Otmar Scherzer

Numerische Mathematik

Vorlesungsskriptum SS 2013

Computational Science Center
Universität Wien
Nordbergstr. 15
1090 Wien

Inhaltsverzeichnis

Dank und Literaturstellen

Diese Vorlesungsskriptum beruht auf dem Buch von Prof. Martin Hanke (Universität Mainz) [4].

Die Literatur zur numerischen Mathematik ist äußerst umfangreich. An dieser Stelle sei auf einige kürzlich erschienene oder wieder aufgelegte Bücher zur Numerischen Mathematik hingewiesen, ohne aber einen Anspruch auf Vollständigkeit zu erheben [6, 8, 7, 5, 2, 1, 3] .

Kapitel 1

Rundefehler, Kondition und Stabilität

1.1 Vektor und Matrixnormen

Wir betrachten sowohl reelle als auch komplexe Vektoren und Matrizen. Meistens ist das unerheblich und wir schreiben der Einfachheit $\mathcal{K}$ für den entsprechenden Zahlenkörper und meinen damit, dass die entsprechenden Resultate in gleicher Weise in $\mathbb{R}$ und $\mathbb{C}$ gelten.

Entsprechend bezeichnet $\mathcal{K}^n$ den Raum der n-dimensionalen Vektoren über $\mathcal{K}$,

$$x = \begin{bmatrix} x_1 \\ \vdots \\ x_n \end{bmatrix}, \quad x_i \in \mathcal{K}\,,$$

und $\mathcal{K}^{m\times n}$ den Raum der $m \times n$ Matrizen über $\mathcal{K}$,

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

Ist $x \in \mathcal{K}^n$, so unterscheiden wir zwischen

$$x^t = [x_1, x_2, ..., x_n] \in \mathcal{K}^{1\times n} \text{ und } x^* = [\overline{x}_1, \overline{x}_2, ..., \overline{x}_n] \in \mathcal{K}^{1\times n};$$

bei x^* sind die Einträge konjugiert komplex. Für $\mathcal{K} = \mathbb{R}$ stimmen x^* und x^t überein. A^t und A^* in $\mathcal{K}^{n\times m}$ sind entsprechend definiert.

Im Raum $\mathcal{K}^n$ greifen wir gelegentlich auf die *kartesische Basis* $\{e_1, ..., e_n\}$ zurück, wobei $e_i = [\delta_{ij}]_{j=1}^n$ den Vektor bezeichnet, der in der i-ten Komponente eine Eins und ansonsten nur Nulleinträge enthält.

$$\delta_{ij} = \begin{cases} 1, & i = j\,, \\ 0, & i \neq j\,, \end{cases}$$

ist das *Kronecker-Symbol*.

Beispiel 1.1. Die am häufigsten verwendeten Normen in $X = \mathcal{K}^n$ sind die

1. *Betragssummennorm:* $\|x\|_1 := \sum_{i=1}^n |x_i|$,
2. *Euklidnorm:* $\|x\|_2 := \sqrt{\sum_{i=1}^n |x_i|^2} = \sqrt{x^* x}\,,$
3. *Maximumnorm:* $\|x\|_\infty := \max_{i=1,...,n} |x_i|$.

Häufigste verwendete Normen in $X = \mathcal{K}^{m\times n}$ sind die

1. $\|A\|_{1,\infty} := \max_{1\leq j\leq n} \sum_{i=1}^m |a_{i,j}|\,.$
2. $\|A\|_{\infty,1} := \max_{1\leq i\leq m} \sum_{j=1}^n |a_{i,j}|\,.$
3. *Frobeniusnorm:* $\|A\|_F := \sqrt{\sum_{i=1}^m \sum_{j=1}^n |a_{i,j}|^2}$.

1.2 Kondition und Rundefehler

Eine Fehlerquelle bei der Implementierung jedes numerischen Algorithmus sind *Daten* - und *Rundefehler*.

Während der Rundefehler in jeder einzelnen *Elementaroperation* (Addition, Multiplikation, Standardfunktionen, ...) in der Regel vernachlässigt werden kann, kann es in einem komplexen Algorithmus zu einer Kumulation der Fehler führen, und sich problematisch auf das berechnete Ergebnis auswirken. Man spricht von einem *stabilen* Algorithmus, wenn keine problematische Fehlerverstärkung auftritt.

Sei $F : \mathbb{R}^n \to \mathbb{R}$. Wir studieren die Auswertung der Funktion F an einem vorgegebenen Vektor x. Aufgrund von Fehlern wird die Auswertung der Funktion F nicht an der Stelle x, sondern an der Stelle $x + \Delta x$ ausgewertet.

Bezeichnen wir mit

$$\Delta y := F(x + \Delta x) - F(x)$$

den Fehler im Ergebnis, so gilt, falls $F \in C^1(\mathbb{R}^n)$ ist, nach dem Mittelwertsatz

$$\begin{aligned}\Delta y &= F(x_1 + \Delta x_1, x_2 + \Delta x_2, ..., x_n + \Delta x_n) - F(x_1, x_2, ..., x_n) \\ &= \sum_{i=1}^{n} \frac{\partial F}{\partial x_i}(\zeta)\Delta x_i \ ,\end{aligned}$$

wobei ζ auf der Strecke zwischen x und $x + \Delta x$ liegt. Ist die Ableitung Lipschitz-stetig, dann gilt sogar

$$\Delta y = \sum_{i=1}^{n} \frac{\partial F}{\partial x_i}(x)\Delta x_i + \mathcal{O}(\varepsilon^2)\,, \tag{1.1}$$

wobei $\varepsilon := \max_{i=1,...,n} |\Delta x_i|$ gesetzt wird.[1]

Um sich die Analyse zu vereinfachen, vernachlässigt man den $\mathcal{O}(\varepsilon^2)$-Term und verwendet den Vektor der *absoluten Konditionszahlen*

$$\mathcal{K}_{\text{abs}} := \left[\left|\frac{\partial F}{\partial x_i}(x)\right|\right]_{i=1,...,n}$$

als ein Maß für die Verstärkung des Fehlers.

Üblicherweise ist die relative Fehlerverstärkung von größerer Bedeutung als die absolute Fehlerverstärkung. Ein Maß für die relative Fehlerverstärkung ergibt sich aus (1.1) unter Vernachlässigung des $\mathcal{O}(\varepsilon^2)$ Terms wie folgt:

$$\frac{\Delta y}{y} = \sum_{i=1}^{n} \frac{\partial F}{\partial x_i}(x)\frac{\Delta x_i}{F(x)} = \sum_{i=1}^{n} \frac{\partial F}{\partial x_i}(x)\frac{x_i}{F(x)}\frac{\Delta x_i}{x_i}\ . \tag{1.2}$$

Der Vektor

$$\mathcal{K}_{\text{rel}} = \left[\left|\frac{\partial F}{\partial x_i}(x)\frac{x_i}{F(x)}\right|\right]_{i=1,...,n}$$

heißt der Vektor der *relativen Konditionszahlen*.

Die Konditionszahlen beschreiben also die Verstärkung der absoluten bzw. relativen Fehler der Eingangsdaten bei der Auswertung der Funktion F. Ein Problem heißt *schlecht konditioniert*, wenn einer der beiden Maxima der Konditionsvektoren signifikant größer als 1 ist. Ansonst nennt man das Problem *gut konditioniert*.

[1] Wir verwenden die Notation $a_\varepsilon = \mathcal{O}(\varepsilon)$, wenn für ein $\varepsilon_0 >$ eine von ε unabhängige Konstante $C > 0$ existiert, so dass für alle $\varepsilon \in (0, \varepsilon_0)$ die Ungleichung $|a_\varepsilon| \leq C\varepsilon$ gilt.

Eine Verallgemeinerung auf mehrdimensionale Funktionen $F : \mathbb{R}^n \to \mathbb{R}^m$ ist leicht möglich, wenn man die einzelnen Komponenten der Funktion betrachtet.

Beispiel 1.2. Addition: Sei $F(x) = x_1 + x_2$, $x = [x_1, x_2]^t$. Dann gilt für die relativen Konditionszahlen

$$(\mathcal{K}_{\text{rel}})_i = \left| \frac{\partial F}{\partial x_i}(x) \frac{x_i}{x_1 + x_2} \right| = \left| \frac{x_i}{F(x)} \right| , \quad i = 1, 2 .$$

Die relativen Konditionszahlen sind somit groß, wenn für $i = 1$ oder $i = 2$ der Betrag von $F(x) = x_1 + x_2$ sehr viel kleiner als der Betrag von x_i ist. Dieses Phänomen bezeichnet man als *Auslöschung*.

Beispielsweise ergibt sich für

$$\begin{aligned} x_1 = 1.000001 , \quad x_2 &= -1 , \\ \Delta x_1 = 0.001 , \quad \Delta x_2 &= 0 , \end{aligned}$$

$$x_1 + x_2 = 0.000001 , \quad (x_1 + \Delta x_1) + (x_2 + \Delta x_2) = 0.001001 .$$

Der absolute Fehler ist gleich groß wie die Fehler in den Daten $0.001001 = |\Delta x_1| + |\Delta x_2|$. Der relative Fehler ($|\Delta y / y| = 0.001001/0.000001 = 1001$) ist signifikant größer.

Multiplikation: Sei $a \neq 0$ fix. Wir betrachten die Multiplikation zweier Zahlen,

$$\begin{aligned} F : \mathbb{R} &\to \mathbb{R} . \\ x &\to ax \end{aligned} \tag{1.3}$$

In diesem Fall lauten die absoluten und relativen Konditionszahlen

$$\mathcal{K}_{\text{abs}} = |F'(x)| = a \text{ und } \mathcal{K}_{\text{rel}} = 1 .$$

1.3 Stabilität

Betrachten wir nun die Implementierung eines Algorithmus zur Auswertung von $F(x)$ mit $F : \mathbb{R}^n \to \mathbb{R}$. Die Modellannahme ist, dass die Implementierung geschrieben werden als $F(x + \Delta x)$.

Der Algorithmus heißt *vorwärts stabil*, wenn eine Konstante C_V existiert, die nicht signifikant größer als 1 ist, sodass [2]

$$\left|\frac{\Delta y}{F(x)}\right| \leq C_V \left\|\frac{\Delta x}{x}\right\|_2 . \tag{1.4}$$

Der Algorithmus heißt *rückwärts stabil*, wenn eine Konstante C_R existiert, die nicht signifikant größer als 1 ist, sodass

$$\left\|\frac{\Delta x}{x}\right\|_2 \leq C_R \left|\frac{\Delta y}{F(x)}\right| . \tag{1.5}$$

Die Konstanten C_V und C_R stehen mit den Konditionszahlen in Verbindung, wie man wie folgt sieht:

$$\begin{aligned}
\left|\frac{\Delta y}{F(x)}\right| &= \left|\frac{F(x+\Delta x)-F(x)}{F(x)}\right| \\
&\approx \left|\frac{F'(x)^t \Delta x}{F(x)}\right| \\
&= \left|\frac{F'(x)^t \left(x \frac{\Delta x}{x}\right)}{F(x)}\right| \\
&\leq \sqrt{n} \underbrace{\max_{i=1,\ldots,n} \left|\frac{\frac{\partial F}{\partial x_i}(x) x_i}{F(x)}\right|}_{\leq \|\mathcal{K}_{\mathrm{rel}}\|_\infty} \left\|\frac{\Delta x_i}{x_i}\right\|_2 .
\end{aligned} \tag{1.6}$$

Das bedeutet also, dass $C_V \approx \sqrt{n}\,\|\mathcal{K}_{\mathrm{rel}}\|_\infty$ gilt.

1.4 Multiplikative Normen

Definition 1.3. Eine Matrixnorm $\|\cdot\|_M$ auf $\mathcal{K}^{n\times n}$ heißt *submultiplikativ*, falls

$$\|AB\|_M \leq \|A\|_M \|B\|_M , \quad \forall A, B \in \mathcal{K}^{n\times n} .$$

Eine Matrixnorm $\|\cdot\|_M$ auf $\mathcal{K}^{n\times n}$ heißt *verträglich* mit der Vektornorm $\|\cdot\|$ auf $\mathcal{K}^n$, falls

$$\|Ax\| \leq \|A\|_M \|x\| , \quad \forall A \in \mathcal{K}^{n\times n}, x \in \mathcal{K}^n .$$

[2] Wir verwenden die Notation, dass die Multiplikation und Division von Vektoren komponentenweise ist.

Definition und Satz 1.4. *Sei $\|\cdot\|$ eine Vektornorm auf $\mathcal{K}^n$. Dann ist*

$$|||A||| := \sup_{x\neq 0} \frac{\|Ax\|}{\|x\|} = \max_{\|x\|=1} \|Ax\|$$

eine Norm auf $\mathcal{K}^{n\times n}$ – die durch $\|\cdot\|$ induzierte Norm.

Die Normeigenschaften sind dabei leicht nachgerechnet.

Beispiel 1.5. Sei $A \in \mathcal{K}^{n\times n}$ und $x \in \mathcal{K}^n$. Dann gilt

$$\begin{aligned}
\|Ax\|_1 &= (\text{Spaltensummennorm}) \\
&= \sum_{i=1}^{n} |(Ax)_i| \\
&= \sum_{i=1}^{n} \left| \sum_{j=1}^{n} a_{ij} x_j \right| \\
&\leq \sum_{i=1}^{n} \sum_{j=1}^{n} |a_{ij}|\, |x_j| \\
&\leq \sum_{j=1}^{n} |x_j| \sum_{i=1}^{n} |a_{ij}| \\
&\leq \sum_{j=1}^{n} |x_j| \max_{j=1,\ldots,n} \sum_{i=1}^{n} |a_{ij}| \\
&= \|x\|_1 \, \|A\|_{1,\infty} \ .
\end{aligned}$$

Also ist $\|\cdot\|_{1,\infty}$ mit $\|\cdot\|_1$ verträglich, was bedeutet, dass

$$\frac{\|Ax\|_1}{\|x\|_1} \leq \|A\|_{1,\infty} \ , \quad \forall x \neq 0 \ . \tag{1.7}$$

Wir zeigen nun, dass $\|A\|_{1,\infty}$ die kleinstmögliche Schranke ist für die (1.7) gilt. Dazu werden wir ein $0 \neq x \in \mathcal{K}^n$ suchen, so dass in (1.7) Gleichheit gilt. Wir wählen nun den Spaltenindex j, für den

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

gilt und setzen für x den j-ten kartesischen Basisvektor e_j. Dann gilt

$$\|A\|_{1,\infty} = \|Ae_j\|_1 = \frac{\|Ae_j\|_1}{\|e_j\|_1} \, .$$

In ähnlicher Weise zeigt man, dass die $\|\cdot\|_{\infty,1}$ durch die Maximumnorm $\|\cdot\|_\infty$ induziert wird.

Lemma 1.6. *Die durch $\|\cdot\|$ induzierte (Matrix–) Norm $|||\cdot|||$ ist submultiplikativ und ist mit der Ausgangsnorm verträglich. Ist $\|\cdot\|_M$ eine andere mit $\|\cdot\|$ verträgliche Norm, dann gilt $|||A||| \leq \|A\|_M \, , \forall A \in \mathcal{K}^{n\times n}$.*

Beweis.

- Für $B \neq 0$ gilt

$$\begin{aligned}
|||AB||| &= \sup_{x\neq 0} \frac{\|ABx\|}{\|x\|} \\
&= \sup_{Bx\neq 0} \left(\frac{\|ABx\|}{\|Bx\|} \frac{\|Bx\|}{\|x\|} \right) \\
&\leq \sup_{Bx\neq 0} \frac{\|ABx\|}{\|Bx\|} \sup_{x\neq 0} \frac{\|Bx\|}{\|x\|} \\
&\leq \sup_{y\neq 0} \frac{\|Ay\|}{\|y\|} \sup_{x\neq 0} \frac{\|Bx\|}{\|x\|} \\
&= |||A||| \, |||B||| \, .
\end{aligned}$$

  Folglich ist die induzierte Norm submultiplikativ.

- Die Verträglichkeit mit der Ausgangsnorm folgt unmittelbar aus der Definition: Demnach ist nämlich

$$|||A||| = \sup_{x\neq 0} \frac{\|Ax\|}{\|x\|} \geq \frac{\|Ax\|}{\|x\|} \text{ für jedes } x \neq 0 \, ,$$

  beziehungsweise $\|Ax\| \leq |||A||| \, \|x\|$.

- Sei $\|\cdot\|_M$ eine andere mit $\|\cdot\|$ verträgliche Norm. Nach Definition 1.4 gilt $|||A||| = \|Ax\|$ für ein gewisses $x \in \mathcal{K}^n$ mit $\|x\| = 1$, und aus der Verträglichkeit der zweiten Matrixnorm folgt daher

$$|||A||| = \|Ax\| \leq \|A\|_M \, \|x\| = \|A\|_M \, .$$

□

Die vermutlich wichtigste Norm in $\mathcal{K}^n$ ist die Euklidnorm. Wir werden uns nun mit der durch die Euklidnorm induzierten Matrixnorm in $\mathcal{K}^{m\times n}$ (muss nicht notwendigerweise eine quadratische Matrix sein) beschäftigen. Diese induzierte Norm heißt *Spektralnorm*

$$\begin{aligned}\|A\|_2 &:= \max_{\|x\|_2=1} \|Ax\|_2 \\ &= \max_{\|x\|_2=1} \sqrt{(Ax)^*(Ax)} \\ &= \max_{\|x\|_2=1} \sqrt{x^*A^*Ax}\ .\end{aligned} \tag{1.8}$$

Bezeichnet $\sigma(C)$ die Menge aller Eigenwerte von C, dann ist der *Spektralradius* gegeben durch

$$\rho(C) := \max\{|\lambda| : \lambda \in \sigma(C)\}\ , \tag{1.9}$$

also dem betragsgrößtem Eigenwert der Matrix C.

Satz 1.7. *Für jede Matrix $A \in \mathcal{K}^{m\times n}$ ist $\|A\|_2 = \sqrt{\rho(A^*A)}$.*

Beweis. A^*A ist hermitesch und positiv semidefinit [3], denn

$$x^*(A^*A)x = \|Ax\|_2^2 \geq 0\ .$$

Demnach existiert eine Orthonormalbasis $\{x_1, ..., x_n\}$ von $\mathcal{K}^n$ aus Eigenvektoren[4] von A^*A mit zugehörigen Eigenwerten

$$\lambda_1 \geq \lambda_2 \geq \cdots \geq \lambda_n \geq 0\ .$$

Jeder Vektor $x \in \mathcal{K}^n$ mit $\|x\|_2 = 1$ lässt sich in dieser Basis entwickeln, das

[3] Eine Matrix $B \in K^{n\times n}$ heißt hermitesch, falls $B^* = B$. Sie heißt positiv semidefinit falls $\|Bx\|_2^2 \geq 0\,, \forall x \in \mathcal{K}^n$.

[4] Wir wollen Eigenvektoren immer so verstehen, dass die Euklidnorm auf 1 normiert ist.

heißt, es existieren $\zeta_i \in \mathcal{K}$, $i = 1, \cdots n$, so dass $x = \sum_{i=1}^n \zeta_i x_i$ und

$$\begin{aligned} 1 = x^*x &= \sum_{i=1}^n \overline{\zeta_i} x_i^* \sum_{j=1}^n \zeta_j x_j \\ &= \sum_{i,j=1}^n \overline{\zeta_i} \zeta_j x_i^* x_j \\ &= \sum_{i,j=1}^n \overline{\zeta_i} \zeta_j \delta_{ij} \\ &= \sum_{i=1}^n |\zeta_i|^2 \ . \end{aligned}$$

Außerdem gilt,

$$\begin{aligned} x^*A^*Ax &= \sum_{i=1}^n \overline{\zeta_i} x_i^* A^*A \sum_{j=1}^n \zeta_j x_j \\ &= \sum_{i,j=1}^n \overline{\zeta_i} x_i^* \zeta_j \lambda_j x_j \\ &= \sum_{i,j=1}^n \overline{\zeta_i} \zeta_j \lambda_j \delta_{ij} \\ &= \sum_{i=1}^n |\zeta_i|^2 \lambda_i \\ &\leq \lambda_1 \sum_{i=1}^n |\zeta_i|^2 \\ &= \lambda_1 \ . \end{aligned}$$

Mit anderen Worten: Es gilt $\max_{\|x\|_2=1} x^*A^*Ax \leq \lambda_1$. Es gilt aber auch für $x = x_1$

$$x_1^*A^*Ax_1 = x_1^*\lambda_1 x_1 = \lambda_1$$

und daher $\max_{\|x\|_2=1} x^*A^*Ax = \lambda_1 = \rho(A^*A)$. Zusammen mit (1.8) folgt daher die Behauptung. □

Lemma 1.8.

$$\|A\|_F^2 = \mathit{Spur}(A^*A) = \sum_{\lambda \in \sigma(A^*A)} \lambda \ .$$

Man beachte den Unterschied zwischen $\|A\|_F^2 = \sum_{\lambda\in\sigma(A^*A)} \lambda$ und $\|A\|_2^2 = \max_{\lambda\in\sigma(A^*A)} \lambda$.

