Numerische Mathematik

(Wintersemester 2003/04, 2004/05, 2005/06, 2010/11)

Dirk Praetorius
Institut für Analysis und Scientific Computing
Technische Universität Wien

Vorwort

Bei dem vorliegenden Skript handelt es sich um die Ausarbeitung der Vorlesung *Numerische Mathematik*, die ich im Wintersemester 2003/04, 2004/05 und 2005/06 an der Technischen Universität Wien gelesen habe. Die Ausarbeitung in LaTeX erfolgte im Wesentlichen parallel zur Vorlesung im Wintersemester 2004/05, und es haben sich sicherlich noch etliche kleinere Tippfehler eingeschlichen. Für Anmerkungen zu Fehlern und Tippfehlern bin ich dankbar.

Mein Dank gilt Frau Claudia Steinwender, die die Abschnitte über die Lagrange-Interpolation ausgearbeitet hat. Herr Stephan Krenn hat eine erste Vorlesungsmitschrift in LaTeX erstellt und mir dadurch sehr viel Arbeit abgenommen. Herrn Alexander Schiftner danke ich für das ausführliche Korrekturlesen der bereits ausgearbeiteten Abschnitte. Auch Herr Michael Szell war so nett, mir seine Anmerkungen zugänglich zu machen.

Im Wintersemester 2005/06 sind mir besonders Frau Satke, Herr Dörsek und Herr Morgenbesser im Gedächtnis geblieben, die mich auf Fehler im Skript hingewiesen haben.

Im Zuge des Wintersemesters 2010/11 habe ich noch einige Tippfehler gefunden und eliminiert. Desweiteren wurden die Abschnitte über Cholesky-Zerlegung und Hermite-Interpolation in die Übung „ausgelagert“.

Dieses Skript ist weiterhin „nur” eine vorläufige Endversion. Eigentlich wollte ich noch MATLAB-Implementierungen und numerische Beispiele zu den einzelnen Kapiteln erstellen, sodass (möglichst) alle vorgestellten numerischen Verfahren mit Experimenten und Anmerkungen zur Realisierung versehen sind. Ich scheiterte aber an meinen eigenen Ansprüchen. Da völlig ungewiss ist, wann ich diese Vorlesung voraussichtlich wieder lese, soll diese Version des Skriptes wenigstens den aktuellen Stand der Korrekturen wiedergeben. `;-)`

`last modified: 26. Dezember 2010`

Inhaltsverzeichnis

Kapitel 1

Grundbegriffe der Numerischen Mathematik

1.1 Gegenstand der Numerischen Mathematik

Von der Realität bis zur Interpretation einer Simulation ist ein weiter Weg.

- Zunächst wird versucht, die Realität mit Hilfe mathematischer Formeln zu beschreiben. Es entsteht das sogenannte **mathematische Modell**. Dies liegt bei physikalischen Problemen in der Regel in Form einer oder mehrerer Differentialgleichungen mit Nebenbedingungen vor.
- Die wenigsten Differentialgleichungen können analytisch gelöst werden. Man muss also durch eine **numerische Simulation** eine Lösung approximieren.
- Aus der numerischen Simulation erhalten wir eine **numerische Lösung**, die dann geeignet interpretiert werden muss.

Regelmäßig zerfällt die numerische Simulation in kleinere **numerische Probleme**, die geeignet zu lösen sind. Die elementarsten numerischen Probleme sind Gegenstand dieser Einführungsveranstaltung.

Jede numerische Simulation ist fehlerbehaftet. Wir unterscheiden dabei wie folgt:

- **Modellfehler**: Regelmäßig ist das mathematische Modell eine Vereinfachung der Realität.
- **Datenfehler**: Die Eingangsdaten der numerischen Simulation erhält man durch Messungen physikalischer Größen. Jede Messung unterliegt einer gewissen Genauigkeit.
- **Rundungsfehler**: Auf Rechnern ersetzen die sogenannten Gleitkommazahlen das Kontinuum $\mathbb{R}$. Da die Menge aller Gleitkommazahlen *endlich* ist, muss es sowohl bei der Eingabe der Eingangsdaten als auch bei der internen Rechnung zu Rundungsfehlern kommen.
- **Verfahrensfehler**: Viele Probleme werden mathematisch in unendlich-dimensionalen Räumen oder mit Limes-Begriffen formuliert. Beides steht im Rechner nicht zur Verfügung und muss daher geeignet diskretisiert werden. Dies führt zu zusätzlichen Fehlern.

In der Vorlesung werden wir uns nur mit den Rundungsfehlern und den Verfahrensfehlern für elementare numerische Probleme beschäftigen. Unter einem **numerischen Problem** verstehen wir dabei eine elementare mathematische Aufgabe, deren Lösung aus der Berechnung von Zahlen besteht, z.B. das Lösen eines Gleichungssystems $Ax = b$ mit regulärer Matrix $A \in \mathbb{K}^{n\times n}$. Ein **Algorithmus** ist eine schematische Methode, um die Lösung eines numerischen Problems zu berechnen, z.B. das Gauß-Verfahren zur Lösung von $Ax = b$. Der **Aufwand eines Algorithmus** beschreibt den Speicherbedarf und die Anzahl an benötigten mathematischen Operationen. Ziel der **Numerischen Mathematik** ist die Entwicklung von Algorithmen, die mit möglichst geringem Aufwand eine möglichst große Klasse numerischer Probleme lösen können.

1.2 Gleitkommazahlen und Rundungsfehler

Die Definition der Gleitkommazahlen erfordert den folgenden Satz aus der Grundvorlesung zur Analysis, siehe z.B. FORSTER [3, Kapitel 5].

Satz 1.1. *Zu fixierter* **Basis** $b \in \mathbb{N}_{\geq 2}$ *und gegebenem* $x \in \mathbb{R}\backslash\{0\}$ *existieren ein* **Vorzeichen** $\sigma \in \{\pm 1\}$, **Ziffern** $a_j \in \{0, 1, \ldots, b-1\}$ *und ein* **Exponent** $e \in \mathbb{Z}$ *mit*

$$x = \Big(\sigma \sum_{k=1}^{\infty} a_k b^{-k}\Big) b^e \quad \textit{und} \quad a_1 \neq 0. \tag{1.1}$$

Man bezeichnet diese Darstellung als **normalisierte Gleitkommadarstellung** *oder* **b-adische Darstellung** *von* x. ■

Bemerkung. Man beachte, dass die Darstellung aufgrund periodischer Darstellungen nicht eindeutig ist. Es gilt beispielsweise $1 = 0.\overline{9}$. □

Beispiel (Dezimalsystem $b = 10$). Unsere alltägliche Zahldarstellung basiert auf dem Dezimalsystem. Die Zahl 147.4 besitzt gemäß Satz 1.1 eine Darstellung $147.4 = 0.1474 \cdot 10^3$. Es gelten also $\sigma = 1$, $a_1 = 1$, $a_2 = 4$, $a_3 = 7$, $a_4 = 4$, $b = 10$ und $e = 3$. ■

Beispiel (Binärsystem $b = 2$). Auf Computern werden Zahlen regelmäßig im Binärsystem dargestellt. Dies hat Konsequenzen. So ist z.B. die Zahl 1/10 im Binärsystem nur durch eine unendliche Reihe darstellbar. ■

Definition. Zu gegebener **Basis** $b \in \mathbb{N}_{\geq 2}$, **Mantissenlänge** $t \in \mathbb{N}$ und **Exponentialschranken** $e_{\min} < 0 < e_{\max}$ definieren wir die Menge der **normalisierten Gleitkommazahlen** $\mathbb{F} := \mathbb{F}(b, t, e_{\min}, e_{\max}) \subset \mathbb{R}$ durch

$$\mathbb{F} = \{0\} \cup \Big\{ \Big(\sigma \sum_{k=1}^{t} a_k b^{-k}\Big) b^e \;\Big|\; \sigma \in \{\pm 1\}, a_j \in \{0, \ldots, b-1\}, a_1 \neq 0, e \in \mathbb{Z}, e_{\min} \leq e \leq e_{\max} \Big\}.$$

Die endliche Summe $a = \sum_{k=1}^{t} a_k b^{-k}$ bezeichnet man als **(normalisierte) Mantisse** einer Gleitkommazahl. □

Dadurch, dass wir bei der Definition von Gleitkommazahlen die unendliche Reihe durch eine endliche Summe ersetzen, sind periodische Darstellungen ausgeschlossen. Man überlegt sich leicht, dass

damit die Darstellung einer normalisierten Gleitkommazahl eindeutig ist. Ferner ist $\mathbb{F}$ offensichtlich eine *endliche Teilmenge* von $\mathbb{R}$. Der folgende Satz stellt die wesentlichen Eigenschaften von $\mathbb{F}$ zur Verfügung.

Satz 1.2. *Für $\mathbb{F} := \mathbb{F}(b, t, e_{\min}, e_{\max})$ gelten die folgenden Aussagen:*
(i) *Zu $x \in \mathbb{F}\backslash\{0\}$ existieren* eindeutig *das Vorzeichen $\sigma \in \{\pm 1\}$, die Mantisse $a = \sum_{k=1}^{t} a_k b^{-k}$ und der Exponent e mit $x = \sigma a b^e$.*
(ii) $x_{\min} := \min\{x \in \mathbb{F} \mid x > 0\} = b^{e_{\min}-1}$, $x_{\max} := \max\{x \in \mathbb{F} \mid x > 0\} = b^{e_{\max}}(1 - b^{-t})$.
(iii) *Für $e_{\min} \le e \le e_{\max}$ und $M := b^t - b^{t-1} \in \mathbb{N}$ gilt*

$$\mathbb{F} \cap [b^{e-1}, b^e) = \{(b^{-1} + jb^{-t})b^e \mid j = 0, \ldots, M-1\}.$$

(iv) *Für $x \in \mathbb{R}$ mit $x_{\min} \le |x| \le x_{\max}$ gilt*

$$\min_{z \in \mathbb{F}} \frac{|z - x|}{|x|} \le \text{eps} := \frac{1}{2} b^{1-t}.$$

Die Zahl eps *heißt* **Maschinengenauigkeit**.

Beweis. (i) sei dem Leser zur Übung überlassen. (ii) Für eine Mantisse $a = \sum_{k=1}^{t} a_k b^{-k}$ gilt

$$b^{-1} \le a \le \sum_{k=1}^{t} (b-1)b^{-k} = \sum_{k=1}^{t} (b^{-k+1} - b^{-k}) = 1 - b^{-t}.$$

Multiplikation mit $b^{e_{\min}}$ bzw. $b^{e_{\max}}$ zeigt die Behauptung. (iii) Für die erste Ziffer $a_1 \in \{1, \ldots, b-1\}$ gibt es nach Definition $(b-1)$ verschiedene Möglichkeiten, für $a_2, \ldots, a_t \in \{0, \ldots, b-1\}$ jeweils b viele Möglichkeiten. Also gibt es insgesamt $(b-1)b^{t-1} = b^t - b^{t-1} = M$ verschiedene Mantissen. Der Abstand zweier benachbarter Mantissen ist gerade b^{-t}, d.h. es gilt $a = b^{-1} + jb^{-t}$ für ein $j = 0, \ldots, M-1$. (iv) Ohne Beschränkung der Allgemeinheit gilt $x \ge 0$ sowie $x \in [b^{e-1}, b^e]$ mit $e_{\min} \le e \le e_{\max}$. Die Gleitkommazahlen $z \in \mathbb{F} \cap [b^{e-1}, b^e]$ liegen gleichverteilt mit Abstand b^{e-t}. Also existiert ein $z \in \mathbb{F}$ mit $|z - x| \le \frac{1}{2} b^{e-t}$. Mit $|x| \ge b^{e-1}$ folgt die Behauptung. ■

Insgesamt ist $\mathbb{F}$ eine endliche Teilmenge von $\widetilde{\mathbb{F}} := [-x_{\max}, -x_{\min}] \cup \{0\} \cup [x_{\min}, x_{\max}]$. Um die Null klafft eine Lücke. Die Abstände zwischen den Gleitkommazahlen nehmen mit dem Betrag zu, der relative Fehler bleibt aber beschränkt.

Bemerkung. Für $x, y \in \mathbb{R}$ bezeichnet man $|x - y|$ als **absoluten Fehler** von y zu x. Ist $x \neq 0$, so bezeichnet man $|x-y|/|x|$ als **relativen Fehler** von y zu x. Die Maschinengenauigkeit eps gibt den maximalen relativen Fehler, wenn eine reelle Zahl $x \in \widetilde{F}$ in eine Gleitkommazahl (durch Rundung) konvertiert wird. □

Für die Menge $\widetilde{\mathbb{F}}$ definieren wir die Rundung $\text{rd} : \widetilde{\mathbb{F}} \to \mathbb{F}$ implizit durch

$$|x - \text{rd}(x)| = \min_{z \in \mathbb{F}} |x - z|, \tag{1.2}$$

wobei $\text{rd}(x)$ das betragsgrößere $z \in \mathbb{F}$ sei, falls die Minimalstelle nicht eindeutig ist. Nach Satz 1.2 existiert zu $x \in \widetilde{F}\backslash\{0\}$ ein $\varepsilon \in \mathbb{R}$ mit $|\varepsilon| \le \text{eps}$ und $\text{rd}(x) = x(1 + \varepsilon)$, denn Umformung zeigt $\varepsilon = (\text{rd}(x) - x)/x$ und daher $|\varepsilon| \le \text{eps}$.

Generalvoraussetzung. Für die Rundungsfehleranalyse machen wir die folgende *idealisierte Annahme:* Die arithmetischen Operationen ($+$, $-$, $\cdot$, $:$) sowie alle mathematischen Funktionen (z.B. sin, cos, $\sqrt{\cdot}$) werden im Rechner *exakt durchgeführt* und liefern als Gleitkomma-Ergebnis das gerundete exakte Ergebnis, z.B.

$$x \oplus y := \mathrm{rd}(x+y) \quad \text{für alle } x, y \in \mathbb{F} \text{ mit } x+y \in \widetilde{\mathbb{F}},$$

wobei $\oplus$ die Rechneraddition bezeichne und wir stets annehmen, dass kein Überlauf (Betrag des Ergebnis $> x_{\max}$) oder Unterlauf (Betrag des Ergebnis $< x_{\min}$) auftrete.

MATLAB-Beispiel: Rundungsfehler

Die Rechnung `x = (116/100) * 100` liefert augenscheinlich als Ergebnis `x = 116`. Rundet man aber mittels `floor(x)` auf die nächste ganze Zahl nach unten, so erhält man als Ergebnis `ans = 115`. Dies ist kein Phänomen der Funktion `floor`, sondern basiert auf Rundungsfehlern in IEEE *double*-Arithmetik! Eingabe von `floor(116)` liefert als Ergebnis `ans = 116`.

Bemerkung (Nicht-Assoziativität der Rechnerarithmetik). Für die Rechnerarithmetik gilt kein Assoziativgesetz, z.B. gilt für $b = 10$ und $t = 2$

$$100 \oplus 4 = \mathrm{rd}(0.1 \cdot 10^3 + 0.4 \cdot 10^1) = \mathrm{rd}(0.104 \cdot 10^3) = 100.$$

Ferner gilt $100 \oplus (4 \oplus 4) = \mathrm{rd}(0.108 \cdot 10^3) = 110$. Insgesamt erhalten wir also

$$(100 \oplus 4) \oplus 4 = 100 \neq 110 = 100 \oplus (4 \oplus 4).$$ □

Bemerkung (Numerische Konvergenz bei mathematischer Divergenz). Die harmonische Reihe $\sum_{k=1}^{\infty} \frac{1}{k}$ ist divergent. In Gleitkommaarithmetik beobachtet man aber Konvergenz, denn sobald $1/j$ klein genug ist, gilt $\frac{1}{j} \oplus \sum_{k=0}^{j-1} \frac{1}{k} = \sum_{k=0}^{j-1} \frac{1}{k}$. □

Bemerkung (Auslöschung). Subtrahiert man zwei annähernd gleiche Zahlen, so werden die rundungsfehlerbehafteten hinteren Stellen plötzlich signifikant, und der relative Fehler ist wesentlich größer als eps. Wir betrachten das folgende Beispiel:

$$0.1236 + 1.234 - 1.356 = 1.3576 - 1.356 = 0.0016 = 0.16 \cdot 10^{-2}.$$

In Rechnerarithmetik mit $b = 10$ und $t = 4$ gilt

$$(0.1236 \oplus 1.234) \ominus 1.356 = 1.358 \ominus 1.356 = 0.2 \cdot 10^{-2}.$$

Der relative Fehler beträgt also 25%, bereits die erste Ziffer ist fehlerbehaftet. □

Denormalisierte Gleitkommazahlen nach IEEE 754-Standard

Für die meisten Rechner, insbesondere für gewöhnliche PCs, gilt der sogenannte IEEE 754-Standard des *Institute of Electrical and Electronics Engineers*. Dabei wird die Menge $\mathbb{F}$ der Gleitkommazahlen erweitert zu

$$\widehat{\mathbb{F}} := \mathbb{F} \cup \left\{ \left(\sigma \sum_{k=1}^{t} a_k b^{-k} \right) b^{e_{\min}} \,\middle|\, \sigma \in \{\pm 1\}, a_j \in \{0, \ldots, b-1\} \right\},$$

d.h. für den Exponenten $e = e_{\min}$ ist auch $a_1 = 0$ erlaubt. $\widehat{\mathbb{F}}$ heißt **denormalisiertes Gleitkommazahlsystem**. Gegenüber $\mathbb{F}$ gelten die zusätzlichen Eigenschaften

$$\widehat{x_{\min}} := \min \left\{ x \in \widehat{\mathbb{F}} \,\middle|\, x > 0 \right\} = b^{e_{\min}-t} < b^{e_{\min}-1} = x_{\min}.$$

sowie

$$\widehat{\mathbb{F}} \cap [-b^{e_{\min}}, b^{e_{\min}}] = \left\{ j b^{e_{\min}-t} \,\middle|\, j = -b^t, \ldots, b^t \right\}$$

Die Standardformate nach IEEE 754-Standard sind das einfach-genaue Format (oder *single*-Format)

$$\widehat{\mathbb{F}}(2, 24, -125, 128) \quad \text{mit} \quad \text{eps} \approx 0.6 \cdot 10^{-7}$$

sowie das doppelt-genaue Format (oder *double*-Format)

$$\widehat{\mathbb{F}}(2, 53, -1021, 1024) \quad \text{mit} \quad \text{eps} \approx 1.11 \cdot 10^{-16}$$

Beide Formate stehen auch in MATLAB zur Verfügung, wobei Variablen in MATLAB standardgemäß vom Typ *double* sind. – Es sei denn, sie werden explizit deklariert.

Bemerkung (Speicherung von Variablen nach IEEE 754-Standard). Eine Variable vom Typ *single* wird in 4 Bytes, d.h. 32 Bits gespeichert:

- Ein Bit speichert das Vorzeichen.
- Die Mantisse für eine normalisierte Gleitkommazahl benötigt 23 Bit bei 24 Ziffern, denn es gilt $a_1 = 1$.
- Der Exponent wird in den restlichen 8 Bit gespeichert.

Mit 8 Bit lassen sich $2^8 = 256$ verschiedene Exponenten darstellen, tatsächlich werden aber durch $e_{\min} = -125$ und $e_{\max} = 128$ nur 254 verschiedene Exponenten benutzt. Die beiden verbleibenden Kombinationen werden wie folgt verwendet: Die Nullbitfolge signalisiert denormalisierte Gleitkommazahlen, d.h. es gilt $a_1 = 0$. Die Einserbitfolge im Exponenten signalisiert einen Ausdruck der Form *Not a Number*, *Overflow* oder *Underflow*.
Die Speicherung einer Variablen vom Typ *double* erfolgt analog in 8 Bytes, d.h. 64 Bits (1 Bit Vorzeichen, 52 Bit Mantisse, 11 Bit Exponent). Weitere Informationen finden sich beispielsweise in PLATO [5, Kapitel 16]. □

1.3 Kondition und Stabilität

In diesem Abschnitt betrachten wir ein abstraktes numerisches Problem der folgenden Gestalt: *Werte eine Funktion $\phi : X \to Y$ bei $x \in X$ aus.* Dabei sind X und Y geeignete normierte Räume. Die **Kondition eines Problems** gibt Aussagen darüber, wie stark sich Änderungen in x, z.B. unvermeidliche Rundungsfehler, auf das Ergebnis $\phi(x)$ auswirken. Ist $\widetilde{\phi}$ ein Algorithmus zur Berechnung von $\phi(x)$, so erhalten wir als Lösung $\widetilde{\phi}(\widetilde{x})$. Die **Stabilität eines Algorithmus** gibt Aussagen darüber, ob der **Fehler der Berechnung** $\|\phi(x) - \widetilde{\phi}(\widetilde{x})\|$ und der **unvermeidliche Fehler** $\|\phi(x) - \phi(\widetilde{x})\|$ von derselben Größenordnung sind. [1]

Definition. Das Problem ist bezüglich dem absoluten Fehler **schlecht konditioniert**, wenn es eine kleine Störung $\widetilde{x}$ von x gibt, z.B. $\widetilde{x} = \mathrm{rd}(x)$, sodass gilt

$$\|\phi(x) - \phi(\widetilde{x})\| \gg \|x - \widetilde{x}\|. \tag{1.3}$$

In diesem Fall kann also eine kleine Störung in x eine große Störung in $\phi(x)$ bewirken. Analog sagt man, dass Problem sei bezüglich dem relativen Fehler schlecht konditioniert, falls anstelle von (1.3) gilt

$$\frac{\|\phi(x) - \phi(\widetilde{x})\|}{\|\phi(x)\|} \gg \frac{\|x - \widetilde{x}\|}{\|x\|}. \tag{1.4}$$

Anderenfalls bezeichnet man ein Problem als **gut konditioniert** (bezüglich relativem oder absolutem Fehler). □

Bemerkung (Konditionszahlen). Ist die Funktion ϕ differenzierbar in x, d.h. es gilt

$$\phi(x) - \phi(\widetilde{x}) = D\phi(x)(x - \widetilde{x}) + o(\|x - \widetilde{x}\|) \quad \text{für } \widetilde{x} \to x,$$

so beschreibt offensichtlich die Ableitung $D\phi(x) \in L(X, Y)$ die Auswirkung von Fehlern. Oft bezeichnet man daher

$$\kappa_{\mathrm{abs}}(x) := \|D\phi(x)\| \tag{1.5}$$

als **absolute Konditionszahl** und

$$\kappa_{\mathrm{rel}}(x) := \frac{\|D\phi(x)\|\|x\|}{\|\phi(x)\|} \tag{1.6}$$

als **relative Konditionszahl**. Bisweilen werden auch **„partielle" relative Konditionszahlen** $\kappa_{jk} := |\partial_k \phi_j(x)||x_k|/|\phi_j(x)|$ betrachtet, vgl. beispielsweise BROKATE [1, Abschnitt 1]. □

Beispiel (Schlechte Kondition bei Auslöschung). Wir betrachten die Funktion $\phi : \mathbb{R}^2 \to \mathbb{R}, (x, y) \mapsto x - y$ und auf $\mathbb{R}^2$ die euklidische Norm $\|\cdot\|_2$. Dann gilt $\|D\phi\|_2 = \sqrt{2}$ und damit

$$\kappa_{\mathrm{rel}} = \sqrt{2}\,\frac{\|(x, y)\|_2}{|x - y|} \gg 0 \quad \text{für } x \approx y,$$

[1] Es wäre sicherlich förderlich, wenn man die Kondition und die Stabilität an der vom Benutzer gewünschten Genauigkeit festmacht, d.h. wenn man beide Begriffe relativ zur Zielgenauigkeit definiert.

d.h. die Subtraktion zweier annähernd gleicher Zahlen ist bezüglich dem relativen Fehler schlecht konditioniert. ■

Beispiel (Kondition einer Matrix). Es sei $\|\cdot\|$ eine Norm auf $\mathbb{K}^n$. Wir verwenden dieselbe Notation für die **induzierte Operatornorm**

$$\|A\| := \sup_{x\in\mathbb{K}^n\setminus\{0\}} \frac{\|Ax\|}{\|x\|} \quad \text{für eine reguläre Matrix } A \in \mathbb{K}^{n\times n}. \tag{1.7}$$

Die Größe $\kappa := \|A\|\|A^{-1}\| \geq 1$ heißt **Konditionszahl** der Matrix bezüglich $\|\cdot\|$. Betrachtet man als numerisches Problem die Lösung des Gleichungssystems $Ax = b$, so ist κ eine scharfe obere Abschätzung der relativen Konditionszahlen: Definiert man die Lösungsfunktion $\phi : \mathbb{K}^n \to \mathbb{K}^n, b \mapsto A^{-1}b$, so hat ϕ die relative Konditionszahl

$$\kappa_{\text{rel}}(b) = \frac{\|D\phi(b)\|\|b\|}{\|\phi(b)\|} = \frac{\|A^{-1}\|\|Ax\|}{\|x\|} \leq \kappa$$

mit $x = A^{-1}b$. Man kann sich leicht überlegen, dass diese Abschätzung scharf ist, d.h. es gilt $\kappa_{\text{rel}}(b) = \kappa$ für mindestens ein $b \in \mathbb{K}^n$. ■

Definition. Das numerische Problem, $\phi : X \to Y$ bei $x \in X$ auszuwerten, erlaubt i.a. mehrere (oder sogar unendlich viele) Realisierungen. Sei $\widetilde{\phi}$ ein Algorithmus, der ϕ realisiert. Ein Algorithmus ist **instabil** bezüglich dem relativen Fehler, falls es eine Störung $\widetilde{x}$ von x gibt, sodass

$$\frac{\|\widetilde{\phi}(\widetilde{x}) - \phi(x)\|}{\|\phi(x)\|} \gg \frac{\|\phi(\widetilde{x}) - \phi(x)\|}{\|\phi(x)\|}.$$

Anderenfalls bezeichnet man einen Algorithmus als **stabil**. □

Beispiel. Wir betrachten das Problem, die Funktion $\phi(x) = \frac{1}{x} - \frac{1}{x+1}$ für große $x \in \mathbb{R}$ auszuwerten. Mögliche Algorithmen sind neben der „naiven" direkten Realisierung $\widetilde{\phi}_1(x) = \frac{1}{x} - \frac{1}{x+1}$, die Berechnung von $\phi(x)$ mittels $\widetilde{\phi}_2(x) = \frac{1}{x(x+1)}$. Bei Algorithmus $\widetilde{\phi}_1$ erwarten wir Instabilität aufgrund von Auslöschungseffekten. Algorithmus $\widetilde{\phi}_2$ erweist sich als stabil.

(Lineare) Vorwärtsanalyse

Wir brauchen eine mathematische Entscheidungsgrundlage, ob ein Algorithmus stabil oder instabil bezüglich unvermeidlichen Rechenfehlern ist. Eine Möglichkeit ist die sogenannte (lineare) Vorwärtsanalyse. Dabei handelt es sich in gewissem Sinne um ein Kochrezept für eine Worst-Case-Analyse:

- Erweitere jeden Rechenschritt um einen $(1+\varepsilon)$-Term mit $|\varepsilon| \leq \text{eps}$, denn nach unserer Voraussetzung liefert die Gleitpunktrechnung das gerundete exakte Ergebnis. Man ersetzt also beispielsweise $x+y$ durch $x \oplus y = (x+y)(1+\varepsilon)$ oder $\sqrt{x}$ durch $\sqrt{x}(1+\varepsilon)$ und erhält insgesamt eine Formel für den Algorithmus in Gleitkommarechnung.

- Wenn $(1+\varepsilon)$-Terme Argumente von Funktionen sind, verwende man die Linearisierung der Funktion mittels Taylor-Formel,

  $$f(x+h) = f(x) + f'(x)h + \mathcal{O}(h^2) \doteq f(x) + f'(x)h,$$

  wobei das Symbol $\doteq$ die Terme höherer Ordnung vernachlässigt (sog. *Gleichheit in erster Näherung*). Beispielsweise gilt $\sqrt{1+\varepsilon} \doteq 1 + \varepsilon/2$ oder $\frac{1}{1+\varepsilon} \doteq 1 - \varepsilon$.

- Höhere Potenzen von ε-Termen werden vernachlässigt, z.B. $(1+\varepsilon_1)(1-\varepsilon_2) \doteq 1 + \varepsilon_1 - \varepsilon_2$.

Dieses Vorgehen liefert (in erster Näherung) eine Formel für $\widetilde{\phi}(\widetilde{x})$. Der Vergleich mit $\phi(x)$ zeigt Stabilität oder Instabilität.

Als Anwendung für die Vorwärtsanalyse verwenden wir die Auswertung der Funktion $\phi(x) = \frac{1}{x} - \frac{1}{x+1}$ für große $x \in \mathbb{R}$.

Behauptung. Die Auswertung von $\phi(x)$ ist gut konditioniert (bzgl. dem relativen Fehler)

Beweis. Es gelten $\phi(x) = \frac{1}{x(x+1)}$ sowie $\phi'(x) = -\frac{2x+1}{(x^2+x)^2}$, und damit ergibt sich

$$\kappa_{\mathrm{rel}}(x) = \frac{\left|\frac{2x+1}{(x^2+x)^2}\right||x|}{\frac{1}{x^2+x}} = \frac{2x^2+x}{x^2+x} = \frac{2+\frac{1}{x}}{1+\frac{1}{x}} \approx 2 \quad \text{für große } x \gg 0.$$

Insbesondere gilt für den unvermeidlichen Fehler $\frac{|\phi(\widetilde{x})-\phi(x)|}{|\phi(x)|} \approx \frac{|\widetilde{x}-x|}{|x|} \leq \mathrm{eps}$ für $\widetilde{x} = \mathrm{rd}(x)$. ■

Behauptung. Naive Realisierung $\widetilde{\phi}_1(x) = \frac{1}{x} - \frac{1}{x+1}$ ist instabil im Sinne der Vorwärtsanalyse.

Beweis. Gleitkommarechnung mit Algorithmus $\widetilde{\phi}_1$ liefert als Ergebnis

$$\widetilde{\phi}_1(\widetilde{x}) = \Big\{\frac{1+\varepsilon_2}{x(1+\varepsilon_1)} - \frac{1+\varepsilon_4}{\{x(1+\varepsilon_1)+1\}(1+\varepsilon_3)}\Big\}(1+\varepsilon_5)$$

mit $\varepsilon_j \in \mathbb{R}$, $|\varepsilon_j| \leq \mathrm{eps}$, wobei wir die fünf Rechenschritte bis zum Endergebnis sukzessive durchgegangen sind und alle Rundungen berücksichtigt haben. Nun verwenden wir die Linearisierungen $\frac{1}{1+\varepsilon_j} \doteq 1 - \varepsilon_j$ sowie $\frac{1}{1+\varepsilon_1+1/x} \doteq 1 - \varepsilon_1 - 1/x$, wobei letztere voraussetzt, dass x vergleichsweise groß ist. Dies liefert

$$\begin{aligned}
\widetilde{\phi}_1(\widetilde{x}) &\doteq \Big\{\frac{1}{x}(1-\varepsilon_1)(1+\varepsilon_2) - \frac{1}{x}(1-\varepsilon_1-1/x)(1-\varepsilon_3)(1+\varepsilon_4)\Big\}(1+\varepsilon_5) \\
&\doteq \frac{1}{x}(1-\varepsilon_1+\varepsilon_2+\varepsilon_5) - \frac{1}{x}(1-\varepsilon_1-\varepsilon_3+\varepsilon_4+\varepsilon_5) + \frac{1}{x^2}(1-\varepsilon_3+\varepsilon_4+\varepsilon_5) \\
&= \frac{1}{x}(\varepsilon_2+\varepsilon_3-\varepsilon_4) + \frac{1}{x^2}(1-\varepsilon_3+\varepsilon_4+\varepsilon_5).
\end{aligned}$$

Nun folgt für $x \gg 0$

$$\begin{aligned}
\frac{|\widetilde{\phi}_1(\widetilde{x}) - \phi(x)|}{|\phi(x)|} &\doteq \Big|(x+1)(\varepsilon_2+\varepsilon_3-\varepsilon_4) + \frac{x+1}{x}(1-\varepsilon_3+\varepsilon_4+\varepsilon_5) - 1\Big| \\
&= \big|(x+1)(\varepsilon_2+\varepsilon_3-\varepsilon_4) + (1+1/x)(-\varepsilon_3+\varepsilon_4+\varepsilon_5) + 1/x\big|.
\end{aligned}$$

Im schlimmsten Fall folgt also

$$\frac{|\widetilde{\phi}_1(\widetilde{x}) - \phi(x)|}{|\phi(x)|} \doteq (x + 2 + 1/x)\,3\text{eps} + 1/x \gg \text{eps} \approx \frac{|\phi(\widetilde{x}) - \phi(x)|}{|\phi(x)|}.$$

Dies zeigt die Nicht-Stabilität. ■

Behauptung. Die Auswertung von $\phi(x)$ in der Form $\widetilde{\phi}_2(x) = \frac{1}{x(x+1)}$ ist stabil.

Beweis. Mit geeigneten $|\varepsilon_j| \leq \text{eps}$ folgt

$$\begin{aligned}
\widetilde{\phi}_2(\widetilde{x}) &= \frac{1+\varepsilon_4}{x(1+\varepsilon_1)\{x(1+\varepsilon_1)+1\}(1+\varepsilon_2)(1+\varepsilon_3)} \\
&= \frac{1}{x^2}\,\frac{1}{1+\varepsilon_1}\,\frac{1}{1+(\varepsilon_1+1/x)}\,\frac{1+\varepsilon_4}{(1+\varepsilon_2)(1+\varepsilon_3)} \\
&\doteq \frac{1}{x^2}\{1-\varepsilon_1-1/x\}(1-\varepsilon_1-\varepsilon_2-\varepsilon_3+\varepsilon_4) \\
&\doteq \frac{1}{x^2}(1-2\varepsilon_1-\varepsilon_2-\varepsilon_3+\varepsilon_4) - \frac{1}{x^3}(1-\varepsilon_1-\varepsilon_2-\varepsilon_3+\varepsilon_4).
\end{aligned}$$

Dies liefert

$$\begin{aligned}
\frac{|\widetilde{\phi}_2(\widetilde{x}) - \phi(x)|}{|\phi(x)|} &\doteq \Big|\frac{x+1}{x}(1-2\varepsilon_1-\varepsilon_2-\varepsilon_3+\varepsilon_4) - \frac{x+1}{x^2}(1-\varepsilon_1-\varepsilon_2-\varepsilon_3+\varepsilon_4) - 1\Big| \\
&= \Big|(1+1/x)(-2\varepsilon_1-\varepsilon_2-\varepsilon_3+\varepsilon_4) - \Big(\frac{1}{x}+\frac{1}{x^2}\Big)(-\varepsilon_1-\varepsilon_2-\varepsilon_3+\varepsilon_4) - \frac{1}{x^2}\Big| \\
&\leq \frac{1}{x^2} + 5\text{eps}(1+1/x)^2 \approx 5\text{eps}
\end{aligned}$$

für große $x \gg 0$. ■

Bemerkung. Die gewählten Linearisierungen sind in gewissem Sinne willkürlich. Anstelle von

$$\frac{1}{x(1+\varepsilon_1)+1} \doteq \frac{1}{x}(1-\varepsilon_1-1/x)$$

könnte man auch die Linearisierung

$$\frac{1}{x(1+\varepsilon_1)+1} = \frac{1}{(x+1)+\varepsilon_1 x} \doteq \frac{1}{x+1} - \frac{\varepsilon_1 x}{(x+1)^2}$$

verwenden, sofern $\varepsilon_1 x$ klein genug ist. Analoge Rechnung führt dann im schlimmsten Fall auf $\frac{|\widetilde{\phi}_1(\widetilde{x})-\phi(x)|}{|\phi(x)|} \doteq (3+7x)\text{eps}$ sowie $\frac{|\widetilde{\phi}_2(\widetilde{x})-\phi(x)|}{|\phi(x)|} \doteq 5\text{eps}$ und sei dem Leser zur Übung überlassen. □

1.4 Verfahrensfehler

Im Wesentlichen gibt es zwei Arten von sogenannten Verfahrensfehlern, die Abbruchfehler und die Diskretisierungsfehler. Abbruchfehler entstehen dann, wenn wir einen konvergenten (aber unendlichen) Algorithmus nach endlich vielen Schritten abbrechen, um in endlicher Zeit ein numerisches Ergebnis zu erhalten.

Beispiel (Berechnung der Quadratwurzel). Für $x > 0$ definieren wir die Folge $(y_n)_{n\in\mathbb{N}}$ induktiv durch $y_1 := \frac{1}{2}(1+x)$ und $y_{n+1} := \frac{1}{2}(y_n + x/y_n)$. Dann gilt $\lim_n y_n = \sqrt{x}$. Man beachte, dass für $x \neq 1$ stets $y_n \neq x$ gilt, denn eine elementare algebraische Umformung zeigt, dass die Gleichheit $\sqrt{x} = \frac{1}{2}(y + x/y)$ genau dann erfüllt ist, wenn bereits $y = \sqrt{x}$ gilt. ■

Ferner steht die Numerik vor dem Problem, dass kontinuierliche Größen wie beispielsweise Ableitung oder Integral vom Rechner entweder nicht analytisch berechnet werden können oder ihre analytische Berechnung mittels Formelmanipulation viel zu aufwändig ist. Man *diskretisiert* daher kontinuierliche Größen: Statt Ableitungen berechnet man Differenzenquotienten, statt Integralen berechnet man gewisse endliche (Riemann-) Summen.

Beispiel (Numerische Differentiation, einseitiger Differenzenquotient). Es sei $f : \mathbb{R} \to \mathbb{R}$ eine differenzierbare Funktion. Um für ein festes $x \in \mathbb{R}$ die Ableitung $\Phi := f'(x)$ zu approximieren, berechnen wir den *einseitigen Differenzenquotienten* $\Phi_h := \frac{f(x+h)-f(x)}{h}$ für ein festes $h > 0$. Nach Definition der Ableitung, gilt

$$|\Phi - \Phi_h| = \left| f'(x) - \frac{f(x+h)-f(x)}{h} \right| \to 0 \quad \text{für } h \to 0.$$

Diese Konvergenz kann aber theoretisch beliebig langsam sein. Man interessiert sich deshalb in der Numerik auch für Aussagen darüber, wie schnell eine diskrete Größe Φ_h gegen die kontinuierliche Größe Φ konvergiert: Für $f \in \mathcal{C}^2(\mathbb{R})$ folgt mit dem Mittelwertsatz die Existenz von η, ζ mit $x \leq \eta \leq \zeta \leq x + h$ mit

$$\left| f'(x) - \frac{f(x+h)-f(x)}{h} \right| = |f'(x) - f'(\zeta)| = |f''(\eta)||x - \zeta| \leq \|f''\|_{\infty,[x,x+h]}\, h = \mathcal{O}(h)$$

für $h \to 0$. ■

Definition. Es sei Φ eine kontinuierliche Größe und Φ_h eine Diskretisierung von Φ, wobei $h > 0$ den **Diskretisierungsparameter** bezeichnet. Eine Abschätzung der Form

$$|\Phi - \Phi_h| = \mathcal{O}(h^\alpha) \tag{1.8}$$

bezeichnet man als **a priori Fehlerabschätzung** für den Diskretisierungsfehler. Die Zahl $\alpha > 0$ bezeichnet man als **Konvergenzordnung**. □

Bemerkung. In der Praxis treten nicht nur ganzzahlige Konvergenzordnungen $\alpha > 0$ auf. Im Beispiel des einseitigen Differenzenquotienten liegt dies an zusätzlichen Eigenschaften der Funktion f. Im Kontext der klassischen Numerischen Mathematik betrachtet man oft die sogenannten **Hölder-stetigen Funktionen** $f \in C^{m,\alpha}(\Omega)$ mit $0 \leq \alpha \leq 1$. Dabei gilt für ein offenes Intervall Ω

$$C^{m,\alpha}(\Omega) := \Big\{ f \in C^m(\Omega) \,\Big|\, \sup_{\substack{x,y\in\Omega \\ x\neq y}} \frac{|f^{(m)}(x) - f^{(m)}(y)|}{|x-y|^\alpha} < \infty \Big\}. \tag{1.9}$$

Offensichtlich ist $C^{0,1}$ gerade die Menge der Lipschitz-stetigen Funktionen. Ist nun $f \in C^{1,\alpha}$ lokal um x, so folgt für den einseitigen Differenzenquotienten offensichtlich

$$\left| f'(x) - \frac{f(x+h)-f(x)}{h} \right| = |f'(x) - f'(\zeta)| = \mathcal{O}(h^\alpha)$$

für $h \to 0$, d.h. wir erhalten eine verminderte Konvergenzordnung, falls f nicht in $\mathcal{C}^2$ ist. ■

Bei vielen Algorithmen stellt sich die Frage, ob es nicht möglich ist, einen Algorithmus zu finden, der unter gewissen Zusatzvoraussetzungen sogar besser konvergiert. Im Falle der numerischen Differentiation ist dies einfach möglich.

Beispiel (Numerische Differentiation, zentraler Differenzenquotient). Anstelle von $\Phi = f'(x)$ berechnen wir $\Phi_h := \frac{f(x+h)-f(x-h)}{2h}$. Schreibt man diesen sogenannten *zentralen Differenzenquotienten* in der Form $\Phi_h = \frac{f(x+h)-f(x)}{2h} + \frac{f(x)-f(x-h)}{2h}$, so gilt offensichtlich

$$\lim_{h\to 0} \Phi_h = \Phi,$$

wenn $f : \mathbb{R} \to \mathbb{R}$ differenzierbar ist bei $x \in \mathbb{R}$. Für $f \in C^2$ folgt analog zu oben

$$|\Phi - \Phi_h| = \mathcal{O}(h) \quad \text{für } h \to 0.$$

Mit Taylor-Entwicklungen für $x \pm h$ ergibt sich

$$f(x \pm h) = f(x) \pm f'(x)h + \frac{f''(x)}{2}\, h^2 \pm \frac{f'''(\zeta_\pm)}{6}\, h^3 \quad \text{mit } x - h \le \zeta_\pm \le x + h \text{ geeignet.}$$

Durch Subtraktion beider Gleichungen folgt

$$|\Phi - \Phi_h| \le \frac{1}{6}\, \|f'''\|_{\infty,[x-h,x+h]}\, h^2 = \mathcal{O}(h^2) \quad \text{für } h \to 0.$$

Sofern f also glatter ist um x, z.B. $f \in C^3$, führt der zentrale Differenzenquotient auf eine höhere Konvergenzordnung. ■

In der Regel sind Rundungsfehler und Verfahrensfehler im Widerstreit. Zur Illustration betrachten wir das folgende Beispiel,

$$e = \exp'(1) \approx \frac{\exp(1+h) - \exp(1-h)}{2h}.$$

Für $h = 10^{-8}$ erwarten wir deshalb einen (relativen) Fehler von der Größenordnung 10^{-16}. Aufgrund von Auslöschung erhalten wir aber in MATLAB einen tatsächlichen relativen Fehler in der Größenordnung 10^{-7}. Pauschal formuliert, gilt die folgende Faustregel: *Ist der Diskretisierungsparameter $h > 0$ klein, so erhalten wir einen kleinen Verfahrensfehler, aber große Rundungsfehler. Ist der Diskretisierungsfehler $h > 0$ groß, so dominiert der Verfahrensfehler gegenüber dem Rundungsfehler.*

Beispiel (Numerische Integration, summierte Trapezregel). Ziel ist die numerische Berechnung des Integrals $\Phi := \int_a^b f\, dx$ mit stetigem Integranden $f : [a, b] \to \mathbb{R}$. Dazu definieren wir die *summierte Trapezregel*

$$\Phi_h := \frac{h}{2}\Big\{ f(a) + 2\sum_{j=1}^{n-1} f(a + jh) + f(b) \Big\} \quad \text{mit } n \in \mathbb{N} \text{ und } h := (b-a)/n.$$

Zur Veranschaulichung definieren wir Knoten $x_j := a + jh$ mit $j = 0, \dots, n$. Verbindet man die Funktionswerte $f(x_{j-1})$ und $f(x_j)$ affin, so ist das Integral dieser affinen Funktion gerade $\frac{h}{2}\big(f(x_{j-1}) + f(x_j)\big)$. De facto ersetzen wir also f durch den interpolierenden affinen Streckenzug und berechnen dann das Integral für diesen exakt. Für $f \in C^2[a,b]$ kann man $|\Phi - \Phi_h| = \mathcal{O}(h^2)$ zeigen. Dazu betrachten wir zunächst ein Teilintervall $[x_{j-1}, x_j]$ von $[a,b]$. Ohne Beschränkung der Allgemeinheit sei $a = 0$ und $j = 1$, d.h. $[x_{j-1}, x_j] = [0,h]$. Es sei $p : [0,h] \to \mathbb{R}$ eine affine Funktion, die f in 0 und h interpoliert, d.h. $p(0) = f(0)$ und $p(h) = f(h)$. Dann gilt $\int_0^h p\,dx = h\,\frac{f(0)+f(h)}{2}$, und wir erhalten

$$\Big| \int_0^h f\,dx - h\,\frac{f(0)+f(h)}{2} \Big| = \Big| \int_0^h (f-p)\,dx \Big| \le h \|f-p\|_{\infty,[0,h]}.$$

Wir müssen also nur noch den Interpolationsfehler abschätzen. Dazu sei $x \in (0,h)$ beliebig. Wir definieren

$$F(y) := \big(f(x) - p(x)\big)y(h-y) - \big(f(y) - p(y)\big)x(h-x).$$

F hat Nullstellen bei $0, x, h$. Nach Mittelwertsatz hat F' zumindest zwei Nullstellen $0 < \zeta_1 < x < \zeta_2 < h$, und schließlich hat F'' zumindest eine Nullstelle $\zeta_1 < \zeta < \zeta_2$. Es folgt damit

$$0 = F''(\zeta) = -2\big(f(x) - p(x)\big) - f''(\zeta)x(h-x).$$

Umformung zeigt

$$f(x) - p(x) = -\frac{f''(\zeta)}{2}\,x(h-x) \quad \text{für } x \in [0,h] \text{ und } \zeta \text{ geeignet}$$

und deshalb $\|f-p\|_{\infty,[0,h]} \le h^2\,\|f''\|_\infty/8$, denn die Parabel $x(h-x)$ nimmt ihr Maximum bei $x = h/2$ an. Auf jedem Intervall gilt also

$$\Big| \int_{x_{j-1}}^{x_j} f\,dx - h\,\frac{f(x_{j-1})+f(x_j)}{2} \Big| \le \frac{\|f''\|_\infty}{8}\,h^3.$$

Summation über alle $n = (b-a)/h$ Teilintervalle beweist schließlich

$$\Big| \int_a^b f\,dx - \Phi_h \Big| \le \frac{(b-a)\|f''\|_\infty}{8}\,h^2 = \mathcal{O}(h^2). \qquad \blacksquare$$

Numerische Bestimmung der Konvergenzordnung

Es sei Φ die kontinuierlich berechnete Größe und Φ_h die diskrete (d.h. über ein numerisches Verfahren berechnete) Größe zum Diskretisierungsparameter $h > 0$. Besitzt das Verfahren Ordnung $\alpha > 0$, so gelten (ansatzweise) für den Fehler $e_h = |\Phi - \Phi_h|$ und Diskretisierungsparameter $h, h/2$

$$e_h = C\,h^\alpha \quad \text{und} \quad e_{h/2} = C\,(h/2)^\alpha$$

mit der Konvergenzordnung $\alpha > 0$ und einer Konstante $C > 0$. Elementare Umformung liefert

$$e_{h/2} = C\,h^\alpha\,2^{-\alpha} = e_h 2^{-\alpha}.$$

Man erhält also die Formeln

$$\alpha = \log(e_h/e_{h/2}) \,/\, \log(2) \quad \text{und} \quad C = e_h/h^\alpha,$$

d.h. die **experimentelle Konvergenzordnung** α (sowie die zugehörige Konstante C) sind *a posteriori* berechenbar, wenn man die Werte von e_h und $e_{h/2}$ kennt. Zur numerischen Verifikation von a priori Fehlerabschätzungen berechnet man mit einer Folge von Schrittweiten $h, h/2, h/4, \ldots$ die zugehörigen Werte von α und C für jeweils zwei aufeinanderfolgende Werte e_h.

h	$e_h^{(1)}$	$\alpha^{(1)}$	$C^{(1)}$	$e_h^{(2)}$	$\alpha^{(2)}$	$C^{(2)}$
1	$1.9525e+00$			$4.7625e-01$		
1/2	$8.0853e-01$	1.27	$1.95e+00$	$1.1469e-01$	2.05	$4.76e-01$
1/4	$3.6996e-01$	1.13	$1.77e+00$	$2.8404e-02$	2.01	$4.63e-01$
1/8	$1.7720e-01$	1.06	$1.61e+00$	$7.0844e-03$	2.00	$4.57e-01$
1/16	$8.6744e-02$	1.03	$1.51e+00$	$1.7701e-03$	2.00	$4.54e-01$
1/32	$4.2919e-02$	1.02	$1.45e+00$	$4.4245e-04$	2.00	$4.53e-01$
1/64	$2.1348e-02$	1.01	$1.41e+00$	$1.1061e-04$	2.00	$4.53e-01$
1/128	$1.0646e-02$	1.00	$1.39e+00$	$2.7652e-05$	2.00	$4.53e-01$
1/256	$5.3161e-03$	1.00	$1.38e+00$	$6.9130e-06$	2.00	$4.53e-01$
1/512	$2.6563e-03$	1.00	$1.37e+00$	$1.7282e-06$	2.00	$4.53e-01$
1/1024	$1.3277e-03$	1.00	$1.36e+00$	$4.3206e-07$	2.00	$4.53e-01$
1/2048	$6.6375e-04$	1.00	$1.36e+00$	$1.0801e-07$	2.00	$4.53e-01$
1/4096	$3.3185e-04$	1.00	$1.36e+00$	$2.7004e-08$	2.00	$4.53e-01$
1/8192	$1.6592e-04$	1.00	$1.36e+00$	$6.7512e-09$	2.00	$4.53e-01$
1/16384	$8.2957e-05$	1.00	$1.36e+00$	$1.6890e-09$	2.00	$4.49e-01$
1/32768	$4.1478e-05$	1.00	$1.36e+00$	$4.1934e-10$	2.01	$4.99e-01$
1/65536	$2.0739e-05$	1.00	$1.36e+00$	$9.1926e-11$	2.19	$3.23e+00$
1/131072	$1.0369e-05$	1.00	$1.36e+00$	$1.9166e-11$	2.26	$7.21e+00$
1/262144	$5.1847e-06$	1.00	$1.36e+00$	$4.8270e-11$	-1.33	$2.90e-18$

Tabelle 1.1: Numerische Differentiation mittels einseitigem und zentralem Differenzenquotienten $\Phi_h^{(1)}$ bzw. $\Phi_h^{(2)}$ zur Approximation von $e = \exp'(1)$: Für den einseitigen Differenzenquotienten beobachten wir die experimentelle Konvergenzordnung 1, d.h. $|e - \Phi_h^{(1)}| = \mathcal{O}(h)$. Den zentrale Differenzenquotient liefert $|e - \Phi_h^{(1)}| = \mathcal{O}(h^2)$, wobei die Konvergenz im letzten Schritt aufgrund von Auslöschungseffekten zusammenbricht.

Beispiel (Numerische Differentiation). Wir betrachten als Beispiel die Approximation von $e = \exp(1)$ durch den einseitigen und zweiseitigen Differenzenquotienten $\Phi_h^{(1)}$ bzw. $\Phi_h^{(2)}$. Tabelle 1.1 gibt die entsprechenden Ergebnisse wieder. Wir beobachten die vorhergesagten Konvergenzordnungen $|e - \Phi_h^{(1)}| = \mathcal{O}(h)$ und $|e - \Phi_h^{(1)}| = \mathcal{O}(h^2)$. Für $h = 2^{-18} = 1/262144 \approx 3.81 \cdot 10^{-6}$ bricht die Konvergenz für den zentralen Differenzenquotienten aufgrund von Auslöschungseffekten zusammen. Man beachte, dass auch die Konstante $C^{(2)}$ a priori vorhergesagt werden kann. Nach der Fehleranalysis für den zentralen Differenzenquotienten gilt $|\Phi - \Phi_h^{(2)}| \leq \frac{1}{6} \,\|\exp'''\|_{\infty,[1-h,1+h]}\, h^2$, und die Konstante vor der h-Potenz konvergiert gegen $e/6 \approx 4.53 \cdot 10^{-1}$. Diese wird experimentell durch $C^{(2)}$ scharf geschätzt. Die Konstante $C^{(1)}$ lässt sich a priori wie folgt vorhersagen: Mit Taylor-Entwicklung gilt $|\Phi - \Phi_h^{(1)}| = \frac{1}{2} \,|\exp''(\zeta)|\, h$ für ein geeignetes $1 \leq \zeta \leq 1 + h$. Wir erhalten also Konvergenz dieser Konstante gegen $e/2 \approx 1.36$. Dies ist gerade die experimentell beobachtete Konstante $C^{(1)}$.

Kapitel 2

Matrixnormen und Konditionierung

2.1 Operatornorm

Lemma 2.1. *Zu fixierten Normen $\|\cdot\|$ auf $\mathbb{K}^m$ bzw. $\mathbb{K}^n$ definieren wir die* **Operatornorm**

$$\|A\| := \sup_{x\in\mathbb{K}^n\setminus\{0\}} \frac{\|Ax\|}{\|x\|} \quad \text{für } A \in \mathbb{K}^{m\times n}. \tag{2.1}$$

Dann gelten die folgenden Aussagen:
(i) *Die Operatornorm $\|\cdot\|$ definiert eine Norm auf $\mathbb{K}^{m\times n}$,*

(ii) $\|A\| = \sup_{\|x\|=1} \|Ax\| = \sup_{\|x\|\le 1} \|Ax\| = \inf\big\{C > 0 \,\big|\, \forall x \in \mathbb{K}^n \quad \|Ax\| \le C\,\|x\|\big\}$,
(iii) *im Fall $A \neq 0$ folgt für $x \in \mathbb{K}^m$ mit $\|x\| \le 1$ und $\|Ax\| = \|A\|$ bereits $\|x\| = 1$,*

(iv) *aufgrund der endlichen Dimension werden alle Infima und Suprema angenommen.*

Beweis in der Übung. ■

Im Folgenden sei stets vorausgesetzt, dass (beliebige) Normen auf $\mathbb{K}^m$, $\mathbb{K}^n$ etc. fixiert seien, und $\|\cdot\|$ bezeichne jeweils die induzierte Operatornorm. Insbesondere betrachten wir für $m = n$ nur den Fall, dass beide Normen übereinstimmen. Das folgende Lemma stellt einige Rechenregeln bereit.

