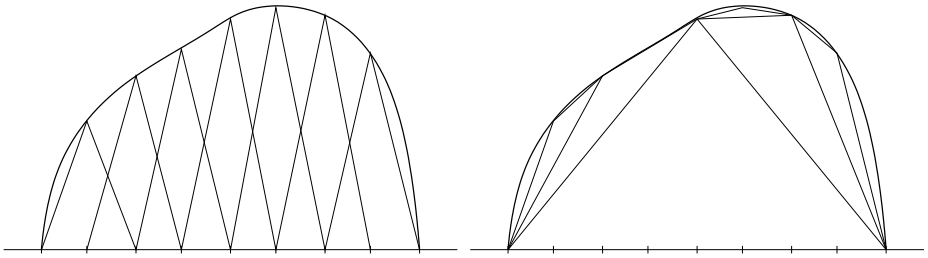


Abbildung 4.2: Nodale und hierarchische Interpolanten

Unter Verwendung der hierarchischen Basis erhalten wir die Quadraturformel

$$Q_n(f) := I(f_{2^L}) = \sum_{\ell=0}^{L} \sum_{i \in \nabla_\ell} \beta_{\ell i} \int_0^1 \varphi_{\ell i}(x) \, \mathrm{d}x = \frac{1}{2} f(0) + \frac{1}{2} f(1) + \sum_{\ell=1}^{L} \sum_{i \in \nabla_\ell} h_\ell \beta_{\ell i}.$$

Der folgende Algorithmus basiert auf dem Zusammenhang zwischen Glattheit und Abfallen der hierarchischen Überschüsse.

Algorithmus 4.24 (Hierarchische Quadratur).
`Input:` $f \in C[0,1]$ und Fehlertoleranz $\varepsilon > 0$
`Output:` Approximation $Q \approx I(f)$

Setze $Q := (f(0) + f(1))/2$ und $(x, h) = (1/2, 1/2)$;
`L1` berechne $E(x,h) := -\frac{1}{2}h^2 Z_h(f)[x]$;
`if` $|E(x,h)| > \varepsilon$ `then`
$Q := Q + hE(x,h)$;
gehe zu `L1` mit $(x,h) := (x - h/2, h/2)$;
gehe zu `L1` mit $(x,h) := (x + h/2, h/2)$;

4.4 Tensorprodukt-Quadratur

Gegeben sei $f : [0,1]^2 \to \mathbb{R}$ analytisch. Das Ziel dieses Abschnitts ist es, Kubaturformeln (im Mehrdimensionalen werden Quadraturformeln als Kubaturformeln bezeichnet) zur Approximation des zweidimensionalen Integrals

$$\int_0^1 \int_0^1 f(x,y)\,\mathrm{d}y\,\mathrm{d}x$$

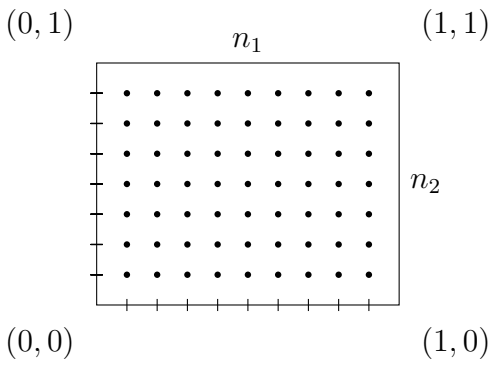
zu konstruieren. Dazu seien für $g : [0,1] \to \mathbb{R}$

$$Q_1(g) = \sum_{i=0}^{n_1} w_i^{(1)} g(x_i) \quad \text{und} \quad Q_2(g) = \sum_{j=0}^{n_2} w_j^{(2)} g(y_j)$$

zwei eindimensionale Quadraturformeln mit Gewichten $w_i^{(1)}$, $w_j^{(2)} \geq 0$ und Stützstellen x_i, y_j. Wir definieren

$$Q(f) := Q_1(Q_2(f_x)) = \sum_{i=0}^{n_1} \sum_{j=0}^{n_2} w_i^{(1)} w_j^{(2)} f(x_i, y_j) = Q_2(Q_1(f_y)).$$