Die Berechnung der Spektralnorm benötigt den größten Eigenwert von A^*A und ist deshalb viel aufwendiger als die Berechnung der $\|\cdot\|_{1,\infty}$ und $\|\cdot\|_{\infty,1}$–Normen. Für viele Anwendungen ist aber eine Abschätzung des größten Eigenwertes von A^*A ausreichend, die man wie folgt erhalten kann:

Satz 1.9. *Für $A \in \mathcal{K}^{m\times n}$ gilt $\|A\|_2 \le \sqrt{\|A\|_{1,\infty}\,\|A\|_{\infty,1}}$.*

Beweis. Nach Satz 1.7 ist $\|A\|_2^2$ der größte Eigenwert von A^*A. Sei x_1 ein zu $\|A\|_2^2$ zugehöriger Eigenvektor (mit $\|x_1\|_2 = 1$) und $\hat{x}_1 = x_1/\|x_1\|_1$.

Dann gilt

$$\|A\|_2^2 = \|Ax_1\|_2^2 = x_1^*A^*Ax_1 = \lambda_1 x_1^* x_1 = \lambda_1 \|x_1\|_2^2 = \lambda_1 \ .$$

Andererseits gilt:

$$\begin{aligned}\lambda_1 &= \lambda_1 \|\hat{x}_1\|_1 \\ &= \|A^*A\hat{x}_1\|_1 \\ &\le \|A^*\|_{1,\infty} \|Ax_1\|_1 \\ &\le \|A^*\|_{1,\infty} \|A\|_{1,\infty} \|\hat{x}_1\|_1 \\ &= \|A^*\|_{1,\infty} \|A\|_{1,\infty} \ .\end{aligned}$$

Da $\|A^*\|_{1,\infty} = \|A\|_{\infty,1}$ gilt, folgt somit:

$$\|A\|_2^2 = \lambda_1 \le \|A\|_{\infty,1} \|A\|_{1,\infty} \ .$$

□

Eine wichtige Anwendung von Matrixnormen ergibt sich bei der Bestimmung der Kondition eines linearen Gleichungssystems.

Sei $A \in \mathcal{K}^{n\times n}$ regulär und Δb ein Eingangsfehler, dann gilt

$$x = A^{-1}b \text{ und } x + \Delta x = A^{-1}(b + \Delta b) = A^{-1}b + A^{-1}\Delta b \ .$$

Also ist der Fehler in der Lösung

$$\Delta x = A^{-1}\Delta b \ .$$

Sind die Matrixnorm $\|\cdot\|_M$ und die Vektornorm $\|\cdot\|$ verträglich, dann gilt

$$\begin{aligned}\frac{\|\Delta x\|}{\|x\|} &= \frac{\|A^{-1}\Delta b\|}{\|x\|} \\ &\leq \left\|A^{-1}\right\|_M \frac{\|\Delta b\|}{\|b\|}\frac{\|Ax\|}{\|x\|} \\ &\leq \left\|A^{-1}\right\|_M \|A\|_M \frac{\|\Delta b\|}{\|b\|}\,.\end{aligned} \tag{1.10}$$

Definition 1.10. Der Faktor

$$\mathrm{cond}_M(A) := \left\|A^{-1}\right\|_M \|A\|_M$$

wird als *Kondition* der Matrix A bzgl. der Matrixnorm $\|\cdot\|_M$ bezeichnet.

Kapitel 2

Eliminiationsalgorithmen

Der grundlegende Baustein aller numerischen Algorithmen ist die Lösung linearer Gleichungssysteme. Prinzipiell unterscheidet man zwischen iterativen Verfahren und Eliminiationsalgorithmen.

2.1 Die LR-Zerlegung

Der wichtigste Algorithmus zur Lösung linearer Gleichungssysteme ist der *Gauß*-Algorithmus, welcher implizit eine Zerlegung einer Koeffizientenmatrix A in zwei Dreiecksmatrizen bewirkt. Deshalb wird er auch LR–Zerlegungs Algorithmus genannt.

Wir betrachten zunächst einen beliebigen Vektor $x = [x_1, ..., x_n]^t \in \mathcal{K}^n$ für den gilt $x_k \neq 0$ ist. Dann definieren wird

$$L^{(k)} = I - l_k e_k^t \,, \tag{2.1}$$

mit $l_k = [0, ..., 0, l_{k+1,k}, ..., l_{n,k}]^t \in \mathcal{K}^n$ mit $l_{jk} = x_j/x_k$, $j = k+1, ..., n$. Somit ist

$$L^{(k)}x = \begin{bmatrix} 1 & 0 & & & \cdots & 0 \\ 0 & \ddots & \ddots & & & \vdots \\ & \ddots & 1 & 0 & & \\ & & -l_{k+1,k} & 1 & \ddots & \\ \vdots & & \vdots & \ddots & \ddots & 0 \\ 0 & \cdots & -l_{n,k} & \cdots & 0 & 1 \end{bmatrix} \begin{bmatrix} x_1 \\ \vdots \\ x_k \\ x_{k+1} \\ \vdots \\ x_n \end{bmatrix} = \begin{bmatrix} x_1 \\ \vdots \\ x_k \\ 0 \\ \vdots \\ 0 \end{bmatrix} .$$

Matrizen der Form $L^{(k)}$ können also benutzt werden, um die unteren $n-k$ Einträge eines Spaltenvektors zu Null zu transformieren.

Sei $A = A^{(1)} = [a_{ij}]_{ij}$ eine $n \times n$–Matrix und $x = [a_{i1}]_i \in \mathcal{K}^n$ die *erste* Spalte von $A^{(1)}$. Wenn $a_{11} \neq 0$ ist, dann gilt mit der Matrix $L^{(1)} = I - l_1 e_1^t$

$$\begin{aligned} L^{(1)}A^{(1)} &= \begin{bmatrix} 1 & 0 & \cdots & & 0 \\ -l_{21} & 1 & 0 & \cdots & 0 \\ -l_{31} & 0 & 1 & & 0 \\ \vdots & & \ddots & \ddots & 0 \\ -l_{n1} & 0 & \cdots & 0 & 1 \end{bmatrix} \begin{bmatrix} a_{11} & a_{12} & \cdots & a_{1n} \\ a_{21} & a_{22} & \cdots & a_{2n} \\ a_{31} & a_{32} & \cdots & a_{3n} \\ \vdots & \vdots & & \vdots \\ a_{n1} & a_{n2} & \cdots & a_{nn} \end{bmatrix} \\ &= \begin{bmatrix} a_{11} & a_{12} & \cdots & a_{1n} \\ 0 & a_{22}^{(2)} & \cdots & a_{2n}^{(2)} \\ 0 & a_{32}^{(2)} & \cdots & a_{3n}^{(2)} \\ \vdots & \vdots & & \vdots \\ 0 & a_{n2}^{(2)} & \cdots & a_{nn}^{(2)} \end{bmatrix} =: A^{(2)} \, . \end{aligned} \tag{2.2}$$

Dies entspricht dem ersten Schritt im Gauß-Algorithmus. Wenn $a_{22}^{(2)} \neq 0$ ist, wählen wir in einem zweiten Schritt ($k = 2$) für x die zweite Spalte von $A^{(2)}$, also $x = [a_{12}, a_{22}^{(2)}, ..., a_{n2}^{(2)}]^t$. Mit der zugehörigen Matrix $L^{(2)} = I - l_2 e_2^t$ ergibt sich dann entsprechend $A^{(3)} = L^{(2)}A^{(2)}$, wobei:

$$L^{(2)} = \begin{bmatrix} 1 & 0 & \cdots & \cdots & \cdots & 0 \\ 0 & 1 & 0 & \cdots & \cdots & 0 \\ 0 & -l_{32} & 0 & \cdots & \cdots & 0 \\ 0 & -l_{42} & 0 & 1 & 0 & \cdots \\ \vdots & \vdots & & \ddots & \ddots & 0 \\ 0 & -l_{n2} & & \cdots & \cdots & 1 \end{bmatrix} \quad A^{(3)} = \begin{bmatrix} a_{11} & a_{12} & a_{13} & \cdots & a_{1n} \\ 0 & a_{22}^{(2)} & a_{23}^{(2)} & \cdots & a_{2n}^{(2)} \\ 0 & 0 & a_{33}^{(3)} & \cdots & a_{3n}^{(3)} \\ 0 & 0 & a_{43}^{(3)} & \cdots & a_{4n}^{(3)} \\ \vdots & \vdots & & & \vdots \\ 0 & 0 & a_{n3}^{(3)} & \cdots & a_{nn}^{(3)} \end{bmatrix} .$$

Geht man auf diese Art und Weise weiter fort (immer vorausgesetzt, dass das *Pivotelement* $a_{ii}^{(i)}$ von Null verschieden ist), dann erhält man nach $(n-1)$ Transformationen schließlich eine obere Dreiecksmatrix $R := A^{(n)}$ und es gilt:

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

Mit anderen Worten: Es ist

$$A = LR \text{ mit } L = L^{(1)-1}L^{(2)-1} \cdots L^{(n-1)-1} \, . \tag{2.3}$$

Die inversen Matrizen $L^{(i)-1}$ können explizit angegeben werden. Aus dem folgenden Resultat sieht man insbesondere, dass L tatsächlich eine Dreiecksmatrix ist.

Satz 2.1. *Es ist* $L^{(i)-1} = I + l_i e_i^t$ *(man beachte* $L^{(i)} = I - l_i e_i^t$*) und* $L = I + l_1 e_1^t + \cdots + l_{n-1} e_{n-1}^t$.

Beweis. Es gilt

$$e_i^t l_j = [0, \cdots, 0, 1, 0, \cdots, 0] \begin{bmatrix} 0 \\ \vdots \\ 0 \\ l_{j+1,j} \\ \vdots \\ l_{n,j} \end{bmatrix} = \begin{cases} 0 & \text{für} \quad i \leq j \\ l_{i,j} & \text{für} \quad i \geq j+1 \end{cases} . \quad (2.4)$$

Daraus folgt zunächst die erste Behauptung, denn

$$\begin{aligned} (I - l_i e_i^t)(I + l_i e_i^t) &= I - l_i e_i^t + l_i e_i^t - l_i e_i^t l_i e_i^t \\ &= (e_i^t l_i = 0 (2.4)) \\ &\quad I - l_i e_i^t + l_i e_i^t \\ &= I \, . \end{aligned}$$

Die spezielle Form von L ergibt sich induktiv: Dazu nehmen wir an, dass für ein $1 \leq k < n$ gilt

$$L^{(1)-1} \cdots L^{(k)-1} = I + l_1 e_1^t + \cdots + l_k e_k^t \, .$$

Für $k = 1$ ist diese Gleichung wegen des ersten Teils des Beweises erfüllt. Aus $L^{(k+1)-1} = I + l_{k+1} e_{k+1}^t$ folgt dann

$$L^{(1)-1} \cdots L^{(k+1)-1} = (I + l_1 e_1^t + \cdots + l_k e_k^t)(I + l_{k+1} e_{k+1}^t) \, ,$$

und wegen (2.4) ergibt dies

$$\begin{aligned} L^{(1)-1} \cdots L^{(k+1)-1} =& I + l_1 e_1^t + \cdots + l_k e_k^t + l_{k+1} e_{k+1}^t + \sum_{i=1}^{k} l_i e_i^t l_{k+1} e_{k+1}^t \\ =& (e_i^t l_{k+1} = 0 \text{ für } i = 1, \cdots, k) \\ & I + l_1 e_1^t + \cdots + l_k e_k^t + l_{k+1} e_{k+1}^t \, . \end{aligned}$$

Damit ist die Induktionsbehauptung auch für $k+1$ erfüllt. □

Wird im Verlaufe des Gauß-Algorithmus ein Pivotelement $a_{ii}^{(i)}$ Null, so bricht der Algorithmus zusammen. Sind hingegen alle Pivotelemente für $i = 1, \cdots, n$ von Null verschieden, dann haben wir insgesamt folgendes Resultat bewiesen:

Satz 2.2. *Falls kein Pivotelement Null wird, bestimmt der Gauß-Algorithmus eine LR-Zerlegung.*

$$A = LR = \begin{bmatrix} 1 & & & & \\ l_{21} & 1 & & & \\ l_{31} & l_{32} & 1 & & \\ \vdots & \vdots & \vdots & \vdots & \\ l_{n1} & l_{n2} & \cdots & l_{n,n-1} & 1 \end{bmatrix} \begin{bmatrix} a_{11} & a_{12} & a_{13} & \cdots & a_{1n} \\ & a_{22}^{(2)} & a_{23}^{(2)} & \cdots & a_{2n}^{(2)} \\ & & a_{33}^{(3)} & \cdots & a_{3n}^{(3)} \\ & 0 & \vdots & \vdots & \\ & & & & a_{nn}^{(n)} \end{bmatrix}$$

in eine linke untere und eine rechte obere Dreiecksmatrix.

Gleichungssysteme mit Dreiecksmatrizen können unmittelbar durch Vorwärts- bzw. Rückwärtssubstitution gelöst werden. Somit ermöglicht die LR-Zerlegung in einfacher Weise die Lösung eines linearen Gleichungssytems $Ax = b$.

Algorithmus 2.3. 1. Zerlege $A = LR$ mit dem Gauß-Algorithmus

2. Löse $Ax = LRx = b$ in zwei Schritten wie folgt:

   - Löse $Ly = b$ durch Vorwärtssubstitution
   - Löse $Rx = y$ durch Rückwärtssubstitution

   Da $Ax = L(Rx) = v$ ist x die gesuchte Lösung des Gleichungssystems $Ax = b$.

Im Folgenden berechnen wir den Aufwand der LR-Zerlegung. Aus (2.2) erkennt man, dass eine Matrix-Matrix-Multiplikation $A^{(k+1)} = L^{(k)}A^{(k)}$ genau $(n-k)^2$ Multiplikationen kostet. Um sich davon an einem Beispiel zu

überzeugen studieren wir den ersten Elliminationsschritt (2.2):

$$L^{(1)}A^{(1)} = \begin{bmatrix} 1 & 0 & \cdots & & 0 \\ -l_{21} & 1 & 0 & \cdots & 0 \\ -l_{31} & 0 & 1 & & 0 \\ \vdots & & \ddots & \ddots & 0 \\ -l_{n1} & 0 & \cdots & 0 & 1 \end{bmatrix} \begin{bmatrix} a_{11} & a_{12} & \cdots & a_{1n} \\ a_{21} & a_{22} & \cdots & a_{2n} \\ a_{31} & a_{32} & \cdots & a_{3n} \\ \vdots & \vdots & & \vdots \\ a_{n1} & a_{n2} & \cdots & a_{nn} \end{bmatrix}$$

$$= \begin{bmatrix} a_{11} & a_{12} & \cdots & a_{1n} \\ 0 & a_{22}^{(2)} & \cdots & a_{2n}^{(2)} \\ 0 & a_{32}^{(2)} & \cdots & a_{3n}^{(2)} \\ \vdots & \vdots & & \vdots \\ 0 & a_{n2}^{(2)} & \cdots & a_{nn}^{(2)} \end{bmatrix} .$$

Klar ist, dass die erste Spalte und die erste Zeile von $A^{(2)}$ nicht berechnet werden müssen. Jeder andere Eintrag von $A^{(2)}$ ergibt sich aus der Formel

$$a_{ij}^{(2)} = -l_{i1}a_{1j} + a_{ij} , \quad \forall i, j = 2, \ldots, n-1 .$$

Um alle Koeffizienten zu berechnen, brauchen wir also $(n-1)^2$-Multiplikationen. Für Untermatrizen geht es analog.

Dazu kommen noch $(n-k)$ Division um den k-ten Spaltenvektor l_{jk} $([a_{jk}^{(k)}]/a_{kk}^{(k)})$ zu bestimmen. Insgesamt ergibt sich also ein Gesamtaufwand in der LR-Zerlegung von

$$\sum_{k=1}^{n-1}(n-k+1)(n-k) = \sum_{j=1}^{n-1}(j+1)j = \frac{(n-1)n(2n-1)}{6} + \frac{n(n-1)}{2}$$
$$= \frac{1}{3}n^3 + \mathcal{O}(n^2)$$

Multiplikationen und Divisionen.

Der Aufwand zur Berechnung der eigentlichen Lösung ist gegenüber dem Aufwand der Berechnung der LR-Zerlegung vernachlässigbar: Dazu ist zunächst für jeden Eintrag von L, der von Null und Eins verschieden ist, eine Multiplikation erforderlich. Die gleiche Anzahl von Multiplikationen werden bei der Rückwärtssubstitution mit R gebraucht, zuzüglich n Divisionen durch die Diagonalelemente. Insgesamt ergibt sich also ein Aufwand von Multiplikationen und Divisionen von

$$\frac{(n-1)^2}{2}(\text{Vorwärts-}) + \frac{(n-1)^2}{2} + n(\text{Rückwärtssubstitution}) = n^2 - n + 1 .$$

Multiplikationen bzw. Divisionen. Wie immer wird der Aufwand der Additionen und Subtraktionen vernachlässigt.

Bewährt hat sich der Gaußalgorithmus mit *partieller* Pivotsuche, bei der im i-ten Teilschritt das Element $a_{ji}^{(i)}$ mit

$$j = \text{argmax}_{i \le k \le n} \frac{\left|a_{ki}^{(i)}\right|}{\sum_{l=i}^{n} \left|a_{kl}^{(i)}\right|} .$$

gewählt wird.

Werden vor dem i-ten Eliminationsschritt beispielsweise die i-te und die j-te Zeile vertauscht, dann erhält man folgenden Eliminationsschritt

$$A^{(i+1)} = L^{(i)} P_{i,j} A^{(i)} , \tag{2.5}$$

wobei

$$P_{i,j} = \left[\begin{array}{ccc|ccc|ccc} 1 & & & & & & & & \\ & \ddots & & & & & & & \\ & & 1 & & & & & & \\ \hline & & & 0 & & & 1 & & \\ & & & & 1 & & & & \\ & & & & & \ddots & & & \\ & & & & & & 1 & & \\ & & & 1 & & & 0 & & \\ \hline & & & & & & & 1 & \\ & & & & & & & & \ddots \\ & & & & & & & & & 1 \end{array}\right] . \tag{2.6}$$

Es gelten folgende Rechenregeln:

- $P_{i,j}A$ entspricht einer Vertauschung der i-ten und j-ten *Zeile* von A.
- $AP_{i,j}$ entspricht einer Vertauschung der i-ten und j-ten *Spalte* von A.
- $P_{i,j}^2 = I$.