Lemma 2.2. (i) *Für Matrizen $A \in \mathbb{K}^{\ell\times m}$ und $B \in \mathbb{K}^{m\times n}$ gilt die* **Idealeigenschaft** $\|AB\| \le \|A\|\|B\|$.
(ii) *Die Identität $\boldsymbol{I} : \mathbb{K}^n \to \mathbb{K}^n$ erfüllt $\|\boldsymbol{I}\| = 1$.*
(iii) *Ist $A \in \mathbb{K}^{n\times n}$ invertierbar, so gilt*

$$\|A^{-1}\| = \Big(\inf_{\|x\|=1} \|Ax\|\Big)^{-1}. \tag{2.2}$$

Beweis in der Übung. ■

Beispiel (Spaltensummennorm $\|\cdot\|_1$). Auf $\mathbb{K}^n$ wird durch $\|x\|_1 := \sum_{j=1}^n |x_j|$ die sog. ℓ_1-Norm definiert. Betrachtet man auf $\mathbb{K}^m$ und $\mathbb{K}^n$ die ℓ_1-Norm, so erhält man als induzierte Operatornorm

die Spaltensummennorm

$$\|A\|_1 = \max_{k=1,\dots,n} \sum_{j=1}^{m} |a_{jk}| \quad \text{für } A \in \mathbb{K}^{m\times n}. \tag{2.3}$$

Für $x \in \mathbb{K}^n$ gilt nämlich

$$\|Ax\|_1 \le \sum_{j=1}^{m} \Big| \sum_{k=1}^{n} a_{jk}x_k \Big| \le \sum_{k=1}^{n} \Big(\sum_{j=1}^{m} |a_{jk}| \Big) |x_k| \le \max_{k=1,\dots,n} \sum_{j=1}^{m} |a_{jk}| \, \|x\|_1$$

Dies zeigt $\|A\|_1 \le \max_{k=1}^{n} \sum_{j=1}^{m} |a_{jk}|$. Wählt man den Index k, bei dem das Maximum angenommen wird und den zugehörigen Standardeinheitsvektor $\mathbf{e}_k \in \mathbb{K}^n$, so folgt die Gleichheit. Insgesamt sehen wir also, dass die Operatornorm in einem der Standardeinheitsvektoren angenommen wird. ■

Beispiel (Zeilensummennorm $\|\cdot\|_\infty$). Auf $\mathbb{K}^n$ wird durch $\|x\|_\infty := \max_{j=1}^{n} |x_j|$ die sog. ℓ_∞-Norm definiert. Betrachtet man auf $\mathbb{K}^m$ und $\mathbb{K}^n$ die ℓ_∞-Norm, so erhält man als induzierte Operatornorm die Zeilensummennorm

$$\|A\|_\infty = \max_{j=1,\dots,m} \sum_{k=1}^{n} |a_{jk}| \quad \text{für } A \in \mathbb{K}^{m\times n}. \tag{2.4}$$

Für $x \in \mathbb{K}^n$ gilt analog zu oben

$$\|Ax\|_\infty \le \max_{j=1,\dots,m} \Big| \sum_{k=1}^{n} a_{jk}x_k \Big| \le \max_{j=1,\dots,m} \sum_{k=1}^{n} |a_{jk}| \, \|x\|_\infty,$$

also $\|A\|_\infty \le \max_{j=1}^{m} \sum_{k=1}^{n} |a_{jk}|$. Um die Gleichheit zu zeigen, fixieren wir zunächst den Index j, für den das Maximum angenommen wird. Dann definieren wir $x \in \mathbb{K}^n$ durch $x_k := \operatorname{sign}(a_{jk})$ mit dem (komplexen) Vorzeichen $\operatorname{sign}(z) = |z|/z$ für $z \neq 0$. Es folgt $\|x\|_\infty = 1$ und $\|Ax\|_\infty = \sum_{k=1}^{n} |a_{jk}|$. ■

Bemerkung. Dass beispielsweise die Spaltensummennorm in einem Eckpunkt der ℓ_1-Einheitssphäre $S_1^n := \big\{x \in \mathbb{K}^n \,\big|\, \|x\|_1 = 1\big\}$ angenommen wird, lässt sich mathematisch wie folgt erklären: Ist $\|\cdot\|$ eine Norm auf $\mathbb{K}^n$, so wird die Operatornorm einer Matrix $A \in \mathbb{K}^{m\times n}$ in einem Extremalpunkt der Einheitssphäre $S := \big\{x \in \mathbb{K}^n \,\big|\, \|x\| = 1\big\}$ angenommen. Dabei heißt ein Punkt Extremalpunkt, wenn er nicht die Konvexkombination von zwei anderen Punkten der Sphäre ist, d.h. die Menge der Extremalpunkte ist

$$E := S \backslash \big\{x \in S \,\big|\, \exists y, z \in S, x \neq y \exists \lambda \in (0,1) : x = \lambda y + (1-\lambda)z\big\}.$$

Dies entspricht anschaulich gerade den Eckpunkten von S. Gilt für $x \in S$ gerade $\|Ax\| = \|A\|$, d.h. die Operatornorm wird in x angenommen, und ist x kein Extremalpunkt, so existieren $y, z \in S$ und $\lambda \in (0,1)$ mit $x = \lambda y + (1-\lambda)z$. Die Dreiecksungleichung

$$\|A\| = \|Ax\| \le \lambda\|Ay\| + (1-\lambda)\|Az\| \le \|A\|$$

zeigt, dass dann die Operatornorm auch in y und z angenommen wird. Abschließend sei angemerkt, dass der Satz von Krein-Milman gerade besagt, dass jeder Nicht-Extremalpunkt in S die Konvexkombination zweier Extremalpunkte ist. □

Definition. Jede Matrix $A \in \mathbb{K}^{n\times n}$ hat (gezählt gemäß algebraischer Vielfachheit) n komplexe Eigenwerte. Den Betrag des betragsgrößten Eigenwerts bezeichnet man als **Spektralradius** von A,

$$\rho(A) := \max\big\{|\lambda| \,\big|\, \lambda \in \mathbb{C} \text{ ist Eigenwert von } A\big\} \tag{2.5}$$

Beispiel (Frobenius-Norm ist keine Operatornorm). Die **Frobenius-Norm**

$$\|A\|_F := \Big(\sum_{j,k=1}^{n} |a_{jk}|^2\Big)^{1/2} \quad \text{für } A \in \mathbb{K}^{n\times n} \tag{2.6}$$

ist eine Hilbert-Norm auf $\mathbb{K}^{n\times n}$, d.h. $(\mathbb{K}^{n\times n}, \|\cdot\|_F)$ ist ein Hilbert-Raum. Ferner gilt $\|A\|_2 \le \|A\|_F$, denn aus der Cauchy-Schwarz-Ungleichung folgt

$$\|Ax\|_2^2 = \sum_{j=1}^{n} \Big|\sum_{k=1}^{n} a_{jk}x_k\Big|^2 \le \sum_{j=1}^{n} \Big(\sum_{k=1}^{n} |a_{jk}|^2\Big)\Big(\sum_{k=1}^{n} |x_k|^2\Big) = \|A\|_F^2\|x\|_2^2.$$

In der Übung sollen Sie zeigen, dass die Frobenius-Norm keine Operatornorm ist, auch wenn man sie umskaliert, d.h. es gibt *keine* Operatornorm $\|\cdot\|$ und *kein* Skalar $\lambda > 0$ mit $\lambda\|\cdot\| = \|\cdot\|_F$. ■

Beispiel (Spektralnorm $\|\cdot\|_2$). Auf $\mathbb{K}^n$ wird durch $\|x\|_2 := \big(\sum_{j=1}^n |x_j|^2\big)^{1/2}$ die sog. ℓ_2-Norm definiert. Betrachtet man auf $\mathbb{K}^m$ und $\mathbb{K}^n$ die ℓ_2-Norm, so erhält man als induzierte Operatornorm die Spektralnorm

$$\|A\|_2 = \sqrt{\rho(\overline{A}^T A)}. \tag{2.7}$$

Der Beweis sei dem Leser zur Übung überlassen. Nutzen Sie, dass für die selbstadjungierte Matrix $B := \overline{A}^T A$, d.h. $B = \overline{B}^T$ nach Linearer Algebra eine Orthonormalbasis des $\mathbb{R}^n$ aus Eigenvektoren $\{u_1, \dots, u_n\}$ von B bezüglich dem **euklidischen Skalarprodukt** $x \cdot y := \sum_{j=1}^n x_j\,\overline{y}_j$ existiert und die zugehörigen Eigenwerte $\lambda_j \in \mathbb{R}$ stets reell sind. ■

Beispiel (Spektralnorm für selbstadjungierte Matrizen). Ist $A \in \mathbb{K}^{n\times n}$ eine selbstadjungierte Matrix, so gilt für die Spektralnorm $\|A\|_2 = \rho(A)$. Dies zeigt man wie folgt: Nach Linearer Algebra ist A diagonalisierbar, d.h. es gilt $A = T\Lambda T^{-1}$ mit einer regulären Matrix T und einer Diagonalmatrix Λ. Hierbei stehen die Eigenwerte von A auf der Diagonalen von Λ und T enthält als Zeilen die entsprechenden Eigenvektoren. Es folgt $\overline{A}^T A = A^2 = T\Lambda^2 T^{-1}$ und damit insbesondere $\rho(A)^2 = \rho(A^2)$, was den Beweis beschließt. ■

2.2 Kondition einer Matrix

In diesem Abschnitt ist $A \in \mathbb{K}^{n\times n}$ eine reguläre Matrix, und $\|\cdot\|$ ist eine fixierte Norm auf $\mathbb{K}^n$ nebst induzierter Operatornorm auf $\mathbb{K}^{n\times n}$.

Definition. Die Größe $\mathrm{cond}(A) := \|A\|\|A^{-1}\|$ bezeichnet man als **Konditionszahl** der regulären Matrix A. Im Fall der ℓ_p-Normen notieren wir die Konditionszahl mit dem entsprechenden Index, z.B. $\mathrm{cond}_\infty(A) := \|A\|_\infty\|A^{-1}\|_\infty$ im Fall der Zeilensummennorm $p = \infty$.

Nach den bewiesenen Eigenschaften der Operatornorm gilt

$$\mathrm{cond}(A) = \max_{\|x\|=1} \|Ax\| / \min_{\|x\|=1} \|Ax\|,$$

d.h. geometrisch gibt die Kondition eine Aussage über die Verzerrung des Raums $\mathbb{K}^n$ durch A.

Wie im ersten Abschnitt gibt die Konditionszahl eine Aussage darüber, wie stark sich der relative Fehler in der rechten Seite $b \in \mathbb{K}^n$ auf die Lösung des Gleichungssystems $Ax = b$ auswirkt.

Lemma 2.3. *Es seien $A \in \mathbb{K}^{n\times n}$ eine reguläre Matrix und $x, \widetilde{x}, b, \widetilde{b} \in \mathbb{K}^n\backslash\{0\}$ Vektoren mit $Ax = b$ und $A\widetilde{x} = \widetilde{b}$. Dann gilt*

$$\frac{\|\widetilde{x} - x\|}{\|x\|} \le \mathrm{cond}(A)\, \frac{\|\widetilde{b} - b\|}{\|b\|}. \tag{2.8}$$

Beweis. Es gilt $\|\widetilde{x} - x\| = \|A^{-1}(\widetilde{b} - b)\| \le \|A^{-1}\|\|\widetilde{b} - b\|$ sowie $\|b\| \le \|A\|\|x\|$. Multipliziert man beide Abschätzungen folgt die Behauptung. ■

Mit etwas größerem Aufwand kann man auch abschätzen, wie sich ein zusätzlicher Fehler bei der Matrix A auf das berechnete Ergebnis auswirkt.

Satz 2.4. *Es seien $A, \widetilde{A} \in \mathbb{K}^{n\times n}$ Matrizen und $x, \widetilde{x}, b, \widetilde{b} \in \mathbb{K}^n$ Vektoren mit $Ax = b$ und $\widetilde{A}\widetilde{x} = \widetilde{b}$. Die Matrix A sei regulär, und es gelte $\|\widetilde{A} - A\| < 1/\|A^{-1}\|$. Dann ist auch $\widetilde{A}$ regulär, und es gilt*

$$\frac{\|x - \widetilde{x}\|}{\|x\|} \le \frac{\mathrm{cond}(A)}{1 - \mathrm{cond}(A)\,\frac{\|\widetilde{A} - A\|}{\|A\|}} \Big(\frac{\|\widetilde{A} - A\|}{\|A\|} + \frac{\|\widetilde{b} - b\|}{\|b\|} \Big) \tag{2.9}$$

Beweis. Das Beweis erfolgt in drei Schritten.

1. Schritt. *Ist $M \in \mathbb{K}^{n\times n}$ mit $\|M\| < 1$, so ist $(\boldsymbol{I} + M)$ regulär, und es gilt*

$$\|(\boldsymbol{I} + M)^{-1}\| \le \frac{1}{1 - \|M\|}. \tag{2.10}$$

Mit der Dreiecksungleichung gilt $\|(\boldsymbol{I} + M)x\| = \|x + Mx\| \ge \|x\| - \|Mx\| \ge (1 - \|M\|)\|x\|$ für alle $x \in \mathbb{K}^n$. Insbesondere gilt $(\boldsymbol{I} + M)x = 0$ genau dann, wenn $x = 0$ gilt, d.h. $(\boldsymbol{I} + M)$ ist injektiv und daher regulär. Mit der Inversen $B := (\boldsymbol{I} + M)^{-1}$ gilt

$$\|B\|(1 - \|M\|) \le \|B\| - \|MB\| \le \|B + MB\| = \|(\boldsymbol{I} + M)B\| = 1.$$

Dies zeigt die Abschätzung (2.10). □

2. Schritt. *$\widetilde{A}$ ist regulär, und es gilt $\|\widetilde{A}^{-1}\| \le \dfrac{\|A^{-1}\|}{1 - \|A^{-1}\|\|\widetilde{A} - A\|}$.* Wir definieren $M := A^{-1}(\widetilde{A} - A)$ und wenden Schritt 1 an, denn es gilt $\|M\| \le \|A^{-1}\|\|\widetilde{A} - A\| < 1$. Also ist $\boldsymbol{I} + M = A^{-1}(A +$

$\widetilde{A} - A) = A^{-1}\widetilde{A}$ regulär und damit auch $\widetilde{A}$. Ferner gilt

$$\|\widetilde{A}^{-1}\| \le \|\widetilde{A}^{-1}A\|\|A^{-1}\| = \|(\boldsymbol{I} + M)^{-1}\|\|A^{-1}\| \le \frac{\|A^{-1}\|}{1 - \|A^{-1}(\widetilde{A} - A)\|}$$
$$\le \frac{\|A^{-1}\|}{1 - \|A^{-1}\|\|\widetilde{A} - A\|}. \qquad \square$$

3. Schritt. *Schließlich zeigen wir Abschätzung* (2.9). Es gilt $\widetilde{A}(x-\widetilde{x}) = \widetilde{A}x-\widetilde{b} = (b-\widetilde{b})-(A-\widetilde{A})x$ und deshalb mit Schritt 2

$$\|x - \widetilde{x}\| \le \|\widetilde{A}^{-1}\|\big(\|b - \widetilde{b}\| + \|A - \widetilde{A}\|\|x\|\big) \le \frac{\mathrm{cond}(A)}{1 - \mathrm{cond}(A)\frac{\|\widetilde{A}-A\|}{\|A\|}}\Big(\frac{\|A - \widetilde{A}\|}{\|A\|}\|x\| + \frac{\|b - \widetilde{b}\|}{\|A\|}\Big)$$

Aus $\|b\| \le \|A\|\|x\|$ folgt $1/\|A\| \le \|x\|/\|b\|$. Die Kombination mit der vorausgegangen Abschätzung und Division durch $\|x\|$ zeigt die Behauptung. ■

Bemerkung. (i) Der Beweis des ersten Schrittes basiert wesentlich auf der endlichen Dimension. Die Aussage gilt aber auch in beliebigen Banach-Räumen und wird dann (nicht viel schwerer) mit Hilfe der Neumannschen Reihe bewiesen.
(ii) Insbesondere zeigt der Satz, dass die Menge $\mathcal{U} := \big\{M \in \mathbb{K}^{n\times n} \,\big|\, M \text{ regulär}\big\}$ eine offene Teilmenge von $\mathbb{K}^{n\times n}$ ist, denn jede hinreichend kleine Störung einer regulären Matrix ist regulär. Wir werden von dieser Feststellung später noch Gebrauch machen. □

2.3 Vorkonditionierung

Beim Lösen des Gleichungssystems $Ax = b$ mit regulärer Matrix $A \in \mathbb{K}^{n\times n}$ werden Rundungsfehler im Wesentlichen mit dem Faktor $\mathrm{cond}(A)$ verstärkt. Dabei bezeichnet wieder $\mathrm{cond}(A) = \|A\|\|A^{-1}\|$ die Konditionszahl bezüglich einer *fixierten* Norm $\|\cdot\|$ auf $\mathbb{K}^n$ (und zugehöriger Operatornorm).

Definition. Ist $A \in \mathbb{K}^{n\times n}$ eine reguläre Matrix, so versteht man unter **Vorkonditionierung** die *numerische Konstruktion* einer regulären Matrix $B \in \mathbb{K}^{n\times n}$, sodass

$$\mathrm{cond}(BA) \le \mathrm{cond}(A)$$

gilt. Man löst in diesem Fall nicht mehr das Gleichungssystem $Ax = b$, sondern das Gleichungssystem $BAx = Bb$, da Rundungsfehler dann weniger verstärkt werden. □

Die Matrix B soll folgende Eigenschaften besitzen:

- Die Berechnung von B ist kostengünstig.
- Die Berechnung von BA ist kostengünstig.

Die beste Wahl von B *wäre* $B = A^{-1}$. In der Regel ist aber A^{-1} nicht bekannt und kann nur sehr aufwändig berechnet werden.

Häufig ergibt sich die Lösung eines Gleichungssystem $A_h x = b$ als letzter Schritt in einem numerischen Diskretisierungsverfahren, und $\mathrm{cond}(A_h)$ hängt vom Diskretisierungsparameter $h > 0$ ab, z.B. $\mathrm{cond}(A_h) = \mathcal{O}(h^{-1})$. Dies wird z.B. bei Interpolationsproblemen der Fall sein, siehe Kapitel 4. In diesem Fall ist man an einer Folge von Vorkonditionierungsmatrizen B_h interessiert, sodass möglichst $\mathrm{cond}(B_h A_h) = \mathcal{O}(1)$ gilt.

In der Regel verwendet man schwach besetzte Matrizen B – im Extremfall Diagonalmatrizen – zur Vorkonditionierung. Für die Zeilensummennorm und Vorkonditionierung mit Diagonalmatrizen ist *Zeilenäquilibrierung* optimal.

Definition. Eine Matrix $A \in \mathbb{K}^{n\times n}$ heißt **zeilenäquilibriert**, falls gilt

$$\sum_{k=1}^{n} |a_{jk}| = 1 \quad \text{für alle } j = 1, \ldots, n. \tag{2.11}$$

Satz 2.5. (i) *Es sei $A \in \mathbb{K}^{n\times n}$ zeilenäquilibriert und regulär. Dann gilt für jede reguläre Diagonalmatrix $D \in \mathbb{K}^{n\times n}$ die Abschätzung* $\mathrm{cond}_\infty(A) \le \mathrm{cond}_\infty(DA)$.
(ii) *Für eine reguläre Matrix $A \in \mathbb{K}^{n\times n}$ definiere die Diagonalmatrix*

$$D := \mathrm{diag}(\alpha_1^{-1}, \ldots, \alpha_n^{-1}) \quad \textit{mit} \quad \alpha_j := \sum_{k=1}^{n} |a_{jk}|.$$

Dann gilt $\mathrm{cond}_\infty(DA) \le \mathrm{cond}_\infty(A)$.

Beweis. (ii) folgt unmittelbar aus (i), denn DA ist zeilenäquilibriert, und damit gilt

$$\mathrm{cond}_\infty(DA) \le \mathrm{cond}_\infty(D^{-1}DA) = \mathrm{cond}_\infty(A).$$

Zum Beweis von (i) beachte man $\|A\|_\infty = \max_{j=1}^{n} \sum_{k=1}^{n} |a_{jk}| = 1$ und insbesondere

$$\|DA\|_\infty = \max_{1\le j\le n} \sum_{k=1}^{n} |d_j a_{jk}| = \max_{1\le j\le n} |d_j| = \|D\|_\infty$$

Daraus folgt

$$\mathrm{cond}_\infty(A) = \|A^{-1}\|_\infty \le \|(DA)^{-1}\|_\infty \|D\|_\infty = \mathrm{cond}_\infty(DA). \qquad \blacksquare$$

Kapitel 3

Eliminationsverfahren zur Lösung linearer Gleichungssysteme

In diesem Abschnitt liegt der Schwerpunkt auf den sogenannten **Eliminationsverfahren** oder **Direkten Lösern**. Im einfachsten Fall ist ein lineares Gleichungssystem $Ax = b$ mit regulärer Matrix $A \in \mathbb{K}^{n\times n}$ und rechter Seite $b \in \mathbb{K}^n$ gegeben und die Lösung $x \in \mathbb{K}^n$ gesucht. Eliminationsverfahren sind Algorithmen, die nach endlich vielen Rechenoperationen die Lösung von $Ax = b$ liefern. Dabei sind i.a. nur die Einträge von A bekannt und *nicht*, ob A regulär ist oder weitere Eigenschaften besitzt. Die Algorithmen müssen in diesem Fall gegebenfalls abbrechen.

Bemerkung (Berechnung der Inversen). Die wichtigste Regel der Numerik ist, dass man die Inverse A^{-1} einer regulären Matrix $A \in \mathbb{K}^{n\times n}$ *nicht* berechnen sollte: Es gibt keinen stabilen Algorithmus zur Berechnung der Inversen und häufig kann auch auf die explizite Berechnung von A^{-1} verzichtet werden. Will man beispielsweise zu gegebenem Vektor $y \in \mathbb{K}^n$ $x = A^{-1}By$ berechnen, so geht man in zwei Schritten vor:

- Man berechnet $z = By$.
- Man löst das Gleichungssystem $Ax = z$.

Falls die Berechnung von A^{-1} *wirklich* nötig ist, berechnet man zunächst eine geeignete Faktorisierung von A (z.B. die LU- oder die QR-Zerlegung). Dann löst man (mittels dieser Faktorisierung) die Gleichungssysteme $Ax^{(j)} = \mathbf{e}_j$ für alle $j = 1, \ldots, n$. Die Matrix $B = (x^{(1)}, \ldots, x^{(n)}) \in \mathbb{K}^{n\times n}$ ist dann die (fehlerbehaftete Approximation der) Inverse A^{-1}. □

3.1 Dreiecksmatrizen

Definition. Eine Matrix $L \in \mathbb{K}^{n\times n}$ heißt **untere Dreiecksmatrix**, falls $\ell_{jk} = 0$ gilt für alle Indizes $j < k$. In diesem Fall verschwinden also alle Einträge oberhalb der Matrix-Diagonalen. Analog dazu heißt eine Matrix $U \in \mathbb{K}^{n\times n}$ **obere Dreiecksmatrix**, falls $u_{jk} = 0$ gilt für $j > k$. Es

gilt also schematisch

$$L = \begin{pmatrix} \ell_{11} & 0 & \dots & \dots & 0 \\ \ell_{21} & \ell_{22} & 0 & \dots & 0 \\ \vdots & \vdots & \ddots & \ddots & \vdots \\ \vdots & \vdots & & \ddots & 0 \\ \ell_{n1} & \ell_{n2} & \dots & \dots & \ell_{nn} \end{pmatrix} \quad \text{und} \quad U = \begin{pmatrix} u_{11} & u_{12} & \dots & \dots & u_{1n} \\ 0 & u_{22} & u_{23} & \dots & u_{2n} \\ \vdots & 0 & \ddots & \ddots & \vdots \\ \vdots & \vdots & \ddots & \ddots & \vdots \\ 0 & 0 & \dots & 0 & u_{nn} \end{pmatrix}.$$

Bemerkung. Für eine Dreiecksmatrix $D \in \mathbb{K}^{n\times n}$ gilt $\det(D) = \prod_{j=1}^{n} d_{jj}$. Insbesondere ist D genau dann regulär, wenn alle Diagonalelemente ungleich Null sind. □

Algorithmus 3.1: Lösung eines oberen Dreieckssystems

Input: reguläre obere Dreiecksmatrix $U \in \mathbb{K}^{n\times n}$, rechte Seite $b \in \mathbb{K}^n$

```
for j = n:-1:1
   x_j = (b_j − Σ_{k=j+1}^{n} u_jk x_k)/u_jj
end
```

Output: Vektor $x \in \mathbb{K}^n$

Behauptung. Algorithmus 3.1 ist wohldefiniert und berechnet in insgesamt n^2 arithmetischen Operationen die eindeutige Lösung $x \in \mathbb{K}^n$ von $Ux = b$.

Beweis. Das Gleichungssystem $Ux = b$ ist äquivalent zu den folgenden n Gleichungen:

$$\begin{aligned} u_{11}x_1 + u_{12}x_2 + \cdots + u_{1n}x_n &= b_1 \\ u_{22}x_2 + \cdots + u_{2n}x_n &= b_2 \\ &\vdots \\ u_{nn}x_n &= b_n \end{aligned}$$

Der Algorithmus löst dieses Gleichungssystem durch Rückwärtseinsetzen, insbesondere sind die im Algorithmus benötigten x_k bereits berechnet. Deshalb und wegen $u_{jj} \neq 0$ ist Algorithmus 3.1 wohldefiniert. Im j-ten Schritt sind jeweils $(n-j)$ Multiplikationen und Subtraktionen sowie 1 Division durchzuführen. Die Gesamtanzahl arithmetischer Operationen ist also $\sum_{j=1}^{n} \big(1 + 2(n-j)\big) = n + 2\sum_{k=1}^{n-1} k = n + 2\,\frac{(n-1)n}{2} = n^2$. ■

Auf analoge Weise löst man ein lineares System mit einer regulären unteren Dreiecksmatrix L.

Algorithmus 3.2: Lösung eines unteren Dreieckssystems

Input: reguläre untere Dreiecksmatrix $L \in \mathbb{K}^{n\times n}$, rechte Seite $b \in \mathbb{K}^n$

```
for j = 1:n
   x_j = (b_j − Σ_{k=1}^{j−1} ℓ_jk x_k)/ℓ_jj
end
```

Output: Vektor $x \in \mathbb{K}^n$

Behauptung. Algorithmus 3.2 ist wohldefiniert und berechnet in insgesamt n^2 arithmetischen Operationen die eindeutige Lösung $x \in \mathbb{K}^n$ von $Lx = b$. ■

Die Menge $\mathcal{U}$ der oberen Dreiecksmatrizen ist abgeschlossen bezüglich der Matrizenmultiplikation. Die Untermenge der regulären oberen Dreiecksmatrizen bildet eine Gruppe.

Lemma 3.3. *Es sei* $\mathcal{U} = \{U \in \mathbb{K}^{n\times n} \mid U$ *obere Dreiecksmatrix*$\}$. *Dann gelten:*
(i) *Für* $A, B \in \mathcal{U}$ *ist das Produkt* $AB \in \mathcal{U}$.
(ii) *Für eine reguläre Matrix* $A \in \mathcal{U}$ *gilt* $B := A^{-1} \in \mathcal{U}$ *und* $b_{jj} = a_{jj}^{-1}$ *für alle* $j = 1, \ldots, n$.

Beweis. (i) Nach Voraussetzung gilt $a_{ij} = 0$ für $i > j$ und $b_{jk} = 0$ für $j > k$, insbesondere erfüllen die Einträge von $C := AB \in \mathbb{K}^{n\times n}$ deshalb

$$c_{ik} = \sum_{j=1}^{n} a_{ij}b_{jk} = \sum_{j=i}^{k} a_{ij}b_{jk}, \tag{3.1}$$

und es folgt $c_{ik} = 0$ für $i > k$, d.h. $C \in \mathcal{U}$.

(ii) Es sei $b^{(k)} \in \mathbb{K}^n$ die k-te Spalte von $B = A^{-1}$. Dann gilt $Ab^{(k)} = \mathbf{e}_k$. Rückwärtseinsetzen zeigt $b_{jk} = b_j^{(k)} = 0$ für $j > k$. Insgesamt folgt deshalb $B \in \mathcal{U}$. Abschließend folgt aus (3.1) und $\boldsymbol{I} = AB$ sofort $1 = a_{jj}b_{jj}$. ■

Das Lemma überträgt sich offensichtlich auf die Menge

$$\mathcal{L} = \{L \in \mathbb{K}^{n\times n} \mid L \text{ untere Dreiecksmatrix}\},$$

indem man von $L \in \mathcal{L}$ zur transponierten Matrix $L^T \in \mathcal{U}$ übergeht.

Lemma 3.4. *Es sei* $\mathcal{L} = \{L \in \mathbb{K}^{n\times n} \mid L$ *untere Dreiecksmatrix*$\}$. *Dann gelten:*
(i) *Für* $A, B \in \mathcal{L}$ *ist das Produkt* $AB \in \mathcal{L}$.
(ii) *Für eine reguläre Matrix* $A \in \mathcal{L}$ *gilt* $B := A^{-1} \in \mathcal{L}$ *und* $b_{jj} = a_{jj}^{-1}$ *für alle* $j = 1, \ldots, n$. ■

3.2 LU-Zerlegung nach Crout

Im ganzen Abschnitt sei $A \in \mathbb{K}^{n\times n}$ eine reguläre Matrix und $b \in \mathbb{K}^n$. Das Ziel ist die Faktorisierung $A = LU$ von A in eine untere Dreiecksmatrix L und eine obere Dreiecksmatrix U, sodass das Gleichungssystem $Ax = b$ mit den Algorithmen aus Abschnitt 3.1 in zwei Schritten gelöst werden kann:

- Berechne die Lösung y von $Ly = b$.
- Berechne die Lösung x von $Ux = y$.

Dann gilt insgesamt $Ax = b$. Der erste Schritt verstärkt einen etwaigen relativen Fehler in b mit dem Faktor $\mathrm{cond}(L)$. Im zweiten Schritt wird der relative Fehler in y mit $\mathrm{cond}(U)$ verstärkt. Insgesamt kann sich ein relativer Fehler in b also mit $\mathrm{cond}(L)\,\mathrm{cond}(U)$ auf den relativen Fehler in x auswirken. Deshalb ist diese Lösungsstrategie nur dann stabil, wenn $\mathrm{cond}(A) \approx \mathrm{cond}(L)\,\mathrm{cond}(U)$. Im allgemeinen gilt aber lediglich $\mathrm{cond}(A) \le \mathrm{cond}(L)\,\mathrm{cond}(U)$.

Definition. Eine Faktorisierung $A = LU$ mit $L \in \mathbb{K}^{n\times n}$ unterer Dreiecksmatrix und $U \in \mathbb{K}^{n\times n}$ oberer Dreiecksmatrix heißt **LU-Zerlegung** von A. □

Bemerkung. Da die Matrix A insgesamt n^2 Einträge hat, eine LU-Zerlegung aber $n^2+n = n(n+1)$ Einträge, kann man die Eindeutigkeit der LU-Zerlegung nur erwarten, wenn man n Zusatzbedingungen an L und U stellt. In der Regel fordert man die **Normalisierung** $\ell_{jj} = 1$ für alle $j = 1, \ldots, n$.

Satz 3.5. *Für die Matrix $A \in \mathbb{K}^{n\times n}$ sind die folgenden Aussagen äquivalent:*
(i) *Alle* **Untermatrizen** $A_k := (a_{ij})_{i,j=1}^k \in \mathbb{K}^{k\times k}$ *sind regulär.*
(ii) *Es existiert eine LU-Zerlegung von A.*
In diesem Fall hat A eine eindeutige **normalisierte LU-Zerlegung** *$A = LU$ mit $\ell_{jj} = 1$ für alle $j = 1, \ldots, n$.*

Beweis. (ii) ⇒ (i): Da A regulär ist, sind L und U regulär und insbesondere L_k und U_k regulär. Aus $A = LU$ folgt insbesondere $A_k = L_kU_k$ für die entsprechenden Untermatrizen, also ist auch A_k regulär. □
(i) ⇒ (ii) wird durch Induktion nach n bewiesen. Wir zeigen, dass eine eindeutige normalisierte LU-Zerlegung existiert. Der Induktionsanfang $n = 1$ ist klar. Im Induktionsschritt existieren eindeutige Dreiecksmatrizen L_{n-1}, U_{n-1} mit $A_{n-1} = L_{n-1}U_{n-1}$ und $\ell_{jj} = 1$ für alle $j = 1, \ldots, n-1$. Mit dem Ansatz

$$A = \begin{pmatrix} A_{n-1} & b \\ c^T & a_{nn} \end{pmatrix}$$

für geeignete $b, c \in \mathbb{K}^{n-1}$ und $a_{nn} \in \mathbb{K}$ ist zu zeigen, dass eindeutige $\ell, u \in \mathbb{K}^{n-1}$ und $\rho \in \mathbb{K}$ existieren, sodass gilt

$$\begin{pmatrix} A_{n-1} & b \\ c^T & a_{nn} \end{pmatrix} = \begin{pmatrix} L_{n-1} & 0 \\ \ell^T & 1 \end{pmatrix} \begin{pmatrix} U_{n-1} & u \\ 0 & \rho \end{pmatrix}. \tag{3.2}$$

Unter der Induktionsvoraussetzung ist (3.2) äquivalent zu den drei linearen Gleichungen

$$b = L_{n-1}u, \quad c = U_{n-1}^T\ell \quad \text{und} \quad a_{nn} = \ell^T u + \rho. \tag{3.3}$$

Da L_{n-1} und U_{n-1} regulär sind, existieren eindeutige Vektoren $\ell, u \in \mathbb{K}^{n-1}$ als Lösungen der ersten beiden Gleichungen von (3.3), und auch die eindeutige Existenz von ρ folgt. ■

Beispiel. Die Matrix $\begin{pmatrix} 0 & 1 \\ 1 & 0 \end{pmatrix}$ hat keine LU-Zerlegung. ■

Beispiel. Es sei $A \in \mathbb{K}^{n\times n}$ eine **positiv definite Matrix**, d.h. $(Ax)\cdot x > 0$ für alle $x \in \mathbb{K}^n\backslash\{0\}$. Dann sind alle Untermatrizen ebenfalls positiv definit. Da positiv definite Matrizen injektiv und deshalb regulär sind, besitzt A eine LU-Zerlegung. ■

Beispiel. Es sei $A \in \mathbb{K}^{n\times n}$ eine **strikt diagonaldominante Matrix**, d.h.

$$\sum_{\substack{k=1\\k\neq j}}^{n} |a_{jk}| < |a_{jj}| \quad \text{für alle } j = 1,\dots,n.$$

Dann sind alle Untermatrizen A_k ebenfalls strikt diagonaldominant. Da strikt diagonaldominante Matrizen injektiv und deshalb regulär sind, besitzt A eine LU-Zerlegung. Wir zeigen die Injektivität: Für $x \in \mathbb{K}^n$ und den Index j mit $\|x\|_\infty = |x_j|$ gilt

$$\|Ax\|_\infty \geq |(Ax)_j| = \Big|\sum_{k=1}^{n} a_{jk}x_k\Big| \geq |a_{jj}||x_j| - \sum_{k\neq j} |a_{jk}||x_k| \geq \Big(|a_{jj}| - \sum_{k\neq j}|a_{jk}|\Big)\|x\|_\infty.$$

Wegen $\lambda_j := \big(|a_{jj}| - \sum\limits_{k\neq j}|a_{jk}|\big) > 0$ folgt wie behauptet $\mathrm{Kern}(A) = \{0\}$. Der Beweis zeigt ferner

$$\inf_{\|x\|_\infty=1} \|Ax\|_\infty \geq \min_{1\leq j\leq n} \lambda_j =: \lambda_0$$

und deshalb $\|A^{-1}\|_\infty \leq \lambda_0^{-1}$. ■

Bemerkung. Das Verfahren, $Ax = b$ mittels LU-Zerlegung zu lösen, ist nicht stabil, wenn die Konditionszahl von L oder U wesentlich schlechter ist als die von A. Dies ist der sogenannte **Standardfehler der Numerik**: Man zerlegt das Ausgangsproblem ϕ in zwei Teilprobleme $\phi = \phi_2 \circ \phi_1$, von denen eines erheblicher schlechter konditioniert ist als das Gesamtproblem.

Beispiel. Wir betrachten die Matrix $A = \begin{pmatrix} \varepsilon & 1 \\ 1 & 0 \end{pmatrix}$ mit $\varepsilon > 0$ klein. Elementare Rechnung zeigt $A^{-1} = \begin{pmatrix} 0 & 1 \\ 1 & -\varepsilon \end{pmatrix}$. Für die Zeilensummennorm gilt demnach $\|A\|_\infty = 1+\varepsilon = \|A^{-1}\|_\infty$, also $\mathrm{cond}_\infty(A) = (1+\varepsilon)^2 \approx 1$. Berechnet man die normalisierte LU-Zerlegung $A = LU$, so erhält man

$$L = \begin{pmatrix} 1 & 0 \\ 1/\varepsilon & 1 \end{pmatrix}, \quad L^{-1} = \begin{pmatrix} 1 & 0 \\ -1/\varepsilon & 1 \end{pmatrix}, \quad U = \begin{pmatrix} \varepsilon & 1 \\ 0 & -1/\varepsilon \end{pmatrix}, \quad U^{-1} = \begin{pmatrix} 1/\varepsilon & 1 \\ 0 & -\varepsilon \end{pmatrix}.$$

Es gelten also $\|L\|_\infty = 1+1/\varepsilon = \|L^{-1}\|_\infty$, $\|U\|_\infty = 1/\varepsilon$ und $\|U^{-1}\|_\infty = 1+1/\varepsilon$, und es folgt $\mathrm{cond}_\infty(L) = (1+1/\varepsilon)^2$, $\mathrm{cond}_\infty(U) = (1+1/\varepsilon)/\varepsilon$. ■

Algorithmus 3.6: Berechnung der LU-Zerlegung nach Crout

Input: Matrix $A \in \mathbb{K}^{n\times n}$ mit LU-Zerlegung

```
for i=1:n
   for k=i:n
      u_ik = a_ik − Σ_{j=1}^{i−1} ℓ_ij u_jk
   end
   for k=i+1:n
      ℓ_ki = (a_ki − Σ_{j=1}^{i−1} ℓ_kj u_ji)/u_ii
   end
end
```

Output: nicht-triviale Einträge der Matrizen $L, U \in \mathbb{K}^{n\times n}$, d.h. ℓ_{jk} für $j > k$ und u_{jk} für $j \le k$.

Behauptung. Der Crout-Algorithmus ist wohldefiniert und berechnet in (asymptotisch) $2n^3/3$ arithmetischen Operationen die nicht-trivialen Einträge der Matrizen L und U der normalisierten LU-Zerlegung. Die berechneten Werte für u_{ik} und ℓ_{ki} können a_{ik} und a_{ki} überschreiben. Dadurch wird kein weiterer Speicherplatz zur Speicherung von L und U benötigt, d.h.

$$\begin{pmatrix} a_{11} & a_{12} & a_{13} & \dots & a_{1n} \\ a_{21} & a_{22} & a_{23} & \dots & a_{2n} \\ a_{31} & a_{32} & a_{33} & \dots & a_{3n} \\ \vdots & & \ddots & \ddots & \vdots \\ a_{n1} & \dots & \dots & a_{n,n-1} & a_{nn} \end{pmatrix} \quad \mapsto \quad \begin{pmatrix} u_{11} & u_{12} & u_{13} & \dots & u_{1n} \\ \boldsymbol{\ell_{21}} & u_{22} & u_{23} & \dots & u_{2n} \\ \boldsymbol{\ell_{31}} & \boldsymbol{\ell_{32}} & u_{33} & \dots & u_{3n} \\ \vdots & & \ddots & \ddots & \vdots \\ \boldsymbol{\ell_{n1}} & \dots & \dots & \boldsymbol{\ell_{n,n-1}} & u_{nn} \end{pmatrix}$$

Beweis der Wohldefiniertheit. Der Algorithmus arbeitet mittels einer **Parkettierung** von A: Die Matrix wird wie folgt eingeteilt

$$\begin{pmatrix} 1a & 1a & 1a & \dots & 1a \\ 1b & 2a & 2a & \dots & 2a \\ 1b & 2b & 3a & \dots & 3a \\ 1b & 2b & 3b & & \\ \vdots & \vdots & \vdots & \textbf{etc.} & \\ 1b & 2b & 3b & & \end{pmatrix}$$

In dieser Einteilung vertreten die Ziffern $1, 2, 3, \dots$ die äußere i-Schleife und die beiden Literale a, b die erste bzw. zweite innere k-Schleife in Algorithmus 3.6.

Mit der Faktorisierung $A = LU$ und $\ell_{ii} = 1$ gilt für $i \le k$

$$a_{ik} = \sum_{j=1}^{n} \ell_{ij} u_{jk} = \sum_{j=1}^{i} \ell_{ij} u_{jk} = u_{ik} + \sum_{j=1}^{i-1} \ell_{ij} u_{jk} \tag{3.4}$$

Für $i > k$ gilt

$$a_{ki} = \sum_{j=1}^{n} \ell_{kj} u_{ji} = \sum_{j=1}^{i} \ell_{kj} u_{ji} = \ell_{ki} u_{ii} + \sum_{j=1}^{i-1} \ell_{kj} u_{ji}. \tag{3.5}$$

Die erste Formel ist nach u_{ik} auflösbar, die zweite kann wegen $u_{ii} \neq 0$ nach ℓ_{ki} aufgelöst werden. Dass alle auftretenden Summanden $\ell_{ij} u_{jk}$ bzw. $\ell_{kj} u_{ji}$ bereits berechnet sind, entnehme man der Parkettierung. ■

Beweis des arithmetischen Aufwands. Fixiert man i und k, so erfordert die Berechnung von u_{ik} jeweils $i-1$ Multiplikationen und Subtraktionen. Die Berechnung von ℓ_{ki} erfordert eine Division sowie jeweils $i-1$ Multiplikationen und Subtraktionen. Im i-ten Schritt des Algorithmus fallen also

$$(n-i+1)\,2(i-1) + (n-i)\,(2(i-1)+1) = (4n+5)i - 4i^2 - (3n+2)$$

arithmetische Operationen an. Also ergeben sich insgesamt

$$\begin{aligned}(4n+5)\sum_{i=1}^{n} i - 4\sum_{i=1}^{n} i^2 - n(3n+2) &= (4n+5)\,\frac{n(n+1)}{2} - 4\,\frac{n(n+1)(2n+1)}{6} - n(3n+2) \\ &= 2\,n^3 - \frac{4}{3}n^3 + \mathcal{O}(n^2) = \frac{2}{3}\,n^3 + \mathcal{O}(n^2)\end{aligned}$$

arithmetische Operationen. Bei Aufwandsangaben wird in der Regel nur der höchste n-Potenz berücksichtigt, d.h. der Aufwand an arithmetischen Operationen für den Crout-Algorithmus beträgt *asymptotisch* $2n^3/3$. ■

Übung. Eine Matrix $A \in \mathbb{K}^{n\times n}$ heißt **Bandmatrix** mit **oberer Bandbreite** $q \in \mathbb{N}_0$ und **unterer Bandbreite** $p \in \mathbb{N}_0$, wenn gilt

$$a_{ik} = 0 \quad \text{für } k + p < i \text{ oder } k > i + q.$$

In diesem Fall gilt also schematisch

$$A = \begin{pmatrix} a_{11} & \dots & a_{1,q+1} & \mathbf{0} & \dots & \mathbf{0} \\ \vdots & \ddots & & \ddots & \ddots & \vdots \\ a_{p+1,1} & & \ddots & & \ddots & \mathbf{0} \\ \mathbf{0} & \ddots & & \ddots & & a_{n-q,n} \\ \vdots & \ddots & \ddots & & \ddots & \vdots \\ \mathbf{0} & \dots & \mathbf{0} & a_{n,n-p} & \dots & a_{nn} \end{pmatrix}.$$

Besitzt die Bandmatrix A eine LU-Zerlegung, so folgere man mit Hilfe von (3.4)–(3.5), dass L untere Bandbreite p und U obere Bandbreite q hat. Der Crout-Algorithmus sowie das Lösen der Dreieckssysteme für Bandmatrizen lässt sich leicht daran adaptieren, dass überflüssige (triviale) Operationen entfallen und nur noch

- $\mathcal{O}(nqp)$ Operationen zur Berechnung der LU-Zerlegung,

- $\mathcal{O}(np)$ Operationen zum Lösen von $Ly = b$,
- $\mathcal{O}(nq)$ Operationen zum Lösen von $Ux = y$

anfallen, d.h. man kann ein Eliminationsverfahren mit *linearem Aufwand* (linear in n) realisieren – anstatt *kubischem Aufwand* für die oben formulierte LU-Zerlegung. Dazu beachte man

$$a_{ik} = \sum_{j=j_0}^{\min\{i,k\}} \ell_{ij} u_{jk}$$

für alle $i \in \{1, \dots n\}$, $k \in \{\max(1, i-p), \dots, \min(i+q, n)\}$ und $j_0 := \max\{1, i-p, k-q\}$. ■

LU-Zerlegung in MATLAB

In MATLAB steht die LU-Zerlegung mit dem vorimplementierten Befehl `lu` zur Verfügung. Intern wird jedoch nicht der Crout-Algorithmus, sondern das Gauß-Verfahren benutzt, das im nächsten Abschnitt eingeführt wird.

3.3 LU-Zerlegung und Gauß-Verfahren

Das **Gauß-Verfahren** (auch: **Gauß-Elimination**) ist das klassische Verfahren zur Lösung linearer Gleichungssysteme $Ax = b$ mit regulärer Matrix $A \in \mathbb{K}^{n\times n}$, das man auch verwendet, wenn man das Gleichungssystem händisch löst: Durch elementare Zeilenumformungen wird die Matrix A auf obere Dreiecksgestalt gebracht.

Algorithmus 3.7: Gauß-Verfahren

Input: $A \in \mathbb{K}^{m\times n}$ regulär, $b \in \mathbb{K}^n$

1. Schritt: Wir lassen die erste Zeile der Matrix $A^{(1)} := A$ unverändert und erhalten eine Matrix $A^{(2)}$, indem wir in den übrigen Zeilen den ersten Eintrag a_{i1} eliminieren. Dazu definieren wir $\ell_{i1} := a_{i1}/a_{11}$ für $2 \le i \le n$ sowie

$$a_{ij}^{(2)} = a_{ij} - \ell_{i1} a_{1j}, \quad b_i^{(2)} = b_i - \ell_{i1} b_1 \quad \text{für } i, j = 2, \dots, n.$$

Mit Matrizen schreibt sich dieser Übergang in der Form $A^{(2)} = L^{(1)} A$, $b^{(2)} = L^{(1)} b$ mit

$$A^{(2)} = \begin{pmatrix} a_{11} & a_{12} & \dots & a_{1n} \\ 0 & a_{22}^{(2)} & \dots & a_{2n}^{(2)} \\ \vdots & \vdots & & \vdots \\ 0 & a_{n2}^{(2)} & \dots & a_{nn}^{(2)}. \end{pmatrix}, \quad b^{(2)} = \begin{pmatrix} b_1 \\ b_2^{(2)} \\ \vdots \\ b_n^{(2)} \end{pmatrix}, \quad L^{(1)} = \begin{pmatrix} 1 & & & \\ -\ell_{21} & 1 & & \\ \vdots & & \ddots & \\ -\ell_{n1} & & & 1 \end{pmatrix},$$

wobei alle nicht angegebenen Einträge von $L^{(1)}$ trivial sind.

2. Schritt: Wir lassen die ersten beiden Zeilen der Matrix $A^{(2)}$ unverändert und erhalten eine

Matrix $A^{(3)}$, indem wir in den übrigen Zeilen den zweiten Eintrag $a_{i2}^{(2)}$ eliminieren. Dazu definieren wir $\ell_{i2} := a_{i2}^{(2)}/a_{22}^{(2)}$ für $3 \le i \le n$ sowie

$$a_{ij}^{(3)} = a_{ij}^{(2)} - \ell_{i2} a_{2j}^{(2)}, \quad b_i^{(3)} = b_i^{(2)} - \ell_{i2} b_2^{(2)} \quad \text{für } i,j = 3,\dots,n.$$

Mit Matrizen schreibt sich dieser Übergang in der Form $A^{(3)} = L^{(2)} A^{(2)}$, $b^{(3)} = L^{(2)} b^{(2)}$ mit

$$A^{(3)} = \begin{pmatrix} a_{11} & a_{12} & a_{13} & \dots & a_{1n} \\ 0 & a_{22}^{(2)} & a_{23}^{(2)} & \dots & a_{2n}^{(2)} \\ 0 & 0 & a_{33}^{(3)} & \dots & a_{3n}^{(3)} \\ \vdots & \vdots & \vdots & & \vdots \\ 0 & 0 & a_{n3}^{(3)} & \dots & a_{nn}^{(3)}. \end{pmatrix}, \quad b^{(3)} = \begin{pmatrix} b_1 \\ b_2^{(2)} \\ b_3^{(3)} \\ \vdots \\ b_n^{(3)} \end{pmatrix}, \quad L^{(2)} = \begin{pmatrix} 1 & & & & \\ & 1 & & & \\ & -\ell_{32} & 1 & & \\ & \vdots & & \ddots & \\ & -\ell_{n2} & & & 1 \end{pmatrix},$$

k-ter Schritt: Vor dem Eliminationsschritt gilt

$$A^{(k)} = \begin{pmatrix} a_{11} & \dots & \dots & \dots & a_{1n} \\ 0 & \ddots & & & \vdots \\ \vdots & 0 & a_{kk}^{(k)} & \dots & a_{kn}^{(k)} \\ \vdots & \vdots & \vdots & & \vdots \\ 0 & 0 & a_{nk}^{(k)} & \dots & a_{nn}^{(k)} \end{pmatrix}, \quad b^{(k)} = \begin{pmatrix} b_1 \\ \vdots \\ b_k^{(k)} \\ \vdots \\ b_n^{(k)} \end{pmatrix}.$$

Mit $\ell_{ik} := a_{ik}^{(k)}/a_{kk}^{(k)}$ für $i = k+1,\dots,n$ und der Eliminationsmatrix

$$L^{(k)} = \begin{pmatrix} 1 & & & & \\ & \ddots & & & \\ & & 1 & & \\ & & -\ell_{k+1,k} & 1 & \\ & & \vdots & & \ddots \\ & & -\ell_{nk} & & 1 \end{pmatrix}$$

definieren wir $A^{(k+1)} = L^{(k)} A^{(k)}$ sowie $b^{(k+1)} = L^{(k)} b^{(k)}$. Die Multiplikation $A^{(k+1)} = L^{(k)} A^{(k)}$ eliminiert gerade die Einträge $a_{jk}^{(k)}$ für $j = k+1,\dots,n$, d.h. es gilt $a_{jk}^{(k+1)} = 0$.

Output: Nach $n-1$ Schritten erhalten wir eine obere Dreiecksmatrix $U := A^{(n)} \in \mathbb{K}^{n\times n}$, eine untere Dreiecksmatrix L mit $\ell_{jj} := 1$ sowie die transformierte rechte Seite $y := b^{(n)} \in \mathbb{K}^n$.

Satz 3.8. *Ist $A \in \mathbb{K}^{n\times n}$ eine reguläre Matrix und $b \in \mathbb{K}^n$, so ist das Gauß-Verfahren genau dann durchführbar, wenn die Matrix A eine LU-Zerlegung besitzt. In diesem Fall berechnet das Verfahren die normalisierte LU-Zerlegung $A = LU$ mit $U = A^{(n)}$ und liefert ferner die modifizierte rechte Seite $y = b^{(n)} = L^{-1}b$. Man erhält also die Lösung $x \in \mathbb{K}^n$ von $Ax = b$, indem man das Gleichungssystem $Ux = y$ löst.*

Beweis. Zunächst nehmen wir an, dass A eine LU-Zerlegung besitzt. Um zu zeigen, dass das Gauß-Verfahren wohldefiniert ist, müssen wir nur zeigen, dass in allen Schritten $a_{kk}^{(k)} \neq 0$ gilt. Wir betrachten die k-te Untermatrix $A_k^{(k)}$ von $A^{(k)} = L^{(k-1)} \cdots L^{(1)} A$,

$$A_k^{(k)} = \begin{pmatrix} a_{11} & a_{12} & \dots & a_{1k} \\ 0 & a_{22}^{(2)} & \dots & a_{2k}^{(2)} \\ \vdots & \ddots & \ddots & \vdots \\ 0 & \dots & 0 & a_{kk}^{(k)} \end{pmatrix}$$

Mit den entsprechenden Untermatrizen gilt $A_k^{(k)} = L_k^{(k-1)} \cdots L_k^{(1)} A_k$. Da die $L_k^{(j)}$ regulär sind, gilt also $\operatorname{rg}(A_k^{(k)}) = \operatorname{rg}(A_k)$. Nach Satz 3.5 über die LU-Zerlegung hat A_k vollen Rang, und deshalb gilt $a_{kk}^{(k)} \neq 0$.
Ist andererseits das Gauß-Verfahren durchführbar, so ist

$$U := A^{(n)} = L^{(n-1)} \cdots L^{(1)} A \in \mathbb{K}^{n\times n}$$

eine obere Dreiecksmatrix, und es ist nur noch zu zeigen, dass $L := (L^{(n-1)} \cdots L^{(1)})^{-1}$ eine normalisierte untere Dreiecksmatrix ist. Mit dem Vektor $\boldsymbol{\ell}_k := (0, \dots, 0, \ell_{k+1,k}, \dots, \ell_{nk}) \in \mathbb{K}^n$ gilt $L^{(k)} = \boldsymbol{I} - \boldsymbol{\ell}_k \mathbf{e}_k^T$. Nutzt man $\mathbf{e}_j^T \boldsymbol{\ell}_k = 0$ für $j \leq k$, so folgt $L^{(k)}(\boldsymbol{I} + \boldsymbol{\ell}_k e_k^T) = \boldsymbol{I}$, d.h.

$$(L^{(k)})^{-1} = \boldsymbol{I} + \boldsymbol{\ell}_k \mathbf{e}_k^T = \begin{pmatrix} 1 & & & & \\ & \ddots & & & \\ & & 1 & & \\ & & \ell_{k+1,k} & 1 & \\ & & \vdots & & \ddots \\ & & \ell_{nk} & & 1 \end{pmatrix}.$$

Mit vollständiger Induktion zeigt man schließlich

$$L := (L^{(n-1)} \cdots L^{(1)})^{-1} = (L^{(1)})^{-1} \cdots (L^{(n-1)})^{-1} = \boldsymbol{I} + \sum_{j=1}^{k} \boldsymbol{\ell}_j \mathbf{e}_j^T,$$

und aus der letzten Gleichheit folgt sofort

$$L = \begin{pmatrix} 1 & 0 & \dots & 0 \\ \ell_{21} & 1 & \ddots & \vdots \\ \vdots & \ddots & \ddots & 0 \\ \ell_{n1} & \dots & \ell_{n,n-1} & 1, \end{pmatrix},$$

d.h. $A = LU$ ist die normalisierte LU-Zerlegung von A.
Insbesondere gelten $U = A^{(n)} = L^{-1}A$ und $y := b^{(n)} = L^{-1}b$, d.h. die (eindeutige) Lösung von $Ax = b$ ist auch die eindeutige Lösung von $Ux = y$. ■

Algorithmus 3.9: Gauß-Elimination

Input: Matrix $A \in \mathbb{K}^{n\times n}$ mit LU-Zerlegung, rechte Seite $b \in \mathbb{K}^n$

```
for k=1:n-1
   for i=k+1:n
      ℓ_ik = a_ik^(k) / a_kk^(k)
      b_i^(k+1) = b_i^(k) − ℓ_ik b_k^(k)
      for j=k+1:n
         a_ij^(k+1) = a_ij^(k) − ℓ_ik a_kj^(k)
      end
   end
end
```

Output: nicht-triviale Einträge der Matrizen $L, U \in \mathbb{K}^{n\times n}$ mit $u_{ij} := a_{ij}^{(i)}$, sowie modifizierte rechte Seite $y \in \mathbb{K}^n$ mit $y_i := b_i^{(i)}$.

Die Verifikation, dass dieser Pseudo-Code die folgenden Eigenschaften hat, sei dem Leser zur Übung überlassen.

Behauptung. Algorithmus 3.9 berechnet in asymptotisch $2n^3/3$ arithmetischen Operationen die normalisierte LU-Zerlegung von A mit $U := A^{(n)}$ sowie die modifizierte rechte Seite $y := b^{(n)} = L^{-1}b$. Bei einer Implementierung werden einfach die oberen Indizes weggelassen, d.h. A und b werden überschrieben, und ℓ_{ik} überschreibt a_{ik}. Dadurch wird kein weiterer Speicherplatz benötigt. ■

Die Gauß-Elimination ist (wie die LU-Zerlegung) nicht für jede reguläre Matrix durchführbar. Um einen Algorithmus zu erhalten, der für jede reguläre Matrix durchführbar ist, erweitern wir das Gauß-Verfahren um eine **Pivot-Strategie**: Dabei wird der Gauß-Algorithmus 3.9 in der k-Schleife vor der i-Schleife wie folgt erweitert:

- Bestimme den Index $p = p(k) \in \{k, \ldots, n\}$ mit $|a_{pk}^{(k)}| = \max\limits_{i=k,\ldots,n} |a_{ik}^{(k)}|$.
- Vertausche Zeilen p und k in $(A^{(k)}, b^{(k)})$ und erhalte $(\widetilde{A}^{(k)}, \widetilde{b}^{(k)})$.
- Führe den Eliminationsschritt für $(\widetilde{A}^{(k)}, \widetilde{b}^{(k)})$ aus und erhalte $(A^{(k+1)}, b^{(k+1)})$.

Bei einer Implementierung wird natürlich keine Zeile wirklich vertauscht, d.h. es wird kein Speicher kopiert, sondern es wird mit einem zusätzlichen **Buchhaltungsvektor** $\pi \in \mathbb{N}^n$ gearbeitet:

- Anfangs gilt $\pi = (1, \ldots, n)$.
- Beim Vertauschen wird lediglich der Inhalt von $\pi(p)$ und $\pi(k)$ vertauscht.

Gegenüber dem einfachen Gauß-Verfahren müssen also lediglich die Zeilenindizes geändert werden: Statt a_{ij} verwenden wir nun $a_{\pi(i),j}$ etc., vgl. Algorithmus 3.11.