Dabei bezeichnet $f_x : [0,1] \to \mathbb{R}$ die Funktion $f_x(y) := f(x,y)$, $y \in [0,1]$, bei festem $x \in [0,1]$. Die Quadraturformel Q wird als **Tensorprodukt-Quadraturformel** bezeichnet. Die Bezeichnung resultiert aus der Struktur der Gitterpunkte, in denen f ausgewertet wird.



Satz 4.25. *Es gilt die Fehlerabschätzung*

$$\left|\int_0^1\int_0^1 f(x,y)\,\mathrm{d}y\,\mathrm{d}x - Q(f)\right| \le \sup_{x\in[0,1]}|E_2(f_x)| + \sup_{y\in[0,1]}|E_1(f_y)|$$

mit $E_1(g) := \left|\int_0^1 g(x)\,\mathrm{d}x - Q_1(g)\right|$ *und* $E_2(g) = \left|\int_0^1 g(y)\,\mathrm{d}y - Q_2(g)\right|$.

Beweis.

$$\begin{aligned}
&\int_0^1\int_0^1 f(x,y)\,\mathrm{d}y\,\mathrm{d}x - Q(f) = \int_0^1\int_0^1 f(x,y)\,\mathrm{d}y\,\mathrm{d}x - \int_0^1 Q_2(f_x)\,\mathrm{d}x + \int_0^1 Q_2(f_x)\,\mathrm{d}x - Q(f)\\
&= \int_0^1 \underbrace{\left[\int_0^1 f(x,y)\,\mathrm{d}y - \sum_{j=0}^{n_2} w_j^{(2)} f(x,y_j)\right]}_{\le E_2(f_x)}\,\mathrm{d}x + \sum_{j=0}^{n_2} w_j^{(2)} \underbrace{\left(\int_0^1 f(x,y_j)\,\mathrm{d}x - \sum_{i=0}^{n_1} w_i^{(1)} f(x_i,y_j)\right)}_{\le E_1(f_{y_j})}\\
&\le \sup_{x\in[0,1]}|E_2(f_x)| + \sup_{y\in[0,1]} E_1(f_y)\cdot\sum_{j=0}^{n_2} w_j^{(2)}.
\end{aligned}$$

Wegen $\sum_{j=0}^{n_2} w_j^{(2)} = 1$ folgt die Behauptung. □

Diese Konstruktion und Satz 4.25 können leicht auf d-dimensionale Integrale, d.h. Integrale über dem Einheitswürfel $[0,1]^d$ verallgemeinert werden. Man beachte, dass der Kubaturfehler wie in Satz 4.25 von der Ordnung des eindimensionalen Quadraturfehlers ist, die Anzahl der Stützstellen aber exponentiell mit der Dimension wächst. Für große d wird das Verhältnis von Genauigkeit zu Aufwand daher schlechter. Man spricht vom **Fluch der Dimensionen**.

4.5 Monte-Carlo-Quadratur

Im folgenden Abschnitt stellen wir die Standard-Methode vor, um den Fluch der Dimensionen zu umgehen. Dazu sei $\{X_i\}_{i\in\mathbb{N}}$ eine Folge von auf $[0,1]^d$ gleichverteilter und unabhängiger Zufallsvariablen. Für die gleichverteilte Zufallsgröße X erhält man für den Erwartungswert von $f(X)$

$$E[f(X)] = \int_{[0,1]^d} f(x)\,\mathrm{d}x =: I(f).$$

Wählt man daher zufällig n Stützstellen $x_1,\dots,x_n \in [0,1]^d$ und setzt

$$Q_n(f) := \frac{1}{n}\sum_{i=1}^n f(x_i),$$

so liefert Q_n im Mittel das richtige Ergebnis, weil

$$E[Q_n(f)] = \frac{1}{n}\sum_{i=1}^n E[f(X_i)] = \frac{1}{n}\sum_{i=1}^n \int_{[0,1]^d} f(x)\,\mathrm{d}x = \int_{[0,1]^d} f(x)\,\mathrm{d}x.$$

Der Kubaturfehler $I(f) - Q_n(f)$ kann beliebig groß sein. Mit wachsendem n wird aber die mittlere Abweichung $E[|I(f) - Q_n(f)|]$ klein.