Lemma 2.4. *Sei $k < i$, $P_{i,j}$ wie in (2.6) und $L^{(k)} = I - l_k e_k^t$ (wie in (2.1)). Dann ist $P_{i,j} L^{(k)} = \tilde{L}^{(k)} P_{i,j}$, wobei $\tilde{L}^{(k)}$ bis auf eine Vertauschung von l_{ik} und l_{jk} wieder die Form (2.1) hat.*

Beweis. Aus $P_{i,j}^2 = I$ folgt

$$P_{i,j}L^{(k)} = P_{i,j}L^{(k)}P_{i,j}^2 = (P_{i,j}L^{(k)}P_{i,j})P_{i,j};$$

Wir zeigen, dass $\tilde{L}^{(k)} := P_{i,j}L^{(k)}P_{i,j}$ die behauptete Gestalt hat. Dazu verwenden wir, dass

$$\tilde{L}^{(k)} = (P_{i,j}L^{(k)})P_{i,j}$$

$$= \left[\begin{array}{ccc|ccc|c} \ddots & & & & & & \\ & 1 & & & & & \\ & -l_{k+1,k} & 1 & & & & \\ & \vdots & \ddots & & & & \\ \hline & -l_{j,k} & & 0 & & 1 & \\ & \vdots & & & 1 & & \\ & \vdots & & & & \ddots & \\ & \vdots & & & & 1 & \\ & -l_{ik} & & 1 & & 0 & \\ \hline & \vdots & & & & & \ddots \\ & -l_{n,k} & & & & & 1 \end{array}\right] P_{i,j}$$

$$= \left[\begin{array}{ccc|ccc|c} \ddots & & & & & & \\ & 1 & & & & & \\ & -l_{k+1,k} & 1 & & & & \\ & \vdots & \ddots & & & & \\ \hline & -l_{j,k} & & 1 & & 0 & \\ & \vdots & & & 1 & & \\ & \vdots & & & & \ddots & \\ & \vdots & & & & 1 & \\ & -l_{ik} & & 0 & & 1 & \\ \hline & \vdots & & & & & \ddots \\ & -l_{n,k} & & & & & 1 \end{array}\right].$$

□

Mit diesem Lemma zeigen wir Konvergenz des Gauß-Algorithmus:

Satz 2.5. *Ist A regulär, dann definiert der Gauß-Algorithmus mit partieller Pivotsuche eine Zerlegung der Matrix $PA = \tilde{L}R$, wobei $R = A^{(n)}$ eine rechte obere Dreiecksmatrix ist, für die gilt $A^{(i+1)} = L^{(i)}P^{(i)}A^{(i)}$, und $P^{(i)}$ eine Permuationsmatrix ist. Die linke untere Dreiecksmatrix ergibt sich durch Vertauschen geeigneter Elemente in den Spalten der Matrix L aus Lemma 2.1.*

Beweis. Nehmen wir zuerst an, dass der Gauß-Eliminiationsalgorithmus mit partieller Pivotsuche niemals zu einem Pivotelement 0 führt. Dann ergibt sich aus (2.5):

$$\begin{aligned} R = A^{(n)} &= L^{(n-1)}P^{(n-1)}A^{(n-1)} = L^{(n-1)}P^{(n-1)}L^{(n-2)}P^{(n-2)}L^{(n-3)}P^{(n-3)}\cdots A \\ &= L^{(n-1)}\tilde{L}^{(n-2)}P^{(n-1)}P^{(n-2)}L^{(n-3)}P^{(n-3)}\cdots A \\ &= L^{(n-1)}\tilde{L}^{(n-2)}\tilde{L}^{(n-3)}P^{(n-1)}P^{(n-2)}P^{(n-3)}\cdots A\,. \end{aligned}$$

Setzen wir $\tilde{L}^{(n-1)} = L^{(n-1)}$, so ergibt sich

$$R = \tilde{L}^{(n-1)}\cdots\tilde{L}^{(1)}P^{(n-1)}\cdots P^{(1)}A = \tilde{L}^{(n-1)}\cdots\tilde{L}^{(1)}PA\,.$$

D.h., auch die Elemente von $\tilde{L}$ unterscheiden sich von den Elementen von L aus Lemma 2.1 lediglich durch Permutationen.

Zu klären bleibt schließlich noch, dass alle Pivotelemente von Null verschieden sind. Wäre etwa das Pivotelement nach dem i-ten Teilschritt tatsächlich Null, dann gilt aufgrund der Auswahlregel $a_{j,i}^{(i)} = 0$, $\forall j \geq i$. Und somit gilt

$$A^{(i)} = \left[\begin{array}{ccc|ccc} a_{1,1} & \cdots & & a_{1,i} & \cdots & a_{1,n} \\ & \ddots & & \vdots & & \vdots \\ \hline & & & 0 & a_{i,i+1} & \cdots & a_{i,n} \\ & & & \vdots & \vdots & \vdots & \vdots \\ & & & 0 & a_{n,i+1} & \cdots & a_{n,n} \end{array}\right].$$

Die Determinante des rechten unteren quadratischen Blocks ist somit Null und daher ist auch die Determinante von $A^{(i)}$ Null. Durch den Produktsatz für Determinanten folgt daraus aber

$$\begin{aligned} 0 = \det(A^{(i)}) &= \det(L^{(i-1)}P^{(i-1)}\cdots L^{(1)}P^{(1)}A) \\ &= \left(\prod_{j=1}^{n}\det(L^{(j)})\prod_{j=1}^{n}\det(P^{(j)})\right)\det(A)\,. \end{aligned}$$

Da die Determinante einer unteren Dreiecksmatrix das Produkt der Diagonaleinträge ist, ist $\det(L^{(j)}) = 1$. Die Permutationsmatrizen $P^{(j)}$ erfüllen $\det(P^{(j)}) = \pm 1$. Somit folgt aus obiger Gleichung, dass A singulär ist. Was im Widerspruch zu unserer Annahme steht. □

Für strikt diagonaldominante Matrizen kann auf die partieller Pivotsuche verzichtet werden, da ohnehin niemals Zeilen vertauscht werden.

Definition 2.6. Eine Matrix A heißt *strikt diagonaldominant*, falls

$$|a_{ii}| > \sum_{j \neq i} |a_{ij}| \,, \qquad \forall i = 1, \cdots, n \,.$$

Satz 2.7. *Ist A strikt diagonaldominant, dann wählt die Pivotsuche in jedem Eliminiationsschritt das Diagonalelement $a_{i,i}^{(i)}$ als Pivotelement aus. Insbesondere existiert also eine LR-Zerlegung von A und A ist nicht singulär.*

Mit der partieller Pivotsuche arbeitet der Gauß-Algorithmus in der Praxis sehr zuverlässig, obwohl immer noch Beispiele konstruiert werden können, bei denen der Algorithm instabil wird.

In solchen Ausnahmefällen kann man die *Totalpivotsuche* verwenden. Dabei wählt man vor dem i-ten Eliminiationsschritt aus dem gesamten rechten unteren Matrixblock (also aus den Indizes (j,k) mit $i \leq j,k \leq n$) das Element $a_{j,k}^{(i)}$ als Pivot-Element aus, das betragsmäßig am *größten* ist. Das entsprechende Element (etwa $a_{j,k}^{(i)}$) wird an die (i,i)-te Position gebracht, indem wie zuvor die Zeilen j und i und zusätzlich noch die Spalten k und i vertauscht werden. Letzteres wird formal dadurch beschrieben, dass A mit einer Permutationsmatrix $Q^{(i)}$ von rechts multipliziert wird ($Q^{(i)}$ sieht wie die Permutationsmatrix in (2.6) aus, wobei k die Rolle von j übernimmt). Entsprechend zu (2.5), ergibt dies die Matrixtransformation

$$A^{(i+1)} = L^{(i)} P^{(i)} A^{(i)} Q^{(i)} \,,$$

und man erhält schließlich die LR-Zerlegung der Matrix PAQ mit $Q = Q^{(1)} \cdots Q^{(n-1)}$.

Wir fassen die wichtigsten Ergebnisse über den Gauß-Algorithmus in einer Tabelle zusammen

Verfahren	Aufwand
ohne Pivotierung	$\frac{1}{3}n^3 + \mathcal{O}(n^2)$
mit Partieller Pivotsuche	$\frac{1}{3}n^3 + \mathcal{O}(n^2)$
mit Totalpivotierung	$\frac{1}{3}n^3 + \mathcal{O}(\sum_{i=1}^{n} i^2)$

2.2 Die Cholesky-Zerlegung

Wir betrachten zunächst eine *Blockversion* der LR-Zerlegung. Dazu partitionieren wir eine gegebene Matrix $A \in \mathcal{K}^{n\times n}$ in die Form

$$A = \begin{bmatrix} A_{11} & A_{12} \\ A_{21} & A_{22} \end{bmatrix} \text{ mit regulärem } A_{11} \in \mathcal{K}^{p\times p} .$$

Dabei ist $A_{12} \in \mathcal{K}^{p\times(n-p)}$, $A_{21} \in \mathcal{K}^{(n-p)\times p}$ und $A_{22} \in \mathcal{K}^{(n-p)\times(n-p)}$. Bei der Block-LR-Zerlegung von A gehen wir analog zum vorigen Abschnitt vor und faktorisieren

$$A = \begin{bmatrix} A_{11} & A_{12} \\ A_{21} & A_{22} \end{bmatrix} = \begin{bmatrix} I & 0 \\ A_{21}A_{11}^{-1} & I \end{bmatrix} \begin{bmatrix} A_{11} & A_{12} \\ 0 & S \end{bmatrix}$$

mit

$$S = A_{22} - A_{21}A_{11}^{-1}A_{12} . \tag{2.7}$$

Definition 2.8. Die $(n-p)\times(n-p)$–Matrix S aus (2.7) heißt *Schurkomplement* von A bzgl. A_{11}.

Die Lösung eines linearen Gleichungssystems $Ax = b$ kann durch (Block)–Vorwärts– und Rückwärtssubstitution erfolgen: Dazu werden die Vektoren x und $b \in \mathcal{K}^n$ in ihren ersten p Komponenten $x_1, b_1 \in \mathcal{K}^p$ und die restlichen Komponenten $x_2, p_2 \in \mathcal{K}^{n-p}$ zerlegt, das heißt wir betrachten das System

$$\begin{bmatrix} I & 0 \\ A_{21}A_{11}^{-1} & I \end{bmatrix} \begin{bmatrix} y_1 \\ y_2 \end{bmatrix} = \begin{bmatrix} b_1 \\ b_2 \end{bmatrix} \text{ (Vorwärtssubstitution)}$$
$$\begin{bmatrix} A_{11} & A_{12} \\ 0 & S \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} = \begin{bmatrix} y_1 \\ y_2 \end{bmatrix} \text{ (Rückwärtssubstitution)) .}$$

Die Vorwärtssubstitution ergibt also Hilfsvektoren

$$\begin{aligned} y_1 &= b_1 , \\ y_2 &= b_2 - A_{21}A_{11}^{-1}b_1 \end{aligned}$$

aus denen dann durch anschließende Rücksubstitution das Ergebnis berechnet wird:

$$\begin{aligned} x_2 &= S^{-1}y_2 = S^{-1}(b_2 - A_{21}A_{11}^{-1}b_1) , \\ x_1 &= A_{11}^{-1}(b_1 - A_{12}x_2) . \end{aligned}$$

Letzteres ist allerdings nur möglich, wenn S regulär ist.

Lemma 2.9. *A sei hermitesch und positiv definit. Dann ist für jedes $1 \leq p \leq n$ die Submatrix A_{11} hermitesch und sowohl A_{11} wie S sind hermitesch und positiv definit.*

Beweis. Wegen

$$\begin{bmatrix} A_{11} & A_{12} \\ A_{21} & A_{22} \end{bmatrix} = A = A^* = \begin{bmatrix} A_{11}^* & A_{21}^* \\ A_{12}^* & A_{22}^* \end{bmatrix}$$

ergibt sich

$$A_{11} = A_{11}^*\,, \quad A_{22} = A_{22}^* \text{ und } A_{12} = A_{21}^*\,.$$

Folglich ist A_{11} hermitesch und für einen beliebigen Vektor $x \in \mathcal{K}^p$ gilt

$$0 \leq \begin{bmatrix} x \\ 0 \end{bmatrix}^* A \begin{bmatrix} x \\ 0 \end{bmatrix} = \begin{bmatrix} x \\ 0 \end{bmatrix}^* \begin{bmatrix} A_{11}x \\ A_{21}x \end{bmatrix} = x^* A_{11} x\,,$$

wobei Gleichheit wegen der positiven Definitheit von A nur dann gelten kann, wenn $x = 0$. Also ist A_{11} ebenfalls positiv definit und A_{11}^{-1} existiert. S ist somit wohldefiniert mit

$$S^* = A_{22}^* - A_{12}^* A_{11}^{-1*} A_{21}^* = A_{22} - A_{21}A_{11}^{-1}A_{12} = S\,.$$

Schließlich definieren wir für einen beliebigen Vektor $y \in \mathcal{K}^{n-p}$ den zugehörigen Vektor $x = -A_{11}^{-1}A_{12}y \in \mathcal{K}^p$ und erhalten

$$\begin{aligned} 0 &\leq \begin{bmatrix} x \\ y \end{bmatrix}^* A \begin{bmatrix} x \\ y \end{bmatrix} \\ &= \begin{bmatrix} x \\ y \end{bmatrix}^* \begin{bmatrix} A_{11}x + A_{12}y \\ A_{21}x + A_{22}y \end{bmatrix} \\ &= \begin{bmatrix} x \\ y \end{bmatrix}^* \begin{bmatrix} -A_{12}y + A_{12}y \\ -A_{21}A_{11}^{-1}A_{12}y + A_{22}y \end{bmatrix} \\ &= \begin{bmatrix} x \\ y \end{bmatrix}^* \begin{bmatrix} 0 \\ Sy \end{bmatrix} \\ &= y^* S y\,, \end{aligned}$$

wobei wiederum Gleichheit nur für $y = 0$ gelten kann. Damit ist S auch positiv definit. □

Beim Gauß-Algorithmus wird die Matrix A in das Produkt $A = LR$ einer linken unteren und einer rechten oberen Dreiecksmatrix zerlegt. Von besonderem Interesse ist der Fall $R = L^*$.

Definition 2.10. Eine Zerlegung $A = LL^*$ mit unterer Dreiecksmatrix L mit positiven Diagonaleinträgen heißt *Cholesky-Zerlegung* von A. [1]

Eine notwendige Bedingung für die Existenz einer Cholesky-Zerlegung gibt das folgende Resultat.

Proposition 2.11. *Hat A eine Cholesky-Zerlegung, dann ist A hermitesch und positiv definit.*

Beweis. Aus $A = LL^*$ folgt unmittelbar

$$A^* = (L^*)^* L^* = LL^* = A \, ;$$

Also ist A hermitesch. Ferner gilt

$$x^* Ax = x^* LL^* x = (L^* x)^* L^* x = \|L^* x\|_2^2 \geq 0 \, , \quad x \in \mathcal{K}^n \, .$$

Dabei gilt Gleichheit genau für $x = 0$, da L positive Diagonaleinträge hat und somit nicht singulär ist - dies folgt aus der Tatsache, dass die Determinante einer Dreiecksmatrix das Produkt der Diagonalelemente ist. Folglich ist A positiv definit. □

Tatsächlich ist diese Bedingung an A auch hinreichend.

Satz 2.12. *Ist A hermitesch und positiv definit, dann existiert eine Cholesky-Zerlegung von A.*

Beweis. Der Beweis wird induktiv über die Dimension der Matrix geführt, wobei für $n = 1$ die Matrix nur aus dem Element a_{11} besteht, das positiv sein muss, da die Matrix $A = a_{11}$ positiv definit ist. Also kann man für $n = 1$ einfach $L = [\sqrt{a_{11}}]$ setzen.

Sei nun die Behauptung für alle Matrizen der Dimension $n - 1$ korrekt und A eine beliebige $n \times n$ Matrix. Dann partitionieren wir

$$A = \begin{bmatrix} a_{11} & A_{12} \\ A_{21} & A_{22} \end{bmatrix} \text{ mit } A_{22} \in \mathcal{K}^{(n-1)\times(n-1)} \text{ und } A_{12} = A_{21}^* \, .$$

[1] Die Cholesky-Zerlegung soll eindeutig sein, was unter der Voraussetzung positiver Diagonaleinträge garantiert werden kann.

Nach Lemma 2.9 ist a_{11} positiv und das Schurkomplement $S = A_{22} - A_{21}A_{12}/a_{11} \in \mathcal{K}^{(n-1)\times(n-1)}$ von A bzgl. a_{11} ist hermitesch und positiv definit; daher existiert $l_{11} := \sqrt{a_{11}} > 0$ und aufgrund der Induktionsannahme hat S eine Cholesky-Zerlegung $S = L_S L_S^*$. Wir setzen

$$L = \begin{bmatrix} l_{11} & 0 \\ A_{21}/l_{11} & L_S \end{bmatrix}, \quad L^* = \begin{bmatrix} l_{11} & A_{12}/l_{11} \\ 0 & L_S^* \end{bmatrix},$$

und dann folgt

$$LL^* = \begin{bmatrix} l_{11} & 0 \\ A_{21}/l_{11} & L_S \end{bmatrix} \begin{bmatrix} l_{11} & A_{12}/l_{11} \\ 0 & L_S^* \end{bmatrix} = \begin{bmatrix} l_{11}^2 & A_{12} \\ A_{21} & B \end{bmatrix}.$$

mit

$$B = \frac{1}{l_{11}^2} A_{21}A_{12} + L_S L_S^* = \frac{1}{a_{11}} A_{21}A_{12} + S = A_{22}\,.$$

Also ist

$$LL^* = \begin{bmatrix} a_{11} & A_{12} \\ A_{21} & A_{22} \end{bmatrix} = A$$

eine Cholesky-Zerlegung von A. □

Die Berechnung der Einträge von L erfolgt sukzessive durch zeilenweise Koeffizientenvergleich bei dem Produkt $A = LL^*$,

$$\begin{bmatrix} a_{11} & a_{12} & \vdots & a_{1n} \\ a_{21} & a_{22} & \vdots & a_{2n} \\ \cdots & \cdots & \cdots & \cdots \\ a_{n1} & a_{n2} & \vdots & a_{nn} \end{bmatrix} = \begin{bmatrix} l_{11} & & \vdots & 0 \\ l_{21} & l_{22} & \ddots & \\ \cdots & & \ddots & \\ l_{n1} & l_{n2} & \vdots & l_{nn} \end{bmatrix} \begin{bmatrix} \overline{l_{11}} & \overline{l_{21}} & \vdots & \overline{l_{n1}} \\ & \overline{l_{22}} & \vdots & \overline{l_{n2}} \\ & & \ddots & \\ 0 & & & \overline{l_{nn}} \end{bmatrix}$$

Auf diese Weise ergeben sich die Einträge von L in der folgenden Weise:

$$\begin{array}{ll} a_{11} = |l_{11}|^2 & l_{11} = \sqrt{a_{11}} \\ & \\ a_{21} = l_{21}\overline{l_{11}} & l_{21} = a_{21}/\overline{l_{11}} \\ a_{22} = |l_{21}|^2 + |l_{22}|^2 & l_{22} = \sqrt{a_{22} - |l_{21}|^2} \\ & \Rightarrow \\ a_{31} = l_{31}\overline{l_{11}} & l_{31} = a_{31}/\overline{l_{11}} \\ a_{32} = l_{31}\overline{l_{21}} + l_{32}\overline{l_{22}} & l_{32} = (a_{32} - l_{31}\overline{l_{21}})/\overline{l_{22}} \\ a_{33} = |l_{31}|^2 + |l_{32}|^2 + |l_{33}|^2 & l_{33} = \sqrt{a_{33} - |l_{31}|^2 + |l_{32}|^2} \\ \vdots & \vdots \end{array}$$

Die Lösbarkeit dieser (nichtlinearen) Gleichungen ist durch den Existenzbeweis (Satz 2.12) gewährleistet, das heißt alle Quadratwurzeln existieren und die resultierenden Diagonalelemente l_{ii} von L sind ungleich Null. Aus dem Algorithmus lässt sich nun sofort folgendes Resultat ableiten

Korollar 2.13. *Die Cholesky-Zerlegung einer hermiteschen positiv definiten Matrix A ist eindeutig bestimmt.*

Wie man aus dem Algorithmus sofort sieht, ist der Aufwand zur Berechnung von l_{ij} (mit $i \geq j$) maximal j Multiplikationen, Divisionen und Wurzeln (der Aufwand zur Berechnung der Additionen wird wieder vernachlässigt). Demnach ergibt sich ein Gesamtaufwand bei der Berechnung der Cholesky-Zerlegung von

$$\sum_{j=1}^{n}(n+1-j)j = \frac{n(n+1)^2}{2} - \frac{n(n+1)(2n+1)}{6} = \frac{1}{6}n^3 + \mathcal{O}(n^2) \; .$$

Die Cholesky-Zerlegung kann genauso eingesetzt werden, wie die LR-Zerlegung. Sie hat den Vorteil, dass sie etwa nur halb so viel kostet.

2.3 Die QR-Zerlegung

Bisher haben wir uns nur mit der Zerlegung von quadratischen Matrizen beschäftigt. Es gibt aber viele Anwendungen - auf die wir später noch zurückkommen werden – die eine Zerlegung von rechteckigen Matrizen erfordern. Ein solcher Algorithmus ist die QR-Zerlegung, mit dem wir uns im folgenden beschäftigen werden. Dazu brauchen wir einige Hilfsmittel.

Definition 2.14. Sei $v \in \mathcal{K}^r \backslash \{0\}$: Dann heißt die Matrix

$$P = I - \frac{2}{v^* v} v v^* \in \mathcal{K}^{r \times r}$$

Householder-Transformation.