Bemerkung. Man beachte, dass das Vertauschen der Zeilen p und k gerade der Matrizenmultiplikation $\widetilde{A}^{(k)} = P^{(k)} A^{(k)}$ mit der elementaren **Permutationsmatrix**

$$P^{(k)} = \begin{pmatrix} 1 & & & & & & & & \\ & \ddots & & & & & & & \\ & & 1 & & & & & & \\ & & & 0 & & & 1 & & \\ & & & & 1 & & & & \\ & & & & & \ddots & & & \\ & & & & & & 1 & & \\ & & & 1 & & & & 0 & \\ & & & & & & & & 1 \\ & & & & & & & \ddots & \\ & & & & & & & & 1 \end{pmatrix} \in \{0,1\}^{n \times n}.$$

entspricht. Hierbei sind gegenüber der Identität $\boldsymbol{I}$ gerade die k-te und p-te Zeile vertauscht im Fall $p \neq k$ und $P^{(k)} = \boldsymbol{I}$ im Fall $p = k$. Offensichtlich ist $P^{(k)}$ regulär mit $P^{(k)} = (P^{(k)})^{-1}$. Der k-te Schritt im Gauß-Verfahren mit Pivotsuche lässt sich formal in der Form

$$A^{(k+1)} = L^{(k)} P^{(k)} A^{(k)}, \quad b^{(k+1)} = L^{(k)} P^{(k)} b^{(k)}$$

schreiben. □

Satz 3.10. *Für jede reguläre Matrix $A \in \mathbb{K}^{n \times n}$ und $b \in \mathbb{K}^n$ ist der Gauß-Algorithmus mit Pivotsuche durchführbar. Ist $P \in \{0,1\}^{n \times n}$ die Permutationsmatrix, die durch den Buchhaltervektor gegeben ist, so wird gerade die normalisierte LU-Zerlegung $PA = LU$ berechnet, und es gilt $|\ell_{ij}| \leq 1$ für alle $1 \leq j \leq i \leq n$. Ferner erhält man die modifizierte rechte Seite $b^{(n)} = L^{-1} P b$.*

Beweis der Wohldefiniertheit. Das Gauß-Verfahren mit Pivotsuche ist genau dann nicht wohldefiniert, wenn es einen Schritt $k = 1, \ldots, n-1$ gibt, sodass $|a_{pk}^{(k)}| = \max_{i=k}^{n} |a_{ik}^{(k)}| = 0$ gilt. In diesem Fall wären also die ersten k Spalten von

$$A^{(k)} = \begin{pmatrix} a_{11} & \ldots & \ldots & \ldots & a_{1n} \\ 0 & \ddots & & & \vdots \\ \vdots & 0 & a_{kk}^{(k)} & \ldots & a_{kn}(k) \\ \vdots & \vdots & \vdots & & \vdots \\ 0 & 0 & a_{nk}(k) & \ldots & a_{nn}^{(k)} \end{pmatrix}$$

linear abhängig. Andererseits haben A und $A^{(k)}$ denselben (vollen) Rang, denn $A^{(k)}$ entsteht aus A durch die Multiplikation mit regulären Permutations- und Eliminationsmatrizen. Demnach führte $a_{pk}^{(k)} = 0$ auf einen Widerspruch zur Regularität von A. Also ist das Verfahren wohldefiniert. ■

Beweis der Aussage über L. Für die Koeffizienten von L gilt nach Definition $\ell_{ik} = a_{ik}^{(k)}/a_{pk}^{(k)}$, da gegebenenfalls *vor* der i-Schleife die Zeilen p und k vertauscht werden. Nach Wahl des Index p folgt $|\ell_{ik}| \leq 1$. ■

Beweisidee für Faktorisierung PA = LU. Für den Beweis der Faktorisierung $PA = LU$ mit $P = P^{(n-1)} \cdots P^{(1)}$ betrachten wir die folgende Entwicklung für $k = 2, 3, 4, \ldots$

$$
\begin{aligned}
A^{(2)} &= L^{(1)} P^{(1)} A, \\
A^{(3)} &= L^{(2)} P^{(2)} A^{(2)} = L^{(2)} [P^{(2)} L^{(1)} P^{(2)}][P^{(2)} P^{(1)}] A, \\
A^{(4)} &= L^{(3)} P^{(3)} A^{(3)} = L^{(3)} [P^{(3)} L^{(2)} P^{(3)}][P^{(3)} P^{(2)} L^{(1)} P^{(2)} P^{(3)}][P^{(3)} P^{(2)} P^{(1)}] A,
\end{aligned}
$$

wobei wir lediglich $P^{(k)} P^{(k)} = \boldsymbol{I}$ ausgenutzt haben. Führt man diese Entwicklung fort und definiert

$$
\widehat{L}^{(k)} := P^{(n-1)} \cdots P^{(k+1)} L^{(k)} P^{(k+1)} \cdots P^{(n-1)},
$$

so erhält man

$$
A^{(n)} = \widehat{L}^{(n-1)} \ldots \widehat{L}^{(1)} PA.
$$

$U := A^{(n)}$ ist eine obere Dreiecksmatrix. Aufgrund der Darstellung $L^{(k)} = \boldsymbol{I} - \boldsymbol{\ell}_k \mathbf{e}_k^T$ überlegt man sich leicht, dass $\widehat{L}^{(k)}$ dieselbe Struktur hat wie $L^{(k)}$, d.h.

$$
\widehat{L}^{(k)} = \begin{pmatrix}
1 & & & & & \\
& \ddots & & & & \\
& & 1 & & & \\
& & -\widehat{\ell}_{k+1,k} & 1 & & \\
& & \vdots & & \ddots & \\
& & -\widehat{\ell}_{nk} & & & 1
\end{pmatrix}.
$$

Mit derselben Argumentation wie für das ursprüngliche Gauß-Verfahren definiert

$$
L := (\widehat{L}^{(n-1)} \ldots \widehat{L}^{(1)})^{-1}
$$

eine normalisierte untere Dreiecksmatrix, d.h. $PA = LU$ ist die normalisierte LU-Zerlegung. Ferner gilt offensichtlich $b^{(n)} = \widehat{L}^{(n-1)} \ldots \widehat{L}^{(1)} Pb = L^{-1} Pb$. ■

Bemerkung. Will man $Ax = b$ lösen und kennt die LU-Zerlegung $PA = LU$, so löst man

- $Ly = Pb$,
- $Ux = y$.

Dann gilt $PAx = LUx = Ly = Pb$, also $Ax = b$. Hat man die LU-Zerlegung von PA über das Gauß-Verfahren mit Pivotsuche erhalten, so gilt $y = b^{(n)} = L^{-1}Pb$, d.h. man muss nur $Ux = b^{(n)}$ lösen. □

Für eine große Klasse von Matrizen ist das Gauß-Verfahren mit Pivotsuche ein stabiles Verfahren und wird in der Praxis sehr häufig eingesetzt. Bei Implementierungen wird die Matrix A regelmäßig überschrieben mit den Einträgen für L und U. Der Buchhaltungsvektor $\pi \in \mathbb{N}^n$ wird als Output zurückgegeben, um L und U aus dem überschriebenen A rekonstruieren zu können. Im folgenden Pseudo-Code sind die Schritt-Indizes bereits weggelassen, d.h. die Matrix A wird mit der oberen Dreiecksmatrix $U = A^{(n)}$ überschrieben. Die Einträge der Matrix L sind aber noch angegeben. Bei einer Implementierung überschreiben die ℓ-Terme die entsprechenden a-Terme (d.h. es ersetzt a_{ik} im Code ℓ_{ik} etc.), sodass lediglich für den Vektor π Speicher angelegt werden muss.

Algorithmus 3.11: Gauß-Elimination mit Pivotsuche

Input: reguläre Matrix $A \in \mathbb{K}^{n\times n}$, rechte Seite $b \in \mathbb{K}^n$

$\pi = (1, \ldots, n)$
`for k=1:n-1`
  Suche $p \in \{k, \ldots, n\}$ mit $|a_{pk}^{(k)}| = \max\left\{|a_{ik}^{(k)}| \,\middle|\, i = k, \ldots, n\right\}$.
  Vertausche $\pi(k)$ und $\pi(p)$.
  `for i=k+1:n`
    $\ell_{\pi(i),k} = a_{\pi(i),k}/a_{\pi(k),k}$
    $b_{\pi(i)} = b_{\pi(i)} - \ell_{\pi(i),k} b_{\pi(k)}$
    `for j=k+1:n`
      $a_{\pi(i),j} = a_{\pi(i),j} - \ell_{\pi(i),k} a_{\pi(k),j}$
    `end`

  `end end`

Output: nicht-triviale Einträge der Matrizen $L, U \in \mathbb{K}^{n\times n}$ mit $u_{ij} := a_{\pi(i),j}$, Buchhaltervektor π sowie modifizierte rechte Seite $y_i := b_{\pi(i)}$.

Bemerkung. Die Suche des Pivotelements p zur Bestimmung des Maximums benötigt $\mathcal{O}(n)$ Operationen pro k-Schritt. Es ergibt sich also lediglich ein Zusatzaufwand von $\mathcal{O}(n^2)$, und das Gauß-Verfahren mit Pivotsuche hat asymptotisch weiterhin den Aufwand $2n^3/3$. □

Praktische Übung. Man implementiere das Gauß-Verfahren mit Pivotsuche, wobei A durch L und U überschrieben werde und b durch $b^{(n)} = L^{-1}Pb$. Schließlich schreibe man ein Programm, das das Gleichungssystem $Ux = b^{(n)}$ löse. Man beachte, dass man dazu die Rückwärtssubstitution um die Buchhalter-Strategie erweitern muss. ■

Bemerkung (Fill-In). Falls A Bandstruktur hat, so hat PA im Allgemeinen keine Bandstruktur mehr. Insbesondere haben die Matrizen L und U der LU-Zerlegung von $PA = LU$ im Allgemeinen keine Bandstruktur, sondern sind *voll besetzt*. Im schlimmsten Fall haben wir also Speicheraufwand $\mathcal{O}(n^2)$ zur Speicherung von L, U und P, wogegen die Ausgangsmatrix A mit Aufwand $\mathcal{O}(n)$ gespeichert werden kann. Man bezeichnet dieses Phänomen als *Fill-In* und vermeidet in der Praxis das Gauß-Verfahren mit Pivotsuche für Matrizen mit Bandstruktur. □

Bemerkung (Berechnung der Determinante). Ist die Faktorisierung $PA = LU$ bekannt, so gilt nach Multiplikationsregel für die Determinante $\det(A) = (-1)^\ell \prod_{j=1}^n u_{jj}$, wobei $\ell \in \mathbb{N}_0$ die Anzahl der Zeilenvertauschungen ist, die P durchführt. Es gelten nämlich $\det(L) = 1$, $\det(U) = \prod_{j=1}^n u_{jj}$ sowie $\det(P) = (-1)^\ell$. □

Gauß-Verfahren in MATLAB

In MATLAB wird die LU-Zerlegung mit dem vorimplementierten Befehl `lu` berechnet. Intern wird dazu das Gauß-Verfahren verwendet – wahlweise mit oder ohne Pivotsuche (abhängig von der Anzahl der Ausgabeparameter).

3.4 Cholesky-Zerlegung

Definition. Eine Matrix $A \in \mathbb{K}^{n\times n}$ heißt **SPD-Matrix**, wenn sie selbstadjungiert, d.h. $A = \overline{A}^T$, und positiv definit ist, d.h. $Ax \cdot x > 0$ für alle $x \in \mathbb{K}^n \backslash \{0\}$. □

Da die positive Definitheit der Matrix die Injektivität impliziert, ist jede SPD-Matrix insbesondere regulär. Für SPD-Matrizen gilt die folgende spezielle LU-Zerlegung mit oberer Dreiecksmatrix $U = \overline{L}^T$.

Satz 3.12. *Ist $A \in \mathbb{K}^{n\times n}$ eine SPD-Matrix, so existiert eine eindeutige untere Dreiecksmatrix $L \in \mathbb{K}^{n\times n}$ mit $A = L\overline{L}^T$ und $\ell_{kk} > 0$ für alle $k \in \{1, \dots, n\}$. Diese Faktorisierung heißt **Cholesky-Zerlegung**. Für die Konditionszahlen gilt* $\operatorname{cond}_2(L) = \operatorname{cond}_2(\overline{L}^T) = \sqrt{\operatorname{cond}_2(A)}$.

Beweis in der Übung. ■

Bemerkung. Die Konditionsgleichheit zeigt, dass bei der Cholesky-Zerlegung die Instabilität, die bei der einfachen LU-Zerlegung durch den Standardfehler auftreten konnte, nicht auftreten kann. Man spricht von der *Stabilität der Cholesky-Zerlegung.*

Algorithmus 3.13: Cholesky-Zerlegung

Input: SPD-Matrix $A \in \mathbb{K}^{n\times n}$

```
for k = 1:n
```
$\ell_{kk} = \left(a_{kk} - \sum_{j=1}^{k-1} |\ell_{kj}|^2\right)^{1/2}$

```
  for i = k+1:n
    ℓ_ik = (a_ik − Σ_{j=1}^{k−1} ℓ_ij ℓ̄_kj)/ℓ_kk
  end
end
```

Output: Einträge ℓ_{jk} für $1 \leq k \leq j \leq n$ der Matrix L der Cholesky-Zerlegung.

Behauptung. Algorithmus 3.13 ist wohldefiniert und berechnet in (asymptotisch) $n^3/3$ Operationen die Einträge ℓ_{jk} für $1 \leq k \leq j \leq n$ der Matrix L. Die Berechnung erfordert nur die Einträge a_{jk} für $j \geq k$. Die Elemente ℓ_{jk} können a_{jk} überschreiben, d.h. es wird kein zusätzlicher Speicher benötigt.

Beweis. Wegen $A = L\overline{L}^T$ und $\ell_{kj} = 0$ für $k < j$ gilt für Indizes $1 \leq k \leq i \leq n$

$$a_{ik} = \sum_{j=1}^{n} \ell_{ij}\overline{\ell}_{kj} = \sum_{j=1}^{k} \ell_{ij}\overline{\ell}_{kj}$$

und wegen $\ell_{kk} \in \mathbb{R}_{>0}$ deshalb insbesondere

$$a_{ik} = \ell_{ik}\ell_{kk} + \sum_{j=1}^{k-1} \ell_{ij}\overline{\ell}_{kj} \quad \text{und} \quad a_{kk} = \ell_{kk}^2 + \sum_{j=1}^{k-1} |\ell_{kj}|^2.$$

Auflösen nach ℓ_{ik} und ℓ_{kk} verifiziert die Formeln in Algorithmus 3.13.

Zur Wohldefiniertheit beachte man, dass der Algorithmus *spaltenweise von links nach rechts* durchgeführt wird. Der Spaltenindex j in den Summen läuft aber nur bis $k-1$, wenn k die aktuelle Spalte ist. Außerdem beachte man, dass a_{ik} lediglich zur Berechnung von ℓ_{ik} benötigt wird und deshalb mit dem berechneten Wert überschrieben werden kann. ■

Cholesky-Zerlegung in MATLAB

In MATLAB ist die Cholesky-Zerlegung vorimplementiert und steht mit dem Befehl `chol` zur Verfügung.

3.5 QR-Zerlegung

Definition. Zu einer Matrix $A \in \mathbb{K}^{m\times n}$ mit $m \geq n$ ist die **QR-Zerlegung** eine Faktorisierung $A = QR$ mit einer orthogonalen Matrix $Q \in \mathbb{K}^{m\times m}$ und einer **verallgemeinerten oberen Dreiecksmatrix** $R \in \mathbb{K}^{m\times n}$, d.h. $r_{jk} = 0$ für $j > k$.

Eine verallgemeinerte obere Dreiecksmatrix lässt sich nach Definition gerade in der Form

$$R = \begin{pmatrix} \widetilde{R} \\ 0 \end{pmatrix} \tag{3.6}$$

schreiben mit einer oberen Dreiecksmatrix $\widetilde{R} \in \mathbb{K}^{n\times n}$ und der Nullmatrix $0 \in \mathbb{K}^{(m-n)\times n}$.

Bemerkung. Die QR-Zerlegung wird zur Lösung schlecht konditionierter Gleichungssysteme eingesetzt. Für eine reguläre Matrix $A \in \mathbb{K}^{n\times n}$ mit Faktorisierung $A = QR$ löst man das Gleichungssystem $Ax = b$ dann in der Form $Rx = \overline{Q}^T b$. Wegen $\mathrm{cond}_2(A) = \mathrm{cond}_2(R)$ führt das Verfahren auf keine weitere Fehlerverstärkung aufgrund der Faktorisierung, d.h. die Lösung von $Ax = b$ mittels QR-Zerlegung ist *stabil*.

Bemerkung. Bisweilen bezeichnet man für $A \in \mathbb{K}^{m\times n}$, $m \geq n$, auch eine Faktorisierung $A = \widetilde{Q}\widetilde{R}$ als QR-Zerlegung, wenn $\widetilde{R} \in \mathbb{K}^{n\times n}$ eine obere Dreiecksmatrix ist und die Spalten von $\widetilde{Q} \in \mathbb{K}^{m\times n}$ orthonormal sind. Beide Definitionen sind äquivalent:
(i) Ist $A = QR$ eine QR-Zerlegung gemäß unserer Definition, so partitionieren wir die Matrizen in der Form (3.6) und $Q = (\widetilde{Q} \,|\, V)$ mit $\widetilde{Q} \in \mathbb{K}^{m\times n}$ und $V \in \mathbb{K}^{m\times(m-n)}$. Offensichtlich gilt dann $A = \widetilde{Q}\widetilde{R}$.
(ii) Ist $A = \widetilde{Q}\widetilde{R}$ eine QR-Zerlegung im zweiten Sinn, so definieren wir R gemäß (3.6) und bemerken, dass wir $\widetilde{Q}$ zu einer orthogonalen Matrix $Q = (\widetilde{Q} \,|\, V) \in \mathbb{K}^{m\times m}$ ergänzen können: Die Spalten $q_1, \ldots, q_n \in \mathbb{K}^m$ von $\widetilde{Q}$ sind orthonormal. Jedes Orthonormalsystem kann zu einer Orthonormalbasis $q_1, \ldots, q_m$ von $\mathbb{K}^m$ ergänzt werden. Die Vektoren $q_{n+1}, \ldots q_m$ bilden die Spalten von V.[1]

Die letzte Bemerkung zeigt, dass man im allgemeinen keine Eindeutigkeit der QR-Zerlegung beweisen kann. Zumindest für reguläre Matrizen ist die QR-Zerlegung aber eindeutig, wenn man die Vorzeichen der Diagonale von R vorschreibt:

Satz 3.14. *Für $m = n$ und $A \in \mathbb{K}^{n\times n}$ regulär sowie einen fixierten Vorzeichenvektor $\sigma \in \mathbb{K}^n$, $|\sigma_j| = 1$, existiert eine eindeutige QR-Zerlegung $A = QR$ mit $r_{jj} = \sigma_j |r_{jj}|$ für alle $j = 1, \ldots, n$.*

Beweis der Eindeutigkeit. Es seien $QR = A = \widetilde{Q}\widetilde{R}$ zwei QR-Zerlegungen von A. Da A regulär ist, sind auch $R, \widetilde{R}$ regulär, und es gilt

$$D := \widetilde{Q}^{-1}Q = \widetilde{R}R^{-1}$$

Die Matrix D ist eine orthogonale obere Dreiecksmatrix. Daraus folgt bereits, dass D dann eine Diagonalmatrix ist: Man betrachte die erste Spalte: Es gilt $a_{j1} = 0$ für alle $j > 1$. Da die Spalte euklidische Länge 1 hat, folgt $|a_{11}| = 1$. Da auch die erste Zeile euklidische Länge 1 hat, folgt $a_{1j} = 0$ für alle $j > 1$. Induktives Vorgehen zeigt die Behauptung.

Es gilt also $D = \mathrm{diag}(d_{11}, \ldots, d_{nn})$ mit $|d_{jj}| = 1$. Genaueres Hinschauen, vgl. (3.1), verifiziert $d_{jj} = \widetilde{r}_{jj}/r_{jj}$ und deshalb $|r_{jj}| = |\widetilde{r}_{jj}|$. Gelten nun $r_{jj} = \sigma_j |r_{jj}|$ und $\widetilde{r}_{jj} = \sigma_j |\widetilde{r}_{jj}|$, so folgt $r_{jj} = \widetilde{r}_{jj}$ und deshalb $D = \boldsymbol{I}$. Schließlich erhalten wir $Q = \widetilde{Q}$ und $R = \widetilde{R}$. ■

Die Existenz einer QR-Zerlegung beweisen wir konstruktiv mit Hilfe des Householder-Verfahrens für *jede* Matrix $A \in \mathbb{K}^{m\times n}$ mit $m \geq n$.

Lemma 3.15. (i) *Zu $w \in \mathbb{K}^n$ definiere die Matrix $W := w\overline{w}^T \in \mathbb{K}^{n\times n}$, d.h. $w_{jk} = w_j \overline{w}_k$. Dann gilt $W = \overline{W}^T$ und $Wx = (x \cdot w)w$ für alle $x \in \mathbb{K}^n$.*
(ii) *ist $w \in \mathbb{K}^n$ mit $\|w\|_2 = 1$ so ist $H := \boldsymbol{I} - 2w\overline{w}^T$ selbstadjungiert, d.h. $H = \overline{H}^T$, und involutorisch, d.h. $H^2 = \boldsymbol{I}$. Insbesondere ist H also orthogonal.*

[1] Ein elementarer Beweis: Die $q_1, \ldots, q_n$ sind orthonormal, insb. linear unabhängig. Also existieren Vektoren $\widetilde{q}_{n+1}, \ldots, \widetilde{q}_m$, sodass $q_1, \ldots, q_n, \widetilde{q}_{n+1}, \ldots, \widetilde{q}_m$ eine Basis von $\mathbb{K}^m$ bilden. Mit dem Gram-Schmidt-Verfahren erhält man eine Orthonormalbasis $q_1, \ldots, q_m$ von $\mathbb{K}^m$.

Beweis. (i) Offensichtlich gilt $W = \overline{W}^T$ und

$$(Wx)_j = \sum_{k=1}^{n} w_{jk}x_k = w_j \sum_{k=1}^{n} x_k\overline{w}_k = (x \cdot w)w_j \text{ für alle } j = 1, \ldots, n.$$

(ii) Mit (i) folgt $H = \overline{H}^T$. Ferner zeigt elementare Rechnung

$$H^2 = (\boldsymbol{I} - 2w\overline{w}^T)(\boldsymbol{I} - 2w\overline{w}^T) = \boldsymbol{I} - 4w\overline{w}^T + 4(w\overline{w}^T)(w\overline{w}^T),$$

und wegen $\overline{w}^T w = \|w\|_2^2 = 1$ heben sich die letzten beiden Terme weg. ■

Die Matrix H aus Lemma 3.15 bezeichnet man als **Householder-Transformation**. Geometrisch handelt es sich bei der Householder-Transformation um eine Spiegelung an der Hyperebene $E = \{x \in \mathbb{K}^n \,|\, x \cdot w = 0\}$. Der Vektor w ist definitionsgemäß gerade ein Normalenvektor auf E. Das folgende Lemma zeigt, dass man w so wählen kann, dass ein vorgegebener Vektor x auf ein Vielfaches des Einheitsvektors $\mathbf{e}_1$ abgebildet wird.

Lemma 3.16. *Es seien $x \in \mathbb{K}^n \backslash \text{span}\{e_1\}$ und $\lambda \in \mathbb{K}$ mit $|\lambda| = 1$ und $\lambda\overline{x}_1 = |x_1|$. Definiert man*

$$w := \frac{x + \sigma\mathbf{e}_1}{\|x + \sigma\mathbf{e}_1\|_2} \text{ mit } \sigma := \lambda\|x\|_2,$$

so gilt $\|w\|_2 = 1$ und $Hx := (\boldsymbol{I} - 2\omega\overline{\omega}^T)x = -\sigma\mathbf{e}_1$.

Beweis. Der Vektor w ist wegen $x + \sigma\mathbf{e}_1 \neq 0$ wohldefiniert, denn $x \notin \text{span}\{\mathbf{e}_1\}$. Es gilt

$$\|x + \sigma\mathbf{e}_1\|_2^2 = \|x\|_2^2 + 2\text{Re}\,(\sigma\mathbf{e}_1 \cdot x) + |\sigma|^2 = 2(x + \sigma\mathbf{e}_1) \cdot x,$$

wobei $|\sigma|^2 = \|x\|_2^2 = x \cdot x$ und $\sigma\mathbf{e}_1 \cdot x = \sigma\overline{x}_1 = \lambda\overline{x}_1\|x\|_2 \in \mathbb{R}$ benutzt wurde. Es folgt

$$2w \cdot x = 2\,\frac{(x + \sigma\mathbf{e}_1) \cdot x}{\|x + \sigma\mathbf{e}_1\|_2} = \|x + \sigma\mathbf{e}_1\|_2 \in \mathbb{R}$$

und insbesondere $w \cdot x = x \cdot w$. Damit ergibt sich

$$2w\overline{w}^T x = 2(x \cdot w)w = x + \sigma\mathbf{e}_1.$$

Insgesamt folgt $Hx = x - 2w\overline{w}^T x = -\sigma\mathbf{e}_1$. ■

Bemerkung. (i) Die Wahl von λ als Vorzeichen von x_1 verhindert Auslöschung in dem Fall, dass $|x_1| \approx \|x\|_2$ und $x \notin \text{span}\{\mathbf{e}_1\}$ gelten.
(ii) Geometrisch interpretiert, ist der Vektor w aus Lemma 3.16 gerade der Richtungsvektor der Winkelhalbierenden zwischen x und $\sigma\mathbf{e}_1$.

Algorithmus 3.17: Householder-Verfahren zur Berechnung der QR-Zerlegung

Input: $A \in \mathbb{K}^{m\times n}$, $m \geq n$

1. Schritt: Falls die erste Spalte $a_1 \in \mathbb{K}^m$ von A ein Vielfaches von $\mathbf{e}_1 \in \mathbb{K}^m$ ist, definiere $H := \boldsymbol{I}_{\mathbb{K}^m}$. Anderenfalls wähle eine Householder-Transformation $H \in \mathbb{K}^{m\times m}$ mit $Ha_1 \in \operatorname{span}\{\mathbf{e}_1\}$. Definiere die Matrix

$$Q^{(1)} := H \in \mathbb{K}^{m\times m}$$

und erhalte $A^{(1)} \in \mathbb{K}^{m\times n}$ durch

$$A^{(1)} := Q^{(1)}A = \begin{pmatrix} a_{11}^{(1)} & a_{12}^{(1)} & \dots & a_{1n}^{(1)} \\ 0 & a_{22}^{(1)} & \dots & a_{2n}^{(1)} \\ \vdots & \vdots & & \vdots \\ 0 & a_{m2}^{(1)} & \dots & a_{mn}^{(1)} \end{pmatrix}$$

2. Schritt: Betrachte nun die Matrix $B \in \mathbb{K}^{(m-1)\times(n-1)}$:

$$B = \begin{pmatrix} a_{22}^{(1)} & \dots & a_{2n}^{(1)} \\ \vdots & & \vdots \\ a_{m2}^{(1)} & \dots & a_{mn}^{(1)} \end{pmatrix}$$

Falls die erste Spalte $b_1 \in \mathbb{K}^{m-1}$ von B ein Vielfaches von $\mathbf{e}_1 \in \mathbb{K}^{m-1}$ ist, definiere $H := \boldsymbol{I}_{\mathbb{K}^{m-1}}$. Anderenfalls wähle eine Householder-Transformation $H \in \mathbb{K}^{(m-1)\times(m-1)}$ mit $Hb_1 \in \operatorname{span}\{\mathbf{e}_1\}$. Definiere die Matrix

$$Q^{(2)} := \begin{pmatrix} \boldsymbol{I}_{\mathbb{K}} & 0 \\ 0 & H \end{pmatrix} \in \mathbb{K}^{m\times m}$$

und erhalte $A^{(2)} \in \mathbb{K}^{m\times n}$ durch

$$A^{(2)} := Q^{(2)}A^{(1)} = \begin{pmatrix} a_{11}^{(1)} & a_{12}^{(1)} & a_{13}^{(1)} & \dots & a_{1n}^{(1)} \\ 0 & a_{22}^{(2)} & a_{23}^{(2)} & \dots & a_{2n}^{(2)} \\ 0 & 0 & a_{33}^{(2)} & \dots & a_{3n}^{(2)} \\ \vdots & \vdots & \vdots & & \vdots \\ 0 & 0 & a_{m3}^{(2)} & \dots & a_{mn}^{(2)} \end{pmatrix}$$

Man beachte, dass die Multiplikation mit $Q^{(2)}$ die erste Zeile der Matrix $A^{(1)}$ nicht ändert.
3. Schritt: Betrachte nun die Matrix $B \in \mathbb{K}^{(m-2)\times(n-2)}$:

$$B = \begin{pmatrix} a_{33}^{(2)} & \dots & a_{3n}^{(2)} \\ \vdots & & \vdots \\ a_{m3}^{(2)} & \dots & a_{mn}^{(2)} \end{pmatrix}$$

Falls die erste Spalte $b_1 \in \mathbb{K}^{m-2}$ von B ein Vielfaches von $\mathbf{e}_1 \in \mathbb{K}^{m-2}$ ist, definiere $H := \boldsymbol{I}_{\mathbb{K}^{m-2}}$. Anderenfalls wähle eine Householder-Transformation $H \in \mathbb{K}^{(m-2)\times(m-2)}$ mit $Hb_1 \in \operatorname{span}\{\mathbf{e}_1\}$. Definiere die Matrix

$$Q^{(3)} := \begin{pmatrix} \boldsymbol{I}_{\mathbb{K}^2} & 0 \\ 0 & H \end{pmatrix} \in \mathbb{K}^{m\times m}$$

und erhalte $A^{(3)} := Q^{(3)}A^{(2)} \in \mathbb{K}^{m\times n}$. Man beachte, dass die Multiplikation mit $Q^{(3)}$ die ersten zwei Spalten von $A^{(2)}$ nicht ändert:

$$A^{(3)} := Q^{(3)}A^{(2)} = \begin{pmatrix} a_{11}^{(1)} & a_{12}^{(1)} & a_{13}^{(1)} & a_{14}^{(1)} & \dots & a_{1n}^{(1)} \\ 0 & a_{22}^{(2)} & a_{23}^{(2)} & a_{24}^{(2)} & \dots & a_{2n}^{(2)} \\ 0 & 0 & a_{33}^{(3)} & a_{34}^{(3)} & \dots & a_{3n}^{(3)} \\ 0 & 0 & 0 & a_{44}^{(3)} & \dots & a_{4n}^{(4)} \\ \vdots & \vdots & \vdots & \vdots & & \vdots \\ 0 & 0 & 0 & a_{m4}^{(3)} & \dots & a_{mn}^{(4)} \end{pmatrix}$$

***k*-ter Schritt:** Betrachte die Matrix $B \in \mathbb{K}^{(m-k+1)\times(n-k+1)}$:

$$B = \begin{pmatrix} a_{kk}^{(k-1)} & \dots & a_{kn}^{(k-1)} \\ \vdots & & \vdots \\ a_{mk}^{(k-1)} & \dots & a_{mn}^{(k-1)} \end{pmatrix}$$

Falls die erste Spalte $b_1 \in \mathbb{K}^{m-k+1}$ von B ein Vielfaches von $\mathbf{e}_1 \in \mathbb{K}^{m-k+1}$ ist, definiere $H := \boldsymbol{I}_{\mathbb{K}^{m-k+1}}$. Anderenfalls wähle eine Householder-Transformation $H \in \mathbb{K}^{(m-k+1)\times(m-k+1)}$ mit $Hb_1 \in \operatorname{span}\{\mathbf{e}_1\}$. Definiere die Matrix

$$Q^{(k)} := \begin{pmatrix} \boldsymbol{I}_{\mathbb{K}^{k-1}} & 0 \\ 0 & H \end{pmatrix} \in \mathbb{K}^{m\times m}$$

und erhalte $A^{(k)} := Q^{(k)}A^{(k-1)} \in \mathbb{K}^{m\times n}$. Man beachte, dass bei der Matrix-Multiplikation mit $Q^{(k)}$ die ersten $k-1$ Spalten von $A^{(k-1)}$ nicht verändert werden.

Output: Nach n Schritten erhalte $R := A^{(n)} \in \mathbb{K}^{m\times n}$, $Q := Q^{(1)}\cdots Q^{(n)} \in \mathbb{K}^{m\times m}$.

Behauptung. Für $A \in \mathbb{K}^{m\times n}$ mit $m \geq n$ ist das Householder-Verfahren durchführbar und liefert nach n Schritten eine QR-Zerlegung von A.

Beweis. Die Wohldefiniertheit des Householder-Verfahrens ist offensichtlich, die Matrix $R = A^{(n)}$ ist eine verallgemeinerte obere Dreiecksmatrix. Es gilt

$$R = A^{(n)} = Q^{(n)}A^{(n-1)} = Q^{(n)}Q^{(n-1)}A^{(n-2)} = \cdots = Q^{(n)}\cdots Q^{(1)}A.$$

Da die Matrizen $Q^{(j)}$ involutorisch sind, folgt

$$A = (Q^{(n)}\cdots Q^{(1)})^{-1}R = Q^{(1)}\cdots Q^{(n)}R,$$

und die Matrix $Q = Q^{(1)}\dots Q^{(n-1)}$ ist orthogonal. ■

Bemerkung. Im Fall $m = n$ berechnet das Householder-Verfahren eine QR-Zerlegung von $A \in \mathbb{K}^{n\times n}$ natürlich in $n-1$ Schritten, da die n-te Spalte nicht mehr behandelt werden muss. □

Bemerkung. (i) Bei der Implementierung fallen Matrix-Matrix-Multiplikationen in der Form

$$A^{(k+1)} = \begin{pmatrix} \boldsymbol{I}_{\mathbb{K}^k} & 0 \\ 0 & H \end{pmatrix} \begin{pmatrix} \mathrm{U} & \mathrm{X} \\ 0 & B \end{pmatrix} = \begin{pmatrix} U & X \\ 0 & HB \end{pmatrix}$$

an, wobei $U \in \mathbb{K}^{k\times k}$ eine obere Dreiecksmatrix ist, $X \in \mathbb{K}^{k\times(n-k)}$ eine im Allgemeinen vollbesetzte Matrix und $B \in \mathbb{K}^{(m-k)\times(n-k)}$ der Matrixblock, der umgeformt wird. $H \in \mathbb{K}^{(m-k)\times(m-k)}$ ist entweder die Identität oder eine geeignete Householder-Transformation. Gegebenenfalls treten also Matrix-Matrix-Multiplikationen vom Typ $HB = (\boldsymbol{I} - 2w\overline{w}^T)B$ auf, die in der Form $(\boldsymbol{I} - 2w\overline{w}^T)B = B - wv^T$ mit $v := 2B^T\overline{w}$ realisiert werden. Dieses Vorgehen reduziert den arithmetischen Aufwand wesentlich gegenüber der direkten Matrix-Matrix-Multiplikation (für die man H gegebenenfalls erst noch aufbauen müsste).
(ii) Bei der Realisierung wird zusätzlicher Speicherplatz benötigt: In der Regel speichert man die Diagonalelemente $(a_{11}^{(1)}, \ldots, a_{n-1,n-1}^{(n-1)})$ in einem Vektor und nutzt den Speicherplatz von $(a_{jj}, \ldots, a_{mj})$, um den Vektor $w^{(j)} \in \mathbb{K}^{m-j}$ im j-ten Schritt zu speichern.
(iii) Der asymptotische Aufwand für die Berechnung der QR-Zerlegung einer $n \times n$-Matrix beträgt $4/3\, n^3$, d.h. die Berechnung der QR-Zerlegung ist doppelt so teuer wie die Berechnung der LU-Zerlegung und viermal so teuer wie die Berechnung der Cholesky-Zerlegung. Sie ist aber *stets* berechenbar und stabil.

QR-Zerlegung in MATLAB

In MATLAB steht die QR-Zerlegung mit dem vorimplementierten Befehl `qr` zur Verfügung.

3.6 Lineare Ausgleichsprobleme

Definition. Es seien eine Matrix $A \in \mathbb{K}^{m\times n}$ und eine rechte Seite $b \in \mathbb{K}^m$ gegeben, d.h. das System $Ax = b$ kann unterbestimmt ($n > m$) oder überbestimmt ($n < m$) sein. Das **Lineare Ausgleichsproblem (LAP)** besteht darin, einen Vektor $x \in \mathbb{K}^n$ mit $\|Ax - b\|_2 = \min\limits_{y\in\mathbb{K}^n} \|Ay - b\|_2$ zu finden. □

Beispiel. Gegeben seien Daten (a_j, b_j) für $j = 1, \ldots, m$. Gesucht ist ein Polynom $p(a) = \sum_{k=0}^n x_k a^k$, das $\sum_{j=1}^m |p(a_j) - b_j|^2$ minimiert. Im einfachsten Fall ist eine Ausgleichsgrade gesucht, d.h. $n = 1$. Definiert man die Matrix $A \in \mathbb{K}^{m\times(n+1)}$ sowie die Vektoren $x \in \mathbb{K}^{n+1}$ und $b \in \mathbb{K}^m$ durch

$$A = \begin{pmatrix} 1 & a_1^1 & \ldots & a_1^n \\ \vdots & \vdots & & \vdots \\ 1 & a_m^1 & \ldots & a_m^n \end{pmatrix}, \quad x = \begin{pmatrix} x_0 \\ \vdots \\ x_n \end{pmatrix}, \quad b = \begin{pmatrix} b_1 \\ \vdots \\ b_m \end{pmatrix},$$

so gelten $p(a_j) = \sum_{k=0}^n x_k a_j^k = (Ax)_j$ und $\|Ax - b\|_2^2 = \sum_{j=1}^m |p(a_j) - b_j|^2$, d.h. das Problem lässt sich als lineares Ausgleichsproblem formulieren. ■

Satz 3.18. (i) *Das lineare Ausgleichsproblem hat mindestens eine Lösung $x \in \mathbb{K}^n$.*
(ii) *Ein Vektor $x \in \mathbb{K}^n$ löst das lineare Ausgleichsproblem genau dann, wenn er Lösung der* **Gaußschen Normalgleichungen** $\overline{A}^T Ax = \overline{A}^T b$ *ist.*
(iii) *Für $m \geq n$ und $\mathrm{rg}(A) = n$ hat das lineare Ausgleichsproblem eine eindeutige Lösung $x \in \mathbb{K}^n$.*

Beweis. Der Beweis wird in mehreren Schritten erbracht. Die (ggf. eindeutige) Lösbarkeit von (LAP) wird über die Gaußschen Normalgleichungen bewiesen. Dazu benötigen wir die folgende Orthogonalzerlegung des $\mathbb{K}^m$.

1. Schritt. *Es gilt* $\mathrm{Bild}(A)^\perp = \mathrm{Kern}(\overline{A}^T)$. Definitionsgemäß gelten $\mathrm{Bild}(A) = \{Ax \,|\, x \in \mathbb{K}^n\} \leq \mathbb{K}^m$ und

$$\begin{aligned}\mathrm{Bild}(A)^\perp &= \{y \in \mathbb{K}^m \,|\, \forall z \in \mathrm{Bild}(A) \quad y \cdot z = 0\} = \{y \in \mathbb{K}^m \,|\, \forall x \in \mathbb{K}^n \quad y \cdot Ax = 0\} \\ &= \{y \in \mathbb{K}^m \,|\, \forall x \in \mathbb{K}^n \quad (\overline{A}^T y) \cdot x = 0\} = \{y \in \mathbb{K}^m \,|\, \overline{A}^T y = 0\} \\ &= \mathrm{Kern}(\overline{A}^T).\end{aligned}$$

Um die vorletzte Gleichheit zu erhalten, wähle man $x = \overline{A}^T y \in \mathbb{K}^n$ und beachte $0 = (\overline{A}^T y) \cdot x = \|\overline{A}^T y\|_2^2$. □

2. Schritt. *Das Problem (GNG) hat mindestens eine Lösung* $x \in \mathbb{K}^n$. Wegen

$$\mathbb{K}^m = \mathrm{Bild}(A) \oplus \mathrm{Bild}(A)^\perp$$

existieren zu $b \in \mathbb{K}^m$ (eindeutige) $v \in \mathrm{Bild}(A)$ und $w \in \mathrm{Bild}(A)^\perp$ mit $b = v + w$. Wähle $x \in \mathbb{K}^n$ mit $Ax = v$. Dann gilt $\overline{A}^T b = \overline{A}^T v = \overline{A}^T Ax$. □

3. Schritt. *Für* $m \geq n$ *und* $\mathrm{rg}(A) = n$ *hat (GNG) genau eine Lösung* $x \in \mathbb{K}^n$. Nach Voraussetzung ist A injektiv. Insbesondere ist $\overline{A}^T A$ eine SPD-Matrix, denn für $x \neq 0$ gilt

$$(\overline{A}^T Ax) \cdot x = (Ax) \cdot (Ax) = \|Ax\|_2^2 > 0.$$

Da positiv definite Matrizen regulär sind, folgt die Behauptung. □

4. Schritt. *Jede Lösung* $x \in \mathbb{K}^n$ *von (GNG) löst auch (LAP).* Sei $y \in \mathbb{K}^n$ beliebig. Dann gilt $Ay - Ax = A(y - x) \in \mathrm{Bild}(A)$. Nach Voraussetzung gilt $Ax - b \in \mathrm{Kern}(\overline{A}^T) = \mathrm{Bild}(A)^\perp$, und es folgt mit Satz von Pythagoras

$$\|Ay - b\|_2^2 = \|A(y - x) + (Ax - b)\|_2^2 = \|A(y - x)\|_2^2 + \|Ax - b\|_2^2 \geq \|Ax - b\|_2^2,$$

d.h. x löst auch (LAP). □

5. Schritt. *Jede Lösung* $x \in \mathbb{K}^n$ *von (LAP) ist auch Lösung von (GNG).* Nach Linearer Algebra gibt es (eindeutige) Vektoren $v \in \mathrm{Bild}(A)$ und $w \in \mathrm{Bild}(A)^\perp$ mit $b = v + w$. Nach Satz von Pythagoras gilt für $x \in \mathbb{K}^n$ somit

$$\|Ax - b\|_2^2 = \|Ax - v\|_2^2 + \|w\|_2^2.$$

Ist $x \in \mathbb{K}^n$ eine Lösung von (LAP), so folgt aus $v \in \mathrm{Bild}(A)$ sofort $Ax = v$, denn diese (zulässige) Wahl von x minimiert die rechte Seite der letzten Gleichheit. Wegen $w \in \mathrm{Bild}(A)^\perp = \mathrm{Kern}(\overline{A}^T)$ folgt $\overline{A}^T b = \overline{A}^T (v + w) = \overline{A}^T v = \overline{A}^T Ax$, d.h. x löst auch (GNG). ■

Im Fall $m \geq n$ und $\mathrm{rg}(A) = n$ können wir die eindeutige Lösung des linearen Ausgleichsproblems entweder mit Hilfe einer Cholesky-Zerlegung von $\overline{A}^T A$ oder einer QR-Zerlegung von A berechnen.

Die Stabilitätsanalyse beider Verfahren benötigt als technisches Hilfsmittel die Singulärwertzerlegung, die in Abschnitt 3.7 vorgestellt wird.

Bemerkung. Im Fall $m \geq n$ und $\mathrm{rg}(A) = n$ kann die Cholesky-Zerlegung von $\overline{A}^T A$ verwendet werden, um die eindeutige Lösung von (GNG) und (LAP) zu berechnen. □

Bemerkung. Es sei $m \geq n$, $\mathrm{rg}(A) = n$ und $A = QR$ eine QR-Zerlegung von A. Partitioniere $R \in \mathbb{K}^{m\times n}$ und $\overline{Q}^T b \in \mathbb{K}^m$ in der Form

$$R = \begin{pmatrix} \widetilde{R} \\ 0 \end{pmatrix}, \quad \overline{Q}^T b = \begin{pmatrix} z \\ r \end{pmatrix}$$

mit $\widetilde{R} \in \mathbb{K}^{n\times n}$ *regulärer* oberer Dreiecksmatrix und Vektoren $z \in \mathbb{K}^n$, $r \in \mathbb{K}^{m-n}$. Dann gilt

$$\|Ax - b\|_2^2 = \|Rx - \overline{Q}^T b\|_2^2 = \|\widetilde{R}x - z\|_2^2 + \|r\|_2^2.$$

Da $\widetilde{R}$ regulär ist, ist die eindeutige Lösung $x \in \mathbb{K}^n$ von $\widetilde{R}x = z$ die eindeutige Lösung von (LAP), und es gilt $\min\limits_{y\in\mathbb{K}^n} \|Ay - b\|_2 = \|r\|_2$. □

Allgemein kann man eine Lösung des linearen Ausgleichsproblems über die Pseudo-Inverse von A berechnen. Der Zusammenhang von Pseudo-Inverser und dem Ausgleichsproblem wird im folgenden Abschnitt ausgearbeitet.

Lineare Ausgleichsprobleme in MATLAB

Lineare Ausgleichsprobleme können in MATLAB einfach mittels Backslash-Operator gelöst werden.

3.7 Singulärwertzerlegung

Satz 3.19. (i) *Zu $A \in \mathbb{K}^{m\times n}$ existieren orthogonale Matrizen $U \in \mathbb{K}^{m\times m}$, $V \in \mathbb{K}^{n\times n}$ und eine* **verallgemeinerte Diagonalmatrix** $\Sigma \in \mathbb{R}^{m\times n}$, *d.h. $\Sigma_{jk} = \sigma_j \delta_{jk}$ mit $A = U\Sigma\overline{V}^T$ und $\sigma_1 \geq \sigma_2 \geq \cdots \geq \sigma_{\min\{m,n\}} \geq 0$. Diese Faktorisierung heißt* **Singulärwertzerlegung** *von A. Die σ_j heißen* **Singulärwerte**.
(ii) *Die Matrix $\Sigma \in \mathbb{R}^{m\times n}$ der Singulärwertzerlegung ist eindeutig, σ_j^2 ist Eigenwert von $\overline{A}^T A$.*
(iii) *Gilt $\mathrm{rg}(A) = r$, so folgt $\sigma_1 \geq \cdots \geq \sigma_r > 0 = \sigma_{r+1} = \cdots = \sigma_{\min\{m,n\}}$.*
(iv) *Es gilt $\|A\|_2 = \sigma_1$.*
(v) *Bezeichnen $u_j \in \mathbb{K}^m$ und $v_j \in \mathbb{K}^n$ die j-ten Spalten von U bzw. V, so folgt*

$$A = \sum_{j=1}^{\mathrm{rg}(A)} \sigma_j u_j \overline{v}_j^T.$$

Bemerkung. Nach (ii) ist die numerische Berechnung der Singulärwertzerlegung im wesentlichen ein Eigenwertproblem. Die numerische Lösung von Eigenwertproblemen wird in der Vorlesung *Numerik von Differentialgleichungen* behandelt. □

Beweis von Satz 3.19. (ii) Ist $A = U\Sigma\overline{V}^T$ in Singulärwertzerlegung gegeben, so gilt $\overline{A}^T A = V\Sigma^T\Sigma V^{-1}$, d.h. $\overline{A}^T A$ und $D := \Sigma^T\Sigma$ sind ähnlich und haben deshalb dieselben Eigenwerte. D ist eine Diagonalmatrix $D = \operatorname{diag}(\sigma_1^2, \dots, \sigma_n^2)$ mit $\sigma_j = 0$ für $j \geq \min\{m, n\}$. Durch die Sortierung $\sigma_1 \geq \cdots \geq \sigma_{\min\{m,n\}}$ sind die Einträge der Matrix Σ also eindeutig festgelegt.
(iii) Die Matrix Σ hat offensichtlich denselben Rang wie A.
(iv) folgt offensichtlich aus (ii) und der Charakterisierung der Spektralnorm.
(v) Mit $r = \operatorname{rg}(A)$ gilt

$$A = (U\Sigma)\overline{V}^T = \big(\sigma_1 u_1, \dots, \sigma_r u_r, 0, \dots, 0\big) \begin{pmatrix} \overline{v}_1^T \\ \vdots \\ \overline{v}_n^T \end{pmatrix} = \sum_{j=1}^{r} \sigma_j u_j \overline{v}_j^T,$$

wobei die Schreibweise spalten- bzw. zeilenweise zu verstehen ist.
(i) Die Matrix $\overline{A}^T A \in \mathbb{K}^{n\times n}$ ist selbstadjungiert. Deshalb existiert eine Orthonormalbasis $v_1, \dots, v_n$ von $\mathbb{K}^n$ aus Eigenvektoren zu $\overline{A}^T A$, und die Eigenwerte μ_j zu v_j sind nicht-negativ. Ohne Beschränkung der Allgemeinheit gilt

$$\mu_1 \geq \mu_2 \geq \cdots \geq \mu_r > 0 = \mu_{r+1} = \cdots = \mu_n$$

mit $r = \operatorname{rg}(\overline{A}^T A)$. Definiere $\sigma_j := \sqrt{\mu_j}$ und $S := \operatorname{diag}(\sigma_1, \dots, \sigma_r) \in \mathbb{R}^{r\times r}$. Definiere die orthogonale Matrix $V = (v_1, \dots, v_n) \in \mathbb{K}^{n\times n}$ und partitioniere V in der Form $V = (V_1 \,|\, V_2)$ mit $V_1 \in \mathbb{K}^{n\times r}$ und $V_2 \in \mathbb{K}^{n\times(n-r)}$. Dann gelten

- S ist regulär.
- $\overline{A}^T A v_j = \sigma_j^2 v_j$, da v_j Eigenvektor zum Eigenwert $\mu_j = \sigma_j^2$ ist, also
- $\overline{A}^T A V_1 = V_1 S^2$.

Wir definieren die Matrix $U_1 := AV_1 S^{-1} \in \mathbb{K}^{m\times r}$. Dann gilt

$$\overline{U}_1^T U_1 = S^{-1}\overline{V}_1^T(\overline{A}^T A V_1)S^{-1} = S^{-1}\overline{V}_1^T(V_1 S^2)S^{-1} = \boldsymbol{I}_{\mathbb{K}^{r\times r}}.$$

Also sind die Spalten $u_1, \dots, u_r \in \mathbb{K}^m$ von U_1 orthonormal, d.h. wir können U_1 zu einer orthogonalen Matrix $U = (U_1 \,|\, U_2) \in \mathbb{K}^{m\times m}$ ergänzen. Abschließend ist nur noch

$$\overline{U}^T A V = \begin{pmatrix} S & 0 \\ 0 & 0 \end{pmatrix} =: \Sigma \tag{3.7}$$

zu zeigen. Dann folgt $A = U\Sigma\overline{V}^T$. Wir zeigen diese Gleichheit blockweise:

$$\overline{U}^T A V = \begin{pmatrix} \overline{U}_1^T \\ \overline{U}_2^T \end{pmatrix} A \begin{pmatrix} V_1 & V_2 \end{pmatrix} = \begin{pmatrix} \overline{U}_1^T A V_1 & \overline{U}_1^T A V_2 \\ \overline{U}_2^T A V_1 & \overline{U}_2^T A V_2 \end{pmatrix}$$

Nach Definition von U_1 gilt $AV_1 = U_1 S$, und deshalb folgt $\overline{U}_1^T A V_1 = \overline{U}_1^T U_1 S = S$. Da die Spalten von $U = (U_1 \,|\, U_2)$ orthonormal sind, gilt $\overline{U}_2^T A V_1 = \overline{U}_2^T U_1 S = 0$. Da die Spalten von V_2 Eigenvektoren von $\overline{A}^T A$ zum Eigenwert 0 sind, gilt $\overline{A}^T A V_2 = 0$. Deshalb folgt $0 = \overline{V}_2^T \overline{A}^T A V_2 = (\overline{AV_2})^T A V_2$.

Insgesamt erhalten wir $B := AV_2 = 0$, denn wäre $\mathrm{rg}(B) \geq 1$, so folgte $\overline{B}^T B \neq 0$. Also ist die zweite Block-Spalte von $\overline{U}^T AV$ ebenfalls Null. ■

Korollar 3.20. (i) *Zu $A \in \mathbb{K}^{m\times n}$ existiert eine eindeutige Matrix $A^+ \in \mathbb{K}^{n\times m}$ mit den folgenden vier Eigenschaften:*
(I1) $A^+A = \overline{A^+A}^T$,
(I2) $AA^+ = \overline{AA^+}^T$,
(I3) $AA^+A = A$,
(I4) $A^+AA^+ = A^+$.
Diese heißt **Pseudo-Inverse** *oder* **Moore-Penrose-Inverse** *von A.*
(ii) *Ist $A \in \mathbb{K}^{n\times n}$ regulär, so gilt $A^+ = A^{-1}$.*
(iii) *Ist $\Sigma \in \mathbb{K}^{m\times n}$ verallgemeinerte Diagonalmatrix mit $\Sigma_{jk} = \sigma_j\delta_{jk}$, so ist Σ^+ ebenfalls eine verallgemeinerte Diagonalmatrix, und es gilt*

$$\Sigma^+_{jk} = \tau_j\delta_{jk} \quad \textit{mit} \quad \tau_j = \begin{cases} \sigma_j^{-1}, & \textit{falls } \sigma_j \neq 0, \\ 0, & \textit{sonst.} \end{cases} \tag{3.8}$$

(iv) *Ist $A = U\Sigma\overline{V}^T \in \mathbb{K}^{m\times n}$ als Singulärwertzerlegung gegeben, so gilt $A^+ = V\Sigma^+\overline{U}^T$.*

Beweis. Der Existenzbeweis für die Pseudo-Inverse ergibt sich mit (iii) und (iv) als Folgerung zur Singulärwertzerlegung. Die Eindeutigkeit folgt mit elementarer algebraischer Manipulation aus den Regeln (I1)–(I4).

1. Schritt. *Beweis der Eindeutigkeit.* Es seien U, V Matrizen mit den Eigenschaften (I1)–(I4).

$$\begin{aligned} U &= UAU = U(AVA)U = UA(VAV)(AVA)U = \overline{UA}^T\overline{VA}^TV\overline{AV}^T\overline{AU}^T \\ &= (\overline{AUA}^T\overline{V}^T)V(\overline{V}^T\overline{AUA}^T) = (\overline{A}^T\overline{V}^T)V(\overline{V}^T\overline{A}^T) = (\overline{VA}^TV)\overline{AV}^T = (VAV)AV \\ &= VAV = V \end{aligned}$$

□

2. Schritt. *Beweis der Existenz gemäß* (iii) *und* (iv). Für eine verallgemeinerte Diagonalmatrix rechnet man leicht nach, dass die Matrix Σ^+ aus (3.8) die Eigenschaften (I1)–(I4) besitzt. Daraus lässt sich dann unmittelbar folgern, dass $V\Sigma^+\overline{U}^T$ die Pseudo-Inverse von $A = U\Sigma\overline{V}^T$ ist. □

3. Schritt. *Verallgemeinerung des Begriffs der Inversen.* Offensichtlich erfüllt die Inverse A^{-1} die Eigenschaften (I1)–(I4). ■

Übung. Zu einem Teilraum V von $\mathbb{K}^n$ bezeichnet $V^\perp := \{x \in \mathbb{K}^n \,|\, \forall y \in V \quad x \cdot y = 0\}$ das **orthogonale Komplement**. $V^\perp$ ist ein Teilraum von $\mathbb{K}^n$, und es gilt $\mathbb{K}^n = V \oplus V^\perp$. Insbesondere existiert eine eindeutige lineare Abbildung $P : \mathbb{K}^n \to \mathbb{K}^n$ mit $P|_V = \boldsymbol{I}$ und $\mathrm{Kern}\, P = V^\perp$, bezeichnet als **Orthogonalprojektion** auf V. Für eine Matrix $A \in \mathbb{K}^{m\times n}$ gilt:
(i) A^+A ist die Orthogonalprojektion auf $\mathrm{Kern}(A)^\perp \leq \mathbb{K}^n$.
(ii) AA^+ ist die Orthogonalprojektion auf $\mathrm{Bild}(A) \leq \mathbb{K}^m$. ■

Satz 3.21. *Zu $A \in \mathbb{K}^{m\times n}$ und $b \in \mathbb{K}^m$ sei*

$$\mathcal{A} := \left\{ x \in \mathbb{K}^n \,\middle|\, \|Ax - b\|_2 = \min_{y\in\mathbb{K}^n} \|Ay - b\|_2 \right\} \neq \emptyset$$

die Lösungsmenge zum linearen Ausgleichsproblem. Dann existiert ein eindeutiges $x \in \mathcal{A}$ mit $\|x\|_2 = \min_{y\in\mathcal{A}} \|y\|_2$, die sog. **Minimum-Norm-Lösung** *des linearen Ausgleichsproblems. Die Pseudo-Inverse erfüllt gerade $x = A^+ b$.*

Beweis. Es sei $A = U\Sigma\overline{V}^T$ eine Singulärwertzerlegung, $x \in \mathbb{K}^n$. Dann gilt

$$\|Ax - b\|_2 = \|\Sigma\overline{V}^T x - \overline{U}^T b\|_2.$$

Da $\overline{V}^T$ orthogonal ist, ist x also genau dann Minimum-Norm-Lösung zu (A, b), wenn $z := \overline{V}^T x$ Minimum-Norm-Lösung zu $(\Sigma, \overline{U}^T b)$ ist. Mit $r := \mathrm{rg}(A)$ und $u_j \in \mathbb{K}^m$ der j-ten Spalte von U gilt

$$\|\Sigma z - \overline{U}^T b\|_2^2 = \sum_{j=1}^{r} |\sigma_j z_j - b \cdot u_j|^2 + \sum_{j=r+1}^{m} |b \cdot u_j|^2.$$

Also definiert

$$z \in \mathbb{K}^n, \quad z_j = \begin{cases} \sigma_j^{-1} b \cdot u_j & \text{für } j \leq r, \\ 0 & \text{sonst,} \end{cases}$$

die *eindeutige* Minimum-Norm-Lösung zu $(\Sigma, \overline{U}^T b)$. Damit ist $x = Vz$ die *eindeutige* Minimum-Norm-Lösung zu (A, b). Mit $v_j \in \mathbb{K}^n$ der j-ten Spalte von V gilt

$$x = Vz = \sum_{j=1}^{r} \sigma_j^{-1} (b \cdot u_j) v_j = \Big(\sum_{j=1}^{r} \sigma_j^{-1} v_j \overline{u}_j^T \Big) b = A^+ b,$$

denn die Summe ist eine Darstellung von $A^+ = V\Sigma^+\overline{U}^T$. ■

Singulärwertzerlegung in MATLAB

In MATLAB kann die Singulärwertzerlegung einer Matrix A mit dem Befehl `svd` berechnet werden. Das folgende Beispiel ist eine nette Anwendung der Singulärwertzerlegung für die Kompression von Bildern.