Satz 4.26. *Sei* $g(x) := I(f) - f(x)$ *und*

$$v_f := \int_{[0,1]^d} g^2(x)\,\mathrm{d}x$$

die Varianz von f*. Dann gilt*

$$E[|I(f) - Q_n(f)|] \leq \sqrt{\frac{v_f}{n}}.$$

Beweis. Für die mittlere quadratische Abweichung gilt zunächst

$$\int_{[0,1]^d} g(x)\,\mathrm{d}x = 0$$

und somit

$$\begin{aligned}
E[|I(f) - Q_n(f)|^2] &= E\left[\left|I(f) - \frac{1}{n}\sum_{i=1}^n f(X_i)\right|^2\right] = E\left[\left|\frac{1}{n}\sum_{i=1}^n (I(f) - f(X_i))\right|^2\right] \\
&= \frac{1}{n^2}\int_{[0,1]^d}\dots\int_{[0,1]^d}\left(\sum_{i=1}^n g(x_i)\right)^2 \mathrm{d}x_n \dots \mathrm{d}x_1 \\
&= \frac{1}{n^2}\sum_{i=1}^n \int_{[0,1]^d}\dots\int_{[0,1]^d} g^2(x_i)\,\mathrm{d}x_n\dots\mathrm{d}x_1 + \frac{1}{n^2}\sum_{i\neq j}\int_{[0,1]^d}\dots\int_{[0,1]^d} g(x_i)g(x_j)\,\mathrm{d}x_n\dots\mathrm{d}x_1 \\
&= \frac{1}{n^2}\sum_{i=1}^n \underbrace{\int_{[0,1]^d} g^2(x_i)\,\mathrm{d}x_i}_{=v_f} + \frac{1}{n^2}\sum_{i\neq j}\underbrace{\int_{[0,1]^d} g(x_i)\,\mathrm{d}x_i}_{=0}\underbrace{\int_{[0,1]^d} g(x_j)\,\mathrm{d}x_j}_{=0} \\
&= \frac{1}{n^2}\sum_{i=1}^n \int_{[0,1]^d} g^2(x_i)\,\mathrm{d}x_i = \frac{v_f}{n}.
\end{aligned}$$

Aus der Hölder-Ungleichung

$$E^2[|I(f) - Q_n(f)|] \leq E[|I(f) - Q_n(f)|^2]$$

folgt die Behauptung. □

Die Größe v_f drückt die Varianz der Funktion f aus. Sie ist unabhängig von n. Daher ist die Konvergenzrate der Monte-Carlo-Quadratur $1/\sqrt{n}$. Diese ist zwar nicht sehr hoch und die Konvergenz liegt nur bzgl. eines schwachen Maßes vor, sie ist aber unabhängig von der Dimension d.

Algorithmus 4.27 (Monte-Carlo-Quadratur).
`Input:` $f \in C[0,1]^d$ und $n \in \mathbb{N}$
`Output:` $Q_n(f) \approx I(f)$

Setze $Q_n(f) = 0$;
`for` $i = 1, \ldots, n$
    erzeuge Zufallsvektor $x_i \in [0,1]^d$ durch d-maliges Auswerten eines Zufallsgenerators;
    $Q_n(f) := Q_n(f) + \frac{1}{n} f(x_i)$;

Die Kosten zur Berechnung von $Q_n(f)$ sind dabei von der Ordnung $O(dn)$. Die Komplexität wächst also linear statt wie bei den Tensor-Produkt-Kubaturregeln exponentiell.

Index