Lemma 2.15. *Die Householder-Transformation P ist eine hermitesche, unitäre Matrix mit*

$$Pv = -v \text{ und } Pw = w \, , \qquad \forall w \in [v]^{\perp} \; .$$

Beweis. Aus der Definition von P folgt

$$P_{ij} = 1\delta_{ij} - \frac{2}{v^*v} v_i \overline{v_j} = 1\delta_{ij} - \overline{\frac{2}{v^*v} v_j \overline{v}_i} = \overline{P_{ji}} = P_{ij}^* \; .$$

Also ist P hermitesch. Außerdem ist P unitär, denn

$$P^*P = P^2 = I - \frac{4}{v^*v} vv^* + \frac{4}{(v^*v)^2} v(v^*v)v^* = I - \frac{4}{v^*v} vv^* + \frac{4}{v^*v} vv^* = I \; .$$

Schließlich ergibt sich für den Vektor v aus der Definition von P und für beliebiges $w \perp v$ (das heißt $v^*w = 0$)

$$\begin{aligned} Pv &= Iv - \frac{2}{v^*v} v(v^*v) = v - 2v = -v \, , \\ Pw &= Iw - \frac{2}{v^*v} v(v^*w) = w - 0 = w \; . \end{aligned}$$

□

Wir konstruieren zunächst eine Householder-Matrix P, die einen beliebigen Vektor $x \in \mathcal{K}^r \backslash \{0\}$ auf ein Vielfaches von $e_1 \in \mathcal{K}^r$ transformiert. Das heißt, wir wollen einen Vektor $v \in \mathcal{K}^r \backslash \{0\}$ finden, sodass gilt

$$Px = x - \frac{2}{v^*v} v(v^*x) = \zeta e_1 \; . \tag{2.8}$$

Aus (2.8) folgt

$$x - \zeta e_1 = \frac{2v^*x}{\|v\|_2^2} v \; . \tag{2.9}$$

Das heißt, dass v proportional zu $x - \zeta e_1$ ist. Tatsächlich erfüllt jedes $v = \lambda(x - \zeta e_1)$, $\lambda \neq 0$ die Gleichung (2.8), wenn $x - \zeta e_1$ die Gleichung (2.8) erfüllt. Setzt man $v = \lambda(x - \zeta e_1)$ in (2.9) ein, so erhält man

$$x - \zeta e_1 = \frac{2\|x\|_2^2 - 2\zeta x_1}{\|x\|_2^2 - 2\zeta x_1 + \zeta^2} (x - \zeta e_1) \, ,$$

Und das impliziert

$$|\zeta| = \|x\|_2 \; . \tag{2.10}$$

Folgende Wahl on (ζ, λ) ist üblich, aber jede andere funktioniert genau so,

$$\zeta = \begin{cases} -\|x\|_2 & \text{für } x_1 = 0 \\ -\frac{x_1}{|x_1|}\|x\|_2 & \text{für } x_1 \neq 0 \end{cases} \quad \text{und } \lambda = \frac{1}{\|x\|_2} \; .$$

Mit dieser Wahl gilt:

$$v = \begin{cases} x/\|x\|_2 + e_1 & \text{für } x_1 = 0\,, \\ \frac{1}{\|x\|_2}\left(x + \frac{x_1\|x\|_2}{|x_1|}e_1\right) & \text{für } x_1 \neq 0 \end{cases}$$

und $Px = \zeta e_1$.

Satz 2.16. *Sei $A \in \mathcal{K}^{m\times n}$ mit $m \geq n$ und Rang$(A) = n$. Dann existiert eine unitäre Matrix $Q \in \mathcal{K}^{m\times m}$ und einer obere Dreiecksmatrix $R \in \mathcal{K}^{m\times n}$ mit*

$$A = QR = Q \left[\begin{array}{ccc} r_{11} & \cdots & r_{1n} \\ & \ddots & \vdots \\ 0 & & r_{nn} \\ \hline 0 & \cdots & 0 \end{array}\right] .$$

Dabei sind $r_{11}, \cdots, r_{nn}$ jeweils von Null verschieden.

Beweis. Wir bestimmen die gesuchte Zerlegung, indem wir in jedem Schritt eine Householder-Transformation von links an A heranmultiplizieren, um sukzessive die Spalten 1 bis n von R zu erhalten. Dies ergibt dann die Darstellung

$$P^{(n)} \cdots P^{(1)} A = R \tag{2.11}$$

mit Householder-Transformationen $P^{(i)}$, und daraus folgt dann die QR-Faktorisierung

$$A = QR \text{ mit } Q = P^{(1)*} \cdots P^{(n)*} = P^{(1)} \cdots P^{(n)} .$$

Im ersten Schritt setzen wir $A^{(1)} = A$ und für x die erste Spalte a_1 von $A^{(1)}$ und bestimmen die Householder-Transformation $P^{(1)} \in \mathcal{K}^{m\times m}$ gemäß (2.9). Es folgt

$$P^{(1)} a_1 = r_{11} e_1\,, \quad r_{11} = \pm\|a_1\|_2 \neq 0\,,$$

beziehungsweise

$$P^{(1)} A = \left[\begin{array}{c|ccc} r_{11} & * & * & * \\ \hline 0 & & A^{(2)} & \end{array}\right] \text{ mit } A^{(2)} \in \mathcal{K}^{(m-1)\times(n-1)} .$$

Nehmen wir an, dass wir nach i Schritten Householder-Matrizen $P^{(1)}, \cdots, P^{(i)}$ konstruiert haben mit

$$P^{(i)} \cdots P^{(1)} A = \left[\begin{array}{ccc|c} r_{11} & \cdots & r_{1i} & \\ & \ddots & \vdots & R^{(i)\prime} \\ 0 & & r_{ii} & \\ \hline 0 & \cdots & 0 & A^{(i+1)} \end{array}\right] \tag{2.12}$$

wobei $R^{(i)\prime} \in \mathcal{K}^{i\times(n-i)}$ und $A^{(i+1)} \in \mathcal{K}^{(m-i)\times(n-i)}$.

Im nächsten Schritt können wir daher für $x \in \mathcal{K}^{m-i}$ die erste Spalte a_{i+1} von $A^{(i+1)}$ wählen und konstruieren die Householder-Matrix $P^{(i+1)\prime} \in \mathcal{K}^{(m-i)\times(m-i)}$ über einen Vektor $v' \in \mathcal{K}^{m-i}$ gemäß (2.9). Auf diese Weise ergibt sich

$$P^{(i+1)\prime}A^{(i+1)} = \left[\begin{array}{c|ccc} r_{i+1,i+1} & * & * & * \\ \hline 0 & \multicolumn{3}{c}{A^{(i+2)}} \end{array}\right]$$

mit $r_{i+1,i+1} = \pm\|a_{i+1}\|_2 \neq 0$. Somit folgt mit

$$P^{(i+1)} := \left[\begin{array}{c|c} I & 0 \\ \hline 0 & P^{(i+1)\prime} \end{array}\right]$$

$$P^{(i+1)}P^{(i)}\cdots P^{(1)}A = \left[\begin{array}{ccc|cc} r_{11} & \cdots & r_{1i} & \multicolumn{2}{c}{\multirow{2}{*}{$R^{(i)\prime}$}} \\ & \ddots & \vdots & & \\ \hline 0 & & r_{ii} & \multicolumn{2}{c}{} \\ \hline 0 & \cdots & 0 & r_{i+1,i+1} & * \ * \ * \\ \hline 0 & \cdots & 0 & 0 & A^{(i+2)} \end{array}\right].$$

Man beachte, dass sich in jedem Schritt die ersten i Zeilen *nicht* verändern. □

Der Beweis ist konstruktiv und ein Algorithmus kann wie im Beweis implementiert werden.

Der Gesamtaufwand der QR Zerlegung ist

$$mn^2 - \frac{1}{3}n^3 + \mathcal{O}(mn) .$$

Bemerkung 2.17. Vergleich der Aufwände bei quadratischen Matrizen:

QR	$\frac{2}{3}n^3 + \mathcal{O}(n^2)$
LR	$\frac{1}{3}n^3 + \mathcal{O}(n^2)$
Cholesky	$\frac{1}{6}n^3 + \mathcal{O}(n^2)$

Kapitel 3

Iterationsverfahren

Wenn die Matrizen sehr groß sind, verbieten sich Eliminationsverfahren wegen ihres hohen Aufwands. Zudem sind die großen, in der Praxis auftretenden Systeme meist dünn besetzt, d.h. nur wenige (etwa 10 Einträge pro Zeile) sind ungleich Null. Typische Beispiele sind Steifigkeitsmatrizen, die bei der Lösung von partiellen Differentialgleichungen auftreten. Obwohl die Matrix eines solchen Problems wegen der Dünnbesetztheit noch gut in den Speicher passen mag, trifft dies für die Faktorisierung L und R nicht mehr zu. In solchen Fällen behilft man sich gerne mit Iterationsverfahren, die das Gleichungssystem zwar nicht exakt, aber hinreichend genau lösen.

Bevor wir konkrete Verfahren vorstellen, wiederholen wir ein fundamentales Resultat aus der Analysis, den Banachschen Fixpunktsatz:

Satz 3.1. *Sei $\Phi : \mathcal{K} \to \mathcal{K}$ eine (nichlineare) bzgl. $\|\cdot\|$ kontrahierende Selbstabbildung einer abgeschlossenen Teilmenge $\mathcal{K} \subseteq \mathcal{K}^n$, d.h.,*

$$\|\Phi(x) - \Phi(y)\| \leq q\|x - y\| \text{ für ein } q < 1 \text{ und alle } x, y \in \mathcal{K} .$$

Dann hat die Fixpunktgleichung $x = \Phi(x)$ genau eine Lösung $\hat{x} \in \mathcal{K}$, und die Iterationsfolge $\{x^{(k)}\}$ mit $x^{(0)} \in \mathcal{K}$ beliebig, $x^{(k+1)} = \Phi(x^{(k)})$ für $k = 0, 1, 2, \cdots$, konvergiert gegen $\hat{x}$ für $k \to \infty$. Darüberhinaus ist für $k \geq 1$

1. $\|x^{(k)} - \hat{x}\| \leq q\|x^{(k-1)} - \hat{x}\|$ *(Monotonie);*
2. $\|x^{(k)} - \hat{x}\| \leq \frac{q^k}{1-q}\|x^{(1)} - x^{(0)}\|$ *(a-priori Schranke);*
3. $\|x^{(k)} - \hat{x}\| \leq \frac{q}{1-q}\|x^{(k)} - x^{(k-1)}\|$ *(a-posteriori Schranke);*

Beweis. 1. Wir wählen einen beliebigen Startwert $x^{(0)} \in \mathcal{K}$ und betrachten die durch $x^{(k+1)} = \Phi(x^{(k)})$, $k = 0, 1, 2, \cdots$, definierte Iterationsreihenfolge. Aufgrund der Kontraktionseigenschaft von Φ gilt für ein beliebiges $k \in \mathbb{N}$, dass

$$\|x^{(k+1)} - x^{(k)}\| = \|\Phi(x^{(k)}) - \Phi(x^{(k-1)})\| \leq q\|x^{(k)} - x^{(k-1)}\| \,. \tag{3.1}$$

Damit ergibt sich induktiv

$$\|x^{(k+1)} - x^{(k)}\| \leq q^k \|x^{(1)} - x^{(0)}\| \,, \quad k \in \mathbb{N} \,. \tag{3.2}$$

Wir zeigen nun, dass die Folge $x^{(k)}$ eine Cauchy-Folge ist. Dazu wählen wir $m, l \in \mathbb{N}$ mit $l > m$ und erhalten aus (3.2)

$$\begin{aligned} \|x^{(l)} - x^{(m)}\| &\leq \|x^{(l)} - x^{(l-1)}\| + \|x^{(l-1)} - x^{(l-2)}\| + \cdots \\ &\quad + \|x^{(m+1)} - x^{(m)}\| \\ &\leq (q^{l-1} + q^{l-2} + \cdots + q^m)\|x^1 - x^{(0)}\| \\ &\leq q^m \frac{1}{1-q} \|x^{(1)} - x^{(0)}\| \,. \end{aligned} \tag{3.3}$$

Da q^m für $m \to \infty$ gegen Null konvergiert, wird der letzte Ausdruck mit hinreichend großem m kleiner als jede positive Zahl ε. Daher ist $\{x^{(k)}\}$ eine Cauchy-Folge mit Grenzwert x. Da alle Iterierten wegen der Selbstabbildungseigenschaft in der *abgeschlossenen* Menge $\mathcal{K}$ bleiben, gehört auch x zu $\mathcal{K}$.

2. Als nächstes wird gezeigt, dass x ein Fixpunkt von Φ ist. Dazu beachte man zunächst, dass Φ (Lipschitz)-stetig ist. Folglich kann man in der Rekursion $x^{(k+1)} = \Phi(x^{(k)})$ den Iterationsindex k gegen ∞ gehen lassen und während die linke Seite gegen x konvergiert, konvergiert die rechte Seite wegen der Stetigkeit von Φ gegen $\Phi(x)$. Also ist $x = \Phi(x)$, bzw. x ist ein Fixpunkt von Φ. Damit ist die Existenz eines Fixpunktes nachgewiesen.

3. Die Eindeutigkeit folgt aus der Kontraktionseigenschaft: Angenommen $\hat{x}$ und x seien zwei Fixpunkte von Φ in $\mathcal{K}$; dann folgt

$$\|x - \hat{x}\| = \|\Phi(x) - \Phi(\hat{x})\| \leq q\|x - \hat{x}\| \,,$$

und dies kann wegen der Voraussetzung $q < 1$ nur gelten, wenn $\|x - \hat{x}\| = 0$, also $x = \hat{x}$ ist. Mit anderen Worten: Φ hat in $\mathcal{K}$ nur einen

Fixpunkt $\hat{x}$ und die Iterationsreihenfolge $\{x^{(k)}\}$ konvergiert für jedes $x^{(0)}$ gegen $\hat{x}$.

4. Es verbleibt der Nachweis der drei Fehlerabschätzungen. Die erste Ungleichung folgt wie vorher die Eindeutigkeit:

$$\|x^{(k)} - \hat{x}\| = \|\Phi(x^{(k-1)}) - \Phi(\hat{x})\| \leq q\|x^{(k-1)} - \hat{x}\| \,.$$

Die zweite Ungleichung folgt aus (3.3): Demnach ist für $m > k$

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

und die Behauptung ergibt sich durch Grenzübergang $m \to \infty$. Zum Beweis der dritten Ungleichung schätzen wir die linke Seite von (3.1) mit der Dreiecksungleichung ab und erhalten zusammen mit $q < 1$

$$\begin{aligned} \|x^{(k+1)} - x^{(k)}\| &\geq \|x^{(k)} - \hat{x}\| - \|x^{(k+1)} - \hat{x}\| \\ &\geq \|x^{(k)} - \hat{x}\| - q\|x^{(k)} - \hat{x}\| \\ &= (1-q)\|x^{(k)} - \hat{x}\| \,. \end{aligned}$$

Eingesetzt in (3.1) folgt daraus die Behauptung des Satzes.

□

Der Banachsche Fixpunktsatz lässt sich nun wie folgt zur Konstruktion konvergenter Iterationsverfahren zur Lösung nichtsingulärer linearer Gleichungssysteme $Ax = b$ mit $A \in \mathcal{K}^{n\times n}$ und $b \in \mathcal{K}^n$ verwenden: Man wählt eine additive Zerlegung von A,

$$A = M - N \,,$$

wobei M invertierbar sein soll und bringt die Gleichung $Ax = b$ auf *Fixpunktgestalt*

$$Mx = Nx + b \text{ bzw. } x = Tx + c \tag{3.4}$$

mit $T = M^{-1}N$ und $c = M^{-1}b$; die rechte Seite $Tx + c$ entspricht also der (hier affin linearen) Funktion $\Phi(x)$ aus Satz 3.1.

Algorithmus 3.2. *(Allgemeines Iterationsprinzip für lineare Gleichungssysteme)*

- `Wähle` $A = M - N$ `mit invertierbarem` M `und` $x^{(0)} \in \mathcal{K}^n$ `beliebig`

- `for` $k = 1, \cdots$ `löse`

$$Mx^{(k)} = Nx^{(k-1)} + b\,. \tag{3.5}$$

Es ist offensichtlich, dass ein solches Verfahren nur dann sinnvoll ist, wenn Gleichungssysteme mit der Matrix M erheblich einfacher zu lösen sind als Gleichungssysteme mit A, und wenn die Matrix-Vektor-Multiplikation mit N billig ist (etwa, wenn N dünnbesetzt ist). Zur Konvergenz dieses Verfahrens gibt der Banachsche Fixpunktsatz die folgende Auskunft:

Satz 3.3. *Ist $\|\cdot\|_M$ eine Matrixnorm, die mit der Vektornorm $\|\cdot\|$ verträglich ist, und ist*

$$\left\|M^{-1}N\right\|_M < 1\,,$$

dann konvergiert das Iterationsverfahren (3.5) für jedes $x^{(0)}$ gegen $A^{-1}b$.

Beweis. Wir setzen $\Phi(x) = Tx + c$ mit $T = M^{-1}N$ und $c = M^{-1}b$. Aus (3.4) ist offensichtlich, dass alle Lösungen von $Ax = b$ auch Fixpunkte der Fixpunktgleichung $x = \Phi(x)$ sind und umgekehrt. $\mathcal{K} = \mathcal{K}^n$ ist abgeschlossen und wegen der Linearität von T folgt

$$\|\Phi(x) - \Phi(z)\| = \|T(x - z)\| \le \|T\|_M \|x - z\|\,,$$

und damit ist auch die zweite Voraussetzung des Banachschen Fixpunktsatzes mit $q = \|M^{-1}N\|_M$ erfüllt. Also konvergiert die Folge $\{x^{(k)}\}$ aus (3.5) gegen den eindeutigen Fixpunkt $\hat{x} = T\hat{x} + c$, also die eindeutige Lösung $\hat{x} = A^{-1}b$ des linearen Gleichungssystems. □

Um die folgenden Konvergenzresultat zu zeigen brauchen wir einige Eigenschaften des Spektralradius einer Matrix.

Lemma 3.4. *Sei $\|\cdot\|$ eine Vektornorm und $|||\cdot|||$ die induzierte Norm. Weiters sei $S \in \mathcal{K}^{n\times n}$ eine reguläre Matrix. Dann wird durch $\|x\|_S = \|Sx\|$ eine Norm auf $\mathcal{K}^n$ definiert. Darüberhinaus ist die von $\|\cdot\|_S$ induzierte Norm gegeben durch*

$$|||A|||_S = \left|\left|\left|SAS^{-1}\right|\right|\right|\,.$$

Beweis. Einfaches nachrechnen. □

Lemma 3.5. *Sei $A \in \mathcal{K}^{n\times n}$ und*

$$\begin{bmatrix} J_1 & & 0 \\ & \ddots & \\ 0 & & J_k \end{bmatrix} = J = V^{-1}AV$$

mit

$$J_j = \begin{bmatrix} \lambda_j & 1 & & 0 \\ & \lambda_j & \ddots & \\ & & \ddots & 1 \\ 0 & & & \lambda_j \end{bmatrix}, \quad \forall j = 1, \ldots, k$$

die Jordansche Normalform.

Weiters sei $S = D^{-1}V^{-1}$ mit

$$D := \begin{bmatrix} \varepsilon & & & 0 \\ & \varepsilon^2 & & \\ & & \ddots & \\ 0 & & & \varepsilon^n \end{bmatrix} \text{ mit } 0 < \varepsilon < 1 \,.$$

Dann gilt

$$|||A|||_S \leq \rho(A) + \varepsilon \,,$$

wobei $|||\cdot|||_S$ die durch $\|\cdot\|_\infty$ und S induzierte Norm ist. [1]

Beweis. Für den Beweis schreiben wir die Eigenwerte $(\lambda_j)_{1\leq j\leq k}$ mit deren Vielfachheit und bezeichnen diese mit $(\mu_i)_{1\leq i\leq n}$und schreiben

$$J = \begin{bmatrix} J_1 & & 0 \\ & \ddots & \\ 0 & & J_k \end{bmatrix} = \begin{bmatrix} \mu_1 & 1 & & 0 \\ 0 & \mu_s & 1(0) & \\ & & \ddots & \\ 0 & & & \mu_n \end{bmatrix} .$$

In obiger Identität bedeutet 1(0), dass das Matrixelement entweder den Wert 1 oder 0 annimmt.