MATLAB-Beispiel: Bildkompression als Anwendung der Singulärwertzerlegung

```
load clown.mat
[U,S,V] = svd(X);
colormap('gray');
for k = 5:5:30
   figure(k);
```

```
    image(U(:,1:k)*S(1:k,1:k)*V(:,1:k)');
end
```

Kondition und Stabilität des linearen Ausgleichsproblems

Über die Pseudo-Inverse lässt sich die Konditionszahl von regulären Matrizen auf rechteckige oder singuläre Matrizen verallgemeinern.

Definition. Es sei $A \in \mathbb{K}^{m\times n}$ eine beliebige Matrix mit $\mathrm{rg}(A) = r$ und Singulärwerten σ_j. Dann bezeichnet $\mathrm{cond}_2(A) := \|A\|_2\|A^+\|_2 = \sigma_1/\sigma_r$ die **Konditionszahl** von A (bezüglich der Spektralnorm). □

Das folgende Lemma untersucht die Kondition des Linearen Ausgleichsproblems. Es zeigt sich, dass der relative Fehler in x sehr groß sein kann, falls der Anteil b_A von b, der im Bild von A liegt, sehr klein ist. Es kann nämlich trotz $\frac{\|b-\widetilde{b}\|_2}{\|b\|_2} \ll 1$ noch $\frac{\|b_A-\widetilde{b}_A\|_2}{\|b_A\|_2} \gg 1$ gelten.

Lemma 3.22. *Es seien $A \in \mathbb{K}^{m\times n}$ und $b, \widetilde{b} \in \mathbb{K}^m$ sowie $x, \widetilde{x} \in \mathbb{K}^n$ Minimum-Norm-Lösungen zu (A, b) bzw. $(A, \widetilde{b})$. Dann gibt es eindeutige $b_A, \widetilde{b}_A \in Bild(A)$ und $b_N, \widetilde{b}_N \in \mathrm{Bild}(A)^\perp = \mathrm{Kern}(\overline{A}^T)$ mit $b = b_A + b_N$ und $\widetilde{b} = \widetilde{b}_A + \widetilde{b}_N$, und es gilt*

$$\frac{\|x-\widetilde{x}\|_2}{\|x\|_2} \le \mathrm{cond}_2(A)\frac{\|b_A-\widetilde{b}_A\|_2}{\|b_A\|_2}.$$

Beweis in der bung. ■

Im vorausgegangenen Abschnitt haben wir gesehen, dass man das lineare Ausgleichsproblem für $m \ge n$ und $\mathrm{rg}(A) = n$ sowohl mit Cholesky-Zerlegung für die Gaußschen Normalgleichungen als auch mit QR-Zerlegung von A lösen kann. Das folgende Lemma zeigt, dass die Lösung von (LAP) mittels Cholesky-Zerlegung von $\overline{A}^T A$ im Allgemeinen nicht stabil ist, denn es gilt $\mathrm{cond}_2(\overline{A}^T A) = \mathrm{cond}_2(A)^2$: Während Rundungsfehler unvermeidlich mit einem Faktor $\mathrm{cond}_2(A)$ verstärkt werden, vgl. Lemma 3.22, können die Rundungsfehler beim Lösen von $\overline{A}^T Ax = \overline{A}^T b$ mit $\mathrm{cond}_2(A)^2$ verstärkt werden.

Lemma 3.23. *Für $A \in \mathbb{K}^{m\times n}$ mit $m \ge n$ gilt* $\mathrm{cond}_2(\overline{A}^T A) = \mathrm{cond}_2(A)^2$.

Beweis. Es sei $A = U\Sigma\overline{V}^T$ eine Singulärwertzerlegung von A. Dann ist $\overline{A}^T A = VD\overline{V}^T$ mit $D := \Sigma^T\Sigma = \mathrm{diag}(\sigma_1^2, \ldots, \sigma_n^2)$ eine Singulärwertzerlegung von $\overline{A}^T A$. Mit $r = \mathrm{rg}(A)$ folgt $\mathrm{cond}_2(\overline{A}^T A) = \sigma_1^2/\sigma_r^2 = \mathrm{cond}_2(A)^2$. ■

Verwendet man die QR-Zerlegung von A zur Lösung des Linearen Ausgleichsproblems, so ist dieses Verfahren stabil, da lediglich Fehlerverstärkung mit Faktor $\mathrm{cond}_2(A)$ auftritt.

Lemma 3.24. *Es sei $A \in \mathbb{K}^{m\times n}$, $m \geq n$ mit gegebener QR-Zerlegung $A = QR$. Es gelte*

$$R = \begin{pmatrix} \widetilde{R} \\ 0 \end{pmatrix}$$

mit einer oberen Dreiecksmatrix $\widetilde{R} \in \mathbb{K}^{n\times n}$. Dann gilt $\mathrm{cond}_2(A) = \mathrm{cond}_2(R) = \mathrm{cond}_2(\widetilde{R})$.

Beweis. Es sei $A = U\Sigma\overline{V}^T$ eine Singulärwertzerlegung von A. Dann ist $R = W\Sigma\overline{V}^T$ mit $W := \overline{Q}^T U \in \mathbb{K}^{m\times m}$ eine Singulärwertzerlegung von R, und es folgt $\mathrm{cond}_2(A) = \mathrm{cond}_2(R)$. Partitioniere

$$Q = (Q_1 \,|\, Q_2), \quad U = (U_1 \,|\, U_2), \quad \Sigma = \begin{pmatrix} \widetilde{\Sigma} \\ 0 \end{pmatrix}$$

mit $Q_1, U_1 \in \mathbb{K}^{m\times n}$, $Q_2, U_2 \in \mathbb{K}^{m\times(m-n)}$ und $\widetilde{\Sigma} \in \mathbb{K}^{n\times n}$. Dann gilt $A = QR = Q_1\widetilde{R}$ und deshalb

$$\widetilde{R} = \overline{Q}_1^T A = \overline{Q}_1^T U\Sigma\overline{V}^T = \overline{Q}_1^T U_1\widetilde{\Sigma}\overline{V}^T,$$

d.h. mit der orthogonalen Matrix $\widetilde{W} := \overline{Q}_1^T U_1 \in \mathbb{K}^{n\times n}$ ist $\widetilde{R} = \widetilde{W}\widetilde{\Sigma}\overline{V}^T$ eine Singulärwertzerlegung von $\widetilde{R}$. Da $\widetilde{\Sigma}$ die volle Diagonale von Σ enthält, folgt $\mathrm{cond}_2(A) = \mathrm{cond}_2(\widetilde{R})$. ■

Kapitel 4

Interpolation

Bei einem **Interpolationsproblem** sind im einfachsten Fall Paare (x_j, y_j) gegeben und *einfache Funktionen* p mit $p(x_j) = y_j$ gesucht. Dabei sind *einfache Funktionen* beispielsweise Polynome, Splines (stückweise Polynome) oder rationale Funktionen. Verwandt, aber mathematisch schwieriger ist das Thema **Approximation**. Gegeben sind dabei eine Norm $\|\cdot\|$ und eine Funktion f, die in der Regel unbekannt ist, z.B. als exakte Lösung einer Differentialgleichung. Gesucht ist wieder eine *einfache Funktion* p, die jetzt aber im Sinne der Norm eine gute Approximation ist, z.B. weil $\|f - p\|$ minimal ist im Vergleich mit anderen einfachen Funktionen.

4.1 Lagrange-Polynominterpolation

Definition. Gegeben seien $n+1$ paarweise verschiedene **Stützstellen** $x_0, \ldots, x_n \in [a, b]$ und zugehörige **Funktionswerte** $y_0, \ldots, y_n \in \mathbb{K}$. Bei der **Lagrange-Interpolationsaufgabe** wird ein Polynom $p \in \mathbb{P}_n := \{\text{Polynome vom Grad } \leq n\}$ gesucht, dessen Werte an den Stützstellen mit den vorgegebenen Funktionswerten übereinstimmen, d.h. $p(x_j) = y_j$ für alle $j = 0, \ldots, n$. Dieses Polynom p bezeichnet man als **Lagrange-Interpolationspolynom**. □

Lemma 4.1. (i) $\mathbb{P}_n$ *ist ein* $\mathbb{K}$*-Vektorraum der Dimension* $\dim \mathbb{P}_n = n + 1$.
(ii) *Die* ***Monome*** $p_j(x) = x^j$ *für* $0 \leq j \leq n$ *bilden eine Basis von* $\mathbb{P}_n$.
(iii) *Die* ***Lagrange-Polynome***

$$L_j(x) := \prod_{\substack{k=0 \\ k \neq j}}^{n} \frac{x - x_k}{x_j - x_k} \in \mathbb{P}_n \quad \text{für } 0 \leq j \leq n$$

bilden eine Basis von $\mathbb{P}_n$*, und mit dem* ***Kronecker-Delta*** δ_{jk} *gilt* $L_j(x_k) = \delta_{jk}$.
(iv) *Die* ***Newton-Polynome****, definiert durch*

$$q_j(x) := \prod_{k=0}^{j-1} (x - x_k) \in \mathbb{P}_j \subseteq \mathbb{P}_n \quad \text{für } 0 \leq j \leq n,$$

bilden eine Basis von $\mathbb{P}_n$.

Beweis. (i), (ii) Die Monome sind linear unabhängig (über $\mathbb{K}$) und spannen den Raum der Polynome auf, d.h. $\mathbb{P}_n = \operatorname{span}\{p_j \mid j = 0, \ldots, n\}$.
(iii) Offensichtlich gilt die Gleichheit $L_j(x_k) = \delta_{jk}$. Also ist nur zu zeigen, dass die Lagrange-Polynome $\{L_0, \ldots, L_n\}$ linear unabhängig sind: Sei $\mu_j \in \mathbb{K}$ mit $\sum_{j=0}^n \mu_j L_j(x) = 0$ für alle $x \in \mathbb{R}$. Durch Einsetzen von $x = x_k$ erhält man $\mu_k = 0$.
(iv) folgt analog zu (iii) durch sukzessives Testen mit $x = x_j$ für $j = 0, \ldots, n$. ■

Übung. Es sei $\{q_0, \ldots, q_n\} \subseteq \mathbb{P}_n$ mit der Eigenschaft $q_j(x_k) = \delta_{jk}$, so folgt schon, dass es sich dabei um die Lagrange-Polynome handeln muss: $q_j = L_j$.

Der folgende Satz zeigt Existenz und Eindeutigkeit des Lagrange-Interpolationspolynoms.

Satz 4.2. (i) *Zu fixierten Stützstellen $a \leq x_0 < \ldots < x_n \leq b$ existiert für alle $y_0, \ldots, y_n \in \mathbb{K}$ ein eindeutiges interpolierendes Polynom $p \in \mathbb{P}_n$ mit der Eigenschaft $p(x_j) = y_j$ für alle $j = 0, \ldots, n$. Dieses wird gegeben durch $p = \sum_{j=0}^n y_j L_j$.*
(ii) *Ist $\{q_0, \ldots, q_n\} \subseteq \mathbb{P}_n$ eine Basis und das Polynom dargestellt als $p = \sum_{j=0}^n \lambda_j q_j$, dann erhält man den Vektor $\lambda := (\lambda_0, \ldots, \lambda_n)$ als eindeutige Lösung des Gleichungssystems*

$$\begin{pmatrix} q_0(x_0) & \ldots & q_n(x_0) \\ \vdots & \ddots & \vdots \\ q_0(x_n) & \ldots & q_n(x_n) \end{pmatrix} \lambda = \begin{pmatrix} y_0 \\ \vdots \\ y_n \end{pmatrix}. \tag{4.1}$$

Die Matrix dieses Gleichungssystems ist regulär. Sie wird als **Vandermonde-Matrix** *bezeichnet.*

Beweis. Offensichtlich ist $p = \sum_{j=0}^n \mu_j L_j$ genau dann Lösung des Interpolationsproblems, wenn $\mu_j = y_j$ gilt. Dies zeigt die eindeutige Existenz des Interpolationspolynoms. Es wird also eine lineare Abbildung $P : \mathbb{K}^{n+1} \to \mathbb{P}_n$ durch die Forderung $(Py)(x_j) = y_j$ für alle $j = 0, \ldots, n$ wohldefiniert. Man definiere ferner die lineare Abbildung T, die ein Polynom an den Stützstellen auswertet,

$$T : \mathbb{P}_n \to \mathbb{K}^{n+1}, \quad p \mapsto \big(p(x_0), \ldots, p(x_n)\big).$$

Es gilt $TP = \boldsymbol{I}$, also ist T bijektiv und invers zu P. Die Matrix in (4.1) ist gerade die darstellende Matrix zur linearen Abbildung T. ■

Bemerkung. Die Kondition der obigen Vandermonde-Matrix hängt *stark* von der gewählten Basis ab. Die Monombasis führt im Allgemeinen auf eine erschreckend große Konditionszahl. □

Häufig ist $y_j = f(x_j)$ das Bild von x_j unter einer Funktion f. Der folgende Satz liefert für hinreichend glatte Funktionen f eine Darstellung des bei der Polynominterpolation auftretenden Fehlers sowie eine elementare Abschätzung.

Satz 4.3. *Es seien die reellwertige Funktion $f \in \mathcal{C}^{n+1}[a, b]$ sowie paarweise verschiedene Stützstellen $x_0, \ldots, x_n \in [a, b]$ gegeben, und $p \in \mathbb{P}_n$ sei das Interpolationspolynom mit $p(x_j) = f(x_j)$ für alle $0 \leq j \leq n$. Es sei $x \in [a, b]$ und $I \subseteq [a, b]$ ein Intervall mit $x_0, \ldots, x_n, x \in I$. Dann existiert ein $\xi \in I$, sodass der Interpolationsfehler an der Stelle x dargestellt werden kann durch*

$$f(x) - p(x) = \frac{f^{(n+1)}(\xi)}{(n+1)!} \prod_{j=0}^n (x - x_j).$$

Beweis. Ohne Beschrnkung der Allgemeinheit gilt $x \neq x_j$ für $0 \leq j \leq n$, da sonst die Aussage trivial ist. Mit dem Polynom

$$\omega(y) := \prod_{j=0}^{n} (y - x_j) \in \mathbb{P}_{n+1}$$

definieren wir eine Funktion $F : I \to \mathbb{R}$ durch

$$F(y) := \big(f(x) - p(x)\big)\,\omega(y) - \big(f(y) - p(y)\big)\,\omega(x).$$

Die Funktion F hat im Intervall I mindestens $n+2$ paarweise verschiedene Nullstellen (bei x_j für $0 \leq j \leq n$ und bei x). Laut Voraussetzung gilt $F \in \mathcal{C}^{n+1}(I)$, und nach dem Mittelwertsatz hat die erste Ableitung $F' \in \mathcal{C}^n(I)$ noch $n+1$ Nullstellen in I. Mittels Induktion folgt weiter, dass $F^{(n+1)}$ noch eine Nullstelle $\xi \in I$ besitzt, d.h.

$$0 = F^{(n+1)}(\xi) = \big(f(x) - p(x)\big)\,(n+1)! - f^{(n+1)}(\xi)\omega(x)$$

Durch Umformen der letzten Gleichung folgt die Behauptung. ■

In einem ersten Schritt zeigen wir, dass man auch für komplexwertige Funktionen den Interpolationsfehler kontrollieren kann.

Korollar 4.4. *Es seien die reell- oder komplexwertige Funktion $f \in \mathcal{C}^{n+1}[a,b]$ sowie paarweise verschiedene Stützstellen $x_0, \ldots, x_n \in [a,b]$ gegeben, und $p \in \mathbb{P}_n$ sei das Interpolationspolynom mit $p(x_j) = f(x_j)$ für alle $0 \leq j \leq n$. Für $x \in [a,b]$ gilt dann*

$$|f(x) - p(x)| \leq C_{\mathbb{K}}\, \frac{\|f^{(n+1)}\|_{\infty,[a,b]}}{(n+1)!} \prod_{j=0}^{n} |x - x_j|, \tag{4.2}$$

wobei $C_{\mathbb{K}} = 1$ ist, falls f reellwertig ist, und $C_{\mathbb{K}} = 2$, falls f komplexwertig ist.

Beweis. Wir betrachten Real- und Imaginärteil getrennt: Die Polynome $\operatorname{Re} p, \operatorname{Im} p \in \mathbb{P}_n$ interpolieren die Funktionen $\operatorname{Re} f, \operatorname{Im} f \in \mathcal{C}^{n+1}[a,b]$. Also gilt mit geeigneten Zwischenstellen $\xi_1, \xi_2 \in [a,b]$

$$f(x) - p(x) = \frac{\operatorname{Re} f^{(n+1)}(\xi_1) + i \operatorname{Im} f^{(n+1)}(\xi_2)}{(n+1)!} \prod_{j=0}^{n} (x - x_j).$$

Betrachtet man nun den Betrag und geht zur Supremumsnorm von $f^{(n+1)}$ über, so folgt die Fehlerabschätzung (4.2). ■

Das folgende Korollar zeigt, dass beliebig oft differenzierbare Funktionen mit gleichmäßig beschränkten Ableitungen durch interpolierende Polynome gut approximiert werden. Überraschenderweise ist dabei die Wahl der Stützstellen beliebig.

Korollar 4.5. *Sei die reellwertige Funktion f unendlich oft differenzierbar, d.h. $f \in \mathcal{C}^\infty[a,b]$, mit gleichmäßig beschränkten Ableitungen $\|f^{(k)}\|_\infty \leq M < \infty$ für alle $k \in \mathbb{N}$. Für alle $n \in \mathbb{N}$ seien paarweise verschiedene $x_0^{(n)}, \ldots, x_n^{(n)} \in [a,b]$ gegeben (d.h. es gibt eine Folge von Unterteilungen des Intervalls) und $p_n \in \mathbb{P}_n$ mit $p_n(x_j^{(n)}) = f_n(x_j^{(n)})$ für alle $0 \leq j \leq n$. Dann folgt*

$$\lim_{n\to\infty} \|f - p_n\|_\infty = 0,$$

d.h. die Folge der interpolierenden Polynome (p_n) ***konvergiert gleichmäßig*** *gegen f.*

Beweis. Mit Abschätzung (4.2) gilt

$$\|f - p_n\|_\infty \leq \frac{\|f^{(n+1)}\|_\infty}{(n+1)!}(b-a)^{n+1} \leq M\frac{(b-a)^{n+1}}{(n+1)!},$$

und der Quotient auf der rechten Seite verschwindet für $n \to \infty$. ■

Lagrange-Polynominterpolation in MATLAB

In MATLAB sind Funktionen zur Lagrange-Polynominterpolation vorimplementiert. Mit der Funktion `polyfit` werden die Koeffizienten des Interpolationspolynoms bezüglich der Monombasis bestimmt. Dies geschieht intern durch Lösen des Vandermonde-Systems 4.1. Wir haben bereits oben angemerkt, dass dieses Vorgehen im allgemeinen *nicht* numerisch stabil ist. Mittels `polyval` kann das Interpolationspolynom an beliebigen Stellen ausgewertet werden.

Anhang: Approximation durch Polynome

In den Anwendungen ist häufig eine glatte Funktion f gegeben, aber die numerische Behandlung bzw. Realisierung erfordert eine diskrete Approximation der Funktion f, zum Beispiel bei der numerischen Integration von f.

Satz 4.6 (Satz von Weierstraß). *Für jede stetige Funktion $f \in \mathcal{C}[a,b]$ existiert eine Folge $(p_n)_{n\in\mathbb{N}}$ von Polynomen mit*

$$\lim_{n\to\infty} \|f - p_n\|_\infty = 0.$$ ■

Dass diese Folge im Allgemeinen *keine* Folge von Interpolationspolynomen ist, zeigt der folgende

Satz 4.7 (Satz von Faber). *Zu jeder Folge von Zerlegungen $(x_0^{(n)}, \ldots, x_n^{(n)})_{n\in\mathbb{N}}$ des Intervalls $[a,b]$ existiert eine stetige Funktion $f \in \mathcal{C}[a,b]$ derart, dass die Folge $(p_n)_{n\in\mathbb{N}}$ der zugehörigen Interpolationspolynome divergiert.* ■

Beispiel (Runge 1901). Man betrachtet die Funktion

$$f(x) = \frac{1}{1+x^2} \text{ auf } [a,b] = [-5,5]$$

sowie die äquidistanten Stützstellen

$$x_j^{(n)} := a + j\frac{b-a}{n} \quad \text{für } 0 \le j \le n, \quad n \in \mathbb{N}.$$

Dann konvergiert die Folge der Interpolationspolynome *nicht* gegen die Funktion f: Am Rand des Intervalls treten starke Oszillationen auf. ■

Satz 4.8. *Zu einer stetigen Funktion $f \in \mathcal{C}[a,b]$ existiert ein eindeutiges Polynom $p \in \mathbb{P}_n$ mit der Eigenschaft*

$$\|f - p\|_\infty = \min_{q \in \mathbb{P}_n} \|f - q\|_\infty,$$

*die sogenannte **Bestapproximation** von f in $\mathbb{P}_n$ bezüglich $\|\cdot\|_\infty$.*

Die Eindeutigkeit dieses bestapproximierenden Polynoms ist nicht einfach zu zeigen, folgt aber aus dem **Alternantensatz von Čebyšev,** siehe Plato [5, §15.5, §15.7]. Die Existenz ist jedoch elementar und folgt aus dem folgenden Lemma.

Lemma 4.9. *Seien X ein normierter Raum und $V \le X$ ein endlichdimensionaler Teilraum. Dann existiert zu jedem $x \in X$ ein $v \in V$ mit $\|x - v\| = \min_{\widetilde{v} \in V} \|x - \widetilde{v}\|$.*

Beweis. Sei $(v_n) \subseteq V$ eine Folge mit

$$\lim_{n\to\infty} \|x - v_n\| = \inf_{\widetilde{v} \in V} \|x - \widetilde{v}\|.$$

Dann gilt wegen der Dreiecksungleichung

$$\|v_n\| \le \|x\| + \|x - v_n\|,$$

und insbesondere ist die Folge (v_n) gleichmäßig beschränkt, weil $\|x - v_n\|$ konvergiert. Da V ein endlichdimensionaler Teilraum ist, hat jede beschränkte Folge in V eine konvergente Teilfolge (Satz von Bolzano-Weierstraß). O.B.d.A. kann also durch Übergang zu einer Teilfolge angenommen werden, dass (v_n) gegen ein $v \in V$ konvergiert. Die Stetigkeit der Norm zeigt abschließend

$$\|x - v\| = \lim_{n\to\infty} \|x - v_n\| = \inf_{\widetilde{v} \in V} \|x - \widetilde{v}\|.$$ ■

Bemerkung. Die Bestapproximation $p \in \mathbb{P}_n$ der Funktion f kann mit dem **Remez-Algorithmus** iterativ berechnet werden, was aber sehr aufwändig ist (siehe Quarteroni [6, §10.8]). In der Praxis verwendet man daher Interpolationspolynome zu „geschickt“ gewählten Knoten: Diese werden so gewählt, dass gilt

$$\max_{x\in[a,b]} \prod_{j=0}^{n} |x - x_j| = \min_{\substack{t_0,\dots,t_n \\ \in[a,b]}} \max_{x\in[a,b]} \prod_{j=0}^{n} |x - t_j|.$$

Im folgenden Abschnitt werden wir sehen, dass die **Čebyšev-Knoten** t_j diese Bedingung erfüllen. □

4.2 Čebyšev-Knoten

In der Fehlerabschätzung der Lagrange-Interpolation tritt der Term

$$\max_{x\in[a,b]} \prod_{j=0}^{n} |x - x_j|$$

auf. In diesem Abschnitt untersuchen wir, wie man die paarweise verschiedenen Stützstellen $x_0, \ldots, x_n \in [a, b]$ so wählt, dass dieser Term minimal wird. Es wird sich zeigen, dass die optimalen Stützstellen durch die transformierten Nullstellen des $(n + 1)$-ten Čebyšev-Polynoms gegeben sind.

Definition. Für $n \in \mathbb{N}_0$ definiere die **Čebyšev-Polynome (der ersten Art)** durch

$$T_n(t) := \cos(n \arccos t) \quad \text{für } t \in [-1, 1]. \qquad \square$$

Lemma 4.10. *Für die Čebyšev-Polynome gelten die folgenden Aussagen:*
(i) $T_n(\cos\theta) = \cos(n\theta)$ *für* $\theta \in [0, \pi]$ *und* $n \in \mathbb{N}_0$.
(ii) *Auf dem Intervall* $[-1, 1]$ *gelten die Identitäten*

$$T_0(x) = 1, \quad T_1(x) = x \quad \textit{und} \quad T_{n+1}(x) = 2xT_n(x) - T_{n-1}(x).$$

(iii) *Der Leitkoeffizient von* $T_n \in \mathbb{P}_n[-1, 1]$ *ist für* $n \geq 1$ *gleich* 2^{n-1}.
(iv) $\|T_n\|_\infty = 1$.
(v) *Das Čebyšev-Polynom* T_n *besitzt im Intervall* $[-1, 1]$ *insgesamt* $(n + 1)$ *lokale Extrema:*

$$T_n(s_j^{(n)}) = (-1)^j \quad \textit{für } s_j^{(n)} := \cos\Big(\frac{j\pi}{n}\Big), \qquad 0 \leq j \leq n.$$

(vi) *Das Čebyšev-Polynom* T_n *besitzt im Intervall* $[-1, 1]$ *insgesamt* n *einfache Nullstellen:*

$$T_n(t_j^{(n)}) = 0 \quad \textit{für } t_j^{(n)} := \cos\Big(\frac{(2j-1)\pi}{2n}\Big), \qquad 1 \leq j \leq n.$$

Beweis. (i) ist klar, (iv), (v) und (vi) folgen aus (i). (iii) folgt offensichtlich induktiv aus (ii).
(ii) Die Darstellungen für T_0 und T_1 ergeben sich offensichtlich aus (i). Für die Herleitung der Rekursionsformel in (ii) wird das folgende Additionstheorem verwendet:

$$\cos(x) + \cos(y) = 2\cos\Big(\frac{x+y}{2}\Big)\cos\Big(\frac{x-y}{2}\Big) \quad \text{für } x, y \in \mathbb{R}.$$

Für $t = \cos\theta \in [-1, 1]$ folgt dann mit (i) $x = (n + 1)\theta$ sowie $y = (n - 1)\theta$

$$2tT_n(t) - T_{n-1}(t) = 2\cos(n\theta)\cos(\theta) - \cos((n-1)\theta) = \cos((n+1)\theta) = T_{n+1}(t). \qquad \blacksquare$$

Der folgende Satz liefert die wesentliche Aussage dieses Abschnitts. Man beachte, dass für die Gültigkeit des Satzes die Stützstellen nicht notwendig verschieden sein müssen.

Satz 4.11. *Mit der* ***affinen Transformation*** $\Psi : [-1,1] \to [a,b]$

$$\Psi(t) := \frac{1}{2}\big[a+b+t(b-a)\big].$$

und den $n+1$ *Nullstellen* $t_1^{(n+1)}, \dots, t_{n+1}^{(n+1)}$ *von* T_{n+1} *gilt*

$$\min_{\substack{x_0,\dots,x_n \\ \in[a,b]}} \max_{x\in[a,b]} \prod_{j=0}^{n} |x-x_j| = \max_{x\in[a,b]} \prod_{j=0}^{n} \big|x - \Psi\big(t_{j+1}^{(n+1)}\big)\big| = \Big(\frac{b-a}{2}\Big)^{n+1} \frac{1}{2^n}.$$

Die Knoten $\psi(t_1^{(n+1)}), \dots, \psi(t_{n+1}^{(n+1)})$ *heißen* **Čebyšev-Knoten** *der Ordnung* n *in* $[a,b]$.

Beweis. Der Beweis folgt in vier Schritten, wobei zunächst $[a,b] = [-1,1]$ betrachtet wird.
1. Schritt. *Es gilt* $\max_{x\in[-1,1]} \prod_{j=0}^{n} \big|x - t_{j+1}^{(n+1)}\big| = 1/2^n$. Aus Lemma 4.10 (iii) und (vi) folgt

$$T_{n+1}(x) = 2^n \prod_{j=0}^{n} \big(x - t_{j+1}^{(n+1)}\big).$$

Mit Lemma 4.10 (iv) folgt daraus

$$1 = \|T_{n+1}\|_\infty = \max_{x\in[-1,1]} 2^n \prod_{j=0}^{n} \big|x - t_{j+1}^{(n+1)}\big|. \qquad \square$$

2. Schritt. *Es gilt* $1/2^n \le \inf_{\substack{x_0,\dots,x_n \\ \in[-1,1]}} \max_{x\in[-1,1]} \prod_{j=0}^{n} |x-x_j|$. Wir beweisen die Ungleichung durch Widerspruch: Angenommen, es existieren Zahlen $x_0, \dots, x_n \in [-1,1]$, sodass bei Definition von $\omega(x) := \prod_{j=0}^{n}(x-x_j)$ die Abschätzung

$$\|\omega\|_{\infty,[-1,1]} = \max_{x\in[-1,1]} \prod_{j=0}^{n} |x-x_j| < \frac{1}{2^n}$$

erfüllt ist. Definiere das Polynom $p \in \mathbb{P}_{n+1}$ mit $p := \frac{1}{2^n} T_{n+1} - \omega$. Zunächst gilt sofort $p \in \mathbb{P}_n$, denn $\frac{1}{2^n} T_{n+1}$ und ω haben als Leitkoeffizienten 1. Ferner hat p mindestens $(n+1)$ Vorzeichenwechsel, also $(n+1)$ Nullstellen, denn nach Lemma 4.10 gilt

$$\frac{1}{2^n} T_{n+1}\big(s_j^{(n+1)}\big) = \frac{(-1)^j}{2^n}, \quad \big|\omega\big(s_j^{(n+1)}\big)\big| < \frac{1}{2^n} \quad \text{für } 0 \le j \le n+1$$

Da $p \in \mathbb{P}_n$ also $n+1$ paarweise verschiedene Nullstellen hat, folgt $p \equiv 0$ – im Widerspruch zu $p\big(s_j^{(n+1)}\big) \neq 0$. Also gilt, wie behauptet,

$$\|\omega\|_{\infty,[-1,1]} \ge \frac{1}{2^n}.$$

Damit ist Satz 4.11 für $[a,b] = [-1,1]$ bewiesen. $\square$

3. Schritt. *Es gilt* $\max_{x\in[a,b]} \prod_{j=0}^{n} \left|x - \Psi\left(t_{j+1}^{(n+1)}\right)\right| = \left(\frac{b-a}{2}\right)^{n+1} \frac{1}{2^n}$. Dies folgt unmittelbar aus

$$\max_{x\in[a,b]} \prod_{j=0}^{n} \left|x - \Psi\left(t_{j+1}^{(n+1)}\right)\right| = \max_{t\in[-1,1]} \prod_{j=0}^{n} \left|\Psi(t) - \Psi\left(t_{j+1}^{(n+1)}\right)\right|, \tag{4.3}$$

da Ψ eine Bijektion ist. □

4. Schritt. *Es gilt* $\left(\frac{b-a}{2}\right)^{n+1} \frac{1}{2^n} \leq \inf_{\substack{x_0,\dots,x_n \\ \in[a,b]}} \max_{x\in[a,b]} \prod_{j=0}^{n} |x - x_j|$. Existieren $x_0, \dots, x_n \in [a,b]$ mit $\max_{x\in[a,b]} \prod_{j=0}^{n} |x - x_j| < \left(\frac{b-a}{2}\right)^{n+1} \frac{1}{2^n}$, so folgt mit den Stützstellen $t_j := \Psi^{-1}(x_j) \in [-1,1]$ und (4.3) ein Widerspruch zu Schritt 2. ■

Bemerkung. (i) Beim Remez-Algorithmus zur Berechnung des bestapproximierenden Polynoms nimmt man in der Regel das Interpolationspolynom zu den Čebyšev-Knoten als Start-Approximation.
(ii) Man kann zeigen, dass für jede Lipschitz-stetige Funktion $f : [a,b] \to \mathbb{R}$ die Folge der Interpolationspolynome zu den Čebyšev-Knoten gleichmäßig auf $[a,b]$ gegen f konvergiert. □

Praktische Übung. Nach Mittelwertsatz ist jede $\mathcal{C}^1$-Funktion auf $[a,b]$ auch Lipschitz-stetig. Nimmt man also die Čebyšev-Knoten (anstatt der äquidistanten Stützstellen) bei der Realisierung des Runge-Beispiels, so erhält man gleichmäßige Konvergenz. Man verifiziere dies numerisch, indem man die Interpolationspolynome zu den Čebyšev-Knoten der Ordnungen $n = 1, \dots, 20$ berechne und zusammen mit $f(x) = 1/(1+x^2)$ über $[-5,5]$ plotte. ■

4.3 Auswertung von Lagrange-Interpolationspolynomen

Im Folgenden werden Algorithmen zur Berechnung der Werte des interpolierenden Polynoms angegeben. Es seien paarweise verschiedene Stützstellen $x_0, \dots, x_n$, Funktionswerte $y_0, \dots, y_n \in \mathbb{K}$ sowie das interpolierende Polynom $p \in \mathbb{P}_n$ mit $p(x_j) = y_j$ gegeben. Zielsetzung dieses Abschnitts ist die Herleitung eines Verfahrens, das das interpolierende Polynom mit dem folgenden Aufwand auswertet:

- Eine **Anlaufrechnung** benötigt $\mathcal{O}(n^2)$ arithmetische Operationen.
- Nachdem die Anlaufrechnung durchgeführt ist, benötigt die **Auswertung** des Polynoms $p(x)$ an der Stelle x noch $\mathcal{O}(n)$ arithmetische Operationen.

Zunächst wird ein direktes Verfahren zur Berechnung von $p(x)$ für *ein* fixiertes $x \in \mathbb{R}$ vorgestellt.

Satz 4.12 (Neville-Verfahren). *Für $j, m \geq 0$ mit $j + m \leq n$ sei $p_{j,m} \in \mathbb{P}_m$ das eindeutige Polynom mit*

$$p_{j,m}(x_k) = y_k \quad \text{für } j \leq k \leq j + m.$$

Dann gilt die Rekursionsformel

$$\begin{aligned} p_{j,0}(x) &= y_j \\ p_{j,m}(x) &= \frac{(x - x_j)p_{j+1,m-1}(x) - (x - x_{j+m})p_{j,m-1}(x)}{x_{j+m} - x_j} \quad \text{für } m > 0 \end{aligned}$$

und insgesamt $p_{0,n}(x) = p(x)$.

Beweis. Die Identität $p_{j,0} \equiv y_j$ ist wegen $p_{j,0} \in \mathbb{P}_0$ mit $p_{j,0}(x_j) = y_j$ offensichtlich richtig. Sei $q(x)$ als rechte Seite der zweiten Identität des Satzes definiert:

$$q(x) := \frac{(x - x_j)p_{j+1,m-1}(x) - (x - x_{j+m})p_{j,m-1}(x)}{x_{j+m} - x_j}.$$

Es gilt $q \in \mathbb{P}_m$ wegen $p_{j+1,m-1}, p_{j,m-1} \in \mathbb{P}_{m-1}$. Außerdem gelten die Gleichheiten

$$q(x_j) = p_{j,m-1}(x_j) = y_j$$

und

$$q(x_{j+m}) = p_{j+1,m-1}(x_{j+m}) = y_{j+m}.$$

Für $j + 1 \leq k \leq j + m - 1$ gilt

$$q(x_k) = \frac{(x_k - x_j)y_k - (x_k - x_{j+m})y_k}{x_{j+m} - x_j} = y_k.$$

Aufgrund der Eindeutigkeit des Interpolationspolynoms gilt daher die Identität $p_{j,m} = q$. ■

Die Abhängigkeiten gemäß der Rekursionsformel sind im folgenden **Neville-Schema** dargestellt.

$$\begin{array}{ccccccccccccc} y_0 & = & p_{0,0}(x) & \longrightarrow & p_{0,1}(x) & \longrightarrow & p_{0,2}(x) & \longrightarrow & \dots & \longrightarrow & p_{0,n}(x) & = & p(x) \\ & & & \nearrow & & \nearrow & & & & \nearrow & & & \\ y_1 & = & p_{1,0}(x) & \longrightarrow & p_{1,1}(x) & & & \nearrow & & & & & \\ & & & \nearrow & & & & & & & & & \\ y_2 & = & p_{2,0}(x) & \longrightarrow & \vdots & & & & & & & & \\ \vdots & & \vdots & & \vdots & \nearrow & & & & & & & \\ y_{n-1} & = & p_{n-1,0}(x) & \longrightarrow & p_{n-1,1}(x) & & & & & & & & \\ & & & \nearrow & & & & & & & & & \\ y_n & = & p_{n,0}(x) & & & & & & & & & & \end{array} \tag{4.4}$$

Bemerkung. Für ein fixiertes $x \in \mathbb{R}$ beträgt der Aufwand zur Berechnung von $p(x)$ mit dem Neville-Schema asymptotisch $\frac{7}{2}n^2$ arithmetische Operationen. Genauer sind $7\frac{n(n+1)}{2}$ Operationen

nötig: Die Berechnung von $p_{j,m}(x)$ im Rekursionsschritt benötigt 4 Subtraktionen, 2 Multiplikationen und 1 Division. Der Faktor $\frac{n(n+1)}{2}$ stammt von den *unbekannten* Einträgen der $(n+1)\times(n+1)$ Dreiecksmatrix im Schema. □

Bemerkung. (i) Beim Neville-Verfahren handelt es sich um ein **Einschrittverfahren**: Die Berechnung des Vektors $(p_{0,k}(x),\dots,p_{n-k,k}(x))$ erfordert lediglich die Werte aus dem vorausgegangen Schritt $(p_{0,k-1}(x),\dots,p_{n-k+1,k-1}(x))$. Dadurch ist bei Realisierung kein zusätzlicher Speicherplatz nötig, wenn man stets die „alten“ Werte (und insbesondere den y-Vektor) überschreibt.
(ii) Es ist problemlos möglich, einen weiteren Interpolationspunkt (x_{n+1},y_{n+1}) hinzuzufügen. In diesem Fall muss lediglich die Diagonale $(p_{n,0}(x),p_{n-1,1}(x),\dots,p_{0,n}(x))$ aus dem Neville-Schema (4.4) zur Verfügung stehen, d.h. man muss diese ggf. speichern. □

Bei der folgenden Realisierung werden die Funktionswerte y_j mit den Werten aus dem Neville-Verfahren überschrieben, um keinen zusätzlichen Speicher zu benutzen.

Algorithmus 4.13: Neville-Verfahren

Input: Stützstellen $x\in\mathbb{R}^n$, Funktionswerte $y\in\mathbb{K}^{n+1}$, `Auswertungspunkt` $t\in\mathbb{R}$

```
for m = 1:n
   for j = 0:n-m
      y_j = ((t - x_j)y_{j+1} - (t - x_{j+m})y_j) / (x_{j+m} - x_j)
   end
end
```

Output: $p(t)=y_0$.

Definition. Bei einem Polynom $p\in\mathbb{P}_n$, dargestellt in der Monombasis $p(x)=\sum_{k=0}^n\lambda_k x^k$ bezeichnen wir den Koeffizienten von x^n als **führenden Koeffizienten** bezüglich $\mathbb{P}_n$. Der **Leitkoeffizient** von p ist gerade der Koeffizient λ_j, für den $\lambda_j\neq 0$ und $\lambda_k=0$ für $k>j$ gelten. □

Satz 4.14 (Dividierte Differenzen). *Für $j, m \geq 0$ mit $j + m \leq n$ definiere*

$$y_{j,0} := y_j$$
$$y_{j,m} := \frac{y_{j+1,m-1} - y_{j,m-1}}{x_{j+m} - x_j} \quad \text{für } m > 0.$$

Dann gilt

(i) *$y_{j,m}$ ist der führende Koeffizient von $p_{j,m} \in \mathbb{P}_m$ aus dem Neville-Verfahren.*

(ii) *Mit $\lambda_j := y_{0,j}$ gilt folgende Darstellung bezüglich der Newton-Basis:*

$$p(x) = \sum_{j=0}^{n} \lambda_j \prod_{k=0}^{j-1} (x - x_k).$$

Beweis. (i) folgt offensichtlich induktiv wegen

$$p_{j,0} \equiv y_j \quad \text{und} \quad p_{j,m}(x) = \frac{(x - x_j)p_{j+1,m-1}(x) - (x - x_{j+m})p_{j,m-1}(x)}{x_{j+m} - x_j}.$$

(ii) Betrachte das Polynom $q_k := p_{0,k} - p_{0,k-1} \in \mathbb{P}_k$: $q_k \in \mathbb{P}_k$ hat den führenden Koeffizienten $y_{0,k}$ und die k Nullstellen $x_0, \ldots, x_{k-1}$. Deshalb gilt

$$q_k = y_{0,k} \prod_{j=0}^{k-1} (x - x_j).$$

Unter Verwendung der Teleskopsumme folgt schließlich die Identität in (ii):

$$p = p_{0,n} = q_0 + \sum_{k=1}^{n} q_k = \sum_{k=0}^{n} y_{0,k} \prod_{j=0}^{k-1} (x - x_j). \qquad \blacksquare$$

Die Abhängigkeiten gemäß der Rekursionsformel sind im folgenden **Schema der Dividierten Differenzen** dargestellt. Gemäß Satz 4.14 ist der schematische Aufbau identisch mit dem Neville-Schema.

$$\begin{array}{ccccccccccc}
y_0 & = & y_{0,0} & \longrightarrow & y_{0,1} & \longrightarrow & y_{0,2} & \longrightarrow & \cdots & \longrightarrow & y_{0,n} \\
 & & & \nearrow & & \nearrow & & & & \nearrow & \\
y_1 & = & y_{1,0} & \longrightarrow & y_{1,1} & & & \nearrow & & & \\
 & & & \nearrow & & & & & & & \\
y_2 & = & y_{2,0} & \longrightarrow & \vdots & & & & & & \\
\vdots & & \vdots & & \vdots & \nearrow & & & & & \\
y_{n-1} & = & y_{n-1,0} & \longrightarrow & y_{n-1,1} & & & & & & \\
 & & & \nearrow & & & & & & & \\
y_n & = & y_{n,0} & & & & & & & &
\end{array} \tag{4.5}$$

Bemerkung. (i) Das Verfahren der Dividierten Differenzen berechnet in $\frac{3}{2}n(n+1)$ arithmetischen Operationen die Koeffizienten λ_j bezüglich der Newton-Basis.
(ii) Man kann den Algorithmus der Dividierten Differenzen so formulieren, dass kein weiterer Speicherplatz benötigt wird.
(iii) De facto ist das Verfahren der Dividierten Differenzen ein Eliminationsverfahren für das Vandermonde-System (4.1) bezüglich der Newton-Basis. Dieser direkte Löser nutzt die zusätzliche Struktur der Vandermonde-Matrix aus, um den eigentlichen Aufwand $\mathcal{O}(n^3)$, vgl. Kapitel 3, auf $\mathcal{O}(n^2)$ zu reduzieren. □

Praktische Übung. Man formuliere einen Algorithmus für das Verfahren der Dividierten Differenzen, bei dem kein weiterer Speicher benötigt wird, sondern der y-Vektor überschrieben wird (vgl. die algorithmische Formulierung des Neville-Verfahrens). ■

Mit der Darstellung $p(x) = \sum_{j=0}^{n} \lambda_j \prod_{k=0}^{j-1}(x - x_k)$ des Interpolationspolynoms gilt:

$$p(x) = \lambda_0 + (x - x_0)[\lambda_1 + (x - x_1)[\lambda_2 + (x - x_2)[\ldots[\lambda_{n-1} + (x - x_{n-1})\lambda_n]\ldots]$$

Das **Horner-Schema** wertet das Polynom p gemäß dieser Formel bei $x \in \mathbb{R}$ aus. Dazu wird die Klammer von hinten nach vorne ausgerechnet.

Algorithmus 4.15: Horner-Schema

Input: λ_j als Ergebnis der Dividierten Differenzen, Auswertnungsstelle $t \in \mathbb{R}$

$y = \lambda_n$
`for k = n-1:-1:0`
  $y = (t - x_k)y + \lambda_k$
`end`

Output: $p(t) = y$.

Behauptung. Das Horner-Schema berechnet in $3n$ arithmetischen Operationen den Funktionswert $p(x) = y$.
Beweis. Dass der Algorithmus $3n$ arithmetischer Operationen bedarf, ist offensichtlich. Es gilt

$$[\ldots[[(x - x_{n-1})\lambda_n + \lambda_{n-1}](x - x_{n-2}) + \lambda_{n-2}](x - x_{n-3}) + \ldots + \lambda_1](x - x_0) + \lambda_0 \stackrel{!}{=} p(x). \quad ■$$

Bemerkung. Will man $p(x)$ nur an einer Stelle auswerten, so verwendet man in der Regel das Neville-Verfahren. Soll $p(x)$ an mehreren Stellen ausgewertet werden, so verwendet man die Dividierten Differenzen in Verbindung mit dem Horner-Schema: Für die Auswertung des Interpolationspolynoms $p(x^{(j)})$ bei $x^{(1)}, \ldots, x^{(N)}$ mittels Neville-Verfahren beträgt der Aufwand $A_{Neville} = \frac{7}{2}Nn(n+1)$, mittels Dividierter Differenzen (und Horner-Schema) lediglich $A_{Newton} = \frac{3}{2}n(n+1) + 3Nn$. Bestechenderweise gilt für $N \geq 1$ stets $A_{Newton} \leq A_{Neville}$, aber das Neville-Verfahren ist (ein wenig) stabiler gegenüber Rundungsfehlern. □

4.4 Spline-Interpolation

Die Polynominterpolation erfordert hohe Annahmen an die Glätte, damit die Folge der Interpolationspolynome gleichmäßig gegen die Funktion konvergiert. Eine Alternative ist es, stückweise Polynome zu betrachten, im einfachsten Fall stückweise affine Funktionen.

Beispiel. Zu Stützstellen $a = x_0 < x_1 < \cdots < x_n = b$ und einer Funktion $f \in \mathcal{C}[a,b]$ ist eine stetige Funktion $s \in \mathcal{C}[a,b]$ gesucht, die stückweise affin ist, d.h.

$$s|_{[x_{j-1},x_j]} \in \mathbb{P}_1 \quad \text{für } j = 1, \ldots, n$$

und f in den Knoten x_j interpoliert, d.h.

$$s(x_j) = f(x_j) \quad \text{für } j = 0, \ldots, n.$$

Dieser **affine Interpolationsspline** ist offensichtlich (eindeutig) stückweise gegeben durch

$$s(x) = f(x_{j-1}) \frac{x - x_j}{x_{j-1} - x_j} + f(x_j) \frac{x - x_{j-1}}{x_j - x_{j-1}} \quad \text{für } x \in [x_{j-1}, x_j]. \tag{4.6}$$

Wir beweisen im Folgenden erste Fehlerabschätzungen für die Spline-Interpolation mit affinen Splines.

Lemma 4.16. *Zu $a = x_0 < x_1 < \cdots < x_n = b$ und $f \in \mathcal{C}^2[a,b]$ definiere den Spline $s \in \mathcal{C}[a,b]$ durch* (4.6)*. Ferner sei die* **lokale Netzweite** *$h : [a,b] \to \mathbb{R}$ definiert durch $h|_{[x_{j-1},x_j)} = x_j - x_{j-1}$. Dann gilt*

$$\|f - s\|_\infty \leq \frac{\sqrt{2}}{8} \|h^2 f''\|_\infty.$$

Beweis. Auf einem Teilintervall $I = [x_{j-1}, x_j]$ ist s das eindeutige Interpolationspolynom vom Grad 1. Also liefert die Fehlerabschätzung für $x \in I$

$$|f(x) - s(x)| \leq \sqrt{2} \frac{\|f''\|_\infty}{2} |(x - x_{j-1})(x - x_j)|.$$

Die Funktion $\omega(x) = |(x - x_{j-1})(x - x_j)|$ nimmt ihr Maximum für $x = (x_{j-1} + x_j)/2$ an. Also gilt $\max\limits_{x \in I} |\omega(x)| = h_j^2/4$. Damit ergibt sich

$$\|f - s\|_{\infty,I} \leq \frac{\sqrt{2}}{8} h_j^2 \|f''\|_{\infty,I},$$

und es folgt die Behauptung. ■

Betrachtet man statt der Maximumsnorm auf $\mathcal{C}[a,b]$ die $\boldsymbol{L^2}$**-Norm**

$$\|f\|_2 := \langle f \,;\, f \rangle_2^{1/2} \quad \text{mit} \quad \langle f \,;\, g \rangle_2 := \int_a^b f \overline{g} \, dx,$$

so erhält man auch Fehlerabschätzungen für $f \in \mathcal{C}^1[a,b]$.

Lemma 4.17. *Unter den Voraussetzungen von Lemma 4.16 gelten*

$$\|f - s\|_2 \le \|hf'\|_2 \quad \text{für } f \in \mathcal{C}^1[a,b],$$
$$\|f - s\|_2 \le \|h^2 f''\|_2 \quad \text{für } f \in \mathcal{C}^2[a,b].$$

Beweis. Definiere $I_j := [x_{j-1}, x_j]$ und betrachte $F = f - s \in \mathcal{C}^1(I_j)$. Wegen $F(x_{j-1}) = 0$ gilt

$$|F(x)| = \Big| \int_{x_{j-1}}^{x} F'\,dx \Big| \le h_j^{1/2} \|F'\|_{2,I_j}$$

nach Cauchy-Schwarz-Ungleichung. Integration über I_j zeigt

$$\|F\|_{2,I_j} \le h_j \|F'\|_{2,I_j}. \tag{4.7}$$

Aufgrund der Interpolationseigenschaft gilt $\int_{I_j} F'\,dx = 0$. Da $s'|_{I_j}$ konstant ist, folgt $s'|_{I_j} = (1/h_j)\int_{I_j} f'\,dx$ und deshalb $\langle f' \,;\, s' \rangle_{2,I_j} = \|s'\|_{2,I_j}^2$. Es ergibt sich

$$\|F'\|_{2,I_j}^2 = \|f'\|_{2,I_j}^2 - 2\,\mathrm{Re}\,\langle f' \,;\, s' \rangle_{2,I_j} + \|s'\|_{2,I_j}^2 = \|f'\|_{2,I_j}^2 - \|s'\|_{2,I_j}^2 \le \|f'\|_{2,I_j}^2.$$

Kombination mit (4.7) zeigt

$$\|F\|_{2,I_j} \le h_j \|f'\|_{2,I_j},$$

und Summation über die Teilintervalle liefert die erste Abschätzung. Um die zweite Abschätzung zu beweisen, müssen wir nur noch $\|F'\|_{2,I_j} \le h_j \|F''\|_{2,I_j}$ zeigen, denn es gilt $F'' = f''$, da s affin ist auf I_j. Wegen $F(x_{j-1}) = 0 = F(x_j)$ hat F' eine Nullstelle $\zeta \in I_j$. Analoges Vorgehen wie oben zeigt

$$|F'(x)| \le (\zeta - x_{j-1})^{1/2} \|F''\|_{2,I_j} \le h_j^{1/2} \|F''\|_{2,I_j} \quad \text{für } x_{j-1} \le x \le \zeta$$

sowie

$$|F'(x)| \le (x_j - \zeta)^{1/2} \|F''\|_{2,I_j} \le h_j^{1/2} \|F''\|_{2,I_j} \quad \text{für } \zeta \le x \le x_j.$$

Integration über I_j liefert

$$\|F'\|_{2,I_j} \le h_j \|F''\|_{2,I_j} = h_j \|f''\|_{2,I_j}. \tag{4.8}$$

Kombination mit (4.7) und Summation über die Teilintervalle zeigt die Behauptung. ■

Definition. Es sei $\Delta = (x_0, \ldots, x_n)$ eine **Zerlegung** von $[a,b]$, d.h. $a = x_0 < x_1 < \cdots < x_n = b$. Zu gegebenen $p, q \in \mathbb{N}_0$ heißt eine Abbildung $s : [a,b] \to \mathbb{K}$ **Spline vom Grad p und Glattheit q bezüglich Δ**, falls $s|_{[x_{j-1},x_j]} \in \mathbb{P}_p$ und $s \in \mathcal{C}^q[a,b]$ gelten. Wir verwenden im Folgenden die Schreibweise $s \in \mathbb{S}_q^p(\Delta)$. Im Spezialfall $q = p-1$ schreiben wir $\mathbb{S}^p(\Delta) := \mathbb{S}_{p-1}^p(\Delta)$. □

Die wichtigsten Beispiele sind **lineare Splines** $\mathbb{S}^1(\Delta)$, **quadratische Splines** $\mathbb{S}^2(\Delta)$ und **kubische Splines** $\mathbb{S}^3(\Delta)$. Mehrdimensionale Splines, d.h. der Definitionsbereich ist ein Gebiet im $\mathbb{R}^d$ anstatt eines Intervalls, bezeichnet man als **Finite Elemente**. Dort hängt die Zulässigkeit der Kombination von Polynomgrad/Glattheit von der Form der Elemente der Zerlegung ab, d.h. eventuell enthält der Finite-Elemente-Raum nur die konstanten Funktionen.

Bei Interpolationssplines gibt es zwei Möglichkeiten, den Fehler $\|f - s\|$ zu verringern:

- Verfeinerung der Zerlegung, sog. ***h*-Methode**,
- Erhöhung des Polynomgrads, sog. ***p*-Methode**.

In der Regel verwendet man die p-Methode dort, wo f glatt ist, in Kombination mit der h-Methode an den Intervallbereichen, wo f nicht glatt ist, sog. ***hp*-Methode**.

Satz 4.18. *Es sei $\Delta = (x_0, \dots, x_n)$ eine Zerlegung von $[a, b]$. Mit den Monomen $p_j(x) = x^j$ und den Funktionen $q_k(x) := \max\{x - x_k, 0\}^m$ definiert*

$$\mathcal{B} := \big\{ p_j, q_k \,\big|\, j = 0, \dots, m,\ k = 1, \dots, n-1 \big\} \tag{4.9}$$

eine Basis von $\mathbb{S}^m(\Delta)$. Insbesondere gilt also $\dim \mathbb{S}^m(\Delta) = m + n$.

Beweis. Offensichtlich gilt $p_j, q_k \in \mathbb{S}^m(\Delta)$. Es ist also nur zu zeigen, dass $\mathcal{B}$ linear unabhängig ist und $\mathbb{S}^m(\Delta) \subseteq \operatorname{span} \mathcal{B}$ gilt.

1. Schritt. *$\mathcal{B}$ ist linear unabhängig.* Seien Skalare $\zeta_j, \mu_k \in \mathbb{K}$ mit $0 = \sum_{j=0}^{m} \zeta_j p_j + \sum_{k=1}^{n-1} \mu_k q_k$ gegeben. Auf dem Teilintervall $[x_0, x_1]$ gilt $q_k = 0$ für alle $1 \le k \le n-1$, also $\sum_{j=0}^{m} \zeta_j p_j = 0$ auf $[x_0, x_1]$. Es folgt $\zeta_j = 0$ für alle $j = 0, \dots, m$, da die Monome linear unabhängig sind. Auf $[x_1, x_2]$ gilt demnach $0 = \sum_{k=1}^{n-1} \mu_k q_k = \mu_1 q_1$, also $\mu_1 = 0$. Induktives Vorgehen liefert $\mu_k = 0$ für $1 \le k \le n-1$. □

2. Schritt. *Es gilt $\mathbb{S}^m(\Delta) \subseteq \operatorname{span} \mathcal{B}$.* Zu $s \in \mathbb{S}^m(\Delta)$ sei $z_1 \in \mathbb{P}_m$ mit $s|_{[x_0,x_1]} = z_1|_{[x_0,x_1]}$. Für $k = 2, \dots, n$ definiere sukzessive

$$z_k := z_{k-1} + \eta_k q_{k-1} \in \operatorname{span} \mathcal{B} \quad \text{mit} \quad \eta_k := \frac{s(x_k) - z_{k-1}(x_k)}{q_{k-1}(x_k)} \in \mathbb{R}. \tag{4.10}$$

Wir zeigen $s = z_n \in \operatorname{span} \mathcal{B}$. Nach Definition gilt $z_k(x_k) = s(x_k)$, d.h. das Residuum $r := s - z_n$ erfüllt $r(x_k) = 0$ für alle Knoten x_k.

- Auf $[x_0, x_1]$ gilt $r = z_1 - z_n$ und $z_1 = z_n$, also $r|_{[x_0,x_1]} = 0$.
- Auf $[x_1, x_2]$ gilt $r \in \mathbb{P}_m$ und $r(x_1) = 0 = r(x_2)$. Wegen $r \in \mathcal{C}^{m-1}[a, b]$ und $r|_{[x_0,x_1]} = 0$, muss x_1 eine m-fache Nullstelle sein, d.h. r hat $m + 1$ Nullstellen in $[x_1, x_2]$. Es folgt $r|_{[x_1,x_2]} = 0$

Induktives Vorgehen zeigt $r|_{[a,b]} = 0$. ■

Gegeben sei eine Zerlegung $\Delta = (x_0, \dots, x_n)$ von $[a, b]$. Gesucht ist ein Interpolationsspline $s \in \mathbb{S}^m(\Delta)$ mit $s(x_j) = y_j$ für alle $j = 0, \dots, n$. Wegen $\dim \mathbb{S}^m(\Delta) = m + n$, sind weitere $m - 1$ Nebenbedingungen zu formulieren. Dies geschieht in der Regel in Form von **Randbedingungen** an die Ableitungen von s. Wir beschränken uns auf den Fall, dass $m-1$ gerade ist, d.h. $m-1 = 2r$ für ein $r \in \mathbb{N}_0$, und stellen die Randbedingungen symmetrisch bei a und b.

(H) **Hermite-Randbedingungen** (oder: **vollständige Randbedingungen**):
$s^{(j)}(a), s^{(j)}(b)$ für $j = 1, \dots, r$ sind zusätzlich vorgegeben.

(N) **Natürliche Randbedingungen**:
Man fordert $s^{(j)}(a) = 0 = s^{(j)}(b)$ für $j = r+1, \dots, 2r$ und $r \le n$.

(P) **Periodische Randbedingungen**:
Man fordert $s^{(j)}(a) = s^{(j)}(b)$ für $j = 1, \dots, 2r$.

Die Forderung von $r \leq n$ im Fall von (N) ist beweistechnischer Natur. In der Regel gilt aber ohnehin $n \gg r$.
Für kubische Splines, d.h. $m = 3$ und $r = 1$, lauten die Randbedingungen also:

(H) $s'(a), s'(b)$ sind zusätzlich vorgegeben.

(N) $s''(a) = 0 = s''(b)$.

(P) $s'(a) = s'(b)$ und $s''(a) = s''(b)$.

Zusammen mit diesen Randbedingungen lässt sich die eindeutige Existenz eines Interpolationssplines beweisen.

Satz 4.19. (i) *Zu $m - 1 = 2r$ vorgegebenen Randbedingungen (H), (P) oder (N) existiert ein eindeutiger interpolierender Spline $s \in \mathbb{S}^m(\Delta)$, der diese Randbedingung erfüllt.*
(ii) *Erfüllt $g \in \mathcal{C}^{(r+1)}[a, b]$ sowohl die Interpolationsbedingung $g(x_j) = s(x_j)$ für alle Knoten x_j sowie dieselben Randbedingungen wie der Interpolationsspline s, so gilt die Orthogonalitätseigenschaft*

$$\|g^{(r+1)} - s^{(r+1)}\|_2^2 = \|g^{(r+1)}\|_2^2 - \|s^{(r+1)}\|_2^2. \tag{4.11}$$

Insbesondere minimiert also s das Energiefunktional $E(g) = \|g^{(r+1)}\|_2$ über alle zulässigen Funktionen (d.h. interpolierend mit derselben Randbedingung), d.h. es gilt die Minimaleigenschaft $\|s^{(r+1)}\|_2 \leq \|g^{(r+1)}\|_2$.

Beweis. Wie üblich müssen wir nur entweder Existenz oder Eindeutigkeit beweisen. Der wesentliche Schritt zum Nachweis der Eindeutigkeit ist der Nachweis der Orthogonalitätseigenschaft (4.11). Wegen $|x - y|^2 = |x|^2 - |y|^2 - 2\mathrm{Re}\,(x - y)\overline{y}$ folgt

$$\|g^{(r+1)} - s^{(r+1)}\|_2^2 = \|g^{(r+1)}\|_2^2 - \|s^{(r+1)}\|_2^2 - 2\,\mathrm{Re} \int_a^b (g^{(r+1)} - s^{(r+1)})\overline{s^{(r+1)}}\,dx.$$

Wir zeigen nun, dass das letzte Integral verschwindet. Dazu verwenden wir mehrfache partielle Integration. Induktiv gilt

$$\begin{aligned}
\int_a^b (g^{(r+1)} - s^{(r+1)})\overline{s^{(r+1)}}\,dx &= \Big[(g^{(r)} - s^{(r)})\overline{s^{(r+1)}}\Big]_a^b - \int_a^b (g^{(r)} - s^{(r)})\overline{s^{(r+2)}}\,dx \\
&= \Big[(g^{(r)} - s^{(r)})\overline{s^{(r+1)}}\Big]_a^b - \Big[(g^{(r-1)} - s^{(r-1)})\overline{s^{(r+2)}}\Big]_a^b + \int_a^b (g^{(r-1)} - s^{(r-1)})\overline{s^{(r+3)}}\,dx \\
&= \sum_{j=0}^{r-1} (-1)^j \Big[(g^{(r-j)} - s^{(r-j)})\overline{s^{(r+j+1)}}\Big]_a^b + (-1)^r \int_a^b (g' - s')\overline{s^{(2r+1)}}\,dx.
\end{aligned}$$

Wegen $s \in \mathbb{S}^m(\Delta)$ mit $m = 2r + 1$ ist $\sigma_j := \overline{s^{(2r+1)}}|_{(x_j, x_{j+1})} \in \mathbb{K}$ konstant. Aufgrund der Interpolationseigenschaft $(g - s)(x_j) = 0$ für alle $j = 0, \dots, n$ folgt

$$\int_a^b (g' - s')\overline{s^{(2r+1)}}\,dx = \sum_{j=0}^{n-1} \sigma_j \int_{x_j}^{x_{j+1}} (g - s)'\,dx = \sum_{j=0}^{n-1} \sigma_j \big[g - s\big]_{x_j}^{x_{j+1}} = 0.$$

Ferner gilt

$$\left[(g^{(r-j)} - s^{(r-j)})\overline{s^{(r+j+1)}}\right]_a^b = 0 \quad \text{für } j = 0, \ldots, r-1,$$

denn im Fall von (H) oder (P) verschwindet der erste Faktor und für (N) verschwindet der zweite Faktor. Damit ist (4.11) bewiesen, und wir wenden uns dem Beweis von (i) zu: Nach (ii) gilt für zwei interpolierende Splines $s, \widetilde{s} \in \mathbb{S}^m(\Delta)$ mit derselben Randbedingung $\|s^{(r+1)} - \widetilde{s}^{(r+1)}\|_2^2 = 0$, also $(s - \widetilde{s})^{(r+1)} = 0$, d.h. $\rho := s - \widetilde{s} \in \mathbb{P}_r$ ist ein Polynom vom Grad r. Wir unterscheiden die verschiedenen Randbedingungen:

(H) Es gilt $\rho^{(j)}(a) = 0$ für $j = 0, \ldots, r$, d.h. ρ hat eine $(r+1)$-fache Nullstelle bei a. Wegen $\rho \in \mathbb{P}_r$ folgt also $\rho = 0$.

(N) Es gelten $\rho(x_j) = 0$ für $j = 0, \ldots, n$ sowie $r \leq n$. Also hat ρ mindestens $(r+1)$ Nullstellen, und es folgt $\rho = 0$.

(P) Es gilt $\rho^{(j)}(a) = \rho^{(j)}(b)$ für $j = 1, \ldots, 2r$. Wegen $\rho \in \mathbb{P}_r$ gilt $\rho^{(r-1)} \in \mathbb{P}_1$. Aus $\rho^{(r-1)}(a) = \rho^{(r-1)}(b)$ folgt deshalb, dass $\rho^{(r-1)}$ konstant ist, d.h. $\rho^{(r-1)} \in \mathbb{P}_0$. Also gilt bereits $\rho \in \mathbb{P}_{r-1}$. Induktives Vorgehen mit demselben Argument zeigt $\rho \in \mathbb{P}_0$. Wegen $s(a) = \widetilde{s}(a)$ folgt $\rho(a) = 0$, also $\rho = 0$.

Damit ist die Eindeutigkeit gezeigt. Die Existenz eines Interpolationssplines folgt aus Dimensionsgründen. ■

Bemerkung. Wir merken an, dass im Fall von natürlichen Randbedingungen (N) keine weiteren Voraussetzungen (in Form von Randbedingungen) an die Funktion g in (4.11) nötig sind.[1]

4.5 Diskrete und Schnelle Fourier-Transformation

Definition. Eine Abbildung $p : [0, 2\pi] \to \mathbb{C}$, $p(x) = \sum_{j=0}^{n-1} c_j \exp(ijx)$, mit Koeffizienten $c_j \in \mathbb{C}$ bezeichnet man als **trigonometrisches Polynom** vom Grad $\leq n-1$. Hier und im Folgenden bezeichnet $i = \sqrt{-1}$ die **imaginäre Einheit**. Es sei $\mathbb{T}_{n-1}$ der $\mathbb{C}$-Vektorraum der trigonometrischen Polynome vom Grad $\leq n-1$. □

Lemma 4.20. *Es gilt* $\dim \mathbb{T}_{n-1} = n$*, und zu paarweise verschiedenen Stützstellen* $x_0, \ldots, x_{n-1}$ *in* $[0, 2\pi)$ *und Funktionswerten* $y_j \in \mathbb{C}$ *existiert ein eindeutiges trigonometrisches Interpolationspolynom* $p \in \mathbb{T}_{n-1}$ *mit* $p(x_j) = y_j$ *für alle* $j = 0, \ldots, n-1$*.*

Beweis. Es sei $p(x) = \sum_{j=0}^{n-1} c_j \exp(ijx)$ ein Polynom in $\mathbb{T}_{n-1}$ mit $p(x_j) = 0$ für alle $j = 0, \ldots, n-1$. Mit $z_k := \exp(ix_k)$ gilt

$$0 = p(x_k) = \sum_{j=0}^{n-1} c_j \exp(ijx_k) = \sum_{j=0}^{n-1} c_j z_k^j = \widetilde{p}(z_k) \quad \text{mit} \quad \widetilde{p}(z) := \sum_{j=0}^{n-1} c_j z^j$$

[1] Eigentlich wäre es auch ganz nett, an dieser Stelle zumindest die allgemeinen Fehlerabschätzungen zu zitieren, wenn sie schon nicht bewiesen werden.

d.h. das Polynom $\widetilde{p} \in \mathbb{P}_{n-1}$ hat die n Nullstellen $z_0, \ldots, z_{n-1}$. Deshalb folgt $\widetilde{p} = 0$, also $c_j = 0$ für alle $j = 0, \ldots, n-1$. Insbesondere folgt nun die lineare Unabhängigkeit der Funktionen $x \mapsto \exp(ijx)$, d.h. $\dim \mathbb{T}_{n-1} = n$. Ferner haben wir gezeigt, dass die (lineare) Auswertung

$$\mathbb{T}_{n-1} \to \mathbb{K}^n, p \mapsto (p(x_0), \ldots, p(x_{n-1}))$$

injektiv ist. Wie in den vorausgegangen Abschnitten folgt deshalb die eindeutige Lösbarkeit des Interpolationsproblems. ■

Im Rest des Abschnitts werden wir nur noch **äquidistante Stützstellen**

$$x_k := \frac{2\pi k}{n} \quad \text{für } k = 0, \ldots, n-1$$

betrachten. Dies führt auf zusätzliche Struktur in der Vandermonde-Matrix, die schließlich mit der **Schnellen Fourier-Transformation** (sog. **Fast Fourier Transform, FFT**) optimal genutzt wird.

Satz 4.21. *Es seien $x_k = 2\pi k/n$ und $y_k \in \mathbb{C}$ für $k = 0, \ldots, n-1$ gegeben. Es sei $p(x) := \sum_{j=0}^{n-1} c_j \exp(ijx)$ das Interpolationspolynom $p \in \mathbb{T}_{n-1}$ mit $p(x_k) = y_k$ für alle k. Mit der n-ten* **Einheitswurzel**

$$\omega_n := \exp\Big(-\frac{2\pi i}{n}\Big) \tag{4.12}$$

und der **Fourier-Matrix** *(oder* **DFT-Matrix***)*

$$V_n \in \mathbb{C}^{n\times n}, \quad V_n := (\omega_n^{jk})_{j,k=0,}^{n-1} \tag{4.13}$$

sowie den Vektoren $c := (c_0, \ldots, c_{n-1}), y := (y_0, \ldots, y_{n-1}) \in \mathbb{C}^n$ gilt

$$\frac{1}{n} V_n y = c,$$

d.h. $\frac{1}{n} V_n$ bildet die Funktionswerte auf die zugehörigen Koeffizienten von p ab, und es gilt $c_j = \frac{1}{n}\sum_{k=0}^{n-1} \omega_n^{jk} y_k$. Die skalierte Matrix $\frac{1}{\sqrt{n}} V_n$ ist symmetrisch und orthogonal, d.h. $\big(\frac{1}{\sqrt{n}} V_n\big)^{-1} = \frac{1}{\sqrt{n}} \overline{V}_n$. Insbesondere gilt also $V_n^{-1} = \frac{1}{n}\overline{V}_n$.

Beweis. Der Vektor c ist Lösung des Vandermonde-Systems $Wc = y$ mit

$$W := \begin{pmatrix} p_0(x_0) & \ldots & p_{n-1}(x_0) \\ \vdots & & \vdots \\ p_0(x_{n-1}) & \ldots & p_{n-1}(x_{n-1}) \end{pmatrix} \in \mathbb{C}^{n\times n} \qquad \text{mit } p_j(x) := \exp(ijx).$$

Die Einträge der Matrix $W = (W_{jk})_{j,k=0}^{n-1}$ erfüllen

$$W_{jk} = \exp\Big(\frac{2\pi i}{n}\, jk\Big) = \exp\Big(\frac{2\pi i}{n}\Big)^{jk} = \omega_n^{-jk} = \overline{\omega}_n^{jk}.$$

Insbesondere folgen $V_n = \overline{W}$ und die Symmetrie von W und V_n. Es sei $W^{(k)} := (W_{jk})_{j=0}^{n-1}$ die k-te Spalte von W. Zunächst gilt

$$W^{(k)} \cdot W^{(k)} = \sum_{j=0}^{n-1} W_{jk}\overline{W}_{jk} = \sum_{j=0}^{n-1} |W_{jk}|^2 = \sum_{j=0}^{n-1} |\omega_n|^{-2jk} = n.$$

Ferner erhalten wir für $k \neq \ell$

$$W^{(k)} \cdot W^{(\ell)} = \sum_{j=0}^{n-1} W_{jk}\overline{W}_{j\ell} = \sum_{j=0}^{n-1} \omega_n^{j(\ell-k)} = \frac{1-\omega_n^{n(\ell-k)}}{1-\omega_n^{\ell-k}} = 0,$$

d.h. die Spalten von W sind orthogonal und haben (euklidische) Länge $\sqrt{n}$. Insgesamt ist $\frac{1}{\sqrt{n}} W$ also eine symmetrische orthogonale Matrix mit Inverser $\sqrt{n}\, W^{-1} = \left(\frac{1}{\sqrt{n}} W\right)^{-1} = \frac{1}{\sqrt{n}} \overline{W}^T = \frac{1}{\sqrt{n}} V_n$, wobei die zweite Gleichheit aufgrund der Orthogonalität gilt. ■

Definition. Die Abbildung $\mathcal{F}_n : \mathbb{C}^n \to \mathbb{C}^n$, $\mathcal{F}_n(x) = V_n x$ bezeichnet man als **Diskrete Fourier-Transformation (DFT)** der Länge n. Zu äquidistanten Stützstellen ist $\frac{1}{n}\mathcal{F}_n(y)$ gerade der Koeffizientenvektor des trigonometrischen Interpolationspolynoms. Die inverse Abbildung $\mathcal{F}_n^{-1}$ heißt **Diskrete Fourier-Rücktransformation**. □

Bemerkung. Die Fourier-Matrix V_n hat nur n verschiedene Einträge, nämlich ω_n^ℓ für $\ell = 0, \ldots, n-1$. Die Berechnung von $\mathcal{F}_n(x)$ benötigt daher die einmalige Berechnung der $n-1$ Skalare $\omega_n, \omega_n^2, \ldots \omega_n^{n-1}$ und anschließend eine Matrix-Vektor-Multiplikation mit V_n, die in $n(2n-1)$ arithmetischen Operationen durchgeführt wird. Insgesamt beträgt der Aufwand bei direkter Realisierung also $\mathcal{O}(n^2)$. Die FFT realisiert die Matrix-Vektor-Multiplikation rekursiv mit einem Aufwand $\frac{3}{2} n \log_2(n)$, sodass insgesamt lediglich $\mathcal{O}(n \log n)$ Operationen notwendig sind. □

Bemerkung. Die Skalierung von $\mathcal{F}_n$ ist in der Literatur uneinheitlich. Wir verwenden $\mathcal{F}_n(x) = V_n x$. Bisweilen findet man aber auch $\frac{1}{\sqrt{n}} V_n$ bzw. $\frac{1}{n} V_n$ bei der Definition von $\mathcal{F}_n$. □

Satz 4.22. *Für $p \in \mathbb{N}$, $n = 2^p$ und $m = n/2$ sei wieder $\omega_n = \exp(-2\pi i/n)$ die n-te Einheitswurzel. Wir definieren die Permutation*

$$\sigma_n : \mathbb{C}^n \to \mathbb{C}^n, \quad \sigma_n(x) = (x_1, x_3, \ldots, x_{n-1}, x_2, x_4, \ldots, x_n).$$

Für $x \in \mathbb{C}^n$ folgt dann mit den Vektoren $a, b \in \mathbb{C}^{n/2}$,

$$a_j = x_j + x_{j+m}, \quad b_j = (x_j - x_{j+m})\omega_n^{j-1} \qquad \text{für } j = 1, \ldots, m = n/2,$$

die Gleichheit

$$\sigma_n(\mathcal{F}_n(x)) = \big(\mathcal{F}_m(a), \mathcal{F}_m(b)\big) \in \mathbb{C}^n, \tag{4.14}$$

d.h. die Auswertung von $\mathcal{F}_n$ ist auf die zweifache Auswertung von $\mathcal{F}_{n/2}$ zurückgeführt.

Korollar 4.23. *Als* **Fast Fourier Transform (FFT)** *bezeichnet man die rekursive Berechnung von $\mathcal{F}_n(x)$ gemäß Satz 4.22. Die Berechnung erfordert zunächst $\mathcal{O}(n)$ Operationen um alle Potenzen ω_n^ℓ, $\ell = 0, \dots, n-1$ zu berechnen. Man beachte, dass $\omega_{n/2}^\ell = \omega_n^{2\ell}$ gilt, sodass alle auftretenden Einheitswurzeln bereits zu Anfang berechnet werden. Die gesamte Rekursion benötigt weniger als $\frac{3}{2} n \log_2(n)$ Operationen.*

Beweis. Wir beweisen den Aufwand des Rekursionsverfahrens durch Induktion nach p. Sei a_p die Gesamtanzahl an Additionen/Subtraktionen und m_p die Gesamtanzahl an Multiplikationen zur Berechnung von $\mathcal{F}_n(x)$, $n = 2^p$. Dann gelten $a_p = p\,2^p$ und $m_p \le \frac{1}{2} p\,2^p$. Der Induktionsanfang $p = 1$ ist klar: $a_1 = 2$, $m_1 = 0$. Betrachte nun den Fall $p+1$: Es gilt $n = 2^{p+1}$, $m = n/2 = 2^p$. Neben der Auswertung von $\mathcal{F}_m(a)$ und $\mathcal{F}_m(b)$ sind jeweils $m = 2^p$ Additionen zur Berechnung der Vektoren a bzw. b nötig sowie $m = 2^p$ Multiplikationen zur Berechnung von b. Somit folgt

$$a_{p+1} = 2a_p + 2 \cdot 2^p = (p+1)\,2^{p+1},$$
$$m_{p+1} = 2m_p + 2^p \le p\,2^p + 2^p = (p+1)2^p = \frac{1}{2}(p+1)2^{p+1}.$$

Insgesamt erhalten wir mit $n = 2^p$ also $a_p + m_p \le \frac{3}{2} p\,2^p = \frac{3}{2}\, n \log_2(n)$. ■

Beweis von Satz 4.22. Nach Definition gilt

$$\mathcal{F}_n(y_0, \dots, y_{n-1}) = \Big(\sum_{\ell=0}^{n-1} \omega_n^{j\ell} y_\ell \,\Big|\, j = 0, \dots, n-1 \Big).$$

Nach Indexverschiebung für j und ℓ ergibt sich

$$\mathcal{F}_n(x_1, \dots, x_n) = \Big(\sum_{\ell=0}^{n-1} \omega_n^{(j-1)\ell} x_{\ell+1} \,\Big|\, j = 1, \dots, n \Big).$$

1. Schritt. *Es gilt $(\mathcal{F}_n(x))_{2j-1} = (\mathcal{F}_m(a))_j$.* Nach Definition der Einheitswurzeln gilt

$$\omega_n^2 = \omega_m, \quad \omega_m^m = 1.$$

Deshalb ergibt sich

$$(\mathcal{F}_n(x))_{2j-1} = \sum_{\ell=0}^{n-1} \omega_n^{(2j-2)\ell} x_{\ell+1} = \sum_{\ell=0}^{m-1} \Big\{ \omega_n^{2(j-1)\ell} x_{\ell+1} + \omega_n^{2(j-1)(\ell+m)} x_{\ell+m+1} \Big\}$$
$$= \sum_{\ell=0}^{m-1} \omega_m^{(j-1)\ell} \underbrace{\big\{ x_{\ell+1} + x_{\ell+m+1} \big\}}_{=a_{\ell+1}} = (\mathcal{F}_m(a))_j$$ □

2. Schritt. *Es gilt $(\mathcal{F}_n(x))_{2j} = (\mathcal{F}_m(b))_j$.* Mit den Einheitswurzeln gilt

$$\omega_n^{(2j-1)\ell} = \omega_n^{2(j-1)\ell} \omega_n^\ell = \omega_m^{(j-1)\ell} \omega_n^\ell$$

sowie wegen $\omega_n^{(2j-1)m} = (\omega_n^m)^{(2j-1)} = (-1)^{(2j-1)} = -1$

$$\omega_n^{(2j-1)(\ell+m)} = \omega_n^{2\ell(j-1)} \omega_n^\ell \omega_n^{(2j-1)m} = -\omega_m^{\ell(j-1)} \omega_n^\ell.$$

Analog zum ersten Schritt erhalten wir nun

$$(\mathcal{F}_n(x))_{2j} = \sum_{\ell=0}^{n-1} \omega_n^{(2j-1)\ell} x_{\ell+1} = \sum_{\ell=0}^{m-1} \left\{ \omega_n^{(2j-1)\ell} x_{\ell+1} + \omega_n^{(2j-1)(\ell+m)} x_{\ell+m+1} \right\}$$
$$= \sum_{\ell=0}^{m-1} \omega_m^{\ell(j-1)} \underbrace{\left\{ x_{\ell+1} - x_{\ell+m+1} \right\} \omega_n^{\ell}}_{=b_{\ell+1}} = (\mathcal{F}_m(b))_j \qquad \square$$

Die Kombination von Schritt 1 und Schritt 2 beweist gerade (4.14). ■

Die FFT kann man auch als *schnelle Matrix-Vektor-Multiplikation* $\mathcal{F}_n(x) = V_n x$ mit der Fourier-Matrix V_n deuten. Anstatt eines Aufwands von $\mathcal{O}(n^2)$ für normale dichtbesetzte Matrizen, erlaubt die Struktur von V_n die Matrix-Vektor-Multiplikation mit einem Aufwand von $\mathcal{O}(n \log(n))$.

Übung. Eine Matrix $A \in \mathbb{K}^{n \times n}$ heißt **zirkulant**, wenn es einen Vektor $(a_0, \ldots, a_{n-1}) \in \mathbb{K}^n$ gibt, sodass die Einträge von A durch

$$A_{jk} = a_{(j-k) \bmod (n)}$$

gegeben sind. Schematisch gilt also

$$A = \begin{pmatrix} a_0 & a_{n-1} & a_{n-2} & \ldots & a_1 \\ a_1 & a_0 & a_{n-1} & \ldots & a_2 \\ a_2 & a_1 & a_0 & \ldots & a_3 \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ a_{n-1} & a_{n-2} & a_{n-3} & \ldots & a_0 \end{pmatrix}.$$

Eine zirkulante Matrix $A \in \mathbb{K}^{n \times n}$ wird durch die Fourier-Matrix V_n diagonalisiert:

$$V_n A V_n^{-1} = \operatorname{diag}(p(1), p(\omega_n), \ldots, p(\omega_n^{n-1})) =: D$$

mit dem Polynom $p(x) = \sum_{j=0}^{n} a_j x^j$. Will man das Gleichungssystem $Ax = b$ lösen, so geschieht dies wegen $A = V_n^{-1} D V_n$ in drei Schritten:

- Berechne $\widetilde{b} = V_n b$.
- Löse $Dy = \widetilde{b}$.
- Berechne $x = V_n^{-1} y$.

Dann gilt $b = V_n^{-1}\widetilde{b} = V_n^{-1} D y = V_n^{-1} D V_n x = Ax$. Mit der FFT lassen sich $\widetilde{b} = V_n b$ und $z = V_n \overline{y}$ in $\mathcal{O}(n \log n)$ Operationen berechnen. Es gilt $V_n^{-1} y = \frac{1}{n} \overline{V}_n y = \frac{1}{n} \overline{z}$. Da D diagonal ist, benötigt das Lösen im zweiten Schritt $\mathcal{O}(n)$ Operationen. Insgesamt haben wir damit ein Eliminationsverfahren für $Ax = b$ konstruiert, das lediglich $\mathcal{O}(n \log n)$ arithmetische Operationen benötigt – anstelle $\mathcal{O}(n^3)$ wie die Eliminationsverfahren in Kapitel 3. Da die Matrizen $\frac{1}{\sqrt{n}} V_n$ und $\sqrt{n} V_n^{-1}$ orthogonal sind, folgt $\operatorname{cond}_2(D) = \operatorname{cond}_2(A)$, sodass die Lösung von $Ax = b$ mit diesem Verfahren stabil ist. ■

Die Diskrete Fourier-Transformation verdankt ihren (bisher nicht motivierten) Namen der Verbindung zur Fourier-Transformation: Für eine Riemann-integrierbare Funktion $f : [0, 2\pi] \to \mathbb{C}$ mit $f(0) = f(2\pi)$ gilt

$$f(x) = \sum_{k=-\infty}^{\infty} c_k \exp(kix) \quad \text{in } L^2(0, 2\pi) \tag{4.15}$$

mit **Fourier-Koeffizienten**

$$c_k = \frac{1}{2\pi} \int_0^{2\pi} f(y) \exp(-kiy)\, dy. \tag{4.16}$$

Ein Beweis findet sich zum Beispiel in FORSTER [3, Abschnitt 23]. Die Reihe in (4.15) bezeichnet man als **Fourier-Reihe**.

Aus der Kenntnis der ersten Fourier-Koeffizienten einer glatten periodischen Funktion kann man unter Verwendung der Fourier-Rücktransformation Näherungen für äquidistante Funktionswerte gewinnen:

Satz 4.24. *Es sei $f \in \mathcal{C}^2[0, 2\pi]$ mit $f(0) = f(2\pi)$, $h := 2\pi/n$ und $x_j := jh$ für $0 \le j \le n-1$. Mit den Fourier-Koeffizienten (4.16) gelten dann*

$$\|(c_0, \ldots, c_{n-1}) - \tfrac{1}{n}\mathcal{F}_n(f(x_0), \ldots, f(x_{n-1}))\|_2 = \mathcal{O}(h^{3/2})$$

sowie

$$\|n\mathcal{F}_n^{-1}(c_0, \ldots, c_{n-1}) - (f(x_0), \ldots, f(x_{n-1}))\|_2 = \mathcal{O}(h).$$

Beweis. Wir betrachten den Integranden

$$g_k \in \mathcal{C}^2[0, 2\pi], \quad g_k(y) := f(y) \exp(-iky)$$

des k-ten Fourier-Koeffizienten. Nach Kapitel 1 gilt für die summierte Trapezregel

$$I_n g_k := \frac{h}{2}\Big\{g(0) + 2\sum_{j=1}^{n-1} g(x_j) + g(2\pi)\Big\} = \frac{2\pi}{n}\sum_{j=0}^{n-1} g(x_j)$$

die a priori Fehlerabschätzung

$$\big|2\pi\, c_k - I_n g_k\big| = \Big|\int_0^{2\pi} g_k\, dx - I_n g_k\Big| = \mathcal{O}(h^2),$$

und deshalb folgt

$$\Big|c_k - \frac{1}{n}\sum_{j=0}^{n-1} g_k(x_j)\Big| = \mathcal{O}(h^2) \quad \text{für alle } k = 0, \ldots, n-1.$$

Es gilt

$$\frac{1}{n}\sum_{j=0}^{n-1} g_k(x_j) = \frac{1}{n}\sum_{j=0}^{n-1} f(x_j)\exp(-ikx_j) = \frac{1}{n}\sum_{j=0}^{n-1} f(x_j)\omega_n^{jk} = \Big(\frac{1}{n}\,\mathcal{F}_n(f(x_0),\dots,f(x_{n-1}))\Big)_k,$$

also

$$\|c - \tfrac{1}{n}\,\mathcal{F}_n(f(x_0),\dots,f(x_{n-1}))\|_2^2 = n\,\mathcal{O}(h^4) = \mathcal{O}(h^3)$$

mit dem Vektor $c = (c_0,\dots,c_{n-1})$. Schließlich folgt der Orthogonalitätseigenschaft von $\frac{1}{\sqrt{n}}\mathcal{F}_n$

$$\begin{aligned}\|n\mathcal{F}_n^{-1}(c) - (f(x_0),\dots,f(x_{n-1}))\|_2^2 &= \|\sqrt{n}\,c - \tfrac{1}{\sqrt{n}}\,\mathcal{F}_n(f(x_0),\dots,f(x_{n-1}))\|_2^2\\ &= n\,\|c - \tfrac{1}{n}\,\mathcal{F}_n(f(x_0),\dots,f(x_{n-1}))\|_2^2\\ &= \mathcal{O}(h^2),\end{aligned}$$

was den Beweis beschließt. ■

Bemerkung. Manchmal werden auch trigonometrische Polynome

$$p(x) = \sum_{k=-n}^{n-1} c_k \exp(ikx)$$

betrachtet. In diesem Fall liefert Indexverschiebung

$$p(x) = \sum_{k=0}^{2n-1} c_{k-n}\exp\big(i(k-n)x\big) = \exp(-inx)\sum_{k=0}^{2n-1} c_{k-n}\exp(ikx).$$

Für äquidistante Knoten $x_j = 2\pi j/(2n)$ folgt dann $\exp(-inx_j) = \exp(-\pi ij) = (-1)^j$, also besitzt die Interpolationsaufgabe $p(x_j) = y_j$ eine eindeutige Lösung. Dies folgt z.B. sofort aus Satz 4.20. Auch alle anderen Resultate können entsprechend umgeschrieben werden. □

Bemerkung. Anstatt die komplexe Exponentialfunktion zu verwenden, kann man die Fourier-Transformation auch über $\mathbb{R}$ mittels Sinus und Cosinus formulieren. Die Notation wird dann aber aufwändiger, vgl. Plato [5, Kapitel 3]. □

Kapitel 5

Extrapolation

5.1 Richardson-Extrapolation

Bei einem Interpolationsproblem versucht man, den Funktionswert $\Phi(x)$ zu approximieren, wobei der Auswertungspunkt x zwischen gegebenen Stützstellen liegt, z.B. $x \in [x_0, x_n]$ mit $x_0 < \cdots < x_n$, wobei $\Phi(x_j)$ bekannt ist. Bei einem Extrapolationsproblem liegt x außerhalb des Intervalls $[x_0, x_n]$.

In diesem Abschnitt lautet die abstrakte Formulierung wie folgt. Gegeben sei eine stetige Funktion $\Phi : [0, 1] \to \mathbb{K}$. Für gewisse $h_j \in (0, 1]$ sei $\Phi(h_j)$ bekannt. Ziel ist es, eine Approximation von $\Phi(0)$ zu berechnen.

Beispiel. Ist f eine stetig differenzierbare Funktion, so betrachten wir bei der numerischen Differentiation den Differenzenquotienten

$$\Phi(h) = \frac{f(x+h) - f(x)}{h} \quad \text{für } h > 0, \tag{5.1}$$

um die unbekannte Ableitung $\Phi(0) = f'(x)$ zu berechnen. Ist die Funktion f hinreichend glatt, so zeigt eine Taylor-Entwicklung

$$f(x+h) = \sum_{j=0}^{n+1} \frac{f^{(j)}(x)}{j!}\, h^j + \mathcal{O}(h^{n+2}),$$

und damit folgt

$$\Phi(h) = \sum_{j=1}^{n+1} \frac{f^{(j)}(x)}{j!}\, h^{j-1} + \mathcal{O}(h^{n+1}) = \Phi(0) + \sum_{k=1}^{n} \frac{f^{(k+1)}(x)}{(k+1)!}\, h^k + \mathcal{O}(h^{n+1}),$$

d.h. wir erhalten eine asymptotische Entwicklung von $\Phi(h)$. ■

Unter der Voraussetzung, dass wir zu Stützstellen $1 \geq h_0 > \cdots > h_n > 0$ die Funktionswerte $\Phi(h_j)$ kennen, wollen wir eine Approximation von $\Phi(0)$ berechnen: Aus naiver Sichtweise ist $\Phi(h_n)$ die beste berechnete Approximation von $\Phi(0)$. Dabei vernachlässigen wir aber die zusätzliche Information von $\Phi(h_j)$ für $j = 0, \ldots, n-1$.

Algorithmus 5.1: Richardson-Extrapolation

Die Funktion $\Phi : [0,1] \to \mathbb{K}$ lasse sich in der Form

$$\Phi(h) = \Phi(0) + \sum_{k=1}^{n} a_k h^{\alpha k} + \mathcal{O}(h^{\alpha(n+1)}) \quad \text{für } h > 0 \tag{5.2}$$

entwickeln, wobei der Skalar $\alpha > 0$ bekannt und neben dem gesuchten Wert $\Phi(0)$ auch die Koeffizienten $a_1, \ldots, a_n \in \mathbb{K}$ unbekannt seien. Sind für $h_0, \ldots, h_n \in (0,1]$ die Funktionswerte $\Phi(h_j)$ bekannt, so existiert das eindeutige Interpolationspolynom $p \in \mathbb{P}_n$ mit $p(h_j^\alpha) = \Phi(h_j)$ für $j = 0, \ldots, n$. Der Funktionswert $p(0)$ kann mit dem Neville-Verfahren berechnet werden und ist eine Approximation von $\Phi(0)$.

Bemerkung. Falls $\Phi(h)$ eine asymptotische Entwicklung (5.2) besitzt, so gilt $|\Phi(h) - \Phi(0)| = \mathcal{O}(h^\alpha)$, und für $a_1 \neq 0$ ist α maximal: Gilt $|\Phi(h) - \Phi(0)| = \mathcal{O}(h^\beta)$, so folgt $\beta \leq \alpha$. □

Nach dem vorausgegangenen Beispiel erfüllt der einseitige Differenzenquotient (5.1) die Voraussetzung (5.2) mit $\alpha = 1$. Bevor wir die Konvergenz der Richardson-Extrapolation beweisen, geben wir noch einige weitere Beispiele.

Beispiel. Wir betrachten den zentralen Differenzenquotienten

$$\Phi(h) = \frac{f(x+h) - f(x-h)}{2h} \quad \text{für } h > 0 \tag{5.3}$$

zur numerischen Berechnung von $\Phi(0) = f'(x)$. Ist f hinreichend glatt, so gilt nach Taylorentwicklung um x

$$f(x+h) = \sum_{j=0}^{2n+2} \frac{f^{(j)}(x)}{j!} h^j + \mathcal{O}(h^{2n+3}) \quad \text{und} \quad f(x-h) = \sum_{j=0}^{2n+2} \frac{f^{(j)}(x)}{j!} (-h)^j + \mathcal{O}(h^{2n+3}).$$

Damit ergibt sich

$$\begin{aligned}
\Phi(h) &= \frac{1}{2h}\left(\sum_{j=0}^{2n+2} \frac{f^{(j)}(x)}{j!} (h^j - (-h)^j) + \mathcal{O}(h^{2n+3}) \right) \\
&= \frac{1}{2h}\left(2 \sum_{\ell=1}^{n+1} \frac{f^{(2\ell-1)}(x)}{(2\ell-1)!} h^{2\ell-1} + \mathcal{O}(h^{2n+3}) \right) \\
&= \sum_{\ell=1}^{n+1} \frac{f^{(2\ell-1)}(x)}{(2\ell-1)!} h^{2\ell-2} + \mathcal{O}(h^{2n+2}) \\
&= f'(x) + \sum_{k=1}^{n} \frac{f^{(2k+1)}(x)}{(2k+1)!} h^{2k} + \mathcal{O}(h^{2(n+1)}).
\end{aligned}$$

Insgesamt erfüllt $\Phi(h)$ also (5.2) mit $\alpha = 2$. ■

Beispiel (Romberg-Verfahren). Als Romberg-Verfahren bezeichnet man die Richardson-Extrapolation der summierten Trapezregel

$$\Phi(h) = \frac{h}{2}\left(f(a) + 2\sum_{j=1}^{n-1} f(x_j) + f(b) \right) \quad \text{mit } h = \frac{b-a}{n} \text{ und Stützstellen } x_j = a + jh \tag{5.4}$$

zur Approximation des Integrals $\Phi(0) = \int_a^b f(x)\,dx$. In Kapitel 1 haben wir uns bereits überlegt, dass $|\Phi(h) - \Phi(0)| = \mathcal{O}(h^2)$ gilt. Die Entwicklung von $\Phi(h)$ in der Form (5.2) mit $\alpha = 2$ heißt *Euler-MacLaurin'sche Summenformel*, siehe PLATO [5, Abschnitt 6.9]. ■

Beispiel (Explizites Euler-Verfahren). Auch die numerische Lösung von Anfangswertproblemen passt in unseren abstrakten Rahmen. Zu gegebener rechter Seite $f(x, y)$ betrachten wir das Anfangswertproblem

$$y'(x) = f(x, y) \quad \text{in } [0, 1] \quad \text{mit gegebenem Anfangswert } y(0) = y_0 \tag{5.5}$$

und gesuchter Lösung $y \in \mathcal{C}^1[0, 1]$. In der Regel interessiert man sich für das Endzeitverhalten $y(1)$. Das einfachste Verfahren zur numerischen Lösung von (5.5) ist das explizite Eulerverfahren. Zu $n \in \mathbb{N}$ und $h := 1/n$ fixiere Stützstellen $x_j = jh$ für $j = 0, \dots, n$. Ziel ist die Berechnung von $y_j \approx y(x_j)$. Dazu ersetzt man die exakte Ableitung $y'(x_j)$ in (5.5) durch den einseitigen Differenzenquotienten $y'(x_j) \approx \frac{y_{j+1} - y_j}{h}$. Umformulierung führt auf das explizite Einschrittverfahren

$$y_{j+1} := y_j + hf(x_j, y_j) \quad \text{für } j = 1, \dots, n. \tag{5.6}$$

In diesem Fall ist

$$\Phi(h) = y_n \approx y(1) = \Phi(0), \tag{5.7}$$

und $\Phi(h)$ erfüllt (5.2) mit $\alpha = 1$. Details folgen in der Vorlesung zur Numerik von Differentialgleichungen bzw. finden sich in PLATO [5, Abschnitt 7.5, 7.6]. ■

Der folgende Satz zeigt, dass die Richardson-Extrapolation exponentiell mit dem Polynomgrad n konvergiert.

Satz 5.2. *Mit einer Konstante $c_1 > 0$ und $\alpha > 0$ erfülle $\Phi : [0, 1] \to \mathbb{K}$ die Entwicklung*

$$\Phi(h) = \Phi(0) + \sum_{j=1}^{n} a_j h^{\alpha j} + a_{n+1}(h) \quad \textit{mit} \quad |a_{n+1}(h)| \le c_1 h^{\alpha(n+1)} \qquad \textit{für } h > 0. \tag{5.8}$$

Es sei $\rho \in (0, 1)$, und die Auswertungspunkte h_k seien gegeben durch $h_k := \rho^k$ für $k = 0, \dots, n$. Es sei $p \in \mathbb{P}_n$ das Interpolationspolynom mit $p(h_k^\alpha) = \Phi(h_k)$ für alle $k = 0, \dots, n$. Dann existiert eine Konstante $c_2 > 0$, die nur von c_1, ρ und α abhängt, mit

$$|p(0) - \Phi(0)| \le c_2\, \rho^{\alpha n(n+1)/2}. \tag{5.9}$$

Bemerkung. Nach Satz 5.2 gilt also $|p(0)-\Phi(0)| = \mathcal{O}(\rho^{\alpha n(n+1)/2})$. Naive Realisierung mit $h_n := \rho^n$ führt lediglich auf $|\Phi(h_n)-\Phi(0)| = \mathcal{O}(\rho^{\alpha n})$.Man beachte, dass der Exponent in diesem Fall lediglich linear in n ist und nicht – wie bei der Richardson-Extrapolation – quadratisch in n. Man sagt deshalb, *die Richardson-Extrapolation vermindere Auslöschungseffekte*, da man für dieselbe Genauigkeit der numerischen Lösung wesentlich größere h_j betrachten kann. Die Richardson-Extrapolation ist aber nur dann anwendbar (und sinnvoll), wenn man die (maximale) Ordnung α des Verfahrens kennt. □

Beweis von Satz 5.2. Es sei $x_\ell := h_\ell^\alpha = \rho^{\ell\alpha}$ und $L_\ell \in \mathbb{P}_n$ das zugehörige Lagrange-Polynom

$$L_\ell(x) = \prod_{\substack{k=0\\k\neq\ell}}^{n} \frac{x - x_k}{x_\ell - x_k}.$$

1. Schritt. *Es gilt* $|p(0)-\Phi(0)| \leq c_1 \sum_{\ell=0}^n \rho^{\ell\alpha(n+1)}|L_\ell(0)|$. Das Polynom p wird als Linearkombination der Lagrange-Polynome dargestellt, $p(x) = \sum_{\ell=0}^n \Phi(h_\ell)L_\ell(x)$. Einsetzen der Entwicklung (5.8) liefert

$$\begin{aligned} p(x) &= \Phi(0)\sum_{\ell=0}^n L_\ell(x) + \sum_{j=1}^n a_j \sum_{\ell=0}^n h_\ell^{\alpha j} L_\ell(x) + \sum_{l=0}^n a_{n+1}(h_\ell)L_\ell(x) \\ &= \Phi(0) + \sum_{j=1}^n a_j x^j + \sum_{\ell=0}^n a_{n+1}(h_\ell)L_\ell(x). \end{aligned}$$

Dabei wurde ausgenutzt, dass mit der eindeutigen Lösbarkeit der Polynominterpolation gerade $\sum_{\ell=0}^n L_\ell(x) = 1$ und $\sum_{\ell=0}^n h_\ell^{\alpha j} L_\ell(x) = \sum_{\ell=0}^n x_\ell^j L_\ell(x) = x^j$ gelten. Für $x = 0$ ergibt sich deshalb

$$|p(0) - \Phi(0)| \leq \sum_{\ell=0}^n |a_{n+1}(h_\ell)||L_\ell(0)|.$$

Mit $h_\ell^{\alpha(n+1)} = \rho^{\ell\alpha(n+1)}$ folgt die Behauptung. □

2. Schritt. $|L_\ell(0)| = \rho^{-\ell\alpha(n+1)}\rho^{\alpha n(n+1)/2} \prod_{\substack{k=0\\k\neq\ell}}^{n} \frac{1}{|1-\rho^{(k-\ell)\alpha}|}$. Elementare Umformung zeigt

$$|L_\ell(0)| = \prod_{\substack{k=0\\k\neq\ell}}^{n} \frac{|x_k|}{|x_\ell - x_k|} = \frac{\prod\limits_{k\neq\ell} \rho^{\alpha k}}{\prod\limits_{k\neq\ell} |\rho^{\alpha\ell} - \rho^{\alpha k}|} = \frac{\prod\limits_{k\neq\ell} \rho^{\alpha k}}{\prod\limits_{k\neq\ell} \left(\rho^{\alpha\ell}|1 - \rho^{\alpha(k-\ell)}|\right)} = \frac{\prod\limits_{k=0}^{n} \rho^{\alpha k}}{\rho^{\ell\alpha(n+1)} \prod\limits_{k\neq\ell} |1 - \rho^{\alpha(k-\ell)}|}$$

Beachtet man

$$\prod_{k=0}^n \rho^{k\alpha} = \rho^{\alpha\sum_{k=0}^n k} = \rho^{\alpha n(n+1)/2},$$

so folgt die Behauptung. □

3. Schritt. *Es gilt* $\displaystyle\prod_{\substack{k=0\\k\neq\ell}}^{n}\frac{1}{|1-\rho^{(k-\ell)\alpha}|}\le\rho^{\alpha\ell(\ell+1)/2}\prod_{k=1}^{n}\frac{1}{(1-\rho^{k\alpha})^2}$.

Die wesentliche Idee ist es, die Menge der positiven und negativen ρ-Potenzen getrennt zu betrachten. Es gilt

$$\big\{k-\ell\,\big|\,k=0,\dots n;k\neq\ell\big\}=\{-\ell,\dots,-1,1,\dots,n-\ell\},$$

und deshalb folgt

$$\prod_{\substack{k=0\\k\neq\ell}}^{n}|1-\rho^{(k-\ell)\alpha}|=\Big(\prod_{k=1}^{\ell}|1-\rho^{-k\alpha}|\Big)\Big(\prod_{k=1}^{n-\ell}|1-\rho^{k\alpha}|\Big).\tag{5.10}$$

Das zweite Produkt in (5.10) kann mittels

$$\prod_{k=1}^{n-\ell}|1-\rho^{k\alpha}|=\prod_{k=1}^{n-\ell}(1-\rho^{k\alpha})\ge\prod_{k=1}^{n}(1-\rho^{k\alpha})$$

abgeschätzt werden. Für das erste Produkt in (5.10) gilt mit $|1-\rho^{-k\alpha}|=\frac{1-\rho^{k\alpha}}{\rho^{k\alpha}}$

$$\prod_{k=1}^{\ell}|1-\rho^{-k\alpha}|=\Big(\prod_{k=1}^{\ell}\rho^{-k\alpha}\Big)\Big(\prod_{k=1}^{\ell}(1-\rho^{k\alpha})\Big)\ge\rho^{-\alpha\ell(\ell+1)/2}\prod_{k=1}^{n}(1-\rho^{k\alpha}),$$

wobei wir wie oben $\prod_{k=1}^{\ell}\rho^{-k\alpha}=\rho^{-\alpha\ell(\ell+1)/2}$ benutzt haben. Insgesamt folgt damit

$$\prod_{\substack{k=0\\k\neq\ell}}^{n}|1-\rho^{(k-\ell)\alpha}|\ge\rho^{-\alpha\ell(\ell+1)/2}\prod_{k=1}^{n}(1-\rho^{k\alpha})^2.$$ □

4. Schritt. *Es gilt* $\displaystyle\prod_{k=1}^{n}\frac{1}{(1-\rho^{k\alpha})^2}\le\exp\Big(\frac{2\rho^{\alpha}}{(1-\rho^{\alpha})^2}\Big)$. Zum Beweis schätzen wir den Logarithmus der linken Seite ab,

$$\log\Big(\prod_{k=1}^{n}\frac{1}{(1-\rho^{k\alpha})^2}\Big)=2\sum_{k=1}^{n}\log\Big(\frac{1}{1-\rho^{k\alpha}}\Big)=2\sum_{k=1}^{n}\log\Big(1+\frac{\rho^{k\alpha}}{1-\rho^{k\alpha}}\Big).$$

Mit den elementaren Abschätzungen $\log(1+t)\le t$ und $\frac{1}{1-\rho^{k\alpha}}\le\frac{1}{1-\rho^{\alpha}}$ folgt weiter

$$\le\frac{2}{1-\rho^{\alpha}}\sum_{k=1}^{n}\rho^{k\alpha}.$$

Mit der geometrischen Reihe $\sum_{k=1}^{\infty}\rho^{k\alpha}=\frac{1}{1-\rho^{\alpha}}-1=\frac{\rho^{\alpha}}{1-\rho^{\alpha}}$ folgt insgesamt

$$\log\Big(\prod_{k=1}^{n}\frac{1}{(1-\rho^{k\alpha})^2}\Big)\le\frac{2\rho^{\alpha}}{(1-\rho^{\alpha})^2}.$$

Anwenden der Exponentialfunktion liefert die Behauptung. □

5. Schritt. *Die Kombination der vorausgegangenen Abschätzungen verifiziert* (5.9). Mit dem ersten und zweiten Beweisschritt erhalten wir

$$|p(0) - \Phi(0)| \le c_1 \rho^{\alpha n(n+1)/2} \sum_{\ell=0}^{n} \prod_{\substack{k=0 \\ k\neq\ell}}^{n} \frac{1}{|1-\rho^{(k-\ell)\alpha}|}$$

Die Summanden werden mit dem dritten und vierten Beweisschritt abgeschätzt zu

$$\prod_{\substack{k=0 \\ k\neq\ell}}^{n} \frac{1}{|1-\rho^{(k-\ell)\alpha}|} \le \rho^{\alpha\ell(\ell+1)/2} \exp\Big(\frac{2\rho^\alpha}{(1-\rho^\alpha)^2}\Big).$$

Damit erhalten wir

$$|p(0) - \Phi(0)| \le c_1 \rho^{\alpha n(n+1)/2} \exp\Big(\frac{2\rho^\alpha}{(1-\rho^\alpha)^2}\Big) \sum_{\ell=0}^{n} \rho^{\alpha\ell(\ell+1)/2} \le c_2 \rho^{\alpha n(n+1)/2}$$

mit der Konstanten $c_2 = \dfrac{c_1}{1-\rho^\alpha} \exp\Big(\dfrac{2\rho^\alpha}{(1-\rho^\alpha)^2}\Big)$. ■

Die Folge der Stützstellen $h_k := \rho^k$ mit fixiertem $\rho \in (0,1)$ bezeichnet man als **Romberg-Folge**. Alternativ kann man auch die **harmonische Folge** $h_k = 1/(k+1)$ oder die **Bulirsch-Folge** nehmen.

5.2 Aitkinsches Δ^2 - Verfahren

Beim Δ^2-Verfahren von Aitkin handelt es sich um ein Verfahren zur **Konvergenzbeschleunigung von Folgen**. Es sei $(x_j)_{j\in\mathbb{N}}$ eine konvergente Folge in $\mathbb{K}$ mit Grenzwert $x \in \mathbb{K}$. Ziel ist die Konstruktion einer Folge $(y_j)_{j\in\mathbb{N}}$, sodass gilt

$$\lim_{j\to\infty} \frac{y_j - x}{x_j - x} = 0, \tag{5.11}$$

d.h. die Folge $(y_j)_{j\in\mathbb{N}}$ konvergiert schneller gegen x als $(x_j)_{j\in\mathbb{N}}$.

Bemerkung. Für eine Folge $(\zeta_j)_{j\in\mathbb{N}}$ definieren wir den Differenzenoperator Δ durch

$$\Delta\zeta_j := \zeta_{j+1} - \zeta_j \quad \text{für } j \in \mathbb{N}. \tag{5.12}$$

Es sei $(x_j)_{j\in\mathbb{N}}$ eine geometrisch konvergente Folge mit Limes $x \in \mathbb{K}$, d.h. es existiert ein Skalar $k \in \mathbb{K}$ mit $|k| < 1$ und

$$x_{j+1} - x = k(x_j - x) \quad \text{für } j \in \mathbb{N}. \tag{5.13}$$

Dann gilt $\lim\limits_{j\to\infty} x_j = x$, und unter der Voraussetzung $x_j \neq x$ für alle $j \in \mathbb{N}$ folgt

$$x = x_j - \frac{(\Delta x_j)^2}{\Delta^2 x_j} = x_j - \frac{(x_{j+1} - x_j)^2}{x_{j+2} - 2x_{j+1} + x_j}, \tag{5.14}$$

d.h. x kann aus drei aufeinanderfolgenden Folgengliedern x_j, x_{j+1}, x_{j+2} berechnet werden. Um dies zu sehen, betrachten wir die Differenz

$$\Big(x_j - \frac{(x_{j+1}-x_j)^2}{x_{j+2}-2x_{j+1}+x_j}\Big) - x = (x_j - x) - \frac{\big((x_{j+1}-x)-(x_j-x)\big)^2}{(x_{j+2}-x)-2(x_{j+1}-x)+(x_j-x)}$$
$$= (x_j - x) - \frac{\big((k-1)(x_j-x)\big)^2}{(k^2-2k+1)(x_j-x)} = 0,$$

wobei wir in der zweiten Gleichheit die geometrisch Folgeneigenschaft ausgenutzt haben. Man beachte, dass wir mit Hilfe der beiden Gleichungen $(x_{j+2}-x) = k(x_{j+1}-x)$ und $(x_{j+1}-x) = k(x_j-x)$ die Unbekannten k und x berechnen können. Es ist jedoch hervorzuheben, dass diese Gleichung wegen der Terms kx nichtlinear ist! □

Satz 5.3. *Die Folge $(x_j)_{j\in\mathbb{N}}$ erfülle $x_j \neq x$ und $x_{j+1} - x = (k+\delta_j)(x_j - x)$ mit einer Nullfolge $(\delta_j)_{j\in\mathbb{N}}$ und einem Skalar $k \in \mathbb{K}$ mit $|k| < 1$, d.h. asymptotisch ist die Folge $(x_j)_{j\in\mathbb{N}}$ geometrisch konvergent. Dann gilt $\lim_{j\to\infty} x_j = x$. Ferner existiert ein Index $j_0 \in \mathbb{N}$, sodass*

$$y_j := x_j - \frac{(\Delta x_j)^2}{\Delta^2 x_j} \tag{5.15}$$

für alle $j \geq j_0$ wohldefiniert ist, d.h. $\Delta^2 x_j \neq 0$, und es gilt (5.11), $\lim_{j\to\infty} \frac{y_j - x}{x_j - x} = 0$.

Beweis. Es sei $q \in (|k|, 1)$. Da $(\delta_j)_{j\in\mathbb{N}}$ eine Nullfolge ist, existiert ein Index $j_1 \in \mathbb{N}$, sodass $|\delta_j| < q - |k|$ für alle $j \geq j_1$ gilt. Ohne Beschränkung der Allgemeinheit sei $j_1 = 1$. Insbesondere folgt $|k+\delta_j| \leq q < 1$. Induktiv erhalten wir $|x_{j+1} - x| \leq q^j |x_1 - x|$, und mit $\lim_{j\to\infty} q^j = 0$ folgt die Konvergenz von $(x_j)_{j\in\mathbb{N}}$.

Wir betrachten nun den Fehlerterm $e_j := x_j - x$. Nach Voraussetzung gilt $e_{j+1} = (k+\delta_j)e_j$, und deshalb folgt für den Nenner der Aitkin-Folge $(y_j)_{j\in\mathbb{N}}$

$$\begin{aligned}\Delta^2 x_j = x_{j+2} - 2x_{j+1} + x_j &= e_{j+2} - 2e_{j+1} + e_j \\ &= e_j\big((k+\delta_{j+1})(k+\delta_j) - 2(k+\delta_j) + 1\big) \\ &= e_j\big((k-1)^2 + (\delta_j\delta_{j+1} + k(\delta_j+\delta_{j+1}) - 2\delta_j)\big)\end{aligned}$$

Wegen $e_j \neq 0$, $(k-1)^2 \neq 0$ und $\mu_j := \delta_j\delta_{j+1} + k(\delta_j+\delta_{j+1}) - 2\delta_j \xrightarrow{j\to\infty} 0$ existiert also ein Index $j_0 \in \mathbb{N}$, so dass y_j für $j \geq j_0$ wohldefiniert ist. Für den Zähler der Aitkin-Folge gilt

$$\Delta x_j = x_{j+1} - x_j = e_{j+1} - e_j = e_j((k+\delta_j) - 1).$$

Insgesamt ergibt sich damit

$$y_j - x = e_j - \frac{e_j^2(k-1+\delta_j)^2}{e_j((k-1)^2+\mu_j)} = e_j\Big(1 - \frac{(k-1+\delta_j)^2}{(k-1)^2+\mu_j}\Big).$$

Division durch e_j zeigt die Behauptung, denn der Bruch konvergiert gegen 1 für $j \to \infty$. ■

Bemerkung. Die Voraussetzungen des Aitkin-Satzes sind defacto für fast jedes numerische Verfahren erfüllt: Im Falle des einseitigen Differenzenquotientens haben wir für $f \in \mathcal{C}^2$ und $e_j = |f'(x) - \Phi(h_j)|$ oben bereits $e_j \leq \|f''\|_{\infty,[x,x+h_j]}\, h_j$ bewiesen. Wenn man den Beweis genauer betrachtet, gilt sogar $e_j = C(1+\lambda_j)h_j$ mit $C = |f''(x)|$ und (λ_j) einer Nullfolge. Setzen wir dieses *typische Konvergenzverhalten* eines numerischen Verfahrens für den Fehler e_j voraus,

$$e_j = C(1+\lambda_j)h_j^\alpha$$

mit einer (unbekannten) Konvergenzrate $\alpha > 0$, so erhalten wir

$$e_{j+1} = C(1+\lambda_{j+1})h_{j+1}^\alpha = \frac{1+\lambda_{j+1}}{1+\lambda_j}\,\frac{h_{j+1}^\alpha}{h_j^\alpha}\,e_j.$$

Normalerweise ergibt sich h_{j+1} aus h_j, beispielsweise durch Halbierung $h_{j+1} = h_j/2$. In diesem Fall sehen wir also

$$e_{j+1} = \frac{1+\lambda_{j+1}}{1+\lambda_j}\,2^{-\alpha}e_j.$$

Definiert man $k := 2^{-\alpha}$ und $\delta_j = 1 - \frac{1+\lambda_{j+1}}{1+\lambda_j}$, so erhalten wir mit $e_{j+1} = (k+\delta_j)e_j$ die Voraussetzung des Aitkin-Satzes. □

Kapitel 6

Quadratur

6.1 Konvergenz von Quadraturverfahren

Im ganzen Kapitel bezeichnet $a, b \in \mathbb{R}$ mit $a < b$ das **Integrationsgebiet**. Es sei $\omega \in L^1(a, b)$ eine **Gewichtsfunktion** mit

$$\omega > 0 \text{ fast überall in } (a, b) \quad \text{und} \quad \int_a^b |x|^j \omega(x)\, dx < \infty \quad \text{für alle } j \in \mathbb{N}. \tag{6.1}$$

Die **numerische Integration** bzw. **Quadratur** ist die approximative Berechnung des Integrals

$$Q : \mathcal{C}[a, b] \to \mathbb{K}, \quad Qf := \int_a^b f(x)\omega(x)\, dx. \tag{6.2}$$

Da $\mathcal{C}[a, b]$ fr das kompakte Intervall $[a, b]$ ein Unterraum von $L^\infty(a, b)$ ist, existiert das Integral Qf. Durch die zweite Voraussetzung an ω ist sichergestellt, dass Qp für alle Polynome $p \in \mathbb{P}(a, b)$ existiert.