[1] Beachte, dass $\|\cdot\|_{\infty,1}$ durch $\|\cdot\|_\infty$ induziert wird. Damit ist $|||A|||_S = \left\|SAS^{-1}\right\|_{\infty,1}$.

Damit gilt

$$\begin{aligned} SAS^{-1} &= D^{-1}V^{-1}AVD \\ &= D^{-1}JD \\ &= \begin{bmatrix} \varepsilon^{-1} & & & 0 \\ & \varepsilon^{-2} & & \\ & & \ddots & \\ 0 & & & \varepsilon^{-n} \end{bmatrix} \begin{bmatrix} \mu_1 & 1 & & 0 \\ 0 & \mu_s & 1(0) & \\ & & \ddots & \\ 0 & & & \mu_n \end{bmatrix} \begin{bmatrix} \varepsilon & & & 0 \\ & \varepsilon^2 & & \\ & & \ddots & \\ 0 & & & \varepsilon^n \end{bmatrix} \\ &= \begin{bmatrix} \mu_1 & \varepsilon & & 0 \\ & \mu_2 & \varepsilon(0) & 0 \\ & \ddots & & \varepsilon \\ 0 & & & \mu_n \end{bmatrix} . \end{aligned}$$

This shows that

$$\left\|SAS^{-1}\right\|_{\infty,1} \leq \max_{1\leq i\leq m} (|\mu_j| + \varepsilon) = \max_{1\leq j\leq k} (|\lambda_j| + \varepsilon) = \rho(A) + \varepsilon \, .$$

□

Korollar 3.6. *Sei $A = M - N$ invertierbar und $T = M^{-1}N$. Dann konvergiert das Iterationsverfahren (3.5) genau dann für jedes $x^{(0)}$ gegen $\hat{x} = A^{-1}b$, wenn $\rho(T) < 1$ erfüllt ist.*

Beweis. Da auf $\mathcal{K}^n$ alle Normen äquivalent sind, reicht es zu zeigen, dass (3.5) bzgl. $\|\cdot\|_\infty$ für jedes $x^{(0)}$ konvergiert, wenn $\rho(T) < 1$ gilt. Die durch $\|\cdot\|_\infty$ induzierte Matrixnorm ist $\|\cdot\|_{\infty,1}$ (vgl. Beispiel 1.5). Damit existiert wegen Lemma 3.5 eine Vektornorm $\|\cdot\|_\varepsilon$ und eine dadurch induzierte Matrixnorm $|||\cdot|||_\varepsilon$ mit

$$|||T|||_\varepsilon \leq \rho(T) + \varepsilon \, .$$

Somit, falls $\rho(T) < 1$ ist, dann gilt für hinreichend kleines ε, $q := |||T|||_\varepsilon \leq \rho(T) + \varepsilon < 1$. Damit ergibt sich eine Beweisrichtung aus Satz 3.3.

Ist umgekehrt $\rho(T) \geq 1$, dann existiert ein Eigenwert λ von T mit $|\lambda| \geq 1$ und zugehörigem Eigenvektor $z \neq 0$. Wählt man $x^{(0)} = A^{-1}b + z$, dann ergibt sich

$$\begin{aligned} x^{(k)} - \hat{x} &= Tx^{(k-1)} + c - \hat{x} \\ &= M^{-1}(Nx^{(k-1)} + b - M\hat{x}) \\ &= M^{-1}(Nx^{(k-1)} - N\hat{x}) \\ &= T(x^{(k-1)} - \hat{x}) \, . \end{aligned}$$

Daraus folgt mit Induktion

$$x^{(k)} - \hat{x} = T^k(x^{(0)} - \hat{x}) = T^k z = \lambda^k z \; . \tag{3.6}$$

Folglich ist $\|x^{(k)} - \hat{x}\| = |\lambda|^k \|z\| \geq \|z\|$ und $x^{(k)}$ konvergiert für $k \to \infty$ nicht gegen $A^{-1}b$. □

Dem Spektralradius von T kommt also bei der Iteration (3.5) eine besondere Bedeutung zu. Gemäß Korollar 3.6 entscheidet der Spektralradius über Konvergenz und Divergenz der Iterationsfolge. Im folgenden studieren wir zwei spezielle Iterationsverfahren, die von großer Bedeutung bei der Lösung von linearen Gleichungssystemen sind:

3.1 Einzel- und Gesamtschrittverfahren

Das einfachste Beispiel eines Iterationsverfahrens zur Lösung eines linearen Gleichungssystems $Ax = b$ mit $A = [a_{ij}]_{ij} \in \mathcal{K}^{n\times n}$ und $b = [b_i]_i \in \mathcal{K}^n$ ist das **Gesamtschrittverfahren** (oder *Jacobi-Verfahren*). Bezeichnen wir mit $x^{(k)} = [x_j^{(k)}]_j \in \mathcal{K}^n$ die Iterierten dieses Verfahrens, dann lautet die Iterationsvorschrift wie folgt:

Algorithmus 3.7. *(Gesamtschrittverfahren)*
`Wähle` $x^{(0)} \in \mathcal{K}^n$ `beliebig.`
`for` $k = 0, 1, \cdots$

- `for` $i = 1, \cdots, n$

$$x_i^{(k+1)} = \frac{1}{a_{ii}} \left(b_i - \sum_{j \neq i} a_{ij} x_j^{(k)} \right) \tag{3.7}$$

`end for`

`until stop`

Offensichtlich muss für die Durchführbarkeit dieser Iterationsvorschrift $a_{ii} \neq 0 \, , i = 1, \cdots, n$ vorausgesetzt werden.

Die Frage nach der Konvergenz werden wir auf Satz 3.3 zurückführen. Dazu zerlegen wir

$$A = D - L - R$$

in eine Diagonal- und eine strikte linke untere und eine strikte rechte obere Dreiecksmatrix. Dann können die n Gleichungen (3.7), $i = 1, \cdots, n$, als eine Vektorgleichung

$$x^{(k+1)} = D^{-1}(b + (L + R)x^{(k)}) \tag{3.8}$$

geschrieben werden. Somit entspricht das Gesamtschrittverfahren der Fixpunktiteration (3.5) mit $M = D$ und $N = L + R$. Die entsprechende Iterationsmatrix

$$\mathcal{J} = M^{-1}N = D^{-1}(L + R)$$

wird *Gesamtschrittoperator* genannt.

Beim *Einzelschritt-* oder *Gauß-Seidel-Verfahren* setzt man in (3.7) alle bereits berechneten Komponenenten von $x^{(k+1)}$ auf der rechten Seite ein:

Algorithmus 3.8. *(Einzelschrittverfahren)* `Wähle` $x^{(0)} \in \mathcal{K}^n$ `beliebig.`
`for` $k = 0, 1, \cdots$

- `for` $i = 1, \cdots, n$

$$x_i^{(k+1)} = \frac{1}{a_{ii}} \left(b_i - \sum_{j<i} a_{ij} x_j^{(k+1)} - \sum_{j>i} a_{ij} x_j^{(k)} \right) \tag{3.9}$$

  `end for`

`until stop`

Der Aufwand ist somit der gleiche wie beim Gesamtschrittverfahren. Entsprechend zu (3.8) erhält man die Matrixformulierung des Einzelschrittverfahren, indem man alle Komponenten von $x^{(k+1)}$ in (3.9) auf die linke Seite bringt. Dann folgt:

$$a_{ii} x_i^{(k+1)} + \sum_{j<i} a_{ij} x_j^{(k+1)} = b_i - \sum_{j>i} a_{ij} x_j^{(k)} \quad i = 1, \cdots n \, .$$

Insbesondere ergibt sich $x^{(k+1)}$ durch Auflösen des Dreieckssystems

$$(D - L)x^{(k+1)} = b + Rx^{(k)} \, . \tag{3.10}$$

Wiederum haben wir eine Fixpunktiteration der Form (3.5), diesmal mit $M = D - L$ und $N = R$. $\mathcal{L} = (D - L)^{-1}R$ wird **Einzelschrittoperator** genannt.

Mit Hilfe des Banachschen Fixpunktsatzes können folgende Aussagen über Konvergenz von Einzel-und Gesamtschrittverfahren bewiesen werden.

Satz 3.9. *Ist A strikt diagonaldominant, dann konvergiert Gesamt- und Einzelschrittverfahren für jeden Startvektor $x^{(0)}$ gegen die eindeutige Lösung von $Ax = b$.*

Beweis. Wir wenden den Satz 3.3 an.

Zunächst betrachten wir das Gesamtschrittverfahren. Aus der strikten Diagonaldominanz von A folgt, dass

$$\|\mathcal{J}\|_{\infty,1} = \|D^{-1}(L+R)\|_{\infty,1} = \max_{i=1\cdots n} \frac{1}{|a_{ii}|} \sum_{j\neq i} |a_{ij}| =: q < 1 \ .$$

Das bedeutet, die Voraussetzung von Satz 3.3 (Banachscher Fixpunktsatz) erfüllt sind, und man erhält Konvergenz des Gesamtschrittverfahren für die $\|\cdot\|_\infty$-Norm.

Beim Einzelschrittverfahren verwenden wir, dass die Zeilensummennorm und müssen nun nachweisen, $\|\cdot\|_{\infty,1}$ durch $\|\cdot\|_\infty$ induziert wird:

$$\|\mathcal{L}\|_{\infty,1} = \max_{\|x\|_\infty=1} \|\mathcal{L}x\|_\infty < 1 \ .$$

Sei also $\|x\|_\infty = 1$ und q wie zuvor definiert. Setzt man $y := \mathcal{L}x$, dann ergeben sich entsprechend zu (3.9) mit $b := 0$, $x^{(k)} := x$ und $y := x^{(k+1)}$, die Komponenten y_i von y:

$$y_i = \frac{1}{a_{ii}} \left(-\sum_{j<i} a_{ij} y_j - \sum_{i<j} a_{ij} x_j \right) \ . \tag{3.11}$$

Wir zeigen nun induktiv, dass $|y_j| \leq q < 1$ für alle $j < i$ gilt.

Für $i = 1$ gilt

$$|y_1| = \left| \frac{1}{a_{11}} \left(\sum_{j>i} a_{ij} x_j \right) \right| \leq \|x\|_\infty \frac{1}{|a_{ii}|} \sum_{j>i} |a_{ij}| \leq q < 1 \ .$$

Im Induktionsschritt $i-1 \to i$ schätzen wir in (3.11) $|y_i|$ mit der Dreiecks-

ungleichung wie folgt ab:

$$\begin{aligned}
|y_i| &\leq \frac{1}{|a_{ii}|}\left(\sum_{j<i}|a_{ij}||y_j| + \sum_{j>i}|a_{ij}||x_j|\right) \\
&\leq \frac{1}{|a_{ii}|}\left(\sum_{j<i}|a_{ij}|q + \sum_{j>i}|a_{ij}|\|x\|_\infty\right) \\
&\leq \frac{1}{|a_{ii}|}\left(\sum_{j<i}|a_{ij}| + \sum_{j>i}|a_{ij}|\right) \\
&= q.
\end{aligned}$$

Für $i = n$ ergibt die Induktionsaussage schließlich $\|y\|_\infty \leq q$, und somit $\|\mathcal{L}\|_{\infty,1} \leq q$.

□

Beispiel 3.10. Gegeben sei das lineare Gleichungssystem $Ax = b$ mit

$$A = \begin{bmatrix} 2 & 0 & 1 \\ 1 & -4 & 1 \\ 0 & -1 & 2 \end{bmatrix}, \quad b = \begin{bmatrix} 1 \\ 4 \\ -1 \end{bmatrix}.$$

Die Lösung lautet $x^t = [1, -1, -1]$.

A ist strikt diagonaldominant: Der Faktor q aus dem Beweis von Satz 3.9 ist $q = 1/2$. Für den Startvektor $x^{(0)} = [1, 1, 1]^t$ ist $\|x^{(0)} - \hat{x}\|_\infty = 2$ und es ergibt sich

- bei dem Gesamtschrittverfahren $x_{\mathcal{J}}^{(1)} = (0, -1/2, 0)^t$ mit Fehler $\|x_{\mathcal{J}}^{(1)} - \hat{x}\|_\infty = 1$;
- bei dem Einzelschrittverfahren $x_{\mathcal{L}}^{(1)} = (0, -3/4, -7/8)^t$ mit Fehler $\|x_{\mathcal{L}}^{(1)} - \hat{x}\|_\infty = 1$.

Bezüglich der Maximumnorm wird der Fehler also tatsächlich in beiden Fällen genau um den Faktor 2 reduziert.

3.2 Das Verfahren der konjugierten Gradienten

Das vermutlich effizienteste Verfahren zur Lösung von linearen Gleichungssystemen $Ax = b$, deren Koeffizientenmatrix $A \in \mathbb{R}^{n\times n}$ symmetrisch und positiv definit ist, ist das Verfahren der konjugierten Gradienten.

Das besagte Verfahren lässt sich nicht in das allgemeine Schema einer Fixpunktiteration einordnen, und braucht deshalb eine andere Konvergenzanalyse.

Sei

$$\Phi(x) = \frac{1}{2}x^t Ax - x^t b : \mathbb{R}^n \to \mathbb{R} .$$

Setzen wir $\hat{x} = A^{-1}b$, dann ergibt eine einfache Rechnung, dass

$$\begin{aligned}\Phi(x) - \Phi(\hat{x}) &= \frac{1}{2}x^t Ax - x^t b - \frac{1}{2}\hat{x}^t A\hat{x} + \hat{x}^t b \\ &= \frac{1}{2}(x-\hat{x})^t A(x-\hat{x}) + \frac{1}{2}\left(x^t A\hat{x} + \hat{x}^t Ax\right) - \hat{x}^t A\hat{x} - x^t b + \hat{x}^t b \\ &= \frac{1}{2}(x-\hat{x})^t A(x-\hat{x}) .\end{aligned}$$

Da A positiv definit ist, ist der letzte Ausdruck nichtnegativ und genau dann Null, wenn $x = \hat{x}$ ist. Mit anderen Worten: Das Funktional Φ hat ein eindeutiges Minimum an der Stelle $x = \hat{x}$.

Definition 3.11. Ist A symmetrisch und positiv definit, dann definiert

$$\langle x, y\rangle_A = x^t Ay , \quad \forall x, y \in \mathbb{R}^n \qquad \|x\|_A := \sqrt{x^t Ax} , \quad \forall x \in \mathbb{R}^n ,$$

ein Salarprodukt und eine Norm in $\mathbb{R}^n$, die so genannte *Energienorm*.

Mit dieser Bezeichnung gilt somit

$$\Phi(x) - \Phi(\hat{x}) = \frac{1}{2}(x-\hat{x})^t A(x-\hat{x}) = \frac{1}{2}\|x-\hat{x}\|_A^2 . \tag{3.12}$$

Geometrisch ist die Funktion $x \to \Phi(x) - \Phi(\hat{x})$ ein Paraboloid mit Nullstelle $\hat{x}$.

Wir konstruieren nun ein Verfahren zur Approximation von $\hat{x}$, welches das Funktional Φ sukzessive minimiert. Ist $x^{(k)}$ die aktuelle Iterierte, dann wird die nächste Iterierte durch

$$x^{(k+1)} = x^{(k)} + \alpha d^{(k)} \tag{3.13}$$

bestimmt, wobei eine geeignete *Suchrichtung* $d^{(k)} \neq 0$ gewählt wird.

Bei vorgegebenem $d^{(k)}$ bestimmen wir den Wert α, für den die Funktion

$$\alpha \to F(\alpha) := \Phi(x^{(k)}+\alpha d^{(k)}) = \Phi(x^{(k)})+\alpha d^{(k)t}Ax^{(k)}+\frac{1}{2}\alpha^2 d^{(k)t}Ad^{(k)}-\alpha d^{(k)t}b\,, \tag{3.14}$$

minimal wird. Aus der Optimmalitätsbedingung $F'(\alpha^{(k)}) = 0$ folgt dann

$$\alpha^{(k)} = \frac{(b-Ax^{(k)})^t d^{(k)}}{d^{(k)t}Ad^{(k)}} =: \frac{r^{(k)t}d^{(k)}}{d^{(k)t}Ad^{(k)}}\,. \tag{3.15}$$

Dabei ist der Nenner ungleich Null, da A positiv definit ist und $d^{(k)} \neq 0$ vorausgesetzt wurde.

Bei dem Verfahren der konjugierten Gradienten macht man den Ansatz

$$d^{(k+1)} = r^{(k+1)} + \beta^{(k)}d^{(k)} \text{ mit } \left\langle d^{(k+1)}, d^{(k)}\right\rangle_A = 0\,. \tag{3.16}$$

Die Bedingung $\left\langle d^{(k+1)}, d^{(k)}\right\rangle_A = 0$ bedeutet, dass die beiden Suchrichtungen $d^{(k+1)}$ und $d^{(k)}$ zueinander orthogonal (bzgl. des inneren Produktes $< \cdot, \cdot >_A$) stehen, was den Namen *Verfahren der konjugierten Richtungen* (oder CG-Verfahren - conjugate gradient method) motiviert.

Aus (3.16) folgt

$$0 = d^{(k+1)t}Ad^{(k)} = r^{(k+1)t}Ad^{(k)} + \beta^{(k)}d^{(k)t}Ad^{(k)}\,,$$

und somit

$$\beta^{(k)} = -\frac{r^{(k+1)t}Ad^{(k)}}{d^{(k)t}Ad^{(k)}}\,. \tag{3.17}$$

Das Verfahren (3.15) mit (3.17) nur dann wohldefiniert, wenn $d^{(k+1)}$ ungleich Null ist.

Wir überlegen uns nun, dass für alle Iterierten $x^{(k)} \neq \hat{x}$, $d^{(k)} \neq 0$. Um dies zu zeigen, weisen wir einige Eigenschaften des Verfahrens der konjugierten Gradienten nach.

Lemma 3.12. *Sei $x^{(0)}$ ein beliebiger Startvektor und $d^{(0)} = r^{(0)}$. Falls $x^{(k)} \neq \hat{x}$ ist für $k = 0, \ldots, m$, dann gilt*

1.

$$r^{(m)t}d^{(j)} = 0\,, \quad \forall 0 \leq j < m\,, \tag{3.18}$$

2.

$$r^{(m)t} r^{(j)} = 0\,, \quad \forall 0 \le j < m\,, \tag{3.19}$$

3.

$$\left\langle d^{(m)}, d^{(j)} \right\rangle_A = 0\,, \quad \forall 0 \le j < m\,. \tag{3.20}$$

Beweis. Da $Ax^{(k+1)} = Ax^{(k)} + \alpha^{(k)} Ad^{(k)}$ ist, folgt

$$r^{(k+1)} = b - Ax^{(k+1)} = b - Ax^{(k)} - \alpha^{(k)} Ad^{(k)} = r^{(k)} - \alpha^{(k)} Ad^{(k)}\,, \quad \forall k \ge 0\,. \tag{3.21}$$

Daher bewirkt die Wahl von $\alpha^{(k)}$ aus (3.15), dass für $x^{(k)} \neq \hat{x}$

$$r^{(k+1)t} d^{(k)} = (r^{(k)} - \alpha^{(k)} Ad^{(k)})^t d^{(k)} = r^{(k)t} d^{(k)} - \alpha^{(k)} d^{(k)t} Ad^{(k)} = 0\,. \tag{3.22}$$

Nun führen wir einen Induktionsbeweis:

$\underline{m = 1}$: Für $k = 0$ folgt aus (3.22) $r^{(1)t} d^{(0)} = 0$ und somit ergibt das die Behauptung (3.18) für $m = 1$ und $j = 0$.

Da $r^{(0)} = d^{(0)}$ folgt aus Behauptung (3.18) sofort auch Behauptung (3.19) für $m = 1$ und $j = 0$.

Wegen der Definition des Verfahrens gilt auch der dritte Teil: $\left\langle d^{(1)}, d^{(0)} \right\rangle_A = 0$.

$\underline{m \to m+1}$: Im Induktionsschritt nehmen wir an, dass *alle drei* Aussagen für ein m erfüllt sind, und dass $x^{(m+1)} \neq \hat{x}$ gilt:

- Zunächst folgt aus (3.22) $r^{(m+1)t} d^{(m)} = 0$.

  Aus (3.21) folgt zusammen mit den beiden Induktionsannahmen (3.18) und (3.20), dass

  $$r^{(m+1)t} d^{(j)} = r^{(m)t} d^{(j)} + \alpha^{(m)} \left\langle d^{(m)}, d^{(j)} \right\rangle_A = 0 - 0\,, \quad \forall 0 \le j < m\,.$$

  Folglich gilt der erste Teil der Behauptung auch für $m + 1$.