Bemerkung. Soll eine Funktion $g : (a, b) \to \mathbb{K}$ integriert werden, so zerlegt man g in das Produkt $g(x) = f(x)\omega(x)$, wobei f der glatte Anteil von g ist und ω der nicht-glatte Anteil. Für $g(x) = \sin(x)/\sqrt{1 - x^2}$ wäre beispielsweise $f(x) = \sin(x)$ und $\omega(x) = 1/\sqrt{1 - x^2}$. □

Definition. Gegeben seien paarweise verschiedene **Stützstellen** (bzw. **Knoten**) $x_j \in \mathbb{R}$ mit $a \leq x_0, \ldots, x_n \leq b$ und Gewichte $\omega_0, \ldots, \omega_n \in \mathbb{K}$. Die Abbildung

$$Q_n : \mathcal{C}[a, b] \to \mathbb{K}, \quad Q_n f := \sum_{j=0}^{n} \omega_j f(x_j) \tag{6.3}$$

heißt **Quadraturformel** auf $[a, b]$ der **Länge** $n \in \mathbb{N}$ (oder: vom Grad n). Im Folgenden wird der Index n immer die Länge (d.h. die Anzahl an Stützstellen ist $n + 1$) der Quadraturformel Q_n bezeichnen. Q_n hat **Exaktheitsgrad** $m \in \mathbb{N}_0$ (oder: ist exakt vom Grad $m \in \mathbb{N}_0$), falls Polynome vom Grad m noch exakt integriert werden, d.h.

$$Q_n p = Qp \quad \text{für alle } p \in \mathbb{P}_m(a, b). \tag{6.4}$$

Das folgende Lemma stellt erste triviale Eigenschaften von Quadraturformeln zusammen.

Lemma 6.1. (i) *Q, Q_n sind lineare und stetige Funktionale auf $\mathcal{C}[a,b]$, d.h. $Q, Q_n \in \mathcal{C}[a,b]^*$, und für die Operatornormen gelten $\|Q\| = \|\omega\|_{L^1(a,b)}$, $\|Q_n\| = \sum_{j=0}^n |\omega_j|$.*
(ii) *Der maximale Exaktheitsgrad von Q_n ist $2n+1$.*
(iii) *Ist Q_n exakt auf $\mathbb{P}_{2n+1}$, so gibt es kein Polynom $p \in \mathbb{P}_{2n+2}$ vom Grad $2n+2$ mit $Qp = Q_n p$.*
(iv) *Ist Q_n exakt vom Grad 0, so folgt $\sum_{j=0}^n \omega_j = \|\omega\|_{L^1(a,b)}$.*
(v) *Im Spezialfall $\omega(x) = 1$ und $a, b \in \mathbb{R}$ gilt $\sum_{j=0}^n \omega_j = b - a$ und $\sum_{j=0}^n \omega_j x_j = (b^2 - a^2)/2$, sofern Q_n exakt vom Grad 1 ist.*

Beweis. (i) Offensichtlich sind Q, Q_n lineare Funktionale, und es ist lediglich die Operatornorm zu berechnen. Mit Hölder-Ungleichung gilt

$$|Qf| \le \|f\|_{L^\infty(a,b)} \|\omega\|_{L^1(a,b)},$$

also $\|Q\| \le \|\omega\|_{L^1(a,b)}$. Mit der konstanten $\mathbb{1}$-Funktion folgt Gleichheit $Q\mathbb{1} = \|\omega\|_{L^1(a,b)}$. Für Q_n folgt mit Dreiecksungleichung

$$|Q_n f| \le \sum_{j=0}^n |\omega_j| |f(x_j)| \le \|f\|_\infty \sum_{j=0}^n |\omega_j|, \tag{6.5}$$

also $\|Q_n\| \le \sum_{j=0}^n |\omega_j|$. Um die Gleichheit zu zeigen, müssen wir wieder eine Funktion $f \in \mathcal{C}[a,b]$ mit $\|f\|_\infty = 1$ und $Q_n f = \sum_{j=0}^n |\omega_j|$ konstruieren. Ohne Beschränkung der Allgemeinheit gilt $x_0 < \cdots < x_n$. Nun wähle f auf $[x_0, x_n]$ als affinen Spline mit der Eigenschaft

$$|f(x_j)| = 1, \quad f(x_j)\omega_j = |\omega_j| \quad \text{für alle } j = 0, \dots, n$$

und setze f auf $(a, x_0]$ und $[x_n, b)$ konstant fort. Dann gilt offensichtlich $f \in \mathcal{C}[a,b]$ und $|f(x)| \le 1$ für alle $x \in \overline{(a,b)}$ mit Gleichheit in (6.5).

(ii) Zum Beweis müssen wir lediglich ein Polynom $p \in \mathbb{P}_{2n+2}$ konstruieren, das nicht exakt integriert wird. Definiere

$$p(x) = \prod_{j=0}^n (x - x_j)^2 \in \mathbb{P}_{2n+2}. \tag{6.6}$$

Dann gilt $Q_n p = 0$. Andererseits gilt $p(x)\omega(x) > 0$ für fast alle $x \in (a,b)$ und deshalb $Qp > 0$.

(iii) Das Residuum $R_n := Q - Q_n : \mathbb{P}_{2n+2} \to \mathbb{K}$ ist eine lineare Abbildung und erfüllt $R_n|_{\mathbb{P}_{2n+1}} = 0$. Sei $\mathcal{B} \subseteq \mathbb{P}_{2n+1}$ eine Basis. Ist q ein Polynom vom Grad $2n+2$, so ist $\mathcal{B} \cup \{q\}$ eine Basis von $\mathbb{P}_{2n+2}$. Wäre $R_n q = 0$, so folgte deshalb $R_n|_{\mathbb{P}_{2n+2}} = 0$. Andererseits gilt aber $R_n p > 0$ mit dem Polynom $p \in \mathbb{P}_{2n+2}$ aus (6.6).

(iv) Die konstante Funktion $\mathbb{1} \in \mathbb{P}_0$ wird exakt integriert. Es gilt $\|\omega\|_{L^1(a,b)} = Q\mathbb{1} = Q_n\mathbb{1} = \sum_{j=0}^n \omega_j$.

(v) Nach (iv) gilt $\sum_{j=0}^n \omega_j = b - a$. Die identische Funktion ist in $\mathbb{P}_1$ und wird deshalb exakt integriert, d.h. es gilt $\sum_{j=0}^n \omega_j x_j = \int_a^b x\,dx = b^2/2 - a^2/2$. ∎

Bemerkung. Bisweilen ist in der Literatur (z.B. PLATO [5]) der Laufindex $j = 1, \dots, n$ statt $j = 0, \dots, n$. Das hat natürlich Konsequenzen für den optimalen Exaktheitsgrad in (ii), (iii). Die Gauß'schen Quadraturformeln haben maximalen Exaktheitsgrad $2n + 1$, siehe Abschnitt 6.3. □

Bemerkung. In der Literatur werden Quadraturformeln auf Standardintervallen gegeben, z.B. auf dem Intervall $[0, 1]$ oder auf $[-1, 1]$, siehe z.B. ABRAMOVITZ oder SECREST, STROUD. Um daraus Quadraturformeln auf $[a, b]$ für $a, b \in \mathbb{R}$ zu erhalten, nutzt man Transformationen, z.B.

$$\Phi : [-1, 1] \to [a, b], \quad \Phi(t) = \frac{1}{2}\big\{a + b + t(b - a)\big\}. \tag{6.7}$$

Mit Transformationssatz gilt dann

$$\int_a^b f\omega \, dx = \int_{-1}^1 f(\Phi(t))\, \omega(\Phi(t))\, |\det D\Phi(t)|\, dt = \frac{b-a}{2} \int_{-1}^1 f(\Phi(t))\, \omega(\Phi(t))\, dt.$$

Ist $\widetilde{Q}_n g = \sum_{j=0}^n \widetilde{\omega}_j g(\widetilde{x}_j)$ eine Quadraturformel auf $[-1, 1]$ zur Gewichtsfunktion $\omega \circ \Phi$, so definiert $Q_n f := \sum_{j=0}^n \omega_j f(x_j)$ mit $\omega_j = (b - a)\widetilde{\omega}_j/2$ und $x_j := \Phi(\widetilde{x}_j)$ eine Quadraturformel auf $[a, b]$. Da die Abbildung Φ affin ist, hat Q_n denselben Exaktheitsgrad wir $\widetilde{Q}_n$. □

Der folgende Satz beweist die Konvergenz von Quadraturformeln, sofern die Polynome asymptotisch korrekt integriert werden. Ferner sehen wir, dass der Quadraturfehler mindestens mit der Ordnung des Bestapproximationsfehlers (bzw. Interpolationsfehlers) fällt.

Satz 6.2. *Zu $a, b \in \mathbb{R}$ und $n \in \mathbb{N}$ sei $Q_n f = \sum_{j=0}^n \omega_j^{(n)} f(x_j^{(n)})$ eine Quadraturformel der Länge n. Hat Q_n Exaktheitsgrad $m \geq 0$, so gilt*

$$|Qf - Q_n f| \leq \Big(\|\omega\|_{L^1(a,b)} + \sum_{j=0}^n |\omega_j^{(n)}| \Big) \min_{p \in \mathbb{P}_m} \|f - p\|_{\infty,[a,b]}.$$

Ferner sind die folgenden beiden Aussagen äquivalent:

(i) *Es gilt $Qf = \lim_{n\to\infty} Q_n f$ für alle $f \in \mathcal{C}[a, b]$.*

(ii) *Es gilt $Qp = \lim_{n\to\infty} Q_n p$ für alle Polynome $p \in \mathbb{P}[a, b]$ sowie ferner $\sup_{n\in\mathbb{N}} \sum_{j=0}^n |\omega_j^{(n)}| < \infty$.*

Beweis. Sei $p \in \mathbb{P}_m$, d.h. $Qp = Q_n p$. Dann folgt

$$\begin{aligned} |Qf - Q_n f| \leq |Qf - Qp| + |Q_n p - Q_n f| &= \Big| \int_a^b (f - p)\omega \, dx \Big| + \Big| \sum_{j=1}^n \omega_j^{(n)} \big(f(x_j^{(n)}) - p(x_j^{(n)}) \big) \Big| \\ &\leq \Big(\|\omega\|_{L^1(a,b)} + \sum_{j=0}^n |\omega_j^{(n)}| \Big) \, \|f - p\|_{\infty,[a,b]}. \end{aligned}$$

Wir betrachten den Banach-Raum $X = (\mathcal{C}[a, b], \|\cdot\|_\infty)$. Nach dem **Satz von Weierstraß** ist $D := \mathbb{P}[a, b]$ ein dichter Teilraum von X, d.h.

$$\forall \varepsilon > 0 \forall x \in X \exists d \in D \quad \|x - d\|_\infty \leq \varepsilon. \tag{6.8}$$

Mit Lemma 6.1 können wir die Aussagen abstrakt formulieren:

(i) $\lim_{n\in\mathbb{N}} Q_n x = Qx$ für alle $x \in X$.
(ii) $\lim_{n\in\mathbb{N}} Q_n d = Qd$ für alle $d \in D$ und $\sup_{n\in\mathbb{N}} \|Q_n\|_{X^*} < \infty$.

Nach dem **Satz von Banach-Steinhaus** gilt

$$C := \sup_{n\in\mathbb{N}} \|Q_n\|_{X^*} < \infty \quad \Longleftrightarrow \quad \forall x \in X \quad \sup_{n\in\mathbb{N}} |Q_n x| < \infty,$$

d.h. Beschränktheit bezüglich der Operatornorm (sog. *gleichmäßige Beschränktheit*) ist äquivalent zur punktweisen Beschränktheit. Daraus folgt unmittelbar die Implikation (i) $\Rightarrow$ (ii). Für die umgekehrte Implikation seien $x \in X$ und $\varepsilon > 0$. Nach (6.8) existiert ein $d \in D$ mit $\|x - d\|_\infty \le \varepsilon/M$ mit $M := 2(\|Q\|_{X^*} + C)$. Wegen (ii) existiert ein $n_0 \in \mathbb{N}$ mit $\|Q_n d - Qd\|_\infty \le \varepsilon/2$ für alle $n \ge n_0$. Es folgt

$$|Qx - Q_n x| \le |Qx - Qd| + |Qd - Q_n d| + |Q_n d - Q_n x| \le \|x - d\|_\infty (\|Q\|_{X^*} + C) + \varepsilon/2 \le \varepsilon.$$

Dies zeigt die behauptete Konvergenz. ■

Bemerkung. Die erste Bedingung in (ii)

$$Qp = \lim_{n\to\infty} Q_n p \quad \text{für alle } p \in \mathbb{P}(a,b)$$

ist für interpolatorische Quadraturformeln Q_n erfüllt, siehe Abschnitt 6.2. Die zweite Bedingung

$$\sup_{n\in\mathbb{N}} \sum_{j=0}^{n} |\omega_j^{(n)}| < \infty$$

ist schwieriger zu erfüllen, da auch negative Gewichte auftreten können. Für die Gaußschen Quadraturformel aus Abschnitt 6.3 sind aber die Gewichte positiv, und deshalb folgt

$$\sum_{j=0}^{n} |\omega_j^{(n)}| = \sum_{j=0}^{n} \omega_j^{(n)} = \|\omega\|_{L^1(a,b)}$$

nach Lemma 6.1, d.h. Gaußsche Quadraturformeln führen immer auf Konvergenz. □

6.2 Interpolatorische Quadraturformeln

Wir verwenden die Notation aus Abschnitt 6.1 mit $\mathbb{K} \in \{\mathbb{R}, \mathbb{C}\}$.

Definition. Eine Quadraturformel Q_n ist **interpolatorisch** (oder: **Interpolationsquadratur**) auf $[a,b]$, falls für jede stetige Funktion $f \in \mathcal{C}[a,b]$ gilt

$$Q_n f = Qp,$$

wobei $p \in \mathbb{P}_n$ das eindeutige Lagrange-Interpolationspolynom mit $p(x_j) = f(x_j)$ für alle $j = 0, \ldots, n$ ist. □

Im Folgenden bezeichnen wieder $L_j \in \mathbb{P}_n$ für $j = 0, \dots, n$ die Lagrange-Polynome

$$L_j(x) = \prod_{\substack{k=0 \\ j \neq k}}^{n} \frac{x - x_k}{x_j - x_k}.$$

Der folgende Satz zeigt insbesondere, dass zu gegebenen Stützstellen x_j eine *eindeutige* Interpolationsquadratur existiert. Wenn der Exaktheitsgrad von Q_n maximal werden soll, so muss Q_n zwangsläufig als Interpolationsquadratur gewählt werden.

Satz 6.3. *Für eine Quadraturformel $Q_n f = \sum_{k=0}^{n} \omega_k f(x_k)$ sind äquivalent:*
(i) *Q_n ist interpolatorisch.*
(ii) *Für die Gewichte ω_j gilt $\omega_j = \int_a^b L_j \omega \, dx$.*
(iii) *Q_n ist mindestens exakt vom Grad n.*

Beweis. (i) $\Rightarrow$ (ii): Nach Voraussetzung gilt für die Lagrange-Polynome gerade

$$\int_a^b L_j \omega \, dx = Q(L_j) = Q_n(L_j) = \sum_{k=0}^{n} \omega_k L_k(x_j) = \omega_j.$$

(ii) $\Rightarrow$ (i): Das Interpolationspolynom $p \in \mathbb{P}_n$ ist gegeben durch $p = \sum_{j=0}^{n} f(x_j) L_j$. Mit $\omega_j = \int_a^b L_j \omega \, dx$ gilt also $Q_n f = Qp$.
(ii) $\Rightarrow$ (iii): Nach Definition ist Q_n exakt für alle Lagrange-Polynome. Da die Lagrange-Polynome $L_0, \dots, L_n$ eine Basis von $\mathbb{P}_n$ bilden, ist Q_n aufgrund der Linearität schon exakt auf $\mathbb{P}_n$.
(iii) $\Rightarrow$ (ii): Da Q_n exakt vom Grad n ist, werden insbesondere die Lagrange-Polynome exakt integriert. ■

Beispiel (Abgeschlossene Newton-Côtes-Formeln). Die Interpolationsquadraturen Q_n auf einem kompakten Intervall $[a, b]$ zu den äquidistanten Knoten $x_j := a + (b - a)\, j/n$ und $j = 0, \dots, n$ nennt man abgeschlossene Newton-Côtes-Formeln. Einige dieser Formeln haben spezielle Namen:

- **Trapezregel** für $n = 1$.
- **Simpson-Regel** oder **Kepler'sche Fassregel** für $n = 2$.
- **Newton'sche 3/8-Regel** für $n = 3$.
- **Milne-Regel** für $n = 4$. □

Beispiel (Offene Newton-Côtes-Formeln). Bei den abgeschlossenen Newton-Côtes-Formeln sind die Intervallgrenzen $a, b \in \mathbb{R}$ Stützstellen. Dies verbietet die Anwendung für Integranden mit schwacher Singularität am Rand. Als offene Newton-Côtes-Formeln bezeichnet man die Interpolationsquadraturen Q_n auf $[a, b]$ zu den äquidistanten Knoten $x_j := a + (b - a)(j + 1)/(n + 2)$ für $j = 0, \dots, n$, d.h. a, b sind gerade keine Stützstellen der Quadraturformel. □

Beispiel (MacLaurin-Formeln). Die Interpolationsquadraturen Q_n zu den äquidistanten Knoten $x_j := a + (b - a)(j + 1/2)/(n + 1)$ bezeichnet man als MacLaurin-Formeln. □

Bemerkung. Die Gewichte ω_j einer Interpolationsquadratur Q_n können durch Lösen eines linearen Gleichungssystems berechnet werden. Es sei $p_0, \ldots, p_n$ eine Basis von $\mathbb{P}_n$. Es gilt

$$\int_a^b p_j \omega \, dx = \sum_{k=0}^{n} p_j(x_k)\omega_k \quad \text{für alle } j = 0, \ldots, n.$$

Diese Gleichung lässt sich als lineares Gleichungssystem formulieren:

$$\begin{pmatrix} p_0(x_0) & \cdots & p_0(x_n) \\ \vdots & & \vdots \\ p_n(x_0) & \cdots & p_n(x_n) \end{pmatrix} \begin{pmatrix} \omega_0 \\ \vdots \\ \omega_n \end{pmatrix} = \begin{pmatrix} \int_a^b p_0 \omega \, dx \\ \vdots \\ \int_a^b p_n \omega \, dx. \end{pmatrix} \tag{6.9}$$

Die Matrix auf der linken Seite von (6.9) ist eine transponierte Vandermonde-Matrix und deshalb regulär, vgl. Satz 4.2. Das Gleichungssystem hat als eindeutige Lösung den Vektor $(\omega_0, \ldots, \omega_n) \in \mathbb{R}^{n+1}$. □

MATLAB-Beispiel:

Wir betrachten das kompakte Intervall $[0, 1]$ mit der Monombasis und Gewichtsfunktion $\omega(x) = 1$. Das Gleichungssystem (6.9) lautet nun

$$\begin{pmatrix} 1 & \cdots & 1 \\ x_0^1 & \cdots & x_n^1 \\ \vdots & & \vdots \\ x_0^n & \cdots & x_n^n \end{pmatrix} \begin{pmatrix} \omega_0 \\ \vdots \\ \vdots \\ \omega_n \end{pmatrix} = \begin{pmatrix} 1 \\ 1/2 \\ \vdots \\ 1/(n+1) \end{pmatrix},$$

und für $x_j = j/n$ erhalten wir die Gewichte der zugehörigen abgeschlossenen Newton-Côtes-Gewichte. Die folgende MATLAB-Funktion berechnet die Stützstellen x_j und die zugehörigen Gewichte ω_j.

```
function [x,omega] = ClosedNewtonCotes(n)
x = [0:n] / n;
A = ( ones(n+1,1)*x ) .^ ( [0:n]'*ones(1,n+1) );
b = 1 ./ [1:n+1]';
omega = A\b;
```

Für $n \geq 8$ treten auch negative Gewichte auf. Der **Satz von Kusmin** besagt sogar, dass gilt

$$\lim_{n\to\infty} \sum_{j=0}^{n} |\omega_j^{(n)}| = \infty,$$

d.h. nach Satz 6.2 bekommen wir also im Allgemeinen *keine* Konvergenz der Quadratur $Q_n f$ gegen das Integral Qf.

Bemerkung. Wie bei der Polynominterpolation ist ein hoher Grad $n \in \mathbb{N}$ nur für glatte Funktionen sinnvoll. Im allgemeinen bewährt es sich aber eher, das Integrationsgebiet zu zerlegen,

$$\int_a^b f\,dx = \sum_{j=0}^{N} \int_{a_j}^{b_j} f\,dx \quad \text{mit } a = a_0 < b_0 = a_1 < \cdots < b_N = b,$$

und auf den Teilintervallen Quadraturformeln niederer Ordnung zu betrachten. Man erhält dann sogenannte **summierte Quadraturformeln**. Dies ist quasi das Analogon zur Interpolation mit Splines anstatt polynomialer Interpolation. □

Der folgende Satz gilt für alle bisherigen Beispiele. Wir sehen, dass man unter gewissen Voraussetzungen an die Stützstellen und die Gewichtsfunktion einen höheren Exaktheitsgrad erhalten kann.

Satz 6.4. *Es seien $a, b \in \mathbb{R}$, und es gelten die Symmetriebedingungen $x_j = a + b - x_{n-j}$ für $j = 0, \dots, n$ sowie $\omega(x) = \omega(a+b-x)$ für fast alle $x \in [a,b]$. Ferner sei $Q_n f := \sum_{j=0}^n \omega_j f(x_j)$ die induzierte Interpolationsquadratur. Dann gelten:*
(i) *Die Gewichte von Q_n sind ebenfalls symmetrisch, d.h. $\omega_j = \omega_{n-j}$ für $j = 0, \dots, n$.*
(ii) *Ist n gerade, so ist Q_n exakt auf $\mathbb{P}_{n+1}$.*

Beweis. (i) Wir betrachten die Quadraturformel $\widetilde{Q}_n f := \sum_{j=0}^n \omega_{n-j} f(x_j)$. Wir zeigen, dass $\widetilde{Q}_n$ exakt ist auf $\mathbb{P}_n$. Dann folgt mit der Eindeutigkeit der Interpolationsquadratur die Gleichheit $Q_n = \widetilde{Q}_n$ und Einsetzen der Lagrange-Polynome zeigt (i). Dazu definieren wir die Polynome

$$p_k(x) = \big(x - (a+b)/2\big)^k \quad \text{sowie} \quad \widetilde{p}_k(x) = p_k(a+b-x) \quad \text{für } k = 0, \dots, n. \tag{6.10}$$

Es gilt $\widetilde{p}_k(x) = \big((a+b)/2 - x\big)^k = (-1)^k p_k(x)$ und deshalb mit der Symmetrie der Knoten

$$\widetilde{Q}_n p_k = \sum_{j=0}^n \omega_{n-j} p_k(x_j) = \sum_{\ell=0}^n \omega_\ell p_k(x_{n-\ell}) = \sum_{\ell=0}^n \omega_\ell \widetilde{p}_k(x_\ell) = Q_n \widetilde{p}_k = Q\widetilde{p}_k.$$

Nun nutzen wir die Symmetrie der Gewichtsfunktion $\omega(x)$ und Substitution und erhalten

$$Q\widetilde{p}_k = \int_a^b p_k(a+b-x)\omega(x)\,dx = \int_a^b p_k(a+b-x)\omega(a+b-x)\,dx = \int_a^b p_k(y)\omega(y)\,dy = Qp_k.$$

Es folgt $\widetilde{Q}_n p_k = Q p_k$, und da $p_0, \dots, p_n$ eine Basis von $\mathbb{P}_n$ bilden, ist $\widetilde{Q}_n$ exakt auf $\mathbb{P}_n$. Wie bereits vorausgenommen, folgt die Behauptung.

(ii) Es sind lediglich die Gleichheiten $Q_n p_{n+1} = 0 = Q p_{n+1}$ zu zeigen. Da n gerade ist und aufgrund der Symmetrie der x_j, folgt $x_{n/2} = (a+b)/2$. Insbesondere gilt mit $p_{n+1}(x_{n/2}) = 0$

$$\begin{aligned} Q_n p_{n+1} &= \sum_{j=1}^{n/2} \omega_{n/2-j}\big(x_{n/2-j} - (a+b)/2\big)^{n+1} + \sum_{j=1}^{n/2} \omega_{n/2+j}\big(x_{n/2+j} - (a+b)/2\big)^{n+1} \\ &= \sum_{j=1}^{n/2} \omega_{n/2-j}\big(x_{n/2-j} - (a+b)/2\big)^{n+1} + \sum_{j=1}^{n/2} \omega_{n/2+j}\big((a+b)/2 - x_{n/2-j}\big)^{n+1} \end{aligned}$$

Mit $\omega_{n/2-j} = \omega_{n/2+j}$ und $\big(x_{n/2-j} - (a+b)/2\big) = (-1)\big((a+b)/2 - x_{n/2-j}\big)$ sieht man, dass die beiden Summen bis auf Vorzeichen übereinstimmen, denn $n+1$ ist ungerade. Wie behauptet folgt $Q_n p_{n+1} = 0$. Für das Integral gilt mit $p_{n+1} = -\widetilde{p}_{n+1}$ und Substitution

$$\begin{aligned} Qp_{n+1} = \int_a^b p_{n+1}(x)\omega(x)\,dx &= -\int_a^b \widetilde{p}_{n+1}(x)\omega(x)\,dx \\ &= -\int_a^b p_{n+1}(a+b-x)\omega(a+b-x)\,dx \\ &= -\int_a^b p_{n+1}(y)\omega(y)\,dy = -Qp_{n+1}, \end{aligned}$$

und deshalb folgt $Qp_{n+1} = 0$. ■

Aus den Fehlerabschätzungen für die Polynominterpolation lassen sich Fehlerabschätzungen für die Quadratur herleiten. Wir betrachten als einfache Beispiele die Trapezregel und die Simpson-Regel für $\omega(x) = 1$ auf einem Kompaktum $[a,b]$.

Satz 6.5. *Es seien $a, b \in \mathbb{R}$ und $Qf := \int_a^b f\,dx$. Wir betrachten die Trapezregel ($n = 1$) und die Simpson-Regel ($n = 2$) für die konstante Gewichtsfunktion $\omega(x) = 1$,*

$$Q_1 f = \frac{b-a}{2}\big[f(a) + f(b)\big] \qquad \textit{und} \qquad Q_2 f = \frac{b-a}{6}\Big[f(a) + 2\,f\Big(\frac{a+b}{2}\Big) + f(b)\Big]$$

(i) *Für eine Funktion $f \in \mathcal{C}^2[a,b]$ gilt*

$$\Big|Qf - Q_1 f\Big| \le \sqrt{2}\,\frac{(b-a)^3}{12}\,\|f''\|_\infty.$$

(ii) *Für eine Funktion $f \in \mathcal{C}^4[a,b]$ gilt*

$$\Big|Qf - Q_2 f\Big| \le \sqrt{2}\,\frac{(b-a)^5}{2880}\,\|f^{(4)}\|_\infty.$$

Beweis. Man beachte, dass die Quadraturen Q_1 und Q_2 abgeschlossene Newton-Côtes-Formeln sind, insbesondere also interpolatorisch. Es sei $p \in \mathbb{P}_1$ mit $f(a) = p(a)$ und $f(b) = p(b)$. Dann gilt $Q_1 f = Qp$ sowie die Interpolationsfehlerabschätzung

$$|f(x) - p(x)| \le \sqrt{2}\,\frac{\|f''\|_\infty}{2}\,|x-a||x-b|.$$

Damit erhalten wir

$$|Q_1 f - Qf| \le \int_a^b |p(x) - f(x)|\,dx \le \frac{\|f''\|_\infty}{2}\int_a^b (x-a)(b-x)\,dx = \frac{\|f''\|_\infty}{2}\,\frac{(b-a)^3}{6}.$$

Dies zeigt (i). Zum Beweis von (ii) sei $p \in \mathbb{P}_3$ das eindeutige Interpolationspolynom mit $f(a) = p(a)$, $f(\frac{a+b}{2}) = p(\frac{a+b}{2})$ und $f(b) = p(b)$ sowie $f'(\frac{a+b}{2}) = p'(\frac{a+b}{2})$. Trivialerweise gilt $Q_2 f = Q_2 p$. Da Q_2

aber auch exakt auf $\mathbb{P}_3$ ist (und nicht nur auf $\mathbb{P}_2$), folgt weiter $Q_2 p = Qp$. Insgesamt folgt

$$|Q_2 f - Qf| \leq \int_a^b |p(x) - f(x)|\,dx \leq \sqrt{2}\,\frac{\|f^{(4)}\|_\infty}{4!} \int_a^b |x-a|\Big|x - \frac{a+b}{2}\Big|^2 |x-b|\,dx$$
$$= \frac{(b-a)^5}{2880}\,\|f^{(4)}\|_\infty$$

aus der Abschätzung des Interpolationsfehlers für diese Hermite-Interpolationsaufgabe. ■

6.3 Gauß'sche Quadraturformeln

In diesem Abschnitt sollen die sogenannten Gauß'schen Quadraturformeln hergeleitet werden, die gerade maximalen Exaktheitsgrad $2n+1$ haben. Wir verwenden die Notation aus Abschnitt 6.1, wobei die Theorie sich zunächst auf $\mathbb{K} = \mathbb{R}$ beschränkt. Der Hauptsatz 6.8 gilt jedoch auch für $\mathbb{K} = \mathbb{C}$ und wird dann auch für $\mathbb{K} \in \{\mathbb{R}, \mathbb{C}\}$ bewiesen. Zur Herleitung der Theorie definieren wir (über $\mathbb{R}$) den Innenproduktraum

$$H := \big\{ f : (a,b) \to \mathbb{R} \text{ integrierbar} \,\big|\, \|f\| < \infty \big\}, \tag{6.11}$$

mit der Norm $\|f\| := \langle f\,;\,f\rangle^{1/2}$, die durch das Skalarprodukt

$$\langle f\,;\,g\rangle := \int_a^b f g \omega\,dx \quad \text{für } f, g \in H \tag{6.12}$$

definiert wird. Aufgrund der Zusatzvoraussetzung (6.1) gilt $\mathbb{P}_n(a,b) \subset H$ für alle $n \in \mathbb{N}_0$. Wir verwenden nun Gram-Schmidt-Orthogonalisierung, um die sogenannten Orthogonalpolynome zu erhalten.

Das Gram-Schmidt-Verfahren zur Orthogonalisierung lässt sich leicht durch vollständige Induktion verifizieren. Vermutlich wurde der Beweis ohnehin in der Vorlesung zur Linearen Algebra erbracht.

Lemma 6.6 (Gram-Schmidt-Orthogonalisierung). *Es sei H ein Innenproduktraum und $x_0, \ldots, x_n \in H$ seien linear unabhängig. Definiert man dann induktiv*

$$p_0 := x_0, \quad p_k := x_k - \sum_{j=0}^{k-1} \frac{\langle x_k\,;\,p_j\rangle}{\|p_j\|^2}\,p_j \quad \textit{für } k = 1, \ldots, n.$$

Dann sind die Vektoren $p_0, \ldots, p_n \in H\backslash\{0\}$ orthogonal, d.h. es gilt $\langle p_j\,;\,p_k\rangle = 0$ für $j \neq k$ und insbesondere linear unabhängig, und $\operatorname{span}\{p_0, \ldots, p_n\} = \operatorname{span}\{x_0, \ldots, x_n\}$. ■

Definition. Es sei $(p_j)_{j\in\mathbb{N}_0}$ die Folge der Polynome $p_j \in \mathbb{P}_j$, die durch Gram-Schmidt-Orthogonalisierung der Monome $(x^j)_{j\in\mathbb{N}_0}$ im (reellen) Innenproduktraum (6.11) gewonnen wird. Man bezeichnet p_j als j-tes **Orthogonalpolynom**. □

Man beachte, dass alle Orthogonalpolynome (wie die Monome) nach Definition den Leitkoeffizient 1 haben. Insbesondere gilt stets $p_0(x) = 1$.

Beispiel (Orthogonalpolynome). Modulo multiplikativer Konstanten gelten die folgenden Aussagen: [1]

- $[a,b] = [-1,1]$, $\omega = 1$ führt auf die **Legendre-Polynome**

$$p_n(x) = \frac{n!}{(2n)!}\,\frac{d^n}{dx^n}\big((x^2-1)^n\big). \tag{6.13}$$

- $[a,b] = [-1,1]$, $\omega(x) = (1-x^2)^{-1/2}$ führt auf die **Čebyšev-Polynome**

$$p_n(x) = \frac{1}{2^{n-1}}\,\cos(n\,\arccos x). \tag{6.14}$$

- $[a,b] = [0,\infty)$, $\omega(x) = \exp(-x)$ führt auf die **Laguerre-Polynome**

$$p_n(x) = (-1)^n \exp(x)\,\frac{d^n}{dx^n}\big(x^n \exp(-x)\big). \tag{6.15}$$

- $[a,b] = (-\infty,\infty)$, $\omega(x) = \exp(-x^2)$ führt auf die **Hermite-Polynome**. ■

Für die Gauß-Quadratur der Länge n nehmen wir als Stützstellen die $n+1$ Nullstellen des $(n+1)$-ten Orthogonalpolynoms $p_{n+1} \in \mathbb{P}_{n+1}$. Dazu müssen wir zunächst zeigen, dass alle Nullstellen einfach sind und im offenen Intervall (a,b) liegen.

Lemma 6.7. *Es seien $x_0,\dots,x_n \in \mathbb{C}$ die gemäß Vielfachheit gezählten Nullstellen des Orthogonalpolynoms $p_{n+1} \in \mathbb{P}_{n+1}$. Dann gelten die folgenden Aussagen:*
(i) *Alle Nullstellen sind einfach, d.h. $x_0,\dots,x_n$ sind paarweise verschieden.*
(ii) *$x_j \in (a,b)$, d.h. die Nullstellen sind reell und liegen im Inneren des Integrationsintervalls.*
(iii) *Mit dem zu x_j gehörigen Lagrange-Polynom gilt $x_j = \langle xL_j\,;\,L_j\rangle/\|L_j\|^2$.*

Beweis. Es seien $x_0,\dots,x_k$ die paarweise verschiedenen *reellen* Nullstellen *ungerader* Vielfachheit von p_{n+1} und $q(x) := \prod_{j=0}^{k}(x-x_j)$ bzw. $q(x) := 1$, falls keine solchen Nullstellen existieren. Insbesondere hat das Polynom $p := p_{n+1}q$ keinen Vorzeichenwechsel in (a,b). Wir nehmen an, es gilt $k < n$ und führen dies zum Widerspruch: Wegen $k < n$ gilt $q \in \mathbb{P}_n$ und deshalb $0 = \langle p_{n+1}\,;\,q\rangle = \int_a^b p_{n+1}q\omega\,dx$. Da $\omega p_{n+1}q = \omega p$ keinen Vorzeichenwechsel in (a,b) hat, folgt $\omega p = 0$ fast überall in (a,b). Da p nicht-trivial ist, müsste also $\omega = 0$ fast überall gelten. Dieser Widerspruch zeigt $k = n$, und es folgen (i) und (ii). Um (iii) zu zeigen, verwenden wir Polynomdivision und erhalten ein Polynom $q \in \mathbb{P}_n\backslash\{0\}$ mit $p_{n+1} = (x-x_j)q$. Es folgt

$$0 = \langle p_{n+1}\,;\,q\rangle = \langle xq\,;\,q\rangle - x_j\langle q\,;\,q\rangle$$

und deshalb $x_j = \langle xq\,;\,q\rangle/\|q\|^2$. Die Polynome q und L_j haben dieselben Nullstellen. Deshalb existiert eine Konstante $c \neq 0$ mit $q = cL_j$. Einsetzen zeigt die Behauptung. ■

Im folgenden Hauptsatz über die Gauß-Quadratur lassen wir wieder den Fall $\mathbb{K} \in \{\mathbb{R},\mathbb{C}\}$ zu. Die eindeutige Gauß-Quadraturformel Q_n hat maximalen Exaktheitsgrad $2n+1$ und ist daher

[1] Vielleicht sollte man auch gleich die 3-Term-Rekursionen dazuschreiben, damit man die Quadraturgewichte und -stützstellen später auch einfach berechnen kann. Für die Čebyšev-Polynome steht das übrigens schon in Kapitel 4.

insbesondere interpolatorisch. Dies liefert im Beweis die Eindeutigkeit. Q_n ist ferner numerisch günstig, da alle Gewichte ω_j positiv sind. Wegen $\sum_{j=0}^n |\omega_j| = \sum_{j=0}^n \omega_j = \|\omega\|_{L^1(a,b)}$ ist (zumindest für $a, b \in \mathbb{R}$) die erste Voraussetzung im Konvergenzsatz 6.2 erfüllt. Die zweite Voraussetzung ist klar, da Q_n interpolatorisch ist, d.h. es gilt Konvergenz $Q_n f \to Qf$ für alle $f \in C[a,b]$, $a, b \in \mathbb{R}$.

Satz 6.8 (Existenz und Eindeutigkeit der Gauß-Quadratur). (i) *Es existiert eine eindeutige Quadraturformel Q_n der Länge n auf $\overline{(a,b)}$ mit maximalem Exaktheitsgrad $2n+1$.*
(ii) *Die Knoten x_j von Q_n sind die $n+1$ Nullstellen des Orthogonalpolynoms $p_{n+1} \in \mathbb{P}_{n+1}$.*
(iii) *Die Gewichte ω_j von Q_n erfüllen $\omega_j = \int_a^b L_j \omega \, dx = \int_a^b L_j^2 \omega \, dx > 0$ mit den Lagrange-Polynomen L_j.*
(iv) *Für $a, b \in \mathbb{R}$ und $f \in \mathcal{C}^{2n+2}[a,b]$ gilt*

$$|Qf - Q_n f| \le \sqrt{2} \, \frac{\|f^{(2n+2)}\|_\infty}{(2n+2)!} \int_a^b \omega(x) \prod_{j=0}^n (x - x_j)^2 \, dx \tag{6.16}$$

Beweis der Existenz. Wir definieren $Q_n f := \sum_{j=0}^n \omega_j f(x_j)$ mit x_j den Nullstellen von p_{n+1} und $\omega_j := \int_a^b L_j \omega \, dx$, wobei die Interpolationsgewichte durch die Stützstellen und den gewünschten Exaktheitsgrad $2n+1 \ge n$ *eindeutig* festgelegt sind. Nun ist für $q \in \mathbb{P}_{2n+1}$ noch $Q_n q = Qq$ zu zeigen. Nach Polynomdivision existieren Polynome $\alpha, \beta \in \mathbb{P}_n$ mit $q = \alpha p_{n+1} + \beta$. Es gilt

$$Qq = \langle q \,;\, 1 \rangle = \underbrace{\langle \alpha \,;\, p_{n+1} \rangle}_{=0} + \langle \beta \,;\, 1 \rangle = Q\beta.$$

Ferner gilt $Q_n q = Q_n \beta$, denn $Q_n(\alpha p_{n+1}) = 0$ aufgrund von $p_{n+1}(x_j) = 0$. Nach Definition ist Q_n interpolatorisch und damit exakt auf $\mathbb{P}_n$, also folgt $Q_n q = Q_n \beta = Q\beta = Qq$. ■

Beweis der Eindeutigkeit. Es sei $\widetilde{Q}_n = \sum_{j=0}^n \widetilde{\omega}_j f(\widetilde{x}_j)$ eine weitere Quadraturformel mit Exaktheitsgrad $2n+1$. Wir zeigen nun $\widetilde{x}_j \in \{x_0, \dots, x_n\}$ für alle $j = 0, \dots, n$. Dann folgt $\{\widetilde{x}_0, \dots, \widetilde{x}_n\} = \{x_0, \dots, x_n\}$ und damit $\widetilde{Q}_n = Q_n$, denn die Gewichte sind durch die Stützstellen eindeutig festgelegt. Um zu zeigen, dass $\widetilde{x}_j$ auch Stützstelle von Q_n ist, definieren wir das Polynom

$$q(x) := \Big\{ \prod_{k=0}^n (x - x_k) \Big\} \Big\{ \prod_{\substack{k=0 \\ k \ne j}}^n (x - \widetilde{x}_k) \Big\} \in \mathbb{P}_{2n+1}.$$

Dann gilt

$$0 = Q_n q = Qq = \widetilde{Q}_n q = \widetilde{\omega}_j q(\widetilde{x}_j) = \underbrace{\widetilde{\omega}_j}_{\ne 0} \Big\{ \prod_{k=0}^n (\widetilde{x}_j - x_k) \Big\} \underbrace{\Big\{ \prod_{\substack{k=0 \\ k \ne j}}^n (\widetilde{x}_j - \widetilde{x}_k) \Big\}}_{\ne 0},$$

und daher existiert ein Index $k = 0, \dots, n$ mit $x_k = \widetilde{x}_j$, was den Beweis der Eindeutigkeit beschließt. ■

Beweis der zusätzlichen Eigenschaften. (i) und (ii) sind bereits bewiesen, und es gilt $\omega_j = \int_a^b L_j \omega \, dx = Q(L_j) = Q_n(L_j)$. Wegen $L_j(x_k) = \delta_{jk} = L_j^2(x_k)$ folgt $Q_n(L_j) = Q_n(L_j^2)$, und

aufgrund von $L_j^2 \in \mathbb{P}_{2n}$ erhalten wir also $\omega_j = Q_n(L_j) = Q_n(L_j^2) = Q(L_j^2) = \int_a^b L_j^2 \omega \, dx > 0$. Dies zeigt (iii). Um (iv) zu zeigen, sei $q \in \mathbb{P}_{2n+1}$ das eindeutige Hermite-Interpolationspolynom mit $q(x_j) = f(x_j)$ und $q'(x_j) = f'(x_j)$ für $0 \le j \le n$. Nach Fehlerabschätzung für das Hermite-Interpolationsproblem gilt

$$|f(x) - q(x)| \le \sqrt{2} \, \frac{\|f^{(2n+2)}\|_\infty}{(2n+2)!} \prod_{j=0}^{n} (x - x_j)^2 \quad \text{für } x \in [a, b].$$

Mit $Q_n f = Q_n q = Q q$ erhalten wir also

$$|Qf - Q_n f| = \Big| \int_a^b \{f(x) - q(x)\} \omega(x) \, dx \Big| \le \sqrt{2} \, \frac{\|f^{(2n+2)}\|_\infty}{(2n+2)!} \int_a^b \omega(x) \prod_{j=0}^{n} (x - x_j)^2 \, dx$$

aus der Monotonie des Integrals. ■

Berechnung von Gauß'schen Quadraturformeln

Zum Abschluss dieses Abschnitts wollen wir noch kurz zeigen, dass die numerische Berechnung von Gauß'schen Quadraturformeln auf ein Eigenwertproblem mit symmetrischer Tridiagonalmatrix $A \in \mathbb{R}_{\text{sym}}^{(n+1)\times(n+1)}$ führt. Der Beweis basiert im Wesentlichen auf der folgenden 3-Term-Rekursion zur Berechnung der Orthogonalpolynome.

Lemma 6.9 (3-Term-Rekursion für Orthogonalpolynome). *Das Orthogonalpolynom p_j ist eindeutig durch die folgende 3-Term-Rekursion*

$$p_0(x) := 1, \quad p_1(x) := x - \beta_0, \quad p_{n+1}(x) := (x - \beta_n)p_n(x) - \gamma_n^2 p_{n-1}(x) \quad \textit{für } n \geq 1 \tag{6.17}$$

mit (reellen) Koeffizienten $\beta_n := \langle xp_n \,;\, p_n\rangle / \|p_n\|^2$, $\gamma_n := \|p_n\|/\|p_{n-1}\|$ gegeben.

Beweis. Der Beweis wird durch Induktion nach n geführt. Der Induktionsanfang $n = 0, 1$ ist klar. Im Induktionsschritt definieren wir für $n \geq 1$ das Polynom

$$q_{n+1}(x) := (x - \beta_n)p_n(x) - \gamma_n^2 p_{n-1}(x)$$

und zeigen, dass q_{n+1} mit dem Orthogonalpolynom p_{n+1} aus dem Gram-Schmidt-Verfahren übereinstimmt. Offensichtlich haben $q_{n+1}, p_{n+1} \in \mathbb{P}_{n+1}$ den Leitkoeffizienten 1, also ist $r := p_{n+1} - q_{n+1} \in \mathbb{P}_n$. Damit ist nur noch zu zeigen, dass $r \in \mathbb{P}_n^\perp$ ist (mit Orthogonalität bezüglich dem Skalarprodukt in H), und es folgt daraus $r = 0$. Nach Definition des Gram-Schmidt-Verfahrens gilt $p_{n+1} \in \mathbb{P}_n^\perp$, also ist nur noch $q_{n+1} \in \mathbb{P}_n^\perp$ zu zeigen. Dazu müssen wir nur nachweisen, dass q_{n+1} orthogonal ist zu $p_0, \ldots, p_n$, denn diese bilden eine Basis von $\mathbb{P}_n$:

- Für $k = 0, \ldots, n-2$ gilt

$$\langle q_{n+1} \,;\, p_k\rangle = \underbrace{\langle p_n \,;\, xp_k\rangle}_{=0} - \beta_n \underbrace{\langle p_n \,;\, p_k\rangle}_{=0} - \gamma_n^2 \underbrace{\langle p_{n-1} \,;\, p_k\rangle}_{=0} = 0.$$

- Für $k = n-1$ gilt

$$\langle q_{n+1} \,;\, p_{n-1}\rangle = \langle p_n \,;\, xp_{n-1}\rangle - \beta_n \underbrace{\langle p_n \,;\, p_{n-1}\rangle}_{=0} - \underbrace{\gamma_n^2 \langle p_{n-1} \,;\, p_{n-1}\rangle}_{=\langle p_n \,;\, p_n\rangle} = \langle p_n \,;\, \underbrace{xp_{n-1} - p_n}_{\in\mathbb{P}_{n-1}}\rangle = 0.$$

- Für $k = n$ gilt

$$\langle q_{n+1} \,;\, p_n\rangle = \underbrace{\langle xp_n \,;\, p_n\rangle - \beta_n \langle p_n \,;\, p_n\rangle}_{=0} - \gamma_n^2 \underbrace{\langle p_{n-1} \,;\, p_n\rangle}_{=0} = 0.$$

Insgesamt folgt also $q_{n+1} \in \mathbb{P}_n^\perp$, also $r = p_{n+1} - q_{n+1} \in \mathbb{P}_n^\perp \cap \mathbb{P}_n$ und deshalb $p_{n+1} = q_{n+1}$. ■

Satz 6.10. *Mit den Konstanten der 3-Term-Rekursion* (6.17) *sind die Eigenwerte der Matrix*

$$A = \begin{pmatrix} \beta_0 & -\gamma_1 & 0 & \dots & 0 \\ -\gamma_1 & \beta_1 & -\gamma_2 & \ddots & \vdots \\ 0 & -\gamma_2 & \ddots & \ddots & 0 \\ \vdots & \ddots & \ddots & \ddots & -\gamma_n \\ 0 & \dots & 0 & -\gamma_n & \beta_n \end{pmatrix} \in \mathbb{R}^{(n+1)\times(n+1)}_{\text{sym}} \tag{6.18}$$

genau die Nullstellen $x_0, \dots, x_n$ des $(n+1)$-ten Orthogonalpolynoms p_{n+1}. Die zugehörigen Gauß-Gewichte werden gegeben durch

$$\omega_k = \Big(\int_a^b \omega \, dx \Big) \Big(\sum_{j=0}^{n} \tau_j^2 p_j^2(x_k) \Big)^{-1} \quad \textit{für } k{=}0,\dots,n \tag{6.19}$$

mit $\tau_0 := 1$ und $\tau_j = (-1)^j/(\gamma_1 \cdots \gamma_j)$ für $j = 1, \dots, n$.

Beweis. Aus notationstechnischen Gründen werden Vektoren und Matrizen im Beweis mit $j, k = 0, \dots, n$ indiziert, und wir definieren $\tau_{n+1} := 0$.

1. Schritt. *Mit $v^{(k)} := \big(\tau_j p_j(x_k)\big)_{j=0}^n \in \mathbb{R}^{n+1}$ gilt $Av^{(k)} = x_k v^{(k)}$, d.h. $v^{(k)}$ ist Eigenvektor zum Eigenwert x_k, denn $v_0^{(k)} = \tau_0 p_0(x_k) = 1$, d.h. $v^{(k)} \neq 0$.* Mit (6.17) für $p_1(x)$ gilt

$$(Av^{(k)})_0 = \beta_0 \underbrace{v_0^{(k)}}_{=1} - \gamma_1 v_1^{(k)} = \beta_0 - \underbrace{\gamma_1 \tau_1}_{=-1} \underbrace{p_1(x_k)}_{=x_k-\beta_0} = x_k = x_k v_0^{(k)}.$$

Für $j = 1, \dots, n-1$ folgt mit (6.17) für $p_{j+1}(x)$

$$\begin{aligned} (Av^{(k)})_j &= -\gamma_j v_{j-1}^{(k)} + \beta_j v_j^{(k)} - \gamma_{j+1} v_{j+1}^{(k)} \\ &= -\underbrace{\gamma_j \tau_{j-1}}_{=-\gamma_j^2 \tau_j} p_{j-1}(x_k) + \beta_j \tau_j p_j(x_k) - \underbrace{\gamma_{j+1}\tau_{j+1}}_{=-\tau_j} p_{j+1}(x_k) \\ &= \tau_j \big\{ \underbrace{\gamma_j^2 p_{j-1}(x_k) + \beta_j p_j(x_k) + p_{j+1}(x_k)}_{=x_k p_j(x_k)} \big\} \\ &= \tau_j x_k p_j(x_k) = x_k v_j^{(k)}. \end{aligned}$$

Dasselbe Argument mit $\tau_{n+1} = 0$ zeigt $(Av^{(k)})_n = x_k v_n^{(k)}$, d.h. $Av^{(k)} = x_k v^{(k)}$. □

2. Schritt. *Es gilt $v^{(j)} \cdot v^{(k)} = 0$ für $j \neq k$,* denn Eigenvektoren symmetrischer Matrizen zu verschiedenen Eigenwerten sind orthogonal: Aus $x_j \neq x_k$ folgt wegen

$$x_j v^{(j)} \cdot v^{(k)} = Av^{(j)} \cdot v^{(k)} = v^{(j)} \cdot Av^{(k)} = x_k v^{(j)} \cdot v^{(k)}$$

umgehend $v^{(j)} \cdot v^{(k)} = 0$. □

3. Schritt. *Es gilt $\omega_k = \big(\int_a^b \omega \, dx \big) / \|v^{(k)}\|_2^2$.* Für $j = 0, \dots, n$ gilt mit der Gauß-Quadratur Q_n

$$\delta_{j0} \int_a^b \omega \, dx = \langle p_j \, ; \, p_0 \rangle = \int_a^b p_j \omega \, dx = Qp_j = Q_n p_j = \sum_{\ell=0}^{n} \omega_\ell p_j(x_\ell).$$

Wegen $\tau_0^2 p_0(x_k) = 1$ für alle $k = 0, \ldots, n$ ergibt sich für fixiertes k

$$\int_a^b \omega \, dx = \sum_{j=0}^{n} \tau_j^2 p_j(x_k) \sum_{\ell=0}^{n} \omega_\ell p_j(x_\ell) = \sum_{\ell=0}^{n} \omega_\ell \sum_{j=0}^{n} \tau_j^2 p_j(x_k) p_j(x_\ell) = \sum_{\ell=0}^{n} \omega_\ell \, v^{(k)} \cdot v^{(\ell)}$$
$$= \omega_k \| v^{(k)} \|_2^2$$

nach Schritt 2. ■

Für großes n erfolgt die Berechnung der x_k numerisch. Verfahren werden in der Vorlesung *Numerik von Differentialgleichungen* vorgestellt bzw. finden sich in der einführenden Literatur zur Numerischen Mathematik, z.b. PLATO [5, Kapitel 12,13].

Kapitel 7

Iterative Lösung von linearen und nichtlinearen Gleichungssystemen

7.1 Fixpunktprobleme

Im folgenden Abschnitt bezeichnet das Tripel (X, Φ, x^*) ein **Iterationsverfahren**: X ist ein metrischer Raum, $\Phi : X \to X$ ist die **Iterationsvorschrift** und $x^* \in X$ ist die gesuchte **Lösung**, d.h. ein Fixpunkt $x^* = \Phi(x^*)$ von Φ. Zu einem **Startwert** $x_0 \in X$ definieren wir die **Iteriertenfolge** $(x_k)_{k\in\mathbb{N}}$ induktiv durch $x_k := \Phi(x_{k-1})$.

Bemerkung. Ist $x = \lim\limits_{k\to\infty} x_k$ der Limes der Iteriertenfolge und Φ stetig bei x, so gilt $\Phi(x) = \lim\limits_{k\to\infty} \Phi(x_k) = \lim\limits_{k\to\infty} x_{k+1} = x$, d.h. x ist Fixpunkt von Φ. □

Ist X ein normierter Raum und $F : X \to X$, so wird die Lösung eines Gleichungssystems $F(x^*) = 0$ [gegeben als Nullstellenproblem] als Fixpunktproblem umformuliert: Mit $\Phi(x) := x - F(x)$ gilt offensichtlich genau dann $\Phi(x) = x$, wenn auch $F(x) = 0$ gilt.

Beispiel. Die nicht-lineare Gleichung $x^2 + \exp(x) = 2$ hat eine eindeutige Lösung $x^* > 0$. Es gilt $x^* \approx 0.53627$. Um x^* iterativ zu berechnen, kann das Problem auf verschiedene Weisen in Fixpunktform geschrieben werden, z.B.

$$\Phi_1(x) = x \pm \big(x^2 + \exp(x) - 2\big), \quad \Phi_2(x) = \sqrt{2 - \exp(x)}, \quad \Phi_3(x) = \log(2 - x^2).$$

Leider erweisen sich die Iterationsverfahren zu Φ_1 und Φ_2 als *nicht* konvergent. Lediglich die Iterationsvorschrift Φ_3 führt auf eine konvergente Iteriertenfolge. Dies hängt, wie wir mit Satz 7.3 sehen werden, mit den Ableitungen $\Phi_\ell^{(k)}(x^*)$ zusammen. Es gilt

$$\Phi_1'(x) = 1 \pm \big(2x + \exp(x)\big), \quad \Phi_2'(x) = -\frac{\exp(x)}{2\sqrt{2 - \exp(x)}}, \quad \Phi_3'(x) = -\frac{2x}{2 - x^2}$$

und deshalb $\Phi_1'(x^*) \approx 1 \pm 2.79$, $\Phi_2'(x^*) \approx 1.59$ sowie $\Phi_3'(x^*) \approx 0.63$. ■

Beispiel (Richardson-Iteration). Ist $A \in \mathbb{K}^{n\times n}$ eine reguläre Matrix und $b \in \mathbb{K}^n$, so gilt

$$Ax^* = b \quad \Longleftrightarrow \quad 0 = F(x^*) := Ax^* - b \quad \Longleftrightarrow \quad x^* = \Phi(x^*) := (\boldsymbol{I} - A)x^* + b.$$

Wir werden später mit Hilfe des Banach'schen Fixpunktsatzes beweisen, dass die Richardson-Iteration genau dann gegen den eindeutigen Fixpunkt x^* von Φ konvergiert, wenn der Spektralradius $\rho(\boldsymbol{I} - A) < 1$ erfüllt, vgl. Satz 7.9. ■

Satz 7.1 (Banach'scher Fixpunktsatz). *Es sei X ein vollständiger metrischer Raum und $\Phi : X \to X$ eine Kontraktion, d.h. es existiert eine Lipschitz-Konstante $L < 1$ mit*

$$d(\Phi(x), \Phi(y)) \leq L\, d(x, y) \quad \textit{für alle } x, y \in X. \tag{7.1}$$

Dann existiert ein eindeutiges $x^* \in X$ mit $\Phi(x^*) = x^*$, und unabhängig vom Startwert $x_0 \in X$ konvergiert die Iteriertenfolge $x_k := \Phi(x_{k-1})$ gegen x^*. Es gelten also insbesondere

$$d(x_k, x^*) \leq L\, d(x_{k-1}, x^*) \tag{7.2}$$

sowie die beiden Fehlerabschätzungen

$$d(x_k, x^*) \leq \frac{L}{1-L}\, d(x_k, x_{k-1}) \leq \frac{L^k}{1-L}\, d(x_1, x_0). \tag{7.3}$$

Beweis. Φ hat höchstens einen Fixpunkt, denn aus der Kontraktivität

$$d(x^*, y^*) = d(\Phi(x^*), \Phi(y^*)) \leq L\, d(x^*, y^*)$$

folgt notwendig $x^* = y^*$. Wir haben bereits gesehen, dass die Iteriertenfolge gegebenenfalls gegen einen Fixpunkt von Φ konvergiert. Da X vollständig ist, ist also nur noch zu zeigen, dass die Iteriertenfolge für jeden Startwert $x_0 \in X$ eine Cauchy-Folge ist. Induktiv zeigt man zunächst $d(x_k, x_{k+1}) \leq L^k d(x_0, x_1)$. Dann folgt für Indizes $m \leq n$ mit Hilfe der geometrischen Reihe

$$d(x_m, x_n) \leq \sum_{k=m}^{n-1} d(x_k, x_{k+1}) \leq d(x_0, x_1) \sum_{k=m}^{n-1} L^k = d(x_0, x_1) L^m \frac{1 - L^{n-m}}{1-L} \leq d(x_0, x_1) \frac{L^m}{1-L},$$

und die rechte Seite verschwindet für $m, n \to \infty$. Dies zeigt die eindeutige Existenz von x^*. Um die Fehlerabschätzung (7.3) zu erhalten, nutzen wir

$$d(x_k, x^*) = d(\Phi(x_{k-1}), \Phi(x^*)) \leq L\, d(x_{k-1}, x^*) \leq L\, d(x_{k-1}, x_k) + L\, d(x_k, x^*)$$

und erhalten die erste Abschätzung in (7.3) durch Umordnen. Die zweite Abschätzung folgt aus $d(x_{k-1}, x_k) \leq L^{k-1} d(x_0, x_1)$. ■

Bemerkung. (i) Die zweite Abschätzung $d(x_k, x^*) \leq \frac{L^k}{1-L}\, d(x_1, x_0)$ in (7.3) bezeichnet man als **a priori Fehlerabschätzung**, da man ohne Kenntnis der exakten Lösung x^* zu gegebener Toleranz $\varepsilon > 0$ eine Höchstanzahl $k \in \mathbb{N}$ a priori angeben kann, sodass $d(x_k, x^*) \leq \varepsilon$ gilt.
(ii) Die erste Abschätzung $d(x_k, x^*) \leq \frac{L}{1-L}\, d(x_k, x_{k-1})$ in (7.3) bezeichnet man als **a posteriori Fehlerabschätzung**, da man zu gegebenem $\varepsilon > 0$ ohne Kenntnis von x^* nach Berechnung von x_k weiß, ob $d(x_k, x^*) \leq \varepsilon$ gilt. In der Regel sind a posteriori Abschätzungen schärfer (und realistischer) als a priori Abschätzungen, da bei der Fehlerschätzung bereits mehr Information zur Verfügung steht.

(iii) In der Praxis interessiert man sich für a priori Fehlerabschätzungen, um mathematisch die Konvergenz eines Verfahrens garantieren zu können. A posteriori Fehlerabschätzungen sind notwendig, um effiziente Abbruchbedingungen (und effiziente Algorithmen) zu entwickeln. □

Definition. Ein Iterationsverfahren (X, Φ, x^*) heißt
(i) **global konvergent**, falls gilt

$$\forall x_0 \in X \quad \lim_{k\to\infty} x_k = x^*,$$

(ii) **lokal konvergent**, falls gilt

$$\exists \varepsilon > 0 \forall x_0 \in U_\varepsilon(x^*) \quad \lim_{k\to\infty} x_k = x^*,$$

(iii) **linear konvergent** bzw. **von Konvergenzordnung $p = 1$**, falls gilt

$$\exists c \in [0,1) \exists \varepsilon > 0 \forall x_0 \in U_\varepsilon(x^*) \forall k \in \mathbb{N} \quad d(x_k, x^*) \le c\, d(x_{k-1}, x^*),$$

(iv) **von Konvergenzordnung $p > 1$**, falls gilt

$$\exists c > 0 \exists \varepsilon > 0 \forall x_0 \in U_\varepsilon(x^*) \forall k \in \mathbb{N} \quad d(x_k, x^*) \le c\, d(x_{k-1}, x^*)^p.$$

Iterationsverfahren von Konvergenzordnung $p = 2$ nennt man **quadratisch konvergent**. Die offene Umgebung $U_\varepsilon(x^*) = \{x \in X \,|\, d(x, x^*) < \varepsilon\}$ bezeichnet man als **Konvergenzbereich** des Iterationsverfahrens. □

Mit der neu eingeführten Notation, können wir den Banach'schen Fixpunktsatz neu formulieren.

Beispiel (Banach'scher Fixpunktsatz). Eine Kontraktion $\Phi : X \to X$ auf einem vollständigen metrischen Raum hat einen eindeutigen Fixpunkt $x^* \in X$. Das Iterationsverfahren (X, Φ, x^*) konvergiert global linear.

Lemma 7.2. *Es sei (X, Φ, x^*) ein Iterationsverfahren der Konvergenzordnung $p \ge 1$. Dann ist (X, Φ, x^*) lokal konvergent und auch von jeder Konvergenzordnung $q \in [1, p]$.*

Beweis. Ist (X, Φ, x^*) linear konvergent, so gilt nach Induktion

$$d(x_k, x^*) \le c^k\, d(x_0, x^*) \quad \text{für alle } k \in \mathbb{N},$$

und wegen $c \in [0, 1)$ verschwindet die rechte Seite für $k \to \infty$. Jedes linear konvergente Verfahren ist also auch lokal konvergent. Nun sei (X, Φ, x^*) von der Konvergenzordnung $p > 1$. Wähle $\varepsilon > 0$ und $c > 0$ gemäß Definition und setze

$$\delta := \min\{\varepsilon, (\tfrac{1}{2c})^{1/(p-1)}\}.$$

Es sei $x_0 \in U_\delta(x^*)$. Wir zeigen induktiv, dass dann gilt

$$d(x_k, x^*) \le 2^{-k}\, d(x_0, x^*) < \delta \quad \text{für alle } k \in \mathbb{N}_0. \tag{7.4}$$

Der Induktionsanfang $k = 0$ ist klar. Im Induktionsschritt gilt also

$$\begin{aligned} d(x_{k+1}, x^*) \le c\, d(x_k, x^*)^p \le c\, 2^{-kp}\, d(x_0, x^*)^{p-1} d(x_0, x^*) &\le c\, 2^{-kp}\, \delta^{p-1}\, d(x_0, x^*) \\ &\le 2^{-(kp+1)}\, d(x_0, x^*) \\ &\le 2^{-(k+1)}\, d(x_0, x^*) < \delta. \end{aligned}$$

Damit ist (7.4) bewiesen, und es folgt insbesondere die lineare Konvergenz, denn

$$d(x_k, x^*) \le c\, d(x_{k-1}, x^*)^{p-1} d(x_{k-1}, x^*) \le c\, \delta^{p-1} d(x_{k-1}, x^*) \le \frac{1}{2}\, d(x_{k-1}, x^*).$$

Dasselbe Vorgehen zeigt für $q \in (1, p)$,

$$d(x_k, x^*) \le c\, d(x_{k-1}, x^*)^{p-q} d(x_{k-1}, x^*)^q \le c\delta^{p-q}\, d(x_{k-1}, x^*)^q$$

und damit Konvergenz der Ordnung q. ■

Globale Konvergenz tritt in der Praxis bei nichtlinearen Iterationsverfahren nicht auf. Da x^* in der Regel unbekannt ist, interessiert man sich für Verfahren, bei denen der Konvergenzbereich (d.h. $\varepsilon > 0$) möglichst groß ist. Die Konvergenzordnung eines Iterationsverfahrens hängt von den Ableitungswerten $\Phi^{(k)}(x^*)$ ab. Der folgende Satz wird hier nur für $X = \mathbb{R}$ bewiesen. Er gilt auch für $X = \mathbb{R}^d$, ist dann aber wesentlich komplizierter zu beweisen, siehe z.B. BROKATE [1, Abschnitt 10].

Satz 7.3. *Es sei $(\mathbb{R}, \Phi, x^*)$ ein Iterationsverfahren, dessen Iterationsvorschrift Φ lokal um den Fixpunkt x^* p-mal stetig differenzierbar sei.*
(i) *Gilt $p = 1$ und $|\Phi'(x^*)| < 1$, so ist $(\mathbb{R}, \Phi, x^*)$ linear konvergent.*
(ii) *Gilt $|\Phi^{(k)}(x^*)| = 0$ für $k = 1, \dots, p-1$, so hat $(\mathbb{R}, \Phi, x^*)$ mindestens Konvergenzordnung p.*
(iii) *Gilt neben* (i) *oder* (ii) *zusätzlich $\Phi^{(p)}(x^*) \neq 0$, so hat $(\mathbb{R}, \Phi, x^*)$ maximal die Konvergenzordnung p, d.h. das Iterationsverfahren hat keine höhere Ordnung als p.*
(iv) *Gilt $|\Phi'(x^*)| > 1$, so gilt*

$$\exists C > 1 \exists \varepsilon > 0 \forall x \in U_\varepsilon(x^*) \quad |x^* - \Phi(x)| \ge C\, |x^* - x|. \tag{7.5}$$

In diesem Fall ist die Iteriertenfolge i.a. also nicht konvergent!

Beweis. (i), (ii) Eine Taylor-Entwicklung von Φ um x^* zeigt

$$\Phi(x) = \sum_{k=0}^{p} \frac{\Phi^{(k)}(x^*)}{k!}(x - x^*)^k + o(|x - x^*|^p) = x^* + \frac{\Phi^{(p)}(x^*)}{p!}(x - x^*)^p + o(|x - x^*|^p),$$

wobei wir die Voraussetzungen an die Ableitungen sowie $x^* = \Phi(x^*)$ benutzt haben. Also gilt

$$\lim_{x \to x^*} \frac{\Phi(x) - x^*}{(x - x^*)^p} = \frac{\Phi^{(p)}(x^*)}{p!},$$

d.h. zu $\varepsilon > 0$ existiert ein $\delta > 0$, sodass gilt

$$\Big| \frac{\Phi(x) - x^*}{(x - x^*)^p} - \frac{\Phi^{(p)}(x^*)}{p!} \Big| \le \varepsilon \quad \text{für alle } x \in U_\delta(x^*). \tag{7.6}$$

Insbesondere folgt aus der Dreiecksungleichung

$$|\Phi(x) - x^*| \le c(p,\varepsilon)\,|x - x^*|^p \quad \text{mit } c(p,\varepsilon) := \Big|\frac{\Phi^{(p)}(x^*)}{p!}\Big| + \varepsilon.$$

Für $p = 1$ muss man $\varepsilon > 0$ klein genug wählen, sodass $c(1,\varepsilon) < 1$ gilt. Dann folgt insbesondere $\Phi(x) \in U_\delta(x^*)$ für $x \in U_\delta(x^*)$. Für einen Startwert $x_0 \in U_\delta(x^*)$ liegt damit die Iteriertenfolge $x_n \in U_\delta(x^*)$, und es gilt lineare Konvergenz. Für $p > 1$ darf $\varepsilon > 0$ beliebig gewählt werden. Ohne Beschränkung der Allgemeinheit gilt $c(p,\varepsilon)\delta^{p-1} \le 1$ für das zugehörige $\delta > 0$. Dann folgt insbesondere wieder $\Phi(x) \in U_\delta(x^*)$ für $x \in U_\delta(x^*)$. Für einen Startwert $x_0 \in U_\delta(x^*)$ liegt damit die Iteriertenfolge $x_n \in U_\delta(x^*)$, und es gilt Konvergenz von Ordnung $p > 1$.
(iii) Nach (7.6) gilt insbesondere

$$|\Phi(x) - x^*| \ge C(p,\varepsilon)\,|x - x^*|^p \quad \text{mit } C(p,\varepsilon) := \Big|\frac{\Phi^{(p)}(x^*)}{p!}\Big| - \varepsilon. \tag{7.7}$$

Ohne Beschränkung der Allgemeinheit sei $\varepsilon > 0$ klein genug, sodass $C(p,\varepsilon) > 0$ gilt. Falls $(\mathbb{R}, \Phi, x^*)$ die Konvergenzordnung q besitzt, so folgt deshalb für $x_{k+1} = \Phi(x_k)$

$$|x_k - x^*|^p \le C(p,\varepsilon)^{-1}|x_{k+1} - x^*| \le C(p,\varepsilon)^{-1} c\,|x_k - x^*|^q$$

mit $c > 0$ gemäß Definition von Konvergenz der Ordnung q. Wegen $\lim\limits_{k\to\infty} |x_k - x^*| = 0$ folgt $q \le p$, da die obere Schranke nicht schneller gegen Null konvergieren kann als die untere.

(iv) Wählt man $\varepsilon > 0$ klein genug, sodass in (7.7) die Konstante $C(1,\varepsilon) > 1$, erfüllt, so folgt (7.5) mit $C = C(1,\varepsilon)$. ■

Bemerkung. Man kann das Aitkin'sche Δ^2-Verfahren verwenden, um die Konvergenzordnung eines Iterationsverfahrens $(\mathbb{R}, \Phi, x^*)$ zu erhöhen, sog. **Verfahren von Steffensen.** Angewendet auf die Iteriertenfolge $x_k = \Phi(x_{k-1})$ liefert das Aitkin-Verfahren (bei leichter Modifikation) unter der Voraussetzung $|\Phi'(x^*)| < 1$ ein neues Iterationsverfahren $(\mathbb{R}, \Psi, x^*)$, und es gilt

- Φ hat Konvergenzordnung $p = 1 \implies \Psi$ hat mindestens Konvergenzordnung 2,
- Φ hat Konvergenzordnung $p > 1 \implies \Psi$ hat mindestens Konvergenzordnung $2p - 1$.

Der Beweis findet sich im Buch von Stoer [7, Abschnitt 5.10]. □

Beispiel (Newton-Verfahren). Es sei $f : \mathbb{R} \to \mathbb{R}$ eine differenzierbare Funktion mit Nullstelle x^*. Um diese Nullstelle zu approximieren, konstruieren wir eine Folge $(x_k)_{k\in\mathbb{N}}$ ausgehend von einem Startpunkt $x_0 \in \mathbb{R}$ induktiv: Zu gegebenem x_k sei x_{k+1} die Nullstelle der Tangente an den Graphen von f im Punkt $(x_k, f(x_k))$, d.h. $x = x_{k+1}$ erfüllt $0 = f(x_k) + f'(x_k)\,(x - x_k)$. Auflösen nach x zeigt

$$x_{k+1} = \Phi(x_k) \quad \text{mit} \quad \Phi(x) = x - f'(x)^{-1} f(x). \tag{7.8}$$

Im folgenden Abschnitt werden wir das Newton-Verfahren im $\mathbb{R}^d$ eingehender untersuchen. Die (lokal) quadratische Konvergenz des Newton-Verfahrens in $\mathbb{R}$ folgt unmittelbar aus Satz 7.3:

Korollar 7.4. *Es sei f lokal um x^* zweimal stetig differenzierbar und $f(x^*) = 0$ sowie $f'(x^*) \neq 0$. Dann ist das Newton-Verfahren wohldefiniert und (mindestens) quadratisch konvergent.*

Beweis. Wegen $f'(x^*) \neq 0$ ist Φ lokal um x^* stetig differenzierbar. Die erste Ableitung erfüllt

$$\Phi'(x) = 1 - \frac{f'(x)^2 - f(x)f''(x)}{f'(x)^2} = \frac{f(x)f''(x)}{f'(x)^2}.$$

Aus $f(x^*) = 0$ folgt $\Phi'(x^*) = 0$, und Satz 7.3 zeigt die Behauptung. ■

Beispiel (Sekantenverfahren). Es sei $f : \mathbb{R} \to \mathbb{R}$ eine Funktion mit Nullstelle x^*. Um diese Nullstelle zu approximieren, konstruieren wir induktiv eine Folge $(x_k)_{k\in\mathbb{N}}$ ausgehend von zwei verschiedenen Punkten $x_0, x_1 \in \mathbb{R}$: Zu gegebenen (verschiedenen) x_{k-1} und x_k sei x_{k+1} die Nullstelle der Gerade durch die beiden Punkte $(x_{k-1}, f(x_{k-1}))$ und $(x_k, f(x_k))$, d.h. $x = x_{k+1}$ erfüllt

$$0 = f(x_k) + \frac{f(x_k) - f(x_{k-1})}{x_k - x_{k-1}}\,(x - x_k),$$

d.h. mit der Steigung $b_k := \frac{f(x_k)-f(x_{k-1})}{x_k-x_{k-1}}$ gilt also

$$x_{k+1} = x_k - b_k^{-1} f(x_k). \tag{7.9}$$

Im abstrakten Rahmen ist also $X = \mathbb{R}^2$. Das Sekantenverfahren gehört in die Klasse der **Quasi-Newton-Verfahren**. Man kann zeigen, dass das Sekantenverfahren für eine $\mathcal{C}^2$-Funktion f mit der Ordnung des Goldenen Schnitts $(1 + \sqrt{5})/2 \approx 1.618$ konvergiert, siehe Carstensen [2, Abschnitt 3.2]. ■

Beispiel (Bisektionsverfahren). Es sei $f : [a, b] \to \mathbb{R}$ eine stetige Funktion mit $f(a)f(b) \leq 0$. Dann hat f eine Nullstelle $x^* \in [a, b]$. Diese kann durch Intervallhalbierung approximiert werden. Wir definieren eine Folge $(a_k, b_k)_{k\in\mathbb{N}}$ ausgehend von $(a_0, b_0) = (a, b)$ induktiv durch

$$c_{k-1} := \frac{a_{k-1} + b_{k-1}}{2}, \quad \text{und} \quad (a_k, b_k) = \begin{cases} (a_{k-1}, c_{k-1}), & \text{falls } f(a_{k-1})f(c_{k-1}) \leq 0 \\ (c_{k-1}, b_{k-1}), & \text{sonst.} \end{cases}$$

Besitzt f in $[a, b]$ eine eindeutige Nullstelle x^*, so ist das Bisektionsverfahren linear konvergent gegen (x^*, x^*), denn in jedem Schritt wird das betrachtete Intervall halbiert. ■

Übung. Man formuliere das Bisektionsverfahren in abstrakter Form (X, Φ, x^*) und beweise, dass (X, Φ, x^*) linear mit Konstante $c = 1/2$ gegen x^* konvergiert, sofern x^* die einzige Nullstelle von f in $[a, b]$ ist. ■

Numerische Bestimmung der Konvergenzordnung

Es sei $(x_k)_{k\in\mathbb{N}}$ die (konvergente) Iteriertenfolge zu einem Iterationsverfahren (X, ϕ, x^*) mit Startwert x_0 und Konvergenzordnung p. Mit $d_k := d(x_k, x^*)$ und dem Ansatz

$$d_k = c\, d_{k-1}^p \quad \text{und} \quad d_{k-1} = c\, d_{k-2}^p$$

erhalten wir durch Division $d_k/d_{k-1} = (d_{k-1}/d_{k-2})^p$ und deshalb

$$p = \frac{\log(d_k/d_{k-1})}{\log(d_{k-1}/d_{k-2})} \quad \text{sowie} \quad c = \frac{d_k}{d_{k-1}^p} \tag{7.10}$$

In der Regel ist aber x^* unbekannt, sodass die benötigten Werte d_{k-2}, d_{k-1}, d_k nicht exakt berechnet werden können. Mit Dreiecksungleichung gelten

$$d(x_k, x_{k+1}) \leq d(x_k, x^*) + c\, d(x_k, x^*)^p \quad \text{und} \quad d(x_k, x^*) \leq d(x_k, x_{k+1}) + c\, d(x_k, x^*)^p.$$

Im Fall $p > 1$ ist $d(x_k, x^*)^p$ verglichen mit $d(x_k, x^*)$ von höherer Ordnung, und insbesondere gilt asymptotisch $d(x_k, x_{k+1}) \approx d(x_k, x^*) = d_k$. Im Fall $p = 1$ gilt nach vorausgegangener Ungleichung

$$d(x_k, x_{k+1}) \leq (1 + c)\, d(x_k, x^*) \quad \text{und} \quad d(x_k, x^*) \leq \frac{1}{1-c} d(x_k, x_{k+1}),$$

d.h. auch in diesem Fall verhält sich $d(x_k, x_{k+1}) \sim d(x_k, x^*) = d_k$. Mit dem Differenzenoperator $\Delta_k := d(x_k, x_{k+1})$ können die **experimentelle Konvergenzordnung** p sowie die Konstante c von (X, Φ, x^*) also durch

$$p = \frac{\log(\Delta_k/\Delta_{k-1})}{\log(\Delta_{k-1}/\Delta_{k-2})} \quad \text{und} \quad c = \frac{\Delta_k}{\Delta_{k-1}^p} \tag{7.11}$$

berechnet werden. Die Bestimmung benötigt also die Berechnung von vier aufeinanderfolgenden Approximationen $x_{k-2}, x_{k-1}, x_k, x_{k+1}$.

MATLAB-Beispiel: Berechnung der experimentellen Konvergenzordnung

Es sei x eine Folge von skalaren Approximationen aus einem skalarwertigen Iterationsverfahren $(\mathbb{R}, \Phi, x^*)$. Dann berechnet die folgende MATLAB-Funktion die experimentelle Konvergenzordnung p sowie die Konstante c.

```
function [p,c] = convorder(x)

n = length(x);
p = zeros(n,1);
c = zeros(n,1);
for j = 4:n
  p(j) = log( abs(x(j)-x(j-1)) / abs(x(j-1)-x(j-2)) ) / ...
         log( abs(x(j-1)-x(j-2)) / abs(x(j-2)-x(j-3)) );
  c(j) = abs(x(j)-x(j-1)) / abs(x(j-1)-x(j-2))^p(j);
end
```

Numerische Realisierung von Verfahren zur Nullstellensuche

Als Anwendung betrachten wir die numerische Bestimmung der Nullstelle $x^* \approx 1/2$ der Funktion

$$f(x) = x^2 + \exp(x) - 2. \tag{7.12}$$

Die Konvergenz jedes iterativen Verfahrens lässt sich in drei Phasen unterteilen:

- **Vorasymptotische Phase.** Der Startwert x_0 ist i.a. zufällig gewählt und liegt daher in aller Regel nicht im Konvergenzbereich des Verfahrens. Es ist daher für $x_0, \ldots, x_{k-1}$ keine Konvergenz zu beobachten.
- **Asymptotische Phase.** Nach k Iterationen, liegt x_k im Konvergenzbereich des Verfahrens, und die Folgenglieder $x_k, \ldots, x_\ell$ zeigen das asymptotische Konvergenzverhalten.
- **Nachasymptotische Phase.** Nach endlich vielen Iterationen unterscheiden sich die x_m für $m \geq \ell$ nur noch aufgrund von Rechenfehlern. Es ist also nicht mehr mit einer Verbesserung der numerischen Lösung zu rechnen. Das Iterationsverfahren sollte beendet werden.

Zum Abbruch eines Iterationsverfahrens zur Nullstellenbestimmung verwenden wir die folgende Heuristik: Wir berechnen zunächst Folgenglieder $x_0, \ldots, x_k$, bis das **Residuum** $|F(x_k)|$ wesentlich kleiner ist als das Anfangsresiduum $|F(x_0)|$, z.B.

- Berechne $x_1, \ldots, x_k$, bis für das Residuum gilt

  $$|F(x_k)| \leq \max\{\tau_{\text{abs}}, \tau_{\text{rel}}|F(x_0)|\},$$

  wobei die **absolute** und **relative Toleranz** $\tau_{\text{abs}}, \tau_{\text{rel}} > 0$ gegebene Parameter sind, z.B. $\tau_{\text{abs}} = 10^{-12}$, $\tau_{\text{rel}} = 10^{-6}$.

Wenn dies der Fall ist, ist davon auszugehen, dass wir bereits in die asymptotischen Phase eingetreten sind. Nun berechnen wir weitere Folgenglieder $x_k, \ldots, x_\ell$, bis $|F(x_\ell)|$ hinreichend genau ist und sich die Iterierten x_ℓ nur noch wenig unterscheiden.

- Berechne $x_k, \ldots, x_\ell$, bis gilt

  $$|F(x_\ell)| \leq \tau_{\text{abs}} \quad \text{und} \quad |x_\ell - x_{\ell-1}| \leq \tau_{\text{abs}} \max\{1, |x_\ell|\},$$

  wobei wir bei der letzten Bedingung berücksichtigt haben, dass $x_\ell = 0$ gelten könnte.

Schritt	x	$\lvert f(x)\rvert$	p	c
1	$1.0000000000e+00$	$1.7183e+00$		
2	$6.3582467285e-01$	$2.9285e-01$		
3	$5.4315669270e-01$	$1.6452e-02$		
4	$5.3729735863e-01$	$6.3824e-05$	2.02	$7.11e-01$
5	$5.3727444952e-01$	$9.7391e-10$	2.01	$6.96e-01$
6	$5.3727444917e-01$	$0.0000e+00$	2.00	$6.68e-01$

Tabelle 7.1: Numerische Ergebnisse zur Bestimmung der Nullstelle x^* von f aus (7.12) mittels Newton-Verfahren und Startwert $x_0 = 1$. Die optimale Konvergenzordnung 2 stellt sich ein, und das Newton-Verfahren konvergiert in 6 Schritten gegen die exakte Lösung.

MATLAB-Beispiel: Newton-Verfahren

Die Funktion f sowie ihre Ableitung f' müssen als MATLAB-Funktionen zur Verfügung stehen. Seit MATLAB 7 geschieht dies für die Funktion f aus (7.12) beispielsweise in der Form

```
f = @(x) x^2+exp(x)-2;
fprime = @(x) 2*x+exp(x);
```

Die folgende Funktion realisiert das Newton-Verfahren (inkl. Abbruchbedingung):

```
function x = newton_method(f,fprime,x,tau_abs,tau_rel)
while abs( f(x(end)) ) > max([tau_abs,tau_rel*abs( f(x(1)) )])
  x(end+1) = newton(x(end),f,fprime);
end
while abs( f(x(end)) ) > tau_abs | ...
      abs( x(end)-x(end-1) ) > tau_abs*max(1,abs(x(end)))
  x(end+1) = newton(x(end),f,fprime);
end

function x = newton(x,f,fprime)
x = x - f(x)/fprime(x);
```

Der Aufruf `x = newton_method(f,fprime,1,1e-12,1e-6);` liefert die numerischen Ergebnisse aus Tabelle 7.1.

Bemerkung. Quadratische Konvergenz bedeutet asymptotisch, dass sich pro Iterationsschritt die Anzahl der Stellen, die von x_n und x^* übereinstimmen, verdoppelt, vgl. Tabelle 7.1. Sobald man in die asymptotische Phase eingetreten ist, sind also nur sehr wenige Iterationsschritte nötig, bis der Fehler $\|x_n - x^*\|/\|x_n^*\| \approx \varepsilon$ auf Rundungsfehlerniveau ist. □

Schritt	x	$\lvert f(x)\rvert$	p	c
1	$0.0000000000e+00$	$1.0000e+00$		
2	$1.0000000000e+00$	$1.7183e+00$		
3	$3.6787944117e-01$	$4.2000e-01$		
4	$4.9203942515e-01$	$1.2225e-01$	3.55	$6.32e-01$
5	$5.4301666517e-01$	$1.6058e-02$	0.55	$1.60e-01$
6	$5.3709785645e-01$	$4.9191e-04$	2.42	$7.92e+00$
7	$5.3727377565e-01$	$1.8764e-06$	1.63	$7.64e-01$
8	$5.3727444925e-01$	$2.2073e-10$	1.58	$5.91e-01$
9	$5.3727444917e-01$	$0.0000e+00$	1.63	$8.57e-01$

Tabelle 7.2: Numerische Ergebnisse zur Bestimmung der Nullstelle x^* von f aus (7.12) mittels Sekanten-Verfahren und Startwert $x_0 = 0$, $x_1 = 1$. Die erwartete Konvergenzrate 1.618 ist im wesentlichen sichtbar. Das Verfahren konvergiert in 9 Schritten gegen die exakte Lösung.

Schritt	x	$\lvert f(x)\rvert$	p	c
1	$5.0000000000e-01$	$1.0128e-01$		
11	$5.3759765625e-01$	$9.0061e-04$	1.00	$5.00e-01$
21	$5.3727483749e-01$	$1.0818e-06$	1.00	$5.00e-01$
31	$5.3727444960e-01$	$1.1817e-09$	1.00	$5.00e-01$
41	$5.3727444917e-01$	$2.5557e-13$	1.00	$5.00e-01$

Tabelle 7.3: Numerische Ergebnisse zur Bestimmung der Nullstelle x^* von f aus (7.12) mittels Bisektionsverfahren auf dem Intervall $[0, 1]$. Wie erwartet beobachten wir lineare Konvergenz mit Konstante 1/2. Nach 41 Schritten ist die Nullstelle x^* mit einer Genauigkeit von 10^{-12} approximiert.

Schritt	x	$\lvert f(x)\rvert$	p	c
1	$1.0000000000e+00$	$1.7183e+00$		
11	$5.4137916241e-01$	$1.1467e-02$	0.97	$5.66e-01$
21	$5.3731367149e-01$	$1.0927e-04$	1.00	$6.27e-01$
31	$5.3727482282e-01$	$1.0409e-06$	1.00	$6.28e-01$
41	$5.3727445273e-01$	$9.9158e-09$	1.00	$6.28e-01$
51	$5.3727444921e-01$	$9.4457e-11$	1.00	$6.28e-01$
61	$5.3727444917e-01$	$8.9972e-13$	1.00	$6.30e-01$

Tabelle 7.4: Numerische Ergebnisse zur Bestimmung der Nullstelle x^* von f aus (7.12) mit Hilfe der Iteration $\Phi(x) = \log(2 - x^2)$ und Startwert $x_0 = 1$. Wir erhalten lineare Konvergenz mit einer Konstanten $c \approx 0.63 \approx \Phi'(x^*)$. Deshalb konvergiert das Verfahren noch langsamer als das Bisektionsverfahren. Erst nach 61 Schritten erhalten wir eine Approximation von x^* mit einer Genauigkeit von 10^{-12}.

MATLAB-Beispiel: Einfache Iterationsverfahren

Die folgenden Funktionen realisieren jeweils einen Iterationsschritt des Sekanten- bzw. Bisektionsverfahrens

```
function x = sekante(x0,x1,f)
b = (f(x1)-f(x0))/(x1-x0);
x = x1-f(x1)/b;

function [a,b] = bisektion(a,b,f)
c = (a+b)/2;
if f(a)*f(c) <= 0
  b=c;
else
  a=c;
end
```

Die anfangs beschriebene Iteration $\Phi_3(x) = \log(2 - x^2)$ kann am MATLAB-Prompt definiert werden

```
phi3 = @(x) log(2-x^2);
```

7.2 Newton-Verfahren zur Lösung nichtlinearer GLS

Das (klassische) **Newton-Verfahren** in $\mathbb{R}^d$ ist durch

$$x_0 \in \mathbb{R}^d, \quad x_n := x_{n-1} - DF(x_{n-1})^{-1} F(x_{n-1}) \quad \text{für } n \in \mathbb{N} \tag{7.13}$$

gegeben. Der folgende Satz betrachtet auch den allgemeineren Fall von sogenannten **gedämpften Newton-Verfahren**, die im Anschluss diskutiert werden sollen.

Satz 7.5. *Es sei $\|\cdot\|$ eine Norm auf $\mathbb{R}^d$, $\Omega \subseteq \mathbb{R}^d$ offen, $F \in C^2(\Omega;\mathbb{R}^d)$ und $x^* \in \Omega$ eine Nullstelle von F derart, dass die Jacobi-Matrix $DF(x^*)$ regulär ist. Ist $\lambda \in (0,1]$ und $\lambda_n \in [\lambda, 1]$, so existiert ein $\varepsilon > 0$ derart, dass die folgenden Aussagen gelten:*
(i) *Für $x \in U_\varepsilon(x^*)$ ist $DF(x)$ regulär.*
(ii) *Für jeden Startwert $x_0 \in U_\varepsilon(x^*)$ ist die Iteriertenfolge $x_n := x_{n-1} - \lambda_n DF(x_{n-1})^{-1} F(x_{n-1})$ wohldefiniert, d.h. $x_n \in U_\varepsilon(x^*)$, und es existiert eine Konstante $c \in (0,1)$ mit*

$$\|x_n - x^*\| \le c\,\|x_{n-1} - x^*\| \quad \textit{für } n \in \mathbb{N}, \tag{7.14}$$

d.h. das gedämpfte Newton-Verfahren ist wohldefiniert und konvergiert linear.
(iii) *Für jeden Startwert $x_0 \in U_\varepsilon(x^*)$ ist die Iteriertenfolge $x_n := x_{n-1} - DF(x_{n-1})^{-1} F(x_{n-1})$ wohldefiniert, d.h. $x_n \in U_\varepsilon(x^*)$, und es existiert eine Konstante $c > 0$ mit*

$$\|x_n - x^*\| \le c\,\|x_{n-1} - x^*\|^2 \quad \textit{für } n \in \mathbb{N}, \tag{7.15}$$

d.h. das Newton-Verfahren ist wohldefiniert und quadratisch konvergent.

Beweis. Wir gliedern den Beweis wieder in mehrere Beweisschritte, wobei wir uns ohne Beschränkung der Allgemeinheit auf die euklidische Norm $\|\cdot\|_2$ beschränken.
1. Schritt. *Es existiert ein $\varepsilon > 0$ mit der Eigenschaft, dass die Jacobi-Matrix $DF(x)$ regulär ist für alle $x \in U_\varepsilon(x^*)$.* Falls $A \in \mathbb{R}^{d\times d}$ regulär ist und $B \in \mathbb{R}^{d\times d}$ mit $\|B - A\|_2 < \|A^{-1}\|_2^{-1}$, so ist auch B regulär. Also ist $\mathcal{U} := \{A \in \mathbb{R}^{d\times d} \,|\, A \text{ regulär}\}$ eine offene Teilmenge von $\mathbb{R}^{d\times d}$ mit $DF(x^*) \in \mathcal{U}$. Die Ableitung $DF : \Omega \to \mathbb{R}^{d\times d}$ ist nach Voraussetzung stetig. Dies zeigt die Behauptung. □

2. Schritt. *Es existiert ein $\varepsilon > 0$, sodass die Suprema*

$$M := \sup_{x\in U_\varepsilon(x^*)} \|DF(x)^{-1}\|_2 \quad \textit{und} \quad \widetilde{M} := \sup_{x\in U_\varepsilon(x^*)} \Big(\sum_{j,k,\ell=1}^{d} \Big| \frac{\partial^2 F_j}{\partial x_k \partial x_\ell}(x) \Big|^2 \Big)^{1/2}$$

endlich sind, d.h. $M, \widetilde{M} < \infty$. Wir nehmen $\varepsilon := \widetilde{\varepsilon}/2$ mit $\widetilde{\varepsilon}$ aus Schritt 1. Dann ist $\overline{U_\varepsilon(x^*)}$ kompakt und in $U_{\widetilde{\varepsilon}}(x^*)$ enthalten. Da stetige Funktionen auf Kompakta ihr Supremum annehmen, folgt $M < \infty$. Wegen $F \in C^2(\Omega;\mathbb{R}^d)$ folgt mit demselben Argument $\widetilde{M} < \infty$. □

3. Schritt. *Für alle $x, y \in U_\varepsilon(x^*)$ gilt $\|F(y) - F(x) - DF(x)(y-x)\|_2 \le \frac{\widetilde{M}}{2}\|y - x\|_2^2$.* Wir verwenden denselben Beweis wir für die Mittelwertsatz in $\mathbb{R}^d$. Für fixierte $x, y \in U_\varepsilon(x^*)$ definieren wir die Funktion $f : [0,1] \to \mathbb{R}^d$, $f(t) := F(x + t(y-x))$. Mit der komponentenweisen Ableitung

von f zeigt partielle Integration

$$\begin{aligned}\int_0^1 (1-t)f''(t)\,dt = f'(1) - f'(0) - \int_0^1 tf''(t)\,dt &= f'(1) - f'(0) - \big[tf'(t)\big]_{t=0}^1 + \int_0^1 f'(t)\,dt \\ &= f(1) - f(0) - f'(0) \\ &= F(y) - F(x) - DF(x)(y-x),\end{aligned}$$

denn aus der Kettenregel folgt $f'(t) = DF(x+t(y-x))(y-x)$, insbesondere also $f'(0) = DF(x)(y-x)$. Insbesondere erhalten wird

$$\|F(y) - F(x) - DF(x)(y-x)\|_2 \le \int_0^1 (1-t)\|f''(t)\|_2\,dt.$$

Nun müssen wir noch die zweiten Ableitungen von f'' berechnen und abschätzen,

$$\begin{aligned}f_j'(t) &= \big(DF(x+t(y-x))(y-x)\big)_j \\ &= \sum_{k=1}^d \frac{\partial F_j}{\partial x_k}(x+t(y-x))(y_k - x_k), \\ f_j''(t) &= \sum_{k=1}^d (y_k - x_k)\, D\frac{\partial F_j}{\partial x_k}(x+t(y-x))(y-x) \\ &= \sum_{k,\ell=1}^d (y_k - x_k)\,\frac{\partial^2 F_j}{\partial x_k \partial x_\ell}(x+t(y-x))(y_\ell - x_\ell).\end{aligned}$$

Mit der Cauchy-Schwarz-Ungleichung folgt

$$\begin{aligned}\|f''(t)\|_2^2 = \sum_{j=1}^d |f_j''(t)|^2 &\le \sum_{j=1}^d \Big(\sum_{k,\ell=1}^d \Big|\frac{\partial^2 F_j}{\partial x_k \partial x_\ell}(x+t(y-x))\Big|^2\Big)\Big(\sum_{k,\ell=1}^d \big|(y_k - x_k)(y_\ell - x_\ell)\big|^2\Big) \\ &\le \widetilde{M}^2 \|x-y\|_2^4.\end{aligned}$$

Damit erhalten wir

$$\|F(y) - F(x) - DF(x)(y-x)\|_2 \le \widetilde{M}\|x-y\|_2^2 \int_0^1 (1-t)\,dt = \frac{1}{2}\widetilde{M}\|x-y\|_2^2.$$

Dies zeigt die Behauptung. □

4. Schritt. *Für alle $x, y \in U_\varepsilon(x^*)$ gilt $\|(y-x) - DF(x)^{-1}\big(F(y) - F(x)\big)\|_2 \le \frac{M\widetilde{M}}{2}\|y-x\|_2^2$,* denn

$$\begin{aligned}\|DF(x)^{-1}\big(F(y) - F(x)\big) - (y-x)\|_2 &= \|DF(x)^{-1}\big(F(y) - F(x) - DF(x)(y-x)\big)\|_2 \\ &\le M\,\|F(y) - F(x) - DF(x)(y-x)\|_2\end{aligned}$$

5. Schritt. *Zu $x \in U_\varepsilon(x^*)$ und $\widetilde{\lambda} \in [\lambda, 1]$ definiere $y := x - \widetilde{\lambda} DF(x)^{-1}F(x)$. Dann gilt*

$$\|y - x^*\|_2 \le \Big\{(1-\lambda) + \frac{M\widetilde{M}}{2}\|x - x^*\|_2\Big\}\|x - x^*\|_2. \tag{7.16}$$

Wegen $F(x^*) = 0$ gilt

$$\begin{aligned} y - x^* &= x - x^* - \widetilde{\lambda} DF(x)^{-1}\{F(x) - F(x^*)\} \\ &= (1-\widetilde{\lambda})(x - x^*) - \widetilde{\lambda} DF(x)^{-1}\big\{\big(F(x) - F(x^*)\big) - DF(x)(x - x^*)\big\}. \end{aligned}$$

Mit Schritt 3 folgt deshalb

$$\|y - x^*\|_2 \le (1-\widetilde{\lambda})\|x - x^*\|_2 + \widetilde{\lambda}\,\frac{M\widetilde{M}}{2}\,\|x - x^*\|_2^2 \le \Big\{(1-\lambda) + \frac{M\widetilde{M}}{2}\,\|x - x^*\|_2\Big\}\|x - x^*\|_2.$$

6. Schritt. *Wählt man $\varepsilon > 0$ klein genug, sodass $(1-\lambda) + \frac{M\widetilde{M}}{2}\,\varepsilon =: c < 1$, so ist das (gedämpfte) Newton-Verfahren mit Startwert $x_0 \in U_\varepsilon(x^*)$ wohldefiniert. Das gedämpfte Newton-Verfahren konvergiert linear, das klassische Newton-Verfahren konvergiert quadratisch.* Die Aussage folgt induktiv aus Schritt 4: Ist $x_n \in U_\varepsilon(x^*)$, so erhalten wir mit $x = x_n$ und $y = x_{n+1}$ die Abschätzung $\|x_{n+1} - x^*\|_2 \le c\,\|x - x^*\|_2 < \varepsilon$, d.h. insbesondere gilt $x_{n+1} \in U_\varepsilon(x^*)$. Dies zeigt zum einen die Wohldefiniertheit und zum anderen die lineare Konvergenz. Für $\lambda = \lambda_n = 1$ folgt aus Schritt 4 sogar $\|x_{n+1} - x^*\|_2 \le c\,\|x - x^*\|_2^2$ ■

Übung. Es sei $\mathcal{U} := \big\{A \in \mathbb{K}^{n\times n} \,\big|\, A \text{ regulär}\big\}$. Man zeige, dass die Abbildung $\mathrm{inv} : \mathcal{U} \to \mathcal{U}, A \mapsto A^{-1}$ Fréchet-differenzierbar und insbesondere stetig ist.
Hinweis. Die Abbildung $b : L(X) \times L(X) \to L(X)$, $b(A,B) = AB$ ist bilinear und deshalb Fréchet-differenzierbar mit $Db(A,B)(H,K) = AK + HB$. Es sei $\mathcal{U} \subseteq L(X)$ die offene Menge aller bijektiven Operatoren. Wir betrachten die Abbildungen $g : \mathcal{U} \to \mathcal{U} \times \mathcal{U}, g(A) = (A, A^{-1})$ und $f : \mathcal{U} \to \mathcal{U}, f = b \circ g$. Es gilt $f(A) = \boldsymbol{I}$. Unter der Voraussetzung, dass $\mathrm{inv} : \mathcal{U} \to \mathcal{U}$ Fréchet-differenzierbar ist, gilt $Dg(A)(B) = (B, D\mathrm{inv}(A)(B))$. Mit der Kettenregel folgt

$$0 = Df(A)(B) = Db(A, A^{-1})(B, D\mathrm{inv}(A)(B)) = AD\mathrm{inv}(A)(B) + BA^{-1}.$$

Umformen zeigt also, dass $D\mathrm{inv}(A)(B) = -A^{-1}BA^{-1}$ gelten muss, falls die Inversion Fréchet-differenzierbar ist. ■

Bei der Konstruktion eines Abbruchkriteriums für eine Nullstellensuche in Abschnitt 7.1 haben wir sowohl die Schrittweite $\|x_{n+1} - x_n\|$ als auch die Norm des Residuums $\|F(x_n)\|$ herangezogen. Für das Newton-Verfahren kann man zeigen, dass sich $\|F(x_n)\|$ genauso verhält wie der Fehler $\|x_n - x^*\|$ und insbesondere ebenfalls quadratisch konvergiert.

Korollar 7.6. *Für das Newton-Verfahren existieren Konstanten $c_1, c_2, c_3 > 0$ mit*

$$c_1^{-1}\|x_n - x^*\| \le \|F(x_n)\| \le c_2\|x_n - x^*\| \quad \textit{und} \quad \|F(x_n)\| \le c_3\|F(x_{n-1})\|^2 \tag{7.17}$$

für alle $n \in \mathbb{N}$, sofern der Startwert $x_0 \in U_\varepsilon(x^*)$ mit hinreichend kleinem $\varepsilon > 0$ erfüllt.

Beweis. Da $x_n \in U_\varepsilon(x^*)$ gilt, reicht es, wenn wir die Aussage für ein beliebiges $y \in U_\varepsilon(x^*)$ zeigen. Nach Schritt 3 im Beweis von Satz 7.5 existiert eine Konstante $C > 0$ mit

$$\|(y - x) - DF(x)^{-1}(F(y) - F(x))\| \le C\,\|y - x\|^2 \quad \text{für alle } x, y \in U_\varepsilon(x^*).$$

Wir nehmen an, dass $C\varepsilon \leq 1/2$ gilt. Für $x = x^*$ und $y \in U_\varepsilon(x^*)$ folgt mit der Dreiecksungleichung

$$\|y - x^*\| \leq C\,\|y - x^*\|^2 + \|DF(x^*)^{-1}F(y)\| \leq \frac{1}{2}\,\|y - x^*\| + \|DF(x^*)^{-1}\|\|F(y)\|,$$

d.h. $c_1 = 2\,\|DF(x^*)^{-1}\|$. Ferner gilt wieder mit Dreiecksungleichung

$$\|F(y)\| \leq \|DF(x^*)\|\|DF(x^*)^{-1}F(y)\| \leq \|DF(x^*)\|\big\{C\,\|y - x^*\|^2 + \|y - x^*\|\big\}$$

und damit $c_2 = \frac{3}{2}\,\|DF(x^*)\|$. Abschließend folgt damit auch

$$\|F(x_n)\| \leq c_2\|x_n - x^*\| \leq c_2c_0\|x_{n-1} - x^*\|^2 \leq c_1^2c_2c_0\|F(x_{n-1})\|^2,$$

wobei $c_0 > 0$ die Konstante aus der quadratischen Konvergenz des Newton-Verfahrens sei. ■

Bemerkung. Offensichtlich gilt die erste Abschtzung in (7.17) für *jedes* Iterationsverfahren zur Nullstellensuche $F(x^*) = 0$, sofern die Funktion F die Voraussetzungen des Newton-Verfahrens erfüllt. □

Selbst für hübsche Funktionen kann man beim Newton-Verfahren *keine globale* Konvergenz erwarten.

Übung. Der Arcustangens ist streng monoton wachsend mit eindeutiger Nullstelle bei $x^* = 0$. Man zeige, dass das Newton-Verfahren divergiert, falls der Startwert x_0 betragsmäßig größer ist als die positive Lösung $y \approx 1.37$ der Gleichung $2y = (1 + y^2)\arctan(y)$. Man mache sich die Behauptung zunächst anhand einer Skizze klar. ■

Um (zumindest für den Arcustangens) globale Konvergenz des Newton-Verfahrens zu erhalten, muss man mit einer Dämpfung arbeiten. Im Arcustangens-Beispiel ist die Crux gerade, dass die Tangente zu flach ist und die Norm des Residuums deshalb von Schritt zu Schritt wächst. Wir zeigen nun, dass man den Dämpfungsparameter so wählen kann, dass das Residuum in jedem Schritt garantiert kleiner wird.

Lemma 7.7. *Es sei $\Omega \subset \mathbb{R}^d$ offen, $F \in C^2(\Omega, \mathbb{R}^d)$ mit $DF(x)$ regulär für alle $x \in \Omega$ und $K \subset \Omega$ kompakt. Dann existieren Konstanten $\lambda_{\max}, \gamma > 0$ mit*

$$\|F\big(x - \lambda DF(x)^{-1}F(x)\big)\|_2^2 \leq \big(1 + \gamma\lambda^2 - 2\lambda\big)\,\|F(x)\|_2^2 \quad \textit{für alle } x \in K, \lambda \in [0, \lambda_{\max}], \tag{7.18}$$

d.h. durch $\lambda > 0$ hinreichend klein kann $\|F(x_n)\|_2 < \|F(x_{n-1})\|_2$ erreicht werden.

Beweis. Da K kompakt ist, sind die beiden folgenden Suprema endlich,

$$c_1 := \sup_{x\in K} \|F(x)\|_2, \quad c_2 := \sup_{x\in K} \|DF(x)^{-1}\|_2,$$

und wir definieren

$$\lambda_{\max} := \min\Big\{1,\ \frac{1}{2c_1c_2}\,\mathrm{dist}(K, \partial\Omega)\Big\},$$

wobei $\operatorname{dist}(K, \partial\Omega) := \inf\big\{\|x-y\|_2 \,\big|\, x \in K, y \in \partial\Omega\big\}$ den Abstand von K zum Rand von Ω bezeichne. Da K kompakt ist, ist $\operatorname{dist}(K, \partial\Omega) > 0$, denn die Abbildung $x \mapsto \operatorname{dist}(\{x\}, \partial\Omega)$ ist stetig. Nun betrachten wir die kompakte Menge

$$\widetilde{K} := \big\{x - \lambda p \,\big|\, x \in K, p \in \mathbb{R}^d, \|p\|_2 \le c_1 c_2, \lambda \in [0, \lambda_{\max}]\big\} \subset \Omega,$$

wobei die Inklusion $\widetilde{K} \subset \Omega$ aus $\|\lambda p\|_2 \le \frac{1}{2}\operatorname{dist}(K, \partial\Omega)$ folgt. Wegen $F \in C^2(\Omega, \mathbb{R}^d)$ definiert $f(y) := \|F(y)\|_2^2$ eine Funktion $f \in C^2(\Omega)$. Nun seien $x \in K$ und $\lambda \in [0, \lambda_{\max}]$ fixiert und

$$p := DF(x)^{-1}F(x).$$

Um die Abschätzung (7.18) zu zeigen, definieren wir $g(t) := f(x - t\lambda p)$. Beachte $g \in C^2[0,1]$ und $x - t\lambda p \in \widetilde{K}$ für $t \in [0,1]$. Partielle Integration zeigt

$$g(1) = g(0) + g'(0) + \int_0^1 (1-t)g''(t)\,dt. \tag{7.19}$$

Offensichtlich gelten

$$g(1) = f(x - \lambda p) = \|F(x - \lambda p)\|_2^2 \quad \text{sowie} \quad g(0) = f(x) = \|F(x)\|_2^2.$$

Zur Berechnung von $g'(0)$ beachten wir $f(y) = F(y) \cdot F(y)$ und erhalten mit der Kettenregel $Df(y) = 2F(y)^T DF(y)$. Mit $g'(t) = -Df(x - t\lambda p)(\lambda p)$ ergibt sich nach Definition von p

$$g'(0) = -2F(x)^T DF(x)(\lambda p) = -2\lambda\|F(x)\|_2^2.$$

Die zweite Ableitung von g erfüllt $g''(t) = (\lambda p)\cdot D^2 f(x - t\lambda p)(\lambda p)$ mit der Hesse-Matrix $D^2 f \in \mathbb{R}^{d\times d}$ und lässt sich deshalb mittels

$$|g''(t)| \le c_3\lambda^2\|p\|_2^2 \le c_2^2 c_3 \lambda^2 \|F(x)\|_2^2, \quad c_3 := \sup_{y \in \widetilde{K}} \|D^2 f(y)\|_2 < \infty,$$

abschätzen. Insgesamt erhalten wir aus (7.19) also die Abschätzung

$$\|F(x - \lambda p)\|_2^2 \le \|F(x)\|_2^2 - 2\lambda\|F(x)\|_2^2 + c_2^2 c_3 \lambda^2 \|F(x)\|_2^2,$$

d.h. es gilt (7.18) mit $\gamma = c_2^2 c_3$.

■

Übung. Man rekapituliere, dass die Funktion $x \mapsto \operatorname{dist}(x, \partial\Omega)$ stetig ist. Man beweise, dass die Menge $\widetilde{K}$ aus dem Beweis von Lemma 7.7 kompakt ist und verifiziere alle aufgetretenen Ableitungen. ■

Um quadratische Konvergenz zu bekommen, sollte man möglichst $\lambda_n = 1$ wählen. Wie uns aber das arctan-Beispiel zeigt, sollten wir stets bemüht sein, über die Wahl des Dämpfungsparameters λ_n gleichzeitig $\|F(x_{n+1})\|_2 < \|F(x_n)\|_2$ sicherzustellen. Dazu verwenden wir folgende Idee:

- Fixiere $q \in (0,1)$.
- In jedem Newton-Schritt suche ein möglichst kleines $\ell \in \mathbb{N}_0$, sodass gilt

$$\|F\big(x_n - q^\ell DF(x_n)^{-1}F(x_n)\big)\|_2 < \|F(x_n)\|_2.$$

- Definiere dann $\lambda_n := q^\ell$.

Da dieses Vorgehen unter Umständen in jedem Schritt viele F-Auswertungen nach sich ziehen kann, berücksichtigen wir noch, dass sich λ_n und λ_{n+1} nicht stark unterscheiden werden, d.h. wir nehmen $\min\{1, \lambda_n/q\} \geq \lambda_n$ als erste Schätzung für λ_{n+1}. Im folgenden Algorithmus führen wir zusätzlich den Parameter $\lambda_{\min}$ ein und brechen die Nullstellensuche ab, wenn der Dämpfungsparameter λ_n kleiner als $\lambda_{\min}$ wird.

Algorithmus 7.8: Gedämpftes Newton-Verfahren

Input: Funktion $F(x)$ mit Nullstelle $x^* \in \mathbb{R}^d$, $x_0 \in \mathbb{R}^d$ Startwert, $q \in (0,1)$, $\lambda_{\min} \in (0,1)$.

$k := 0$, $\lambda_0 := 1$
`repeat`
  Berechne $A := DF(x_k)$, $b := -F(x_k)$
  Löse das Gleichungssystem $Ap_k = b$
  `while (` $\lambda_k \geq \lambda_{\min}$ `and` $\|F(x_k + \lambda_k p_k)\|_2 \geq \|F(x_k)\|_2$ `)`
    $\lambda_k := q\lambda_k$
  `end`
  `if` $\lambda_k < \lambda_{\min}$
    Abbruch mit Fehlermeldung
  `else`
    $x_{k+1} := x_k + \lambda_k p_k$, $\lambda_{k+1} := \min\{1, \lambda_k/q\}$, $k := k+1$
  `end`
`until (` Approximation x_{k+1} von x^* hinreichend genau `)`

Bemerkung. Wir betrachten den Fall $\lambda_{\min} = 0$. Führt die `while`-Schleife zur Bestimmung des Dämpfungsparameters auf eine Endlosschleife, so ist nach Lemma 7.7 die Jacobi-Matrix $DF(x_k)$ nicht regulär. Anderenfalls würde bewiesenermaßen irgendwann $\|F(x_k+\lambda_k p_k)\|_2 < \|F(x_k)\|_2$ gelten. Für $d = 1$ konvergiert unser gedämpftes Newton-Verfahren also entweder gegen eine Nullstelle x^* von F oder gegen ein x^* mit $F'(x^*) = 0$, d.h. gegen ein Extremum bzw. einen Wendepunkt. □

Bemerkung. Da die Berechnung von $DF(x_k)$ im Extremfall auf Aufwand $\mathcal{O}(n^2)$ führt, interessiert man sich manchmal für Verfahren, die zwar langsamer konvergieren, aber in jedem Iterationsschritt numerisch günstiger sind.

- **Vereinfachte Newton-Verfahren (engl. *fast frozen Newton*).** Die Jacobi-Matrix $DF(x_k)$ wird nicht mehr in allen Iterationsschritten berechnet. Im Extremfall betrachten wir also

  $$x_n := x_{n-1} - DF(x_0)^{-1} F(x_{n-1}) \quad \text{für } n \in \mathbb{N}.$$

  Eine Konvergenzanalysis findet sich beispielsweise in Heuser [4, Kapitel 189].

- **Quasi-Newton-Verfahren.** Man ersetzt die Matrix $DF(x_n)$ durch eine billigere Approximation B_n, d.h.

$$x_n := x_{n-1} - B_{n-1}^{-1} F(x_{n-1}) \quad \text{für } n \in \mathbb{N}.$$

Beispiele hierfür sind das Sekantenverfahren in $\mathbb{R}$ oder das Broyden-Verfahren in $\mathbb{R}^d$, vgl. STOER [7, Kapitel 5.4.3]. □

7.3 Stationäre Iterationsverfahren zur Lösung linearer GLS

In diesem Abschnitt sei $A \in \mathbb{K}^{n\times n}$ eine reguläre Matrix, $b \in \mathbb{K}^n$ und $x^* \in \mathbb{K}^n$ die exakte Lösung von $Ax^* = b$. Bei **stationären linearen Iterationsverfahrens** konstruiert man eine Matrix $M \in \mathbb{K}^{n\times n}$ und einen Vektor $c \in \mathbb{K}^n$, sodass x^* der eindeutige Fixpunkt der Iterationsvorschrift $\Phi(x) := Mx + c$ ist. Mit dem Banach'schen Fixpunktsatz zeigt man, dass das Iterationsverfahren $(\mathbb{R}^n, \Phi, x^*)$ genau dann global linear konvergiert, wenn für den Spektralradius $\rho(M) < 1$ gilt, siehe Satz 7.9. Bei einem **instationären linearen Iterationsverfahren** variieren die Matrix M und/oder der Vektor c in jedem Iterationsschritt.