- Wegen (3.16) ist $r^{(j)} = d^{(j)} - \beta^{(j-1)} d^{(j-1)}$, $1 \le j \le m$, und $r^{(0)} = d^{(0)}$. Damit gilt wegen des ersten Teils des schon Bewiesenen

  $$r^{(m+1)t} r^{(j)} = \underbrace{r^{(m+1)t} d^{(j)}}_{=0} - \beta^{(j-1)} \underbrace{r^{(m+1)t} d^{(j-1)}}_{=0} = 0\,, \quad \forall 1 \le j \le m\,. \tag{3.23}$$

- Zum Beweis des dritten Teils der Behauptung: $\left\langle d^{(m+1)}, d^{(m)} \right\rangle_A = 0$ folgt unmittelbar aus der Definition des Verfahrens.
  Aus der Induktionsannahme und (3.16) folgt, dass

$$\begin{aligned} \left\langle d^{(m+1)}, d^{(j)} \right\rangle_A &= \left\langle r^{(m+1)}, d^{(j)} \right\rangle_A + \beta^{(m)} \left\langle d^{(m)}, d^{(j)} \right\rangle_A \\ &= r^{(m+1)t} A d^{(j)} , \quad \forall 0 \leq j < m . \end{aligned}$$

  Daraus folgt nun: Ersetzt man nun $Ad^{(j)}$ gemäß (3.21) und (3.16), dann ergibt sich

$$\begin{aligned} & \alpha^{(j)} \left\langle d^{(m+1)}, d^{(j)} \right\rangle_A \\ =& d^{(m+1)t} \underbrace{\alpha^{(j)} A d^{(j)}}_{-(r^{(j+1)} - r^{(j)})} \\ =& \underbrace{-r^{(m+1)t}(r^{(j+1)} - r^{(j)})}_{=0 (3.23)} + \beta^{(m)} \alpha^{(j)} \underbrace{\left\langle d^{(m)}, d^{(j)} \right\rangle_A}_{=(3.20)0} \quad \forall 0 \leq j < m . \end{aligned}$$

  Wenn wir noch zeigen, dass $\alpha^{(j)} \neq 0$ für $0 \leq j < m$, dann ist die Behauptung vollständig bewiesen.
  Nehmen wir an, es gelte $\alpha^{(j)} = 0$ für ein j mit $0 \leq j < m$: Wegen (3.15) bedeutet dies

$$\begin{aligned} 0 &= r^{(j)t} d^{(j)} \\ &= r^{(j)t}(r^{(j)} + \beta^{(j-1)} d^{(j-1)}) \\ &= r^{(j)t} r^{(j)} + \beta^{(j-1)} r^{(j)t} d^{(j-1)} \\ &\underbrace{=}_{(3.23)} \|r^{(j)}\|_2^2 \qquad \forall 0 < j < m . \end{aligned}$$

  Für $j = 0$ gilt per Definition des CG-Verfahrens

$$0 = r^{(0)t} d^{(0)} = \|r^{(0)}\|_2^2 .$$

  Da $x^{(j)} \neq \hat{x}$, muss auch $\|r^{(j)}\|_2 \neq 0$ für alle $0 \leq j < m$ gelten. Also erhalten wir einen Widerspruch zur Annahme $\alpha^{(j)} = 0$. Somit ist $\left\langle d^{(m+1)}, d^{(j)} \right\rangle_A = 0$ für alle $0 \leq j < m + 1$.

□

Gemäß Lemma 3.12, Behauptung (3.20) sind alle Suchrichtungen paarweise A-konjugiert. Gemäß Lemma 3.12, Behauptung (3.19) sind alle Residuen linear unabhängig. Daher ergibt sich nach spätestens $n = \dim(A)$ Schritten $r^{(n)} = 0$, also $\hat{x} = x^{(n)}$.

Korollar 3.13. *Nach spätestens $n = \dim(A)$ Schritten findet das CG-Verfahren die exakte Lösung $\hat{x}$.*

In der Praxis ist diese Ergebnis nicht von allzu großer Bedeutung, da das CG-Verfahren in erster Linie eingesetzt wird und wesentlich weniger als n-Iterationsschritte benötigt.

Das CG-Verfahren hat Optimalitätseigenschaften, die wir im folgenden Satz zusammenfassen:

Satz 3.14. *Sei $k = 1, 2, \cdots$ fix. Darüberhinaus sei $d^{(0)} = r^{(0)}$ und $x^{(k)} \neq \hat{x}$ die k-te Iterierte des CG-Verfahrens. Dann liegt $x^{(k)}$ in dem Raum*

$$x^{(k)} \in x^{(0)} + \left[r^{(0)}, \cdots, r^{(k-1)}\right] = x^{(0)} + \left[r^{(0)}, Ar^{(0)}, \cdots, A^{(k-1)}r^{(0)}\right] \ . \quad (3.24)$$

Unter allen Elementen x dieser Menge minimiert $x^{(k)}$ die Zielfunktion Φ. Der Raum

$$\mathcal{K}_k(A, r^{(0)}) = \left[r^{(0)}, Ar^{(0)}, \cdots, A^{(k-1)}r^{(0)}\right]$$

wird Krylov-Raum *der Dimension k von A bezüglich $r^{(0)}$ genannt.*

Beweis. Wir beweisen zunächst induktiv (nach j), dass

$$d^{(j)} \in \left[r^{(0)}, \cdots, r^{(k-1)}\right] \ , \quad j = 0, \cdots, k-1 \ . \quad (3.25)$$

- Für $j = 0$ ist $d^{(0)} = r^{(0)}$ und damit gilt der Induktionsanfang.
- Für $j = 1, \ldots, k-1$ gelte die Induktionsvoraussetzung:

  $$d^{(s)} \in \left[r^{(0)}, \cdots, r^{(k-1)}\right] \ , \quad s = 0, \cdots, j-1 \ .$$

  Wir zeigen, dass $d^{(j)} \in \left[r^{(0)}, \cdots, r^{(k-1)}\right]$ gilt. Aus (3.16) folgt, dass $d^{(j)} = r^{(j)} + \beta^{(j-1)}d^{(j-1)}$

  $$d^{(j)} \in \left[r^{(j)}, d^{(j-1)}\right] \subseteq \left[r^{(j)}, r^{(0)}, \cdots, r^{(k-1)}\right] = \left[r^{(0)}, \cdots, r^{(k-1)}\right] \ .$$

  und somit gilt zusammenfassend

  $$\left[d^{(0)}, \cdots, d^{(k-1)}\right] \subseteq \left[r^{(0)}, \cdots, r^{(k-1)}\right] \ .$$

Solange $x^{(k)} \neq \hat{x}$ ist, folgt aus Lemma 3.12, dass jede der beiden Mengen $\{d^{(j)}\}_{j=0}^{k-1}$ und $\{r^{(j)}\}_{j=0}^{k-1}$ aus linear unabhängigen Vektoren besteht. Demnach ist

$$\left[d^{(0)}, \cdots, d^{(k-1)}\right] = \left[r^{(0)}, \cdots, r^{(k-1)}\right] \ . \tag{3.26}$$

Nun beweisen wir (3.24): Aus (3.13) folgt

$$x^{(k)} = x^{(0)} + \sum_{j=0}^{k-1} \alpha^{(j)} d^{(j)} \in x^{(0)} + \left[r^{(0)}, \cdots, r^{(k-1)}\right] \ .$$

Damit haben wir die erste Inklusion in (3.24) gezeigt. Jetzt zeigen wir induktiv, dass

$$r^{(j)} \in \left[r^{(0)}, \cdots, A^{k-1} r^{(0)}\right] \ , \quad j = 0, \cdots, k-1 \ . \tag{3.27}$$

- Es gilt $r^{(0)} \in \left[r^{(0)}, \cdots, A^{k-1} r^{(0)}\right]$, weshalb die Induktionsvoraussetzung gilt.

- Für $j = 1, \ldots, k-1$ gelte die Induktionsvoraussetzung für $j-1$:

  $$r^{(s)} \in \left[r^{(0)}, \cdots, A^{k-1} r^{(0)}\right] \ , \quad s = 0, \cdots, j-1 \ .$$

  Wir zeigen, dass $r^{(j)} \in \left[r^{(0)}, \cdots, A^{k-1} r^{(0)}\right]$ gilt.

  Da $j - 1 \leq k - 2$ gilt, folgt aus (3.25)

  $$d^{(j-1)} \in \left[r^{(0)}, \cdots, r^{(k-2)}\right] \ .$$

  Somit folgt aus der Induktionsannnahme die Beziehung

  $$d^{(j-1)} \in \left[r^{(0)}, \cdots, r^{(k-2)}\right] \subseteq \left[r^{(0)}, \cdots, A^{k-2} r^{(0)}\right] \ .$$

  Wiederum, unter Verwendung von $j - 1 \leq k - 2$, folgt aus (3.27) und (3.21) die Inklusion

  $$r^{(j)} = r^{(j-1)} + \alpha^{(j-1)} A d^{(j-1)} \in \left[r^{(0)}, \cdots, A^{k-1} r^{(0)}\right] \ .$$

  Demnach ist

  $$\left[r^{(0)}, \cdots, r^{(k-1)}\right] \subseteq \left[r^{(0)}, Ar^{(0)}, \cdots, A^{k-1} r^{(0)}\right] \ ,$$

  Da $r^{(0)}, \cdots, r^{(k-1)}$ linear unabhängig sind, hat die rechte Seite maximale Dimension k. Also stimmen die beiden Mengen überein. Damit gilt (3.24).

Zum Beweis der Minimialeigenschaft: Aus Korollar 3.13 folgt schließlich die Existenz eines Iterationsindex $m \leq n$ (m muss nicht unbedingt mit n übereinstimmen), so dass

$$\hat{x} = x^{(m)} = x^{(0)} + \sum_{j=0}^{m-1} \alpha^{(j)} d^{(j)} \ .$$

Folglich ist

$$\hat{x} - x^{(k)} = \sum_{j=k}^{m-1} \alpha^{(j)} d^{(j)} \ .$$

Für ein beliebiges Element x von $x^{(0)} + \left[r^{(0)}, Ar^{(0)}, \cdots, A^{(k-1)}r^{(0)}\right]$ gilt wegen (3.26)

$$\hat{x} - x = \hat{x} - x^{(k)} + \sum_{j=0}^{k-1} \delta_j d^{(j)} \ ,$$

mit geeigneten $\delta_j \in \mathbb{R}$.

Da die Suchrichtungen nach Lemma 3.12 A-konjugiert sind, folgt daher aus dem Satz von Pythagoras

$$\begin{aligned}
\Phi(x) - \Phi(\hat{x}) &= \left(\frac{1}{2}x^t Ax - x^t b\right) - \left(\frac{1}{2}\hat{x}^t A\hat{x} - \hat{x}^t b\right) \\
&= \frac{1}{2}\|x - \hat{x}\|_A^2 \\
&= \frac{1}{2}\|x^{(k)} - \hat{x}\|_A^2 + \frac{1}{2}\left\|\sum_{j=0}^{k-1} \delta_j d^{(j)}\right\|_A^2 \\
&= \Phi(x^{(k)}) - \Phi(\hat{x}) + \frac{1}{2}\left\|\sum_{j=0}^{k-1} \delta_j d^{(j)}\right\|_A^2 \ .
\end{aligned}$$

Demnach ist $\Phi(x) \geq \Phi(x^{(k)})$ mit Gleichheit genau dann wenn $x = x^{(k)}$. Also is $x^{(k)}$ das minimierende Element auf dem Krylov-Raum. □

Für die Implementierung des CG-Verfahrens in der Praxis verwendet man nicht die Gleichungen (3.15) und (3.17) für $\alpha^{(k)}$ und $\beta^{(k)}$, sondern die folgenden Darstellungen (3.28) und (3.29).

Zur Herleitung dieser Formeln verwenden wir, dass aufgrund von Lemma 3.12 und (3.16) gilt:

$$r^{(k)t}d^{(k)} = r^{(k)t}r^{(k)} + \beta^{(k-1)}r^{(k)t}d^{(k-1)} = r^{(k)t}r^{(k)} \ .$$

In (3.15) eingesetzt, ergibt sich

$$\alpha^{(k)} = \frac{\|r^{(k)}\|_2^2}{d^{(k)t}Ad^{(k)}} \ . \tag{3.28}$$

Aus (3.21), (3.28) und Lemma 3.12 folgt

$$\begin{aligned} r^{(k+1)t}Ad^{(k)} &\underbrace{=}_{(3.21)} -\frac{1}{\alpha^{(k)}}(r^{(k+1)t}r^{(k+1)} - r^{(k+1)t}r^{(k)}) \\ &\underbrace{=}_{\text{Lemma 3.12}} -\frac{1}{\alpha^{(k)}}\|r^{(k+1)}\|_2^2 \\ &= -\frac{\|r^{(k+1)}\|_2^2}{\|r^{(k)}\|_2^2}d^{(k)t}Ad^{(k)} \ . \end{aligned}$$

Anstelle von (3.17) kann man daher die Formel

$$\beta^{(k)} = \frac{\|r^{(k+1)}\|_2^2}{\|r^{(k)}\|_2^2} \tag{3.29}$$

verwenden. Damit lautet das CG-Verfahren in der zu implementierenden Form wie folgt:

Algorithmus 3.15. *(Verfahren der konjugierten Gradienten)*

`Wähle` $x^{(0)}$ `beliebig; setze` $r^{(0)} = b - Ax^{(0)}$, $d^{(0)} = r^{(0)}$.

`for` $k = 0, 1, 2, \cdots$

$$\begin{aligned} \alpha^{(k)} &= \frac{\|r^{(k)}\|_2^2}{d^{(k)t}Ad^{(k)}} \\ x^{(k+1)} &= x^{(k)} + \alpha^{(k)}d^{(k)} \\ r^{(k+1)} &= r^{(k)} - \alpha^{(k)}Ad^{(k)} \\ \beta^{(k)} &= \frac{\|r^{(k+1)}\|_2^2}{\|r^{(k)}\|_2^2} \\ d^{(k+1)} &= r^{(k+1)} + \beta^{(k)}d^{(k)} \end{aligned}$$

`until stop`

Kapitel 4

Eigenwerte

In diesem Kapitel beschäftigen wir uns mit der Berechnung der Eigenwerte einer Matrix $A \in \mathbb{C}^{n\times n}$. Ist die Matrix A reell, dann sind Eigenwerte und Eigenvektoren im Regelfall noch immer complex, und deshalb bietet es sich an gleich den allgemeineren Fall von komplexen Matrizen zu studieren.

Die Eigenwertgleichung lautet

$$Ax = \lambda x \quad x \in \mathbb{C}^n \setminus \{0\},\ \lambda \in \mathbb{C}\,,$$

und ist *nichtlinear*: wir wissen aus der linearen Algebra, dass die Nullstellen des charakteristischen Polynoms

$$p(\lambda) = \det(A - \lambda I)$$

die Eigenwerte von A sind. Aus dieser Beziehung lässt sich sofort erkennen, dass die Eigenwertbestimmung ein nichtlineares Problem ist.

Die Eigenvektoren einer Matrix A zu einem Eigenwert λ gehören zu einem linearen Raum.

Zuerst fassen wir einige Bemerkungen aus der linearen Algebra zusammen:

- Ist λ ein Eigenwert der invertierbaren Matrix $A \in \mathbb{C}^{n\times n}$ zum Eigenvektor x, so ist $\frac{1}{\lambda}$ Eigenwert von A^{-1} zum Eigenvektor x. Darüberhinaus ist $\overline{\lambda}$ ein Eigenwert von A^*.

- Im Allgmeinen sind Eigenwerte von reellen Matrizen nicht reell. Die Eigenwerte von hermiteschen Matrizen sind reell.

- Üblicherweise normiert man den Eigenvektor auf Norm 1.

- Sind λ_i die Eigenwerte der Matrix $A \in \mathbb{C}^{n\times n}$, so gilt

$$\sum_{i=1}^{n} \lambda_i = \text{Spur}(A) \text{ und } \prod_{i=1}^{n} \lambda_i = \det(A)\,,$$

  wobei bei mehrfachen Eigenwerten die Vielfachheit zu beachten ist.

- Ist die Matrix echt positiv definit, so sind die Eigenwerte reell und echt größer Null.

- Sei $A \in \mathbb{C}^{n\times n}$ mit n linear unabhängigen Eigenvektoren, $\{x_i : i = 1, \ldots, n\}$. Dann gilt

$$A = X\Lambda X^{-1}\,,$$

  wobei die i-te Spalte der $i - te$ Eigenvektor von A (mit Eigenwert λ_i) ist. In diesem Fall nennt man die Matrix $A \in \mathbb{C}^{n\times n}$ *diagonalisierbar*.

  Beachte, dass die Eigenvektoren einer regulären Matrix nicht orthogonal stehen: Die Matrix

$$\begin{bmatrix} 1 & 1 \\ 0 & 1 \end{bmatrix}$$

  kann nicht diagonalisiert werden. Beachte, dass 1 ein doppelter Eigenwert zum Eigenvektor $(1, 0)$.

- Kann bei einer diagonalisierbaren Matrix X zudem unitär gewählt werden, dann heißt A *normal*. Normale Matrizen lassen sich durch die Gleichung $AA^* = A^*A$ charakterisieren. Hermitesche Matrizen sind also ein Spezialfall der normalen Matrizen.

- Ist A hermitesch so sind Eigenvektoren zu verschiedenen Eigenwerten orthogonal. Bei nicht hermiteschen Matrizen muss das nicht gelten.

4.1 Eigenwerteinschließung

Im diesem Teil diskutieren wir relativ leicht zu berechnende Abschätzungen für Eigenwerte.

Satz 4.1. (Satz von Gerschgorin) *Sei $A = [a_{ij}] \in \mathbb{C}^{n\times n}$ und λ ein beliebiger Eigenwert von A. Dann gilt*

$$\lambda \in \bigcup_{i=1}^{n} \mathcal{K}_i := \bigcup_{i=1}^{n} \left\{ \zeta \in \mathbb{C} : |\zeta - a_{ii}| \leq \sum_{j\neq i} |a_{ij}| \right\} . \tag{4.1}$$

Beweis. Sei $Ax = \lambda x$ mit $x = [x_i] \neq 0$. Dann existiert ein i mit $|x_j| \leq |x_i|$ für alle $j \neq i$. Bezeichnet $(Ax)_i$ die i-te Komponente von Ax, dann folgt

$$\lambda x_i = (Ax)_i = \sum_{j=1}^{n} a_{ij} x_j$$

und somit ist

$$|\lambda - a_{ii}| = \left| \sum_{j\neq i} a_{ij} \frac{x_j}{x_i} \right| \leq \sum_{j\neq i} |a_{ij}| .$$

Also ist $\lambda \in \mathcal{K}_i \subseteq \bigcup_{j=1}^{n} \mathcal{K}_j$. □

Für $\overline{\lambda} \in \sigma(A^*)$ gilt der Satz von Gerschgorin entsprechend, nämlich

$$\overline{\lambda} \in \bigcup_{i=1}^{n} \mathcal{K}_i = \bigcup_{i=1}^{n} \left\{ \zeta : |\zeta - \overline{a}_{ii}| \leq \sum_{j=1, j\neq i}^{n} |a_{ji}| \right\}$$

oder äquivalent

$$\lambda \in \bigcup_{i=1}^{n} \overline{\mathcal{K}}_i = \bigcup_{i=1}^{n} \left\{ \zeta : |\zeta - a_{ii}| \leq \sum_{j=1, j\neq i}^{n} |a_{ji}| \right\}$$

Weitere Einschließungssätze beruhen auf dem Konzept des Wertebereichs einer Matrix.

Definition 4.2. Unter dem *Wertebereich* einer Matrix $A \in \mathbb{C}^{n\times n}$ versteht man die Menge aller *Rayleigh-Quotienten* $\frac{x^*Ax}{x^*x}$ mit $x \in \mathbb{C}^n \backslash \{0\}$,

$$\mathcal{W}(A) := \left\{ \zeta = \frac{x^*Ax}{x^*x} : x \in \mathbb{C}^n \backslash \{0\} \right\} \subseteq \mathbb{C} .$$

Der Wertebereich beinhaltet insbesondere die Eigenwerte der Matrix.

Lemma 4.3. *1. $\mathcal{W}(A)$ ist zusammenhängend.*

2. *Ist A hermitesch, dann ist $\mathcal{W}(A)$ das reelle Intervall $[\lambda_{\min}, \lambda_{\max}]$.*

3. *Ist A* schiefsymmetrisch*, d.h., $A^* = -A$, dann ist $\mathcal{W}(A)$ ein rein imaginäres Intervall, nämlich die konvexe Hülle aller Eigenwerte von A.*

Beweis. 1. Liegen ζ_0 und ζ_1 im Wertebereich $\mathcal{W}(A)$, dann existieren $x_0, x_1 \in \mathbb{C}\backslash\{0\}$ mit

$$\zeta_0 = \frac{x_0^* A x_0}{x_0^* x_0}\,, \quad \zeta_1 = \frac{x_1^* A x_1}{x_1^* x_1}\,.$$

Nehmen wir an, dass $0 \neq \zeta_0 \neq \zeta_1 \neq 0$, dann sind x_0 und x_1 linear unabhängig; sind nämlich zwei Vektoren x_0 und x_1 linear abhängig, so ist x_0 ein Vielfaches von x_1, und folglich ist $\zeta_0 = \zeta_1$. Damit gilt

$$0 \notin [x_0, x_1] := \{x_t = x_0 + t(x_1 - x_0) : t \in [0,1]\}\,.$$

Die Konseqeunz daraus ist, dass die Abbildung $t \to \frac{x_t^* A x_t}{x_t^* x_t}$ stetig ist, und

$$\zeta_t := \frac{x_t^* A x_t}{x_t^* x_t}\,, \quad 0 \leq t \leq 1\,,$$

ist eine stetige Kurve in $\mathcal{W}(A)$, die ζ_0 mit ζ_1 verbindet.

2. Wir wählen eine Orthonormalbasis $\{x_i\}_{i=1}^n$ in $\mathbb{C}^n$ mit $Ax_i = \lambda x_i$, wobei die Eigenwerte von A absteigend sortiert sein sollen, $\lambda_1 \geq \lambda_2 \geq \cdots \geq \lambda_n$. Für einen beliebigen Vektor $x = \sum_{i=1}^n \zeta_i x_i \in \mathbb{C}^n$, $\zeta_i \in \mathbb{C}$, gilt dann

$$x^* A x = \sum_{i,j=1}^n \overline{\zeta}_i \zeta_j x_i^* A x_j = \sum_{i,j=1}^n \overline{\zeta}_i \zeta_j \lambda_j x_i^* x_j = \sum_{i=1}^n \lambda_i |\zeta_i|^2\,,$$

und wegen der Anordnung der Eigenwerte folgt

$$\lambda_n \|x\|_2^2 = \lambda_n \sum_{i=1}^n |\zeta_i|^2 \leq \sum_{i=1}^n \lambda_i |\zeta_i|^2 \leq \lambda_1 \sum_{i=1}^n |\zeta_i|^2 = \lambda_1 \|x\|_2^2\,.$$

Damit ist zunächst gezeigt, dass $\mathcal{W}(A) \subseteq [\lambda_n, \lambda_1]$ gilt. Da $\mathcal{W}(A)$ nach dem ersten Teil dieses Satzes auch zusammenhängend ist, folgt die zweite Behauptung.

3. Wegen $A^* = -A$ ist iA hermitesch, denn

$$(iA)^* = \overline{i} A^* = -iA^* = iA\,.$$

Ferner ist

$$\mathcal{W}(A) = \mathcal{W}(-iiA) = -i\mathcal{W}(iA) = -i\left[\lambda_{\min}, \lambda_{\max}\right] ,$$

wobei $\lambda_{\min}$ bzw. $\lambda_{\max}$ der kleinste, bzw., größte Eigenwert von iA ist. Damit sind $-i\lambda_{\min}$ und $-i\lambda_{\max}$ der betragsleinste, bzw. -größte Eigenwert von A.

Daher folgt die Behauptung aus dem zweiten Teil des Satzes.

□

Für jede beliebige Matrix $A \in \mathbb{C}^{n\times n}$ ist

$$A = \frac{A + A^*}{2} + \frac{A - A^*}{2}$$

eine Zerlegung in die hermitesche Matrix $\frac{A+A^*}{2}$ und die schiefsymmetrische Matrix $\frac{A-A^*}{2}$. Diese Zerlegung ist die Grundlage des folgenden Einschließungssatzes.

Satz 4.4. (Satz von Bendixon)

$$\sigma(A) \subseteq \mathcal{R} := \mathcal{W}\left(\frac{A + A^*}{2}\right) \oplus \mathcal{W}\left(\frac{A - A^*}{2}\right) \tag{4.2}$$

enthalten.

Beweis. Wegen Lemma 4.3 reicht es zu zeigen, dass $\mathcal{W}(A) \subseteq \mathcal{R}$. Für $x \in \mathbb{C}^n\backslash\{0\}$ gilt

$$\begin{aligned}\frac{x^*Ax}{x^*x} &= \frac{x^*\left(\frac{A+A^*}{2} + \frac{A-A^*}{2}\right)x}{x^*x} \\ &= \frac{x^*\frac{A+A^*}{2}x}{x^*x} + \frac{x^*\frac{A-A^*}{2}x}{x^*x} \\ &\in \mathcal{W}\left(\frac{A + A^*}{2}\right) \oplus \mathcal{W}\left(\frac{A - A^*}{2}\right) .\end{aligned}$$

□

Beispiel 4.5. Wir wenden die Resultate aus den Sätzen von Gerschgorin und Bendixson auf die Matrix

$$A = \begin{bmatrix} 4 & 0 & -3 \\ 0 & -1 & 1 \\ -1 & 1 & 0 \end{bmatrix}$$

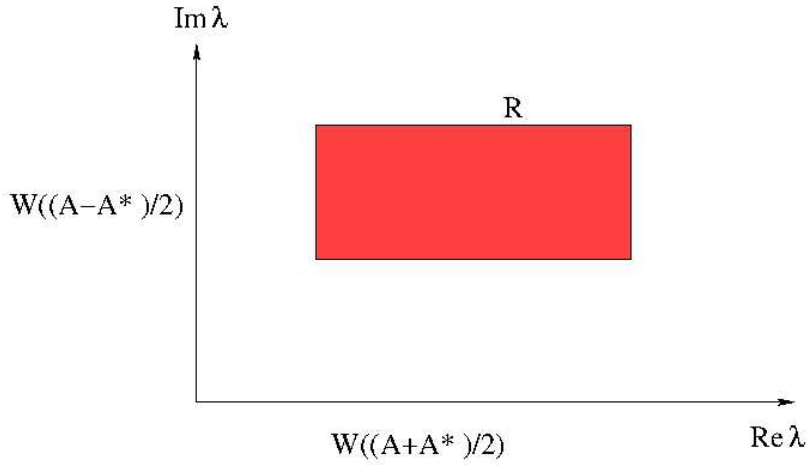

Abbildung 4.1: Satz von Bendixon

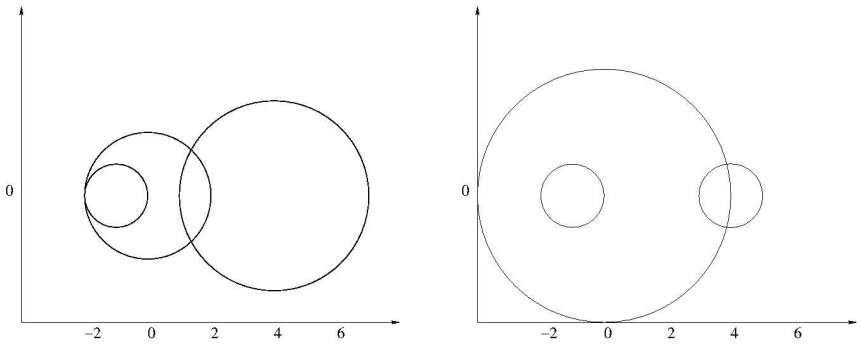

Abbildung 4.2: Gerschgorinkreise für A (links) und A^* rechts

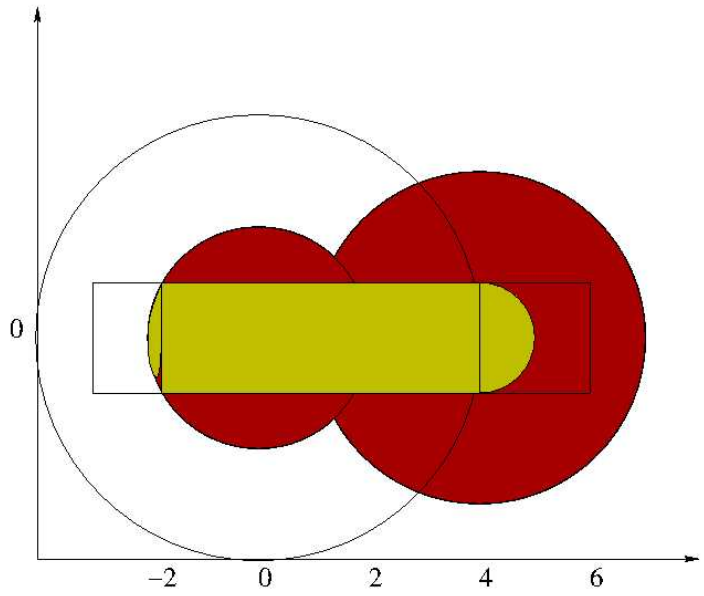

Abbildung 4.3: Alle Einschließungen gemeinsam

an. Abbildung 4.2 zeigt die Gerschgorinkreise für A und A^*. Zur Anwendung des Satz von Bendixon benötigen wir den symmetrischen und den schiefsymmetrsichen Anteil von A,

$$H = \frac{A + A^t}{2} = \begin{bmatrix} 4 & 0 & -2 \\ 0 & -1 & 1 \\ -2 & 1 & 0 \end{bmatrix}, \quad S = \frac{A - A^t}{2} = \begin{bmatrix} 0 & 0 & -1 \\ 0 & 0 & 0 \\ 1 & 0 & 0 \end{bmatrix}.$$

Die Spektren von H und S könnten beispielsweise auch mit Hilfe des Satzes von Gerschgorin eingeschlossen werden: Auf diese Weise erhält man das etwas gröbere Dreieck

$$\tilde{\mathcal{R}} = [-3, 6] \times [-i, i] \supset \mathcal{R} \supset \sigma(A) .$$

Folglich muss das Spektrum von A im Schnitt *aller* drei Einschlussmengen liegen. Dies ergibt die dunkelste (gelbe) Menge in Abbildung 4.3. Tatsächlich ist das Spektrum

$$\sigma(A) = \{-1.7878, 0.1198, 4.6679\} .$$

Für hermitesche Matrizen ist noch folgendes Resultat von großer Bedeutung:

Satz 4.6. (Maxmin Prinzip von Courant-Fischer) *Sei $A = A^* \in \mathbb{C}^{n\times n}$ und $\{z_1, \ldots, z_n\} \subseteq \mathbb{C}^n$ ein orthonormales System. Ferner seien $\lambda_1 \geq \lambda_2 \geq \cdots \geq \lambda_n$ die absteigend sortierten Eigenwerte von A mit zugehörigen Eigenvektoren x_i. Dann ist*

$$\min_{0\neq x\in[z_1,\ldots,z_k]} \frac{x^*Ax}{x^*x} \leq \lambda_k \tag{4.3}$$

und es gilt Gleichheit für $[z_1, \ldots, z_k] = [x_1, \ldots, x_k]$.

Beweis.

- Aus Lemma 4.3, wonach $\mathcal{W}(A) = [\lambda_{\min}, \lambda_{\max}] = [\lambda_n, \lambda_1]$ ist, folgt die Behauptung für $k = 1$.

- Es gelte $k \geq 2$. Sei

$$0 \neq x = \mathcal{X}_k := \left\{ x = \sum_{i=1}^{k} \zeta_i z_i \text{ mit } x \perp x_i \,, \forall i = 1, \ldots, k-1 \right\} .$$

Da $\{x_i\}$ eine orthonormale Basis ist, gibt es Koeffizienten $\chi \in \mathbb{C}$, sodass $x = \sum_{i=1}^{n} \chi_i x_i$ ist, und somit gilt

$$\begin{aligned}
x^*Ax &= x^* \sum_{i=1}^{n} \chi_i \lambda_i x_i \\
&= \sum_{i=1}^{n} \lambda_i \chi_i x^* x_i \\
&= \sum_{i=k}^{n} \lambda_i \chi_i \sum_{j=1}^{n} \overline{\chi}_j x_j^* x_i \\
&= \sum_{i=k}^{n} \lambda_i |\chi_i|^2 \\
&\leq \lambda_k \sum_{i=k}^{n} |\chi_i|^2 \\
&\leq \lambda_k \|x\|_2^2 \qquad \forall x \in \mathcal{X}_k \,.
\end{aligned}$$

Hieraus folgt

$$\min_{0\neq x\in[z_1,\ldots,z_k]} \frac{x^*Ax}{x^*x} \leq \min_{0\neq x\in\mathcal{X}_k} \frac{x^*Ax}{x^*x} \leq \lambda_k \,.$$

- Der Beweis des zweiten Teils. Falls $[z_1, \ldots, z_k] = [x_1, \ldots, x_k]$ ist, dann kann jedes $x \in [z_1, \ldots, z_k]$ geschrieben werden als $x = \sum_{i=1}^{k} \chi_i x_i$ und es folgt

$$x^* A x = \sum_{i=1}^{k} \lambda_i \chi_i \sum_{j=1}^{k} \overline{\chi}_j x_j^* x_i = \sum_{i=1}^{k} \lambda_i |\chi_i|^2 \geq \lambda_k \sum_{i=1}^{k} |\chi_i|^2 = \lambda_k \|x\|_2^2 \, .$$

  Somit ist

$$\min_{x \in [z_1, \ldots, z_k], x \neq 0} \frac{x^* A x}{x^* x} \geq \lambda_k \, ,$$

  und demnach gilt in diesem Fall Gleichheit in (4.3).

□

Unter den Voraussetzungen von Satz 4.6 lautet ein entsprechendes Maximumprinzip wie folgt

$$\max_{x \perp [z_1, \ldots, z_k], x \neq 0} \frac{x^* A x}{x^* x} \geq \lambda_{k+1} \, ,$$

mit Gleichheit für $[z_1, \ldots, z_k] = [x_1, \ldots, x_k]$.

4.2 Potenzmethode

Das erste von uns betrachtete konstruktive Verfahren zur Berechnung einzelner Eigenwerte und Eigenvektoren ist die *Potenzmethode von v. Mises*.

Wir beschränken uns auf $n \times n$-Matrizen mit betragsmäßig angeordneten Eigenwerten λ_i, $i = 1, \ldots, n$, für die gilt:

$$|\lambda_1| > \cdots > |\lambda_n| \geq 0 \, .$$

Sei v_i ein zu λ_i gehöriger Eigenvektoren von A. Wir nehmen an, dass die Eigenvektoren $\{v_i\}$ linear unabhängig sind. Dann kann jeder Vektor $x \in \mathbb{C}^n$ in diese Basis entwickelt werden,

$$x = \sum_{i=1}^{n} \zeta_i v_i \, . \tag{4.4}$$

Demnach ist

$$A^k x = \sum_{i=1}^{n} \lambda_i^k \zeta_i v_i \, . \tag{4.5}$$

Das v. Mises-Verfahren beruht auf der asymptotischen Darstellung

$$A^k x \approx \lambda_1^k \zeta_1 v_1 \,, \quad k \to \infty \,.$$

Algorithmus 4.7. `Bestimme einen Näherungsvektor` $z^{(0)}$ `mit` $\|z^{(0)}\|_2 = 1$

- `for` $k = 1, 2, \ldots$

$$\tilde{z}^{(k)} := A z^{(k-1)} \,, \quad z^{(k)} := \frac{\tilde{z}^{(k)}}{\|\tilde{z}^{(k)}\|_2} \,, \tag{4.6}$$

  `end for`

Die Iterierten der Potenzmethode haben folgende Eigenschaften:

Satz 4.8. *Zusätzlich zu den allgemeinen Voraussetzungen dieses Kapitels gelte (4.4) mit* $\zeta_1 \neq 0$*. Dann gilt:*

$$\|\tilde{z}^{(k)}\|_2 = |\lambda_1| + O(q^k) \,, \quad k \to \infty \,. \tag{4.7}$$

Ferner gilt für $k \to \infty$*:*

$$\|z^{(k)} - \mathit{Sign}(\lambda_1^k \zeta_1) v_1\|_2 = O(q^k) \,,$$

wobei $\mathit{Sign}(x) = \frac{x}{|x|}$ *bezeichnet.*

Beweis. Wir setzen $q := \left|\frac{\lambda_2}{\lambda_1}\right| < 1$.

Im Beweis verwenden wir die Identität

$$z^{(k)} = \frac{\tilde{z}^{(k)}}{\|\tilde{z}^{(k)}\|_2} = \frac{A z^{(k-1)}}{\|A z^{(k-1)}\|_2} = \frac{A \tilde{z}^{(k-1)}}{\|A \tilde{z}^{(k-1)}\|_2} = \frac{A^2 z^{(k-2)}}{\|A^2 z^{(k-2)}\|_2} \,.$$

Mit Induktion folgt somit

$$z^{(k)} = \frac{A^k z^{(0)}}{\|A^k z^{(0)}\|_2} \,.$$

Aus (4.5) folgt mit $x = z^{(0)}$

$$A^k z^{(0)} = \lambda_1^k \zeta_1 (v_1 + w^{(k)}) \text{ mit } w^{(k)} = \sum_{i=2}^{n} \left(\frac{\lambda_i}{\lambda_1}\right)^k \frac{\zeta_i}{\zeta_1} v_i$$

und

$$\|w^{(k)}\|_2 \leq q^k \sum_{i=2}^{n} \left|\frac{\zeta_i}{\zeta_1}\right| = O(q^k) \; . \tag{4.8}$$

Damit ist

$$z^{(k)} = \frac{A^k x}{\|A^k x\|_2} = \mathrm{Sign}(\lambda_1^k \zeta_1) \frac{v_1 + w^{(k)}}{\|v_1 + w^{(k)}\|_2} = \mathrm{Sign}(\lambda_1^k \zeta_1) v_1 + e^{(k)} \tag{4.9}$$

mit

$$e^{(k)} = \frac{\mathrm{Sign}(\lambda_1^k \zeta_1)}{\|v_1 + w^{(k)}\|_2} \left(w^{(k)} + (1 - \|v_1 + w^{(k)}\|_2) v_1 \right) \; .$$

Nun ist $\|v_1\|_2 - \|w^{(k)}\|_2 \leq \|v_1 + w^{(k)}\|_2 \leq \|v_1\|_2 + \|w^{(k)}\|_2$ und $\|v_1\|_2 = 1$, so dass aus (4.8) folgt, dass

$$\left|1 - \|v_1 + w^{(k)}\|_2\right| = \left|\|v_1\|_2 - \|v_1 + w^{(k)}\|_2\right| \leq \|w^{(k)}\|_2 = O(q^{(k)}) \quad k \to \infty \; .$$

Daraus folgt $\|e^{(k)}\|_2 = O(q^{(k)})$ für $k \to \infty$, und es ist gezeigt, dass für $z^{(k)}$ und $\tilde{z}^{(k+1)}$ für $k \to \infty$ gilt

$$\begin{aligned} z^{(k)} &= \mathrm{Sign}(\lambda_1^k \zeta_1) v_1 + O(q^k) \, , \\ \tilde{z}^{(k+1)} &= \lambda_1 \mathrm{Sign}(\lambda_1^k \zeta_1) v_1 + O(q^k) \; . \end{aligned}$$

Daraus folgt unmittelbar die Behauptung. □

Bemerkung 4.9.

- Die Normierung $\tilde{z}^{(k)} \to z^{(k)}$ in (4.6) ist sinnvoll, wenn auch nicht notwendig.
- Die Voraussetzung $\zeta_1 \neq 0$ kann natürlich nicht a–priori überprüft werden, da die Eigenvektoren nicht bekannt sind.

Die Potenzmethode von v. Mises kann in dieser Form nur verwendet werden, um λ_1 zu bestimmen. Zur Berechnung anderer Eigenwerte von A kann jedoch A geeignet transformiert werden.

Beispiel 4.10. 1. Ist $\lambda_n \neq 0$ und verwendet man A^{-1} statt A in (4.6), dann wird dies *inverse Iteration* genannt. A^{-1} hat die Eigenwerte λ_i^{-1} mit

$$|\lambda_n^{-1}| > |\lambda_{n-1}^{-1}| > \cdots > |\lambda_1^{-1}|$$

mit den gleichen Eigenvektoren wie A. Also approximiert die inverse Iteration $|\lambda_n^{-1}|$ und den zugehörigen Eigenvektor v_n.

2. Ist λ eine Näherung an einen Eigenwert von A, liegt aber selbst nicht im Spektrum $\sigma(A)$, dann ergibt (4.6) mit $(A-\lambda I)^{-1}$ anstellen von A die *gebrochene Iteration von Wielandt* . $(A-\lambda I)^{-1}$ hat die Eigenwerte $(\lambda_i - \lambda)^{-1}, i = 1, \ldots n$.

4.3 Singuläre Werte

Satz 4.11. *Sei $A \in \mathbb{C}^{m\times n}$ mit rang$(A) = p \leq \min\{m, n\}$. Dann existiert eine* Singulärwertzerlegung *(SVD), das ist ein Triple*

$$\{(\sigma_i, \vec{u}_j, \vec{v}_k) : i = 1, \ldots, p\,, j = 1, \ldots, m\,, k = 1, \ldots, n\}\,,$$

mit reellen Zahlen $\{\sigma_i : i = 1, \ldots, p\}$,

$$\sigma_1 \geq \sigma_2 \geq \ldots \geq \sigma_p > 0$$

und Orthonormalbasen $\{\vec{u}_j\}_{j=1}^m$ and $\{\vec{v}_k\}_{k=1}^n$ von $\mathbb{C}^m$ und $\mathbb{C}^n$, sodass

$$\begin{aligned} A\vec{v}_i = \sigma_i\vec{u}_i\,, \quad A^*\vec{u}_i = \sigma_i\vec{v}_i \quad i = 1, \ldots, p\,, \\ A\vec{v}_k = 0\,, \quad A^*\vec{u}_j = 0 \quad j, k > p\,. \end{aligned}$$

In Matrixschreibweise kann man die SVD kompakter schreiben: Sei

$$U := [\vec{u}_1, \ldots, \vec{u}_m] \in \mathbb{R}^{m\times m}\,, \quad V := [\vec{v}_1, \ldots, \vec{v}_n] \in \mathbb{R}^{n\times n}\,,$$

und

$$\Sigma := \left[\begin{array}{ccc|c} \sigma_1 & & 0 & 0 \\ & \ddots & & \vdots \\ 0 & & \sigma_p & 0 \\ \hline & 0 \;\vdots\; 0 & & 0 \end{array}\right]$$

Die Matrizen U und V sind *unitär*, d.h.

$$U^*U = UU^* = I \in \mathbb{R}^{m\times m} \text{ und } V^*V = VV^* = I \in \mathbb{R}^{n\times n}\,.$$

Somit gilt

$$A = U\Sigma V^* \text{ and } A^* = V\Sigma U^*\,. \tag{4.10}$$

Damit gilt

$$A^*A = V\Sigma^2V^* \text{ wobei } \Sigma^2 = \text{diag}(\sigma_1^2, \sigma_2^2, \ldots, \sigma_n^2)\,,$$

$$\Sigma^2 V^* x = \lambda V^* x \ .$$

Dies bedeutet aber, dass $V^* x = e_j$ sein muss (also $x = V e_j$) und $\lambda = \sigma_j^2$.

Daher ist ein naheliegender Algorithmus zur numerischen Bestimmung einer Singulärwertzerlegung der Matrix die numerische Lösung des symmetrischen Eigenwertproblems zu $A^* A$. Allerdings ist ein solcher Algorithmus numerisch instabil. Stabile Algorithmen beruhen auf Householder Transformationsmatrizen.

Korollar 4.12. *Sei $A \in \mathbb{R}^{n \times n}$ symmetrisch mit rang$(A) = p \leq n$. Dann gilt*

$$A = U \Sigma U^* \, ,$$

wobei U unitär ist. In diesem Fall sind die Koeffizienten σ_i die Eigenwerte von A.

Kapitel 5

Nichtlineare Gleichungen

Nach den linearen Gleichungen untersuchen wir nun nichtlineare Gleichungen in einer oder mehreren reellen Variablen. Nichtlineare Gleichungen werden zumeist als Nullstellenaufgaben formuliert, d.h., gesucht ist die Nullstelle der Abbildung

$$f : \mathcal{D}(f) \subseteq \mathbb{R}^n \to \mathbb{R}^n \ .$$

Durch die Transformation $f(x) := g(x) - x$ kann jede nichtlineare Gleichung $g(x) = x$ unmittelbar in eine solche Nullstellenaufgabe übergeführt werden.

Da nichtlineare Gleichungen in der Regel nicht mehr geschlossen gelöst werden können, kommen nur Näherungsverfahren in Frage. Häufig besteht dabei ein Iterationsschritt in der Lösung eines linearen Teilproblems.

5.1 Konvergenzordnung

Wir unterscheiden verschiedene Konvergenzbegriffe bei iterativen Lösungsverfahren.

Definition 5.1. Sei $\{\varepsilon_k\}_{k\in\mathbb{N}_0}$ eine positive reelle Nullfolge. Man sagt, dass die Konvergenz dieser Folge (mindestens) die Ordnung $p \geq 1$ hat, wenn ein $C > 0$ und ein $k_0 \in \mathbb{N}$ existiert, so dass

$$\varepsilon_{k+1} \leq C\varepsilon_k^p, \quad \forall k \geq k_0 \ . \tag{5.1}$$

Für $p = 1$ wird vorausgesetzt, dass $C < 1$ ist.

Entsprechend wird die Konvergenzordnung einer konvergenten Folge $\{x^{(k)}\} \subseteq \mathbb{R}^n$ mit Grenzwert $\hat{x}$ über die Konvergenzordnung der Fehlerfolge

$$\varepsilon_k = \|x^{(k)} - \hat{x}\|$$

definiert.

Beispiel 5.2.

1. Der Fall $p = 1$ ist von besonderer Bedeutung und uns bereits im Zusammenhang mit dem Banachschen Fixpunktsatz begegnet. Die Fixpunktiteration
$$x^{(k+1)} = Tx^{(k)} + c\,, \quad T \in \mathbb{R}^{n\times n},\ c, x^{(0)} \in \mathbb{R}^n\,,$$
hat die Konvergenzordnung $p = 1$ (und nicht mehr), falls $0 < \rho(T) < 1$.

2. Der Fall $p = 2$ (quadratische Konvergenz) wird uns im Zusammenhang mit dem *Newton-Verfahren* begegnen.

3. p braucht nicht unbedingt eine ganze Zahl sein. Ein entsprechendes Beispiel ist das *Sekantenverfahren*.

4. Die Folge $\varepsilon_k := k^{-\nu}$, $\nu \in \mathbb{R}^+$, konvergiert gegen 0, erfüllt aber keine Ungleichung der Form (5.1). Man spricht in diesem Fall von *sublinearer Konvergenz* (langsamer als linear).

   Entsprechend heißt eine Nullfolge $\{\varepsilon_k\}$ **superlinear konvergent** (schneller als linear), falls
$$\frac{\varepsilon_{k+1}}{\varepsilon_k} \to 0\,, \quad k \to \infty\,.$$
   Insbesondere ist das Verfahren von der Ordnung 1.

Bei nichtlinearen Verfahren ist es wichtig zwischen *lokaler* und *globaler* Konvergenz zu unterscheiden:

Definition 5.3. Ein Iterationsverfahren $x^{(k+1)} = \Phi(x^{(k)})$ mit einer Funktion $\Phi : \mathcal{D}(\Phi) \subseteq \mathbb{R}^n \to \mathbb{R}^n$ heißt *lokal konvergent* gegen $\hat{x} \in \mathbb{R}^n$, falls eine offene Umgebung $\mathcal{U} \subseteq \mathcal{D}(\Phi)$ um $\hat{x}$ existiert, so dass für Startvektoren $x^{(0)} \in \mathcal{U}$ die resultierende Folge $\{x^{(k)}\}$ gegen $\hat{x}$ konvergiert.

Man spricht von *globaler Konvergenz*, falls $\mathcal{U} = \mathcal{D}(\Phi) = \mathbb{R}^n$ ist.

Satz 5.4. *Sei $\emptyset \neq \mathcal{D}(\Phi)$ offen. Die Funktion $\Phi : \mathcal{D}(\Phi) \subseteq \mathbb{R}^n \to \mathbb{R}^n$ sei stetig differenzierbar mit Fixpunkt $\hat{x}$. Ferner sei $\|\cdot\|$ eine Norm in $\mathbb{R}^n$ und $|||\cdot|||$ eine verträgliche Matrixnorm mit der Eigenschaft, dass*
$$|||\nabla\Phi(x)||| < 1 \qquad \forall x \in \mathcal{D}(\Phi)\,.$$
Dann konvergiert das Iterationsverfahren
$$x^{(k+1)} = \Phi(x^{(k)})\,, \quad k = 0, 1, 2, \cdots$$
(mindestens) lokal linear gegen $\hat{x}$.

Beweis. Wegen der Stetigkeit von $\nabla\Phi$ existiert eine abgeschlossene Kugel $\overline{B_\rho(\hat{x})} \subseteq \mathcal{D}(\Phi)$, sodass

$$|||\nabla\Phi(x)||| \leq q < 1\,, \quad \forall x \in \overline{B_\rho(\hat{x})}\,.$$

Aus dem Mittelwertsatz in $\mathbb{R}^n$ folgt

$$\Phi(y) - \Phi(x) = \int_0^1 \nabla\Phi(x + t(y-x))(y-x)\,dt\,,$$

und somit gilt

$$\|\Phi(y) - \Phi(x)\| = \int_0^1 |||\nabla\Phi(x + t(y-x))|||\,\|y-x\|\,dt \leq q\|y-x\|\,.$$

Das bedeutet, dass Φ eine Kontraktion auf $\overline{B_\rho(\hat{x})}$ ist. Für $y = \hat{x}$ sieht man ferner, dass Φ eine Selbstabbildung der Menge $\overline{B_\rho(\hat{x})}$ ist, denn für $x \in \overline{B_\rho(\hat{x})}$ ist

$$\|\Phi(x) - \hat{x}\| = \|\Phi(x) - \Phi(\hat{x})\| \leq q\|x - \hat{x}\| \leq q\rho < \rho\,,$$

also ist auch $\Phi(x) \in \overline{B_\rho(\hat{x})}$. Damit folgt die Behauptung des Satzes aus dem Banachschen Fixpunktsatz. □

Die Existenz des Fixpunktes braucht man nur für die Selbstabbildungseigenschaften in $\overline{B_\rho(\hat{x})}$.

Satz 5.5. *Sei $p = 2, 3, \ldots$. Die Funktion $\Phi : [a,b] \to \mathbb{R}$ sei $(p+1)$-mal stetig differenzierbar in (a,b). Sei $\hat{x} \in (a,b)$ ein Fixpunkt von Φ, d.h., $\Phi(\hat{x}) = \hat{x}$.*

- *Gilt*
$$0 = \Phi'(\hat{x}) = \cdots = \Phi^{(p-1)}(\hat{x}) \text{ und } \Phi^{(p)}(\hat{x}) \neq 0\,, \tag{5.2}$$
dann konvergiert das Iterationsverfahren $x^{(k+1)} = \Phi(x^{(k)})$ lokal genau mit der Ordnung p gegen $\hat{x}$.

- *Gilt*
$$0 = \Phi'(\hat{x}) = \cdots = \Phi^{(p-1)}(\hat{x}) = \Phi^{(p)}(\hat{x})\,, \tag{5.3}$$
dann konvergiert das Iterationsverfahren $x^{(k+1)} = \Phi(x^{(k)})$ lokal mit der Ordnung $p+1$ gegen $\hat{x}$.

Beweis. Zuerst wenden wir Satz 5.4 auf ein Kugel $\mathcal{B}_r(\hat{x})$ mit $\overline{\mathcal{B}_r(\hat{x})} \subseteq (a,b)$ an, für die gilt:

$$|\Phi'(x)| \leq q < 1\,, \qquad \forall x \in \overline{\mathcal{B}_r(\hat{x})}\,.$$

Dies ist wegen (5.2), bzw. (5.3) möglich. Darüberhinaus gilt wegen Satz 5.4, dass für jeden Startwert $x^{(0)} \in \overline{\mathcal{B}_r(\hat{x})}$, $x^{(k)} \to \hat{x}$.

Mit Taylorentwicklung ergibt sich nun

$$\Phi(x^{(k)}) = \Phi(\hat{x}) + \sum_{i=1}^{p} \frac{\Phi^{(i)}(\hat{x})}{i!}(x^{(k)} - \hat{x})^i + \frac{\Phi^{(p+1)}(\zeta)}{(p+1)!}(x^{(k)} - \hat{x})^{p+1};$$

für ein ζ zwischen $\hat{x}$ und $x^{(k)}$. Da $\hat{x}$ ein Fixpunkt ist, ist nach Voraussetzung

$$x^{(k+1)} = \Phi(x^{(k)}) = \hat{x} + \frac{\Phi^{(p)}(\hat{x})}{p!}(x^{(k)} - \hat{x})^p + \frac{\Phi^{(p+1)}(\zeta)}{(p+1)!}(x^{(k)} - \hat{x})^{p+1}\,. \quad (5.4)$$

- Im ersten Fall betrachten wir die Kugel auf $\mathcal{B}_s(\hat{x})$ mit $0 < s \leq r$ und

$$s < \frac{p+1}{2}\left|\frac{\Phi^{(p)}(\hat{x})}{\Phi^{(p+1)}(\zeta)}\right|\,, \qquad \forall \zeta \in \overline{\mathcal{B}_s(\hat{x})}\,. \quad (5.5)$$

  Da $x^{(k)}$ gegen $\hat{x}$ konvergiert existiert ein Index $k_0 \in \mathbb{N}$, sodass

$$x^{(k)} \in \overline{\mathcal{B}_s(\hat{x})}\,, \qquad \forall k \geq k_0\,.$$

  Wir nehmen nun an, dass $k \geq k_0$ gilt. Damit folgt aus (5.4) und der Dreiecksungleichung

$$|x^{(k+1)} - \hat{x}| \geq \left|\frac{\Phi^{(p)}(\hat{x})}{p!}(x^{(k)} - \hat{x})^p\right| - \left|\frac{\Phi^{(p+1)}(\zeta)}{(p+1)!}(x^{(k)} - \hat{x})^{p+1}\right|\,.$$

  Damit folgt aus (5.5)

$$\frac{1}{2}\frac{|\Phi^{(p)}(\hat{x})|}{p!}|x^{(k)} - \hat{x}|^p < |x^{(k+1)} - \hat{x}| < \frac{3}{2}\frac{|\Phi^{(p)}(\hat{x})|}{p!}|x^{(k)} - \hat{x}|^p\,.$$

  Damit ist die Konvergenzordnung genau p.

- Im zweiten Fall erhält man aus der Taylorentwicklung (5.4), dass ein ζ zwischen $\hat{x}$ und $x^{(k)}$, sodass

$$x^{(k+1)} - \hat{x} = \Phi(x^{(k)}) - \Phi(\hat{x}) = \frac{\Phi^{(p+1)}(\zeta)}{(p+1)!}(x^{(k)} - \hat{x})^{p+1}\,,$$

und damit gilt

$$\begin{aligned}\left|x^{(k+1)} - \hat{x}\right| &= \left|\frac{\Phi^{(p+1)}(\zeta)}{(p+1)!}\right| \left|x^{(k)} - \hat{x}\right|^{p+1} \\ &\leq C \left|x^{(k)} - \hat{x}\right|^{p+1} \ .\end{aligned}$$

Damit ist die Konvergenzordnung mindestens Konvergenzordnung p+1.

□

5.2 Das Newton-Verfahren reeller Funktionen

Im folgenden studieren wir Iterationsverfahren zur Lösung der eindimensionalen Nullstellengleichung

$$f(\hat{x}) = 0 \ . \tag{5.6}$$

Hier ist f eine reellwertige Funktion einer Variablen $x \in [a, b] \subseteq \mathbb{R}$.

Um die allgemeinen Ergebnisse dieses Kapitels nutzen zu können, bringen wir die Gleichung (5.6) zuerst in Fixpunktform. Wir wählen als Ansatz

$$x = x + q(x)f(x) =: \Phi(x)$$

mit einer glatten Funktion q, von der wir zunächst nur voraussetzen, dass $q(x)$ in einer Umgebung von $\hat{x}$ von Null verschieden ist. Um Satz 5.5 anwenden zu können, fordern wir zunächst, dass $\Phi'(x)$ verschwindet. Also, dass

$$\Phi'(\hat{x}) = 1 + q'(\hat{x})f(\hat{x}) + q(\hat{x})f'(\hat{x}) = 0 \ .$$

Da $f(\hat{x}) = 0$, erhalten wir die Bedingung

$$\Phi'(\hat{x}) = 1 + q(\hat{x})f'(\hat{x}) + q'(\hat{x}) \underbrace{f(\hat{x})}_{=0} = 0 \ ,$$

oder

$$q(\hat{x}) = -\frac{1}{f'(\hat{x})} \ .$$

Dies ist natürlich nur für $f'(\hat{x}) \neq 0$ möglich. Mit der Wahl $q(x) = -1/f'(x)$ erhalten wir das Newton-Verfahren.

Satz 5.6. *Sei $f \in C^3[a,b]$ und $\hat{x} \in (a,b)$ mit $f(\hat{x}) = 0$ und $f'(\hat{x}) \neq 0$. Dann konvergiert das* Newton-Verfahren

$$x^{(k+1)} = x^{(k)} - \frac{f(x^{(k)})}{f'(x^{(k)})} \tag{5.7}$$

lokal quadratisch gegen $\hat{x}$.

Beweis. Der Fixpunktoperator

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

hat folgende Eigenschaften:

- $\Phi(\hat{x}) = \hat{x}$,
- $\Phi'(\hat{x}) = \frac{f(\hat{x})f''(\hat{x})}{(f'(\hat{x}))^2} = 0$.

Damit folgt aus Satz 5.5, dass das Newton-Verfahren lokal mindestens quadratisch konvergent ist . □

Leider ist die Konvergenz des Newton-Verfahrens in der Regel nur lokal. Nur in Ausnahmefällen kann globale Konvergenz garantiert werden. Eine solche Ausnahme ist der Fall einer konvexen Funktion f.

Satz 5.7. *Sei $I \subseteq \mathbb{R}$ ein Intervall und $f : I \to \mathbb{R}$ monoton wachsend und konvex mit (eindeutiger) Nullstelle $\hat{x} \in I$. Dann konvergiert das Newton-Verfahren für alle $x^{(0)} \in I$ mit $x^{(0)} \geq \hat{x}$ monoton gegen $\hat{x}$.*

Beweis. Wir nehmen an, dass für ein $k \geq 0$ die aktuelle Iterierte $x^{(k)} \geq \hat{x}$ ist und beweisen die Induktionsbehauptung

$$\hat{x} \leq x^{(k+1)} \leq x^{(k)}\,. \tag{5.8}$$

Hierzu wenden wir die Konvexitätsbedingung an, dass für alle $u, v \in I$ und $0 \leq \alpha \leq 1$

$$f(\alpha u + (1-\alpha)v) \leq \alpha f(u) + (1-\alpha)f(v)$$

gilt auf $u = x^{(k+1)}$ und $v = x^{(k)}$. Dies ergibt

$$\alpha f(x^{(k+1)}) \geq f(\alpha x^{(k+1)} + (1-\alpha)x^{(k)}) - (1-\alpha)f(x^{(k)})\,,$$

beziehungsweise

$$f(x^{(k+1)}) \geq \frac{f(x^{(k)} + \alpha(x^{(k+1)} - x^{(k)})) - f(x^{(k)})}{\alpha} + f(x^{(k)}) \; .$$

Dies gilt nach Voraussetzung für alle $\alpha \in (0, 1]$. Durch Grenzübergang $\alpha \to 0$ folgt somit

$$f(x^{(k+1)}) \geq f'(x^{(k)})(x^{(k+1)} - x^{(k)}) + f(x^{(k)}) \; .$$

Man beachte, dass eine konvexe Funktion differenzierbar ist, weshalb wir im Satz diese Eigenschaft nicht extra voraussetzen müssen. Die rechte Seite ist aufgrund der Newton Vorschrift (5.7) Null. Also ist $f(x^{(k+1)})$ nichtnegativ und wegen der Monotonie folglich