Bemerkung. (i) Die iterative Lösung eines linearen Gleichungssystems ist dann nötig, wenn A zwar schwach besetzt ist (d.h. vorwiegend Null-Einträge hat), aber A keine weitere erkennbare Struktur besitzt (z.B. Bandstruktur), sodass das direkte Lösen mittels Eliminationsverfahren auf einen Aufwand $\mathcal{O}(n^3)$ führt. Dies ist bereits für $n = 10^4$ wenig akzeptabel. Man beachte allerdings, dass moderne Standard-PCs (z.B. Pentium 4 mit 3GHz Taktung) ca. $6 \cdot 10^9$ arithmetische Gleitkommaoperationen pro Sekunde leisten, d.h. vom Rechenaufwand her wird ein Gleichungssystem mit $n = 1000 = 10^3$ Unbekannten in ca. 1 Sekunde gelöst.
(ii) Iterative Verfahren liefern in der Regel „zuverlässigere" Lösungen von Gleichungssystemen mit schlechter Kondition, da Rechenfehler *nicht* stören, solange die berechneten Approximationen im Konvergenzbereich des Verfahrens bleiben. Man sagt deshalb auch, dass iterative Verfahren *sich selbst stabilisieren.* □

Beispiel (Richardson-Iteration). Bei der Richardson-Iteration ist $M = \boldsymbol{I} - \lambda A$ und $c = \lambda b$, wobei der Parameter $\lambda \in \mathbb{K}$ geeignet gewählt werden muss, sodass $\rho(M) < 1$ gilt. ■

Übung. Man überlege sich ein möglichst einfaches Beispiel A, sodass die Iterationsmatrix M der Richardson-Iteration unabhängig von λ stets $\rho(M) \geq 1$ erfüllt. In diesem Fall ist nach Satz 7.9 im allgemeinen mit der Divergenz des Verfahrens zu rechnen. ■

Der Nachteil bei der Richardson-Iteration ist der, dass man $\rho(\boldsymbol{I} - \lambda A) < 1$ de facto nicht überprüfen kann. In diesem Abschnitt wollen wir daher noch weitere Verfahren betrachten, bei denen man leichter sehen kann, ob das Verfahren konvergiert oder nicht.

Im gesamten Abschnitt bezeichne $D \in \mathbb{K}^{n\times n}$ die Diagonale von A,

$$d_{jk} = \begin{cases} a_{jj} & \text{für } j = k, \\ 0 & \text{für } j \neq k, \end{cases}$$

$L \in \mathbb{K}^{n\times n}$ die Einträge unterhalb der Diagonale

$$\ell_{jk} = \begin{cases} a_{jk} & \text{für } j > k, \\ 0 & \text{für } j \le k, \end{cases}$$

$U \in \mathbb{K}^{n\times n}$ die Einträge oberhalb der Diagonale

$$u_{jk} = \begin{cases} a_{jk} & \text{für } j < k, \\ 0 & \text{für } j \le k. \end{cases}$$

Es gilt also insbesondere $A = D + L + U$.

Beispiel. Bei der **Jacobi-Iteration** ist $M := -D^{-1}(A - D)$, $c := D^{-1}b$. Dann gilt

$$b = Ax = Dx + (A - D)x \quad \Longleftrightarrow \quad Dx = -(A - D)x + b,$$

d.h. x^* ist der eindeutige Fixpunkt von $\Phi(x) = Mx + c$. Da D eine Diagonalmatrix ist, hängt $x_j^{(\ell+1)}$ von $x_1^{(\ell)}, \dots, x_n^{(\ell)}$ ab. Man nennt die Jacobi-Iteration deshalb auch **Gesamtschrittverfahren**. Ein Iterationsschritt besteht also aus

- Berechnung von $y = -(A - D)x + b$ in $\mathcal{O}(n^2)$ arithmetischen Operationen,
- direktes Lösen des Gleichungssystems $Dx = y$ in $\mathcal{O}(n)$ arithmetischen Operationen.

Jeder Iterationsschritt erfordert also $\mathcal{O}(n^2)$ arithmetische Operationen. ■

Beispiel. Bei der **Gauß-Seidel-Iteration** ist $M := -(L + D)^{-1}U$, $c := (L + D)^{-1}b$. Dann gilt

$$b = Ax = (L + D)x + Ux \quad \Longleftrightarrow \quad (L + D)x = -Ux + b,$$

d.h. x^* ist der eindeutige Fixpunkt von $\Phi(x) = Mx + c$. Da $L + D$ eine untere Dreiecksmatrix ist (Vorwärtsubstitution!), hängt $x_j^{(\ell+1)}$ von $x_1^{(\ell+1)}, \dots, x_{j-1}^{(\ell+1)}, x_j^{(\ell)}, \dots, x_n^{(\ell)}$ ab. Man spricht deshalb auch vom **Einzelschrittverfahren**. Ein Iterationsschritt besteht aus

- Berechnung von $y = -Ux + b$ in $\mathcal{O}(n^2)$ arithmetischen Operationen,
- direktes Lösen des Gleichungssystems $(L + D)x = y$ in $\mathcal{O}(n^2)$ arithmetischen Operationen.

Ein Iterationsschritt erfordert also wieder nur $\mathcal{O}(n^2)$ arithmetische Operationen. ■

7.3.1 Konvergenz der stationären linearen Iteration

Der folgende Satz charakterisiert die Konvergenz von stationären linearen Iterationsverfahren und zeigt insbesondere sofort, dass wir für die Richardson-Iteration $\rho(\boldsymbol{I} - \lambda A) < 1$ benötigen, um Konvergenz erwarten zu können.

Satz 7.9. *Für eine Matrix $M \in \mathbb{K}^{n\times n}$ sind die folgenden beiden Aussagen äquivalent:*
(i) *Der Spektralradius erfüllt $\rho(M) < 1$,*
(ii) *Für alle $c \in \mathbb{K}^n$ und $\Phi(x) := Mx + c$ ist das Iterationsverfahren $(\mathbb{K}^n, \Phi, x^*)$ global linear konvergent und hat einen eindeutigen Fixpunkt x^*.*

***Beweis von Satz 7.9,* (ii) $\Longrightarrow$ (i) *für* $\mathbb{K} = \mathbb{C}$.** Wir definieren die Iterationsvorschrift $\Phi(x) := Mx + c$. Es sei $\|\cdot\|$ eine beliebige Norm auf $\mathbb{C}^n$. Wähle den Startwert so, dass $x_0 - x^*$ Eigenvektor zum Eigenwert $\lambda \in \mathbb{C}$ ist mit $|\lambda| = \rho(M)$. Induktiv zeigt man, dass gilt

$$x_k - x^* = \Phi(x_{k-1}) - \Phi(x^*) = M(x_{k-1} - x^*) = M^k(x_0 - x^*) = \lambda^k(x_0 - x^*).$$

Nach Voraussetzung gilt deshalb

$$\rho(M)^k\|x_0 - x^*\| = |\lambda^k|\|x_0 - x^*\| = \|x_k - x^*\| \to 0 \quad \text{für } k \to \infty,$$

und es folgt $\rho(M) < 1$. ■

***Beweis von Satz 7.9,* (ii) $\Longrightarrow$ (i) *für* $\mathbb{K} = \mathbb{R}$.** Wir müssen nur zeigen, dass die Voraussetzung (ii) auch für $\mathbb{C}$ gilt. Dazu sei $c \in \mathbb{C}^n$ gegeben und $x_0 = a_0 + ib_0 \in \mathbb{C}^n$ ein beliebiger Startvektor mit $a_0, b_0 \in \mathbb{R}^n$. Es gilt $a_k := \operatorname{Re} x_k = M(\operatorname{Re} x_{k-1}) + \operatorname{Re} c$ sowie $b_k := \operatorname{Im} x_k = M(\operatorname{Im} x_{k-1}) + \operatorname{Im} c$, da die Einträge von M reell sind. Nach Voraussetzung konvergieren die reellen Folgen $a_k, b_k \in \mathbb{R}^n$ unabhängig von a_0, b_0 gegen eindeutige Grenzwerte $a, b \in \mathbb{R}^n$. Insgesamt konvergiert also x_k gegen den eindeutigen Grenzwert $x^* = a + ib \in \mathbb{C}^n$ unabhängig von x_0, und es gilt $x^* = Mx^* + c$. ■

Der Beweis der Implikation (i) $\Longrightarrow$ (ii) benötigt neben dem Banachschen Fixpunktsatz noch eine Charakterisierung des Spektralradius.

Lemma 7.10. *Für jede Matrix $M \in \mathbb{K}^{n\times n}$ gilt*

$$\rho(M) = \inf\big\{\|M\| \,\big|\, \|\cdot\| \text{ Operatornorm, die von einer Norm auf } \mathbb{C}^n \text{ induziert wird}\big\}.$$

Beweis. Um $\leq$ zu beweisen, sei $\lambda \in \mathbb{C}$ ein Eigenwert mit $|\lambda| = \rho(M)$ und $x \in \mathbb{K}^n$ ein zugehöriger Eigenvektor. Dann gilt für jede Norm $\|\cdot\|$ auf $\mathbb{C}^n$

$$\rho(M)\|x\| = \|Mx\| \leq \|M\|\|x\|.$$

Die Abschätzung $\geq$ ist weniger elementar. Nach Linearer Algebra ist jede Matrix über $\mathbb{C}$ trigonalisierbar, d.h. es existiert eine reguläre Matrix $T \in \mathbb{C}^{n\times n}$, sodass gilt

$$R := T^{-1}MT = \begin{pmatrix} r_{11} & r_{12} & \dots & r_{1n} \\ 0 & r_{22} & \dots & r_{2n} \\ \vdots & \ddots & \ddots & \vdots \\ 0 & \dots & 0 & r_{nn} \end{pmatrix}.$$

Da R und M ähnlich sind, haben sie dieselben Eigenwerte. Da die Eigenwerte von R auf der Diagonalen stehen, gilt insbesondere

$$\rho(M) = \rho(R) = \max_{j=1,\dots,n} |r_{jj}|.$$

Zu $\varepsilon > 0$ definieren wir die Diagonalmatrix $D_\varepsilon := \operatorname{diag}(1, \varepsilon, \dots, \varepsilon^{n-1}) \in \mathbb{R}^{n\times n}$ und eine zugehörige Norm $\|x\|_\varepsilon := \|D_\varepsilon^{-1}T^{-1}x\|_\infty$ auf $\mathbb{C}^n$. Die induzierte Operatornorm auf $\mathbb{C}^{n\times n}$ erfüllt gerade $\|M\|_\varepsilon = \|D_\varepsilon^{-1}T^{-1}MTD_\varepsilon\|_\infty = \|D_\varepsilon^{-1}RD_\varepsilon\|_\infty$ und lässt sich daher als Zeilensummennorm berechnen. Mit

$$D_\varepsilon^{-1}RD_\varepsilon = \begin{pmatrix} r_{11} & \varepsilon r_{12} & \dots & \varepsilon^{n-1}r_{1n} \\ 0 & r_{22} & \dots & \varepsilon^{n-2}r_{2n} \\ \vdots & \ddots & \ddots & \vdots \\ 0 & \dots & 0 & r_{nn} \end{pmatrix}$$

erhalten wir

$$\|M\|_\varepsilon = \max_{j=1,\dots,n} \left(|r_{jj}| + \sum_{k=j+1}^{n} \varepsilon^{k-j} |r_{jk}| \right) \xrightarrow{\varepsilon \to 0} \max_{j=1,\dots,n} |r_{jj}| = \rho(M).$$

Der Grenzübergang für $\varepsilon \to 0$ beschließt den Beweis. ■

Übung. Es seien $D \in \mathbb{K}^{n\times n}$ eine reguläre Matrix, und $\|\cdot\|$ notiere sowohl eine Norm auf $\mathbb{K}^n$ als auch die induzierte Operatornorm. Dann definiert auch $\|x\|_D := \|Dx\|$ eine Norm auf $\mathbb{K}^n$, und die induzierte Operatornorm erfüllt $\|M\|_D = \|DMD^{-1}\|$. ■

***Beweis von Satz 7.9,* (i) $\Longrightarrow$ (ii).** Wegen $\rho(M) < 1$ existiert nach Lemma 7.10 eine induzierte Operatornorm mit $\|M\| < 1$. Damit definiert $\Phi(x) := Mx + c$ eine Kontraktion auf $\mathbb{K}^n$, denn $\|\Phi(x) - \Phi(y)\| = \|Mx - My\| \le \|M\|\|x - y\|$. Die Behauptung folgt aus dem Banachschen Fixpunktsatz 7.1. ■

7.3.2 Konvergenz von Jacobi- und Gauß-Seidel-Iteration

Um die Konvergenz der Richardson-Iteration mathematisch zu beweisen, muss der Spektralradius der Iterationsmatrix bestimmt werden. Dies ist im Allgemeinen nicht analytisch möglich und numerisch sehr aufwändig. Es ist daher das Ziel, aus einfacheren algebraischen Eigenschaften der Matrix A auf die Konvergenz der Jacobi- bzw Gauß-Seidel-Iteration zu schließen. Anders als in Satz 7.9 sind die Ergebnisse in diesem Abschnitt hinreichende Konvergenzbedingungen, i.a. aber nicht notwendig.

Definition. Eine Matrix $A \in \mathbb{K}^{n\times n}$ ist **diagonaldominant**, falls

$$\sum_{\substack{k=1 \\ k\neq j}}^{n} |a_{jk}| \le |a_{jj}| \tag{7.20}$$

für alle Zeilen $j = 1, \dots, n$ gilt, wobei die Ungleichung für mindestens einen Index $1 \le j_0 \le n$ strikt ist, d.h. es gilt $<$ in (7.20). Die Matrix A ist **strikt diagonaldominant**, falls die strikte Ungleichung $<$ in (7.20) für alle Indizes $j = 1, \dots, n$ gilt. □

Satz 7.11. *Die Matrix $A \in \mathbb{K}^{n\times n}$ sei strikt diagonaldominant. Dann ist A bijektiv, und Jacobi-Verfahren und Gauß-Seidel-Verfahren sind wohldefiniert und konvergent. Bezeichnen $M^{(J)}, M^{(GS)} \in \mathbb{K}^{n\times n}$ die Verfahrensmatrizen für das Jacobi- bzw. das Gauß-Seidel-Verfahren, so gilt für die Zeilensummennormen $\|M^{(GS)}\|_\infty \le \|M^{(J)}\|_\infty < 1$ und damit sowohl $\rho(M^{(GS)}) < 1$ als auch $\rho(M^{(J)})$.*

Beweis. Falls A strikt diagonaldominant ist, gilt insbesondere $a_{jj} \neq 0$ für alle $j = 1, \dots, n$, und deshalb sind die Matrizen $M^{(J)} = -D^{-1}(A - D)$ und $M^{(GS)} = -(L + D)^{-1}U$ wohldefiniert. Ferner gilt nach Definition von $M^{(J)}$

$$\sum_{\substack{k=1 \\ j\neq k}}^{n} |M^{(J)}_{jk}| = \sum_{\substack{k=1 \\ j\neq k}}^{n} \frac{|a_{jk}|}{|a_{jj}|} < 1 \quad \text{für alle } j = 1, \dots, n.$$

Mit der Zeilensummenorm folgt $\rho(M^{(J)}) \leq \|M^{(J)}\|_\infty < 1$ und daraus die Konvergenz des Jacobi-Verfahrens. Insgesamt ist nun nur noch zu zeigen, dass $\|M^{(GS)}\|_\infty \leq \|M^{(J)}\|_\infty$ gilt. Wir zeigen dies in mehreren Schritten.

1. Schritt. *Ist $M \in \mathbb{K}^{n\times n}$ mit $\rho(M) < 1$, so ist $\boldsymbol{I} - M$ regulär, und die Inverse lässt sich über die Neumann'sche Reihe darstellen, $(\boldsymbol{I} - M)^{-1} = \sum_{k=0}^{\infty} M^k$.* Wegen $\rho(M) < 1$ existiert eine induzierte Operatornorm mit $\|M\| < 1$, und

$$\Big\| \sum_{k=i}^{j} M^k \Big\| \leq \sum_{k=i}^{j} \|M\|^k \xrightarrow{i,j\to\infty} 0$$

zeigt die Existenz von $\sum_{k=0}^{\infty} M^k$. Multiplikation der Reihe mit $(\boldsymbol{I} - M)$ beweist

$$(\boldsymbol{I} - M) \sum_{k=0}^{\infty} M^k = \sum_{k=0}^{\infty} M^k - \sum_{k=1}^{\infty} M^k = \boldsymbol{I}.$$

Hieraus folgt die Behauptung. □

Im Folgenden bezeichne $|\cdot|$ den komponentenweisen Betrag von Matrizen und Vektoren, und auch $\leq$ sei komponentenweise verstanden.

2. Schritt. *Es gilt $\rho(D^{-1}L) = 0 = \rho(|D^{-1}L|)$.* Die Matrizen $D^{-1}L$ und $|D^{-1}L|$ sind Dreiecksmatrizen mit trivialer Diagonale. Da die Eigenwerte einer Dreiecksmatrix auf der Diagonalen stehen, folgt die Behauptung. □

3. Schritt. *Es gelten $|(\boldsymbol{I} + D^{-1}L)^{-1}| \leq (\boldsymbol{I} - |D^{-1}L|)^{-1}$ sowie $\boldsymbol{I} \leq (\boldsymbol{I} - |D^{-1}L|)^{-1}$.* Aufgrund von Schritt 2 können wir die Inverse als Neumann'sche Reihen darzustellen. Es gelten

$$|(\boldsymbol{I} + D^{-1}L)^{-1}| = \Big| \sum_{k=0}^{\infty} (-D^{-1}L)^k \Big| \leq \sum_{k=0}^{\infty} |D^{-1}L|^k = (\boldsymbol{I} - |D^{-1}L|)^{-1}$$

sowie

$$(\boldsymbol{I} - |D^{-1}L|)^{-1} = \boldsymbol{I} + \sum_{k=1}^{\infty} |D^{-1}L|^k \geq \boldsymbol{I}.$$ □

4. Schritt. *Wegen $\|M^{(J)}\|_\infty \leq 1$ gilt $\|M^{(GS)}\|_\infty \leq \|M^{(J)}\|_\infty$.* Mit dem Vektor $\mathbf{e} = (1, \ldots, 1) \in \mathbb{K}^n$ gilt $\|M\|_\infty = \||M|\mathbf{e}\|_\infty$. Also ist $|M^{(GS)}|\mathbf{e} \leq \|M^{(J)}\|_\infty \mathbf{e}$ zu zeigen. Da die Matrizen L und U keine gemeinsamen nicht-trivialen Einträge haben, gilt

$$|M^{(J)}| = |D^{-1}L + D^{-1}U| = |D^{-1}L| + |D^{-1}U|.$$

Ferner gilt $D + L = D(\boldsymbol{I} + D^{-1}L)$, also

$$(D + L)^{-1} = (\boldsymbol{I} + D^{-1}L)^{-1} D^{-1}.$$

Zusammen mit der komponentenweisen Abschätzung $|ST| \leq |S||T|$ und Schritt 3 erhalten wir

$$\begin{aligned} |M^{(GS)}| = |(D + L)^{-1}U| &= |(\boldsymbol{I} + D^{-1}L)^{-1} D^{-1}U| \\ &\leq |(\boldsymbol{I} + D^{-1}L)^{-1}||D^{-1}U| \\ &\leq (\boldsymbol{I} - |D^{-1}L|)^{-1} \big[(|M^{(J)}| - \boldsymbol{I}) + (\boldsymbol{I} - |D^{-1}L|) \big] \\ &= \boldsymbol{I} + (\boldsymbol{I} - |D^{-1}L|)^{-1} (|M^{(J)}| - \boldsymbol{I}). \end{aligned}$$

Deshalb erhalten wir

$$\begin{aligned}|M^{(GS)}|\mathbf{e} \leq \mathbf{e} + (\boldsymbol{I} - |D^{-1}L|)^{-1}(|M^{(J)}| - \boldsymbol{I})\mathbf{e} &\leq \mathbf{e} + (\boldsymbol{I} - |D^{-1}L|)^{-1}(\|M^{(J)}\|_\infty \mathbf{e} - \mathbf{e}) \\ &= \mathbf{e} + (\|M^{(J)}\|_\infty - 1)(\boldsymbol{I} - |D^{-1}L|)^{-1}\mathbf{e} \\ &\leq \mathbf{e} + (\|M^{(J)}\|_\infty - 1)\mathbf{e} \\ &= \|M^{(J)}\|_\infty \mathbf{e},\end{aligned}$$

wobei wir in der letzten Abschätzung $\|M^{(J)}\|_\infty \leq 1$ und $(\boldsymbol{I} - |D^{-1}L|)^{-1} \geq \boldsymbol{I}$ ausgenutzt haben. Abschließend folgt also $\|M^{(GS)}\|_\infty = \||M^{(GS)}|\mathbf{e}\|_\infty \leq \|M^{(J)}\|_\infty$. ■

Satz 7.12. *Ist A^T strikt diagonaldominant, so ist das Jacobi-Verfahren wohldefiniert und konvergent.*

Beweis. Analog zum vorherigen Beweis gilt für die transponierte Verfahrensmatrix $\rho(M^{(J)}) \leq \|M^{(J)}\|_1 = \|(M^{(J)})^T\|_\infty < 1$, denn die Spaltensummennorm von $M^{(J)}$ ist gerade die Zeilensummennorm der transponierten Matrix. ■

Die strikte Diagonaldominanz ist eine starke Voraussetzung an Matrizen, die bereits von den Matrizen bei der Spline-Interpolation verletzt wird. Diese sind aber wenigstens diagonaldominant und ferner irreduzibel.

Definition. Eine Matrix $A \in \mathbb{K}^{n\times n}$ heißt **reduzibel**, falls es nichtleere Indexmengen J, K gibt mit $J \cap K = \emptyset$, $J \cup K = \{1, \ldots, n\}$ sowie $a_{jk} = 0$ für alle $j \in J$ und $k \in K$. Eine Matrix heißt **irreduzibel**, falls sie nicht reduzibel ist. □

Beispiel. Die Matrix $\begin{pmatrix} 1 & 2 & 0 \\ 3 & 4 & 0 \\ 5 & 6 & 7 \end{pmatrix}$ ist reduzibel mit $J = \{1, 2\}$ und $K = \{3\}$. ■

Bemerkung. Ist eine reguläre Matrix $A \in \mathbb{K}^{n\times n}$ reduzibel, so lässt sich das Lösen des Gleichungssystems $Ax = b$ in zwei Schritte aufteilen:

- Im ersten Schritt lösen wir für $j \in J$ das Gleichungssystem $b_j = \sum_{\ell=1}^n a_{j\ell}x_\ell = \sum_{\ell\in J} a_{j\ell}x_\ell$.
- Für $k \in K$ gilt $b_k = \sum_{\ell\in J} a_{k\ell}x_\ell + \sum_{\ell\in K} a_{k\ell}x_\ell$. Da die Summanden der ersten Summe alle bekannt sind, lösen wir im zweiten Schritt $b_k - \sum_{\ell\in J} a_{k\ell}x_\ell = \sum_{\ell\in K} a_{k\ell}x_\ell$ für alle $k \in K$. □

Übung. Die Matrix $A \in \mathbb{R}^{n\times n}$ sei irreduzibel und diagonaldominant, und für die Diagonalelemente gelte $a_{jj} > 0$ für $j = 1, \ldots, n$. Dann gilt $\operatorname{Re}\lambda > 0$ für alle (komplexen) Eigenwerte λ von A. Ist A zusätzlich symmetrisch, so ist A positiv definit. ■

Satz 7.13. *Ist $A \in \mathbb{K}^{n\times n}$ irreduzibel und diagonaldominant, so sind Jacobi- und Gauß-Seidel-Verfahren wohldefiniert und konvergent.*

Zum Beweis von Satz 7.13 werden wir direkt den Spektralradius der beiden Verfahrensmatrizen abschätzen. Die wesentliche Beobachtung dabei ist, dass jede irreduzible und diagonaldominante Matrix bereits regulär ist.

Lemma 7.14. *Jede irreduzible und diagonaldominante Matrix $M \in \mathbb{K}^{n\times n}$ ist regulär und erfüllt $m_{jj} \neq 0$ für alle $j = 1, \ldots, n$.*

Beweis. Wir nehmen an, M sei nicht regulär, d.h. es existiert ein $x \in \mathbb{K}^n \backslash \{0\}$ mit $Mx = 0$. Insbesondere folgt aus der Dreiecksungleichung

$$|m_{jj}||x_j| \leq \sum_{\substack{\ell=1 \\ \ell\neq j}}^{n} |m_{j\ell}||x_\ell| \quad \text{für alle } j = 1, \ldots, n. \tag{7.21}$$

Wir definieren die Indexmengen $J := \big\{ j \,\big|\, |x_j| = \|x\|_\infty \big\}$ und $K := \big\{ k \,\big|\, |x_k| < \|x\|_\infty \big\}$. Offensichtlich gilt $J \cap K = \emptyset$, $J \cup K = \{1, \ldots, n\}$ und $J \neq \emptyset$. Wäre $K = \emptyset$, so könnte man in (7.21) die x_j- und x_ℓ-Terme herausstreichen und erhielte einen Widerspruch zur Diagonaldominanz. Also gilt auch $K \neq \emptyset$, und aufgrund der Irreduzibilität von M existieren Indizes $j \in J$ und $k \in K$ mit $m_{jk} \neq 0$. Mit diesen ergibt sich

$$|m_{jj}| \leq \sum_{\substack{\ell=1 \\ \ell\neq j}} |m_{j\ell}| \frac{|x_\ell|}{|x_j|} < \sum_{\substack{\ell=1 \\ \ell\neq j}} |m_{j\ell}|,$$

denn der Quotient ist stets ≤ 1 wegen $|x_j| = \|x\|_\infty$ und < 1 für $\ell = k \in K$. Also erhalten wir einen Widerspruch zur Diagonaldominanz von M, d.h. M ist regulär. Gäbe es schließlich ein triviales Diagonalelement $m_{jj} = 0$, so folgte aus der Diagonaldominanz, dass bereits die j-te Zeile die Nullzeile wäre. Da M regulär ist, folgt insbesondere $m_{jj} \neq 0$ für alle $j = 1, \ldots, n$. ■

Beweis von Satz 7.13 für das Jacobi-Verfahren. Wegen $a_{jj} \neq 0$ für alle $j = 1, \ldots, n$ ist $M^{(J)} = -D^{-1}(A-D)$ wohldefiniert. Um $\rho(M^{(J)}) < 1$ zu zeigen, beweisen wir, dass $M := M^{(J)} - \lambda \boldsymbol{I}$ für $\lambda \in \mathbb{C}$ mit $|\lambda| \geq 1$ regulär ist. Da A irreduzibel ist, ist auch $A - D$ irreduzibel, denn es wird lediglich die Diagonale verändert. $M^{(J)}$ entsteht durch zeilenweise Multiplikation von $A - D$ mit Werten $\neq 0$. Deshalb ist auch $M^{(J)}$ irreduzibel. Da M und $M^{(J)}$ sich nur auf der Diagonale unterscheiden, ist M irreduzibel. Aufgrund der Diagonaldominanz von A gilt

$$\sum_{\substack{k=1 \\ j\neq k}}^{n} |m_{jk}| = \sum_{\substack{k=1 \\ j\neq k}}^{n} |m_{jk}^{(J)}| = \sum_{\substack{k=1 \\ j\neq k}}^{n} \frac{|a_{jk}|}{|a_{jj}|} \leq 1 \leq |\lambda| = |m_{jj}| \quad \text{für alle } j = 1, \ldots, n,$$

und für mindestens einen Index j gilt diese Ungleichung strikt. Also ist M auch diagonaldominant und nach Lemma 7.14 insgesamt regulär. ■

Beweis von Satz 7.13 für das Gauß-Seidel-Verfahren. Die Wohldefiniertheit von $M^{(GS)} = -(L+D)^{-1}U$ ist wieder klar. Wir betrachten $M := M^{(GS)} - \lambda \boldsymbol{I}$ für $\lambda \in \mathbb{C}$ mit $|\lambda| \geq 1$. Durch Multiplikation mit $-(L+D)$ sieht man, dass M genau dann regulär ist, wenn $\widetilde{M} := U + \lambda L + \lambda D$ regulär ist. Offensichtlich erbt $\widetilde{M}$ die Irreduzibilität von $A = D + L + U$. Ferner ist $\widetilde{M}$ diagonaldominant,

denn es gilt

$$\sum_{\substack{k=1\\k\neq j}}^{n} |\widetilde{m}_{jk}| = |\lambda| \sum_{k=1}^{j-1} |a_{jk}| + \sum_{k=j+1}^{n} |a_{jk}| \leq |\lambda| \sum_{\substack{k=1\\k\neq j}}^{n} |a_{jk}| \leq |\lambda||a_{jj}| = |\widetilde{m}_{jj}| \quad \text{für } j = 1, \ldots, n$$

mit strikter Ungleichung für mindestens einen Index j. Nach Lemma 7.14 ist $\widetilde{M}$ regulär. Insgesamt erhalten wir wie zuvor $\rho(M^{(GS)}) < 1$. ■

7.4 Krylov-Verfahren zur Lösung linearer GLS

Im gesamten Abschnitt sei $A \in \mathbb{K}^{n\times n}$ eine reguläre Matrix, $b \in \mathbb{K}^n$ und $x^* \in \mathbb{K}^n$ mit $Ax^* = b$.

Bemerkung. Es kommt häufig vor, dass die Matrix A in einem komprimierten Datenformat vorliegt, das keinen schnellen Zugriff auf auf die Einträge A_{jk} erlaubt, dennoch aber z.B. die Matrix-Vektor-Multiplikation in $\mathcal{O}(n\log(n))$ Operationen durchführen lässt. In diesem Fall verbieten sich die Jacobi- und die Gauß-Seidel-Iteration, bei denen ja in jedem Schritt ein einfacheres lineares Problem direkt gelöst werden muss. Man interessiert sich deshalb für iterative Löser, die lediglich von der Matrix-Vektor-Multiplikation Gebrauch machen, aber keine explizite Kenntnis der Einträge A_{jk} erfordern. □

Definition. Für $\ell \in \mathbb{N}$ bezeichnet man die Vektorräume

$$\mathcal{K}_\ell := \mathcal{K}_\ell(A, b) := \operatorname{span}\{b, Ab, A^2b, \ldots, A^{\ell-1}b\} \subseteq \mathbb{K}^n$$

als **Krylov-Räume** bezüglich (A, b). □

Lemma 7.15. *Für $\ell \in \mathbb{N}$ sind äquivalent:*
(i) $\dim \mathcal{K}_{\ell+1} \leq \ell$,
(ii) $\mathcal{K}_\ell = \mathcal{K}_{\ell+1}$,
(iii) $A(\mathcal{K}_\ell) \subseteq \mathcal{K}_\ell$,
(iv) $x^* \in \mathcal{K}_\ell$.
Insbesondere existiert also ein $\ell \in \mathbb{N}$ mit $x^* \in \mathcal{K}_\ell$. Definiere $\ell^* = \min\left\{\ell \in \mathbb{N} \,\middle|\, x^* \in \mathcal{K}_\ell\right\}$.

Beweis. (i)⇒(ii): Wir wählen den minimalen Index $m < \ell$, sodass die Menge $\{b, Ab, \ldots, A^m b\}$ linear abhängig ist. Dann existieren Skalare $\lambda_j \in \mathbb{K}$ mit $A^m b = \sum_{j=0}^{m-1} \lambda_j A^j b$. Insbesondere folgt also

$$A^\ell b = A^{\ell-m}(A^m b) = \sum_{j=0}^{m-1} \lambda_j A^{\ell-m+j} b \in \operatorname{span}\{A^{\ell-m}b, \ldots, A^{\ell-1}b\} \subseteq \mathcal{K}_\ell.$$

(ii)⇒(iii): Für $j = 0, \ldots, \ell-1$ gilt $A(A^j b) \in \mathcal{K}_{\ell+1} = \mathcal{K}_\ell$. Da $\mathcal{K}_\ell$ durch diese Vektoren aufgespannt wird, folgt $A(\mathcal{K}_\ell) \subseteq \mathcal{K}_\ell$.

(iii)⇒(iv): Die Abbildung $A : \mathcal{K}_\ell \to \mathcal{K}_\ell$ ist wohldefiniert, linear und injektiv. Insbesondere ist $A : \mathcal{K}_\ell \to \mathcal{K}_\ell$ bijektiv. Wegen $Ax^* = b \in \mathcal{K}_\ell$ folgt $x^* \in \mathcal{K}_\ell$.

(iv)⇒(i): Es gilt $x^* \in \mathcal{K}_\ell = \operatorname{span}\{b, Ab, A^2b, \dots, A^{\ell-1}b\}$, also $b = Ax^* \in \operatorname{span}\{Ab, A^2b, \dots, A^\ell b\}$. Insbesondere ist also $\{b, Ab, A^2b, \dots, A^\ell b\}$ linear abhängig, und es folgt $\dim \mathcal{K}_{\ell+1} \leq \ell$. ■

Das folgende Lemma ist ein Standardresultat für Hilbert-Räume und gilt grundsätzlich auch im Fall unendlich-dimensionaler Räume. Für endlich-dimensionale Hilbert-Räume kann der Beweis elementar geführt werden.

Lemma 7.16. *Es sei X ein endlich-dimensionaler Hilbert-Raum und Y ein Teilraum von X. Zu $x \in X$ existiert ein eindeutiges $Px \in Y$ mit*

$$\|x - Px\|_X = \min_{y \in Y} \|x - y\|_X. \tag{7.22}$$

Man bezeichnet $Px \in Y$ als **Bestapproximation von x in Y**. *Der Operator $P : X \to Y$ ist linear und erfüllt*

$$Py = y \quad \text{und} \quad \langle x \,;\, y\rangle = \langle Px \,;\, y\rangle \quad \text{für alle } y \in Y \text{ und } x \in X. \tag{7.23}$$

P heißt **Orthogonalprojektion auf Y**. *Ist $y_1, \dots, y_n$ eine Orthonormalbasis von Y, so gilt*

$$Px = \sum_{j=1}^{n} \langle x \,;\, y_j\rangle y_j. \tag{7.24}$$

Insbesondere gilt die **Parseval-Gleichung** $\|Px\|_X^2 = \sum_{j=1}^n |\langle x \,;\, y_j\rangle|^2$ *für alle $x \in X$.*

Beweis. Wir gliedern den Beweis lediglich in die einzelnen Schritte und überlassen deren Verifikation dem geneigten Leser:
1. Schritt. *Aufgrund eines Kompaktheitsschlusses gibt es ein Element $Px \in Y$, das* (7.22) *löst.*
2. Schritt. *Ein Element $Px \in Y$ erfüllt genau dann* (7.22)*, wenn gilt $\forall y \in Y \quad \langle x - Px \,;\, y\rangle = 0$.*
3. Schritt. *Das $Px \in Y$ mit* (7.22) *ist eindeutig, insbesondere ist $P : X \to Y$ wohldefiniert.*
4. Schritt. *Die Abbildung $P : X \to Y$ ist linear und besitzt die Eigenschaften* (7.23)*.*
5. Schritt. *Schreibt man Px als Linearkombination der y_j, so folgen* (7.24) *und insbesondere die Parseval-Gleichung.* ■

Bemerkung. Für eine SPD-Matrix $A \in \mathbb{K}^{n\times n}_{\mathrm{sym}}$ definiert

$$\langle x \,;\, y\rangle_A := \langle x \,;\, Ay\rangle_2 \quad \text{für } x, y \in \mathbb{K}^n \tag{7.25}$$

ein Skalarprodukt auf $\mathbb{K}^n$. Die zugehörige Norm ist $\|x\|_A = \langle x \,;\, x\rangle_A^{1/2}$. Das **CG-Verfahren** (*conjugate gradient method*) besteht darin, für $\ell \in \mathbb{N}$ die Bestapproximation $x_\ell \in \mathcal{K}_\ell(A, b)$ von x^* bezüglich $\|\cdot\|_A$ zu bestimmen, d.h.

$$\text{finde } x_\ell \in \mathcal{K}_\ell(A,b) \quad \text{mit} \quad \|x^* - x_\ell\|_A = \min_{y \in \mathcal{K}_\ell(A,b)} \|x^* - y\|_A. \tag{7.26}$$

Lemma 7.16 garantiert die eindeutige Existenz von x_ℓ. Die Berechnung von x_ℓ ist möglich, ohne x^* zu kennen: Dazu konstruieren wir mittels Gram-Schmidt-Orthogonalisierung in Satz 7.17 eine

Folge $d_0, \ldots, d_{\ell-1}$ von paarweise $\langle \cdot\,;\,\cdot \rangle_A$-orthogonalen Vektoren mit $\mathcal{K}_\ell(A,b) = \operatorname{span}\{d_0, \ldots, d_{\ell-1}\}$. Mit (7.24) folgt dann

$$x_\ell = \sum_{j=0}^{\ell-1} \frac{\langle x^* \,;\, d_j \rangle_A}{\langle d_j \,;\, d_j \rangle_A}\, d_j = \sum_{j=0}^{\ell-1} \frac{\langle b \,;\, d_j \rangle_2}{\langle d_j \,;\, d_j \rangle_A}\, d_j.$$

In dieser Formulierung sieht man, dass x_ℓ allein aus der Kenntnis von b berechnet werden kann, d.h. die unbekannte exakte Lösung x^* wird nicht benötigt. □

Übung. Es sei X ein endlich-dimensionaler Hilbert-Raum mit Basis $x_1, \ldots, x_n \in X$. Man zeige, dass dann die **Massenmatrix** $M \in \mathbb{K}^{n\times n}$, definiert durch $M_{jk} := \langle x_j \,;\, x_k \rangle$, eine SPD-Matrix ist.■

Bemerkung. Für eine reguläre Matrix $A \in \mathbb{K}^{n\times n}$ besteht das **CGNR-Verfahren** (*cg norm residual method*) in der Verwendung des CG-Verfahrens zur Lösung von $A^T A x^* = A^T b$. Es gilt

$$\|x^* - y\|_{A^TA}^2 = \langle x^* - y \,;\, A^TA(x^* - y) \rangle_2 = \langle A(x^* - y) \,;\, A(x^* - y) \rangle_2 = \|b - Ay\|_2^2,$$

und deshalb folgt für die Iteriertenfolge

$$\mathbf{x}_\ell \in \mathcal{K}_\ell(A^TA, A^Tb) \quad \text{mit} \quad \|b - Ax_\ell\|_2 = \min_{y \in \mathcal{K}_\ell(A^TA, A^Tb)} \|b - Ay\|_2. \tag{7.27}$$

Die Berechnung von x_ℓ erfordert also offensichtlich nicht die Kenntnis von x^*, sondern lediglich die der rechten Seite b. Die eindeutige Existenz von x_ℓ folgt aus dem CG-Verfahren. □

Bemerkung. Für eine reguläre Matrix $A \in \mathbb{K}^{n\times n}$ besteht das **GMRES-Verfahren** (*generalized minimal residual method*) darin,

$$x_\ell \in \mathcal{K}_\ell(A,b) \quad \text{mit} \quad \|b - Ax_\ell\|_2 = \min_{y \in \mathcal{K}_\ell(A,b)} \|b - Ay\|_2 \quad \text{für } \ell \in \mathbb{N}_0 \tag{7.28}$$

zu berechnen. Bei der Definition der Iteriertenfolge handelt es sich also in jedem Schritt um ein lineares Ausgleichsproblem. Da die Restriktion von A auf $\mathcal{K}_\ell(A,b)$ injektiv ist, können wir Satz 3.18 anwenden und erhalten die eindeutige Existenz von x_ℓ. □

Wegen $x^* \in \mathcal{K}_{\ell^*}$ berechnen alle drei Verfahren in ℓ^* Schritten die exakte Lösung $x_{\ell^*} = x^*$. Man kann diese sog. **Krylov-Verfahren** also nicht nur als iterative Verfahren, sondern auch als Eliminationsverfahren deuten. Im Folgenden betrachten wir nur das CG-Verfahren. Für das GMRES-Verfahren verweisen wir auf die Literatur, z.B. Plato [5, Abschnitte 11.6, 11.7, 11.8], oder auf die Vorlesung *Iterative Lösung großer Gleichungssysteme*.

Satz 7.17. *Es seien $A \in \mathbb{K}^{n\times n}_{\mathrm{sym}}$ eine SPD-Matrix und $x_\ell \in \mathcal{K}_\ell$ mit (7.26) für $\ell \geq 1$. Es bezeichne*

$$r_0 := b \quad \textit{und} \quad r_\ell := b - Ax_\ell \quad \textit{für } \ell \in \mathbb{N},\ \ell \leq \ell^*$$

das **Residuum**. *Offensichtlich gilt genau dann $r_\ell = 0$, wenn $x_\ell = x^*$ gilt, d.h. $\ell = \ell^*$. Dann ist $\{r_0, \ldots, r_{\ell-1}\}$ eine Basis von $\mathcal{K}_\ell$. Das Gram-Schmidt-Verfahren liefert eine Orthogonalbasis $\{d_0, \ldots, d_{\ell-1}\}$ von $\mathcal{K}_\ell$ bezüglich $\langle \cdot\,;\,\cdot \rangle_A$. Für $\ell < \ell^*$ gelten dann die folgenden Aussagen:*
(i) *$d_0 = b$, $d_{\ell+1} = r_{\ell+1} + \beta_\ell d_\ell$ mit $\beta_\ell := \|r_{\ell+1}\|_2^2 / \|r_\ell\|_2^2$.*
(ii) *$x_{\ell+1} = x_\ell + \alpha_\ell d_\ell$, $r_{\ell+1} = r_\ell - \alpha_\ell A d_\ell$ mit $\alpha_\ell := \|r_\ell\|_2^2 / \|d_\ell\|_A^2$.*

Bevor wir Satz 7.17 beweisen, formulieren wir unsere Erkenntnis in Form eines Algorithmus.

Algorithmus 7.18: CG-Verfahren

Input: SPD-Matrix $A \in \mathbb{K}^{n\times n}$, Vektor $b \in \mathbb{K}^n$

Definiere $r_0 := b$, $d_0 := b$, $x_0 := 0$, $\ell = 0$.
(1) Abbruch, falls $r_\ell = 0$.
(2) Definiere $\alpha_\ell := \|r_\ell\|_2^2/\|d_\ell\|_A^2, \quad x_{\ell+1} = x_\ell + \alpha_\ell d_\ell, \quad r_{\ell+1} = r_\ell - \alpha_\ell A d_\ell$.
(3) Definiere $\beta_\ell := \|r_{\ell+1}\|_2^2/\|r_\ell\|_2^2, \quad d_{\ell+1} := r_{\ell+1} + \beta_\ell d_\ell$.
(4) Update $\ell \mapsto \ell + 1$, Rücksprung nach (1).

Output: Lösung $x^* = x_\ell$ von $Ax^* = b$ sowie Index $\ell^* = \ell$.

Bemerkung. Algorithmus 7.18 bricht (mathematisch, *nicht* numerisch) nach endlich vielen Schritten ab und bestimmt dabei ℓ^*. Bei der Implementierung sollte die Bedingung $r_\ell = 0$ in (1) durch eine geeignete Abbruchbedingung ersetzt werden, z.B. $|r_\ell| \le \tau_{\text{abs}}$ mit gegebener absoluter Toleranz $\tau_{\text{abs}} > 0$. □

Bemerkung. Nach (i) und (v) aus Satz 7.17 gilt

$$\|x^* - x_{\ell+1}\|_A = \min_{\lambda\in\mathbb{K}} \|x^* - (x_\ell + \lambda d_\ell)\|_A,$$

d.h. die Schrittweite $\lambda = \alpha_\ell$ ist optimal. □

Beweis von Satz 7.17. Wir gliedern den Beweis der Übersicht halber wieder in mehrere Schritte.
1. Schritt. *Es gilt $\langle r_k\,;\,y\rangle_2 = 0$ für $y \in \mathcal{K}_k$. Insbesondere ist $\{r_0,\dots,r_{\ell-1}\}$ eine Orthogonalbasis von $\mathcal{K}_\ell$ bezüglich dem euklidischen Skalarprodukt.* Offensichtlich gelten $r_k \in \mathcal{K}_{k+1}$ und

$$\langle r_k\,;\,y\rangle_2 = \langle A(x^* - x_k)\,;\,y\rangle_2 = \langle x^* - x_k\,;\,y\rangle_A = 0 \quad \text{für all } y \in \mathcal{K}_k.$$

Insbesondere sind die r_k also paarweise ℓ_2-orthogonal und damit linear unabhängig. Wegen $\{r_0,\dots,r_{\ell-1}\} \subseteq \mathcal{K}_\ell$ und $\dim \mathcal{K}_\ell = \ell$ folgt die Behauptung. □
2. Schritt. *Gram-Schmidt-Orthogonalisierung von $\{r_0,\dots,r_{\ell-1}\}$ liefert eine $\langle\cdot\,;\,\cdot\rangle_A$-Orthogonalbasis $\{d_0,\dots,d_{\ell-1}\}$ von $\mathcal{K}_\ell$ mit $d_0 = r_0 = b$ und $d_{k+1} = r_{k+1} + \widetilde{\beta}_k d_k$, $\widetilde{\beta}_k := -\langle r_{k+1}\,;\,d_k\rangle_A/\|d_k\|_A^2$. Ferner gilt nach Definition des Gram-Schmidt-Verfahrens $\langle d_k\,;\,y\rangle_A = 0$ für $y \in \mathcal{K}_k$.* Nach Gram-Schmidt-Orthogonalisierung gelten $d_0 = r_0$ und

$$d_{k+1} = r_{k+1} - \sum_{j=0}^{k} \frac{\langle r_{k+1}\,;\,d_j\rangle_A}{\|d_j\|_A^2}\, d_j \quad \text{induktiv für } k \ge 0.$$

Wegen $r_j \in \mathcal{K}_{j+1}$ gilt induktiv $d_j \in \mathcal{K}_{j+1}$. Für $j \le k-1$ folgt $d_j \in \mathcal{K}_k$, also $Ad_j \in \mathcal{K}_{k+1}$ und mit Schritt 1 deshalb

$$\langle r_{k+1}\,;\,d_j\rangle_A = \langle r_{k+1}\,;\,Ad_j\rangle_2 = 0 \quad \text{für } j \le k-1,$$

d.h. die Summe reduziert sich auf den letzten Summanden für $j = k$.

3. Schritt. *Mit* $\widetilde{\alpha}_k := \langle x^* \,;\, d_k\rangle_A / \|d_k\|_A^2$ *gelten* $x_{k+1} = x_k + \widetilde{\alpha}_k d_k$ *und* $r_{k+1} = r_k - \widetilde{\alpha}_k A d_k$. Mit der Orthogonalbasis $\{d_0, \ldots, d_k\}$ von $\mathcal{K}_{k+1}$ können wir x_{k+1} explizit schreiben als

$$x_{k+1} = \sum_{j=0}^{k} \frac{\langle x^* \,;\, d_j\rangle_A}{\|d_j\|_A^2}\, d_j = x_k + \frac{\langle x^* \,;\, d_k\rangle_A}{\|d_k\|_A^2}\, d_k,$$

denn x_k besitzt dieselbe Darstellung mit Laufindex $j = 0, \ldots, k-1$. Dies zeigt $x_{k+1} = x_k + \widetilde{\alpha}_k d_k$, und nach Definition $r_{k+1} = b - Ax_{k+1}$ folgt unmittelbar $r_{k+1} = r_k - \widetilde{\alpha}_k A d_k$. □

Damit ist der Beweis de facto abgeschlossen. Es fehlt nur noch, die Gleichheiten $\alpha_k = \widetilde{\alpha}_k$ und $\beta_k = \widetilde{\beta}_k$ nachzurechnen.

4. Schritt. *Es gilt* $\widetilde{\alpha}_k := \langle x^* \,;\, d_k\rangle_A / \|d_k\|_A^2 = \|r_k\|_2^2 / \|d_k\|_A^2 =: \alpha_k$. Nach Schritt 2 gilt $d_k = r_k + \widetilde{\beta}_{k-1} d_{k-1}$. Wegen Schritt 1 und $d_{k-1} \in \mathcal{K}_k$ gilt $\langle r_k \,;\, d_{k-1}\rangle_2 = 0$. Deshalb folgt

$$\|r_k\|_2^2 = \langle r_k \,;\, r_k\rangle_2 = \langle r_k \,;\, d_k\rangle_2 = \langle b - Ax_k \,;\, d_k\rangle_2 = \langle x^* - x_k \,;\, d_k\rangle_A = \langle x^* \,;\, d_k\rangle_A,$$

denn es gilt $x_k \in \mathcal{K}_k$ und damit $\langle x_k \,;\, d_k\rangle_A = 0$. □

5. Schritt. *Es gilt* $\widetilde{\beta}_k := -\langle r_{k+1} \,;\, d_k\rangle_A / \|d_k\|_A^2 = \|r_{k+1}\|_2^2 / \|r_k\|_2^2 =: \beta_k$. Nach Schritt 3 und 4 gilt $r_{k+1} = r_k - \alpha_k A d_k$. Mit $\langle r_{k+1} \,;\, r_k\rangle_2 = 0$ folgt

$$\|r_{k+1}\|_2^2 = -\alpha_k \langle r_{k+1} \,;\, Ad_k\rangle_2 = -\alpha_k \langle r_{k+1} \,;\, d_k\rangle_A = \widetilde{\beta}_k \|r_k\|_2^2$$

nach Definition von $\widetilde{\beta}_k$ und α_k. Umstellen zeigt die Behauptung. ■

Im Gegensatz zu den stationären linearen Iterationsverfahren aus Abschnitt 7.3 kann man für das CG-Verfahren a priori angeben, wie viele Iterationsschritte höchstens notwendig sind, um eine gewünschte Genauigkeit zu erreichen.

Satz 7.19. *Unter den Voraussetzungen des CG-Verfahrens gilt mit* $\kappa := \sqrt{\operatorname{cond}_2(A)}$ *für alle* $\ell \in \mathbb{N}$ *die relative a priori Fehlerabschätzung*

$$\|x^* - x_\ell\|_A \leq 2\|x^*\|_A \left(\frac{\kappa - 1}{\kappa + 1}\right)^\ell \tag{7.29}$$

und

$$\|x^* - x_\ell\|_2 \leq 2\kappa\|x^*\|_2 \left(\frac{\kappa - 1}{\kappa + 1}\right)^\ell. \tag{7.30}$$

Beweis. Jede selbstadjungierte Matrix ist diagonalisierbar über $\mathbb{K}$, d.h. es existiert eine $\langle\cdot\,;\,\cdot\rangle_2$-Orthonormalbasis $v_1, \ldots, v_n$ von $\mathbb{K}^n$ aus Eigenvektoren zu A, und die Eigenwerte λ_j sind reell. Aufgrund der positiven Definitheit sind alle Eigenwerte ferner positiv. Es sei $\lambda_j > 0$ der Eigenwert zum Eigenvektor v_j. Ohne Einschränkung gilt $\lambda_1 \geq \cdots \geq \lambda_n > 0$. Aufgrund der Selbstadjungiertheit folgt $\operatorname{cond}_2(A) = \lambda_1/\lambda_n$, denn $\|A\|_2 = \rho(A) = \lambda_1$ und $\|A^{-1}\|_2 = \rho(A^{-1}) = \lambda_n^{-1}$. Für einen Vektor $x = \sum_{j=1}^n \alpha_j v_j \in \mathbb{K}^n$ gilt dann

$$\|x\|_2^2 = \sum_{j=1}^n \alpha_j^2 \quad \text{und} \quad \|x\|_A^2 = \langle Ax \,;\, x\rangle_2 = \sum_{j=1}^n \lambda_j \alpha_j^2 \tag{7.31}$$

Deshalb folgt die Normäquivalenz

$$\sqrt{\lambda_n}\,\|x\|_2 \le \|x\|_A \le \sqrt{\lambda_1}\,\|x\|_2 \quad \text{für } x \in \mathbb{K}^d$$

und für $x^* \neq 0$ daraus die Abschätzung

$$\frac{\|x^* - x\|_2}{\|x^*\|_2} \le \frac{\sqrt{\lambda_1}}{\sqrt{\lambda_n}} \frac{\|x^* - x\|_A}{\|x^*\|_A} = \kappa \frac{\|x^* - x\|_A}{\|x^*\|_A}.$$

Es ist also nur (7.29) zu zeigen. Im Fall $\operatorname{cond}_2(A) = \lambda_1/\lambda_n = 1$ folgt $A = \lambda_1 \boldsymbol{I}$ und deshalb sofort $x_1 = x^*$. Also können wir ohne Beschränkung der Allgemeinheit $\kappa > 1$ annehmen. Die Verifikation von (7.29) folgt in 2 Schritten:

1. Schritt. *Es gilt* $\|x^* - x_\ell\|_A \le \|x^*\|_A \inf\limits_{p\in\mathbb{P}_\ell, p(0)=1} \max\limits_{j=1,\dots,n} |p(\lambda_j)|$. Es sei $p \in \mathbb{P}_\ell$ mit $p(0) = 1$. Da das Polynom $1 - p$ dann eine Nullstelle bei $t = 0$ hat, gilt

$$q \in \mathbb{P}_{\ell-1}, \quad q(t) := \frac{1 - p(t)}{t}$$

nach Polynomdivision. Definiere $x := q(A)b \in \mathcal{K}_\ell$. Es gilt

$$p(A)x^* = (1 - Aq(A))x^* = x^* - q(A)Ax^* = x^* - x.$$

Mit der Darstellung $x^* = \sum_{j=1}^n \alpha_j v_j$ folgt $p(A)x^* = \sum_{j=1}^n \alpha_j p(A)v_j = \sum_{j=1}^n \alpha_j p(\lambda_j)v_j$. Mit (7.31) erhalten wir daraus

$$\begin{aligned}\|x^* - x\|_A^2 = \|p(A)x^*\|_A^2 = \sum_{j=1}^n \lambda_j \alpha_j^2 p(\lambda_j)^2 &\le \max_{k=1,\dots,\ell} |p(\lambda_k)|^2 \sum_{j=1}^n \lambda_j \alpha_j^2 \\ &= \max_{k=1,\dots,\ell} |p(\lambda_k)|^2 \, \|x^*\|_A^2.\end{aligned}$$

2. Schritt. *Es gibt ein Polynom* $p \in \mathbb{P}_\ell$ *mit* $p(0) = 1$ *und* $p(\lambda_j) \le 2\Big(\dfrac{\kappa - 1}{\kappa + 1}\Big)^\ell$ *für alle* $j = 1, \dots, n$. Mit dem Čebyšev-Polynom $T_\ell(t) := \arccos(\ell \cos(t)) \in \mathbb{P}_\ell$ erster Art auf $[0, 1]$ definieren wir

$$p(\lambda) := \frac{q(\lambda)}{q(0)}, \quad q(\lambda) := T_\ell\Big(\frac{\lambda_1 + \lambda_n - 2\lambda}{\lambda_1 - \lambda_n}\Big)$$

Offensichtlich gilt $p \in \mathbb{P}_\ell$ mit $p(0) = 1$. Wegen $\|T_\ell\|_{\infty,[-1,1]} = 1$ und $\kappa = \lambda_1/\lambda_n$ gilt

$$\max_{\lambda_n \le \lambda \le \lambda_1} |p(\lambda)| \le \Big|T_\ell\Big(\frac{\lambda_1 + \lambda_n}{\lambda_1 - \lambda_n}\Big)\Big|^{-1} = \Big|T_\ell\Big(\frac{\operatorname{cond}_2(A) + 1}{\operatorname{cond}_2(A) - 1}\Big)\Big|^{-1}.$$

Direkte Rechnung für das Čebyšev-Polynom T_ℓ zeigt schließlich

$$\Big|T_\ell\Big(\frac{\kappa^2 + 1}{\kappa^2 - 1}\Big)\Big| \ge \frac{1}{2}\Big(\frac{\kappa + 1}{\kappa - 1}\Big)^\ell,$$

siehe Plato [5, Lemma 11.18]. Hieraus folgt die Behauptung. ■

Literaturverzeichnis

[1] M. Brokate: *Praktische Analysis*, Vorlesungsskript Christian-Albrechts-Universität zu Kiel, 1998.

[2] C. Carstensen: *Numerische Mathematik 2*, Vorlesungsskript TU Wien, 2002.

[3] O. Forster: *Analysis 1*, Vieweg ^{4}1992.

[4] H. Heuser: *Lehrbuch der Analysis, Teil 2*, Teubner Stuttgart ^{8}1993.

[5] R. Plato: *Numerische Mathematik kompakt*, Vieweg 2000

[6] A. Quarteroni, R. Sacco, F. Saleri: *Numerical Mathematics*, Springer 2000

[7] Stoer: *Numerische Mathematik*, Springer ^{7}1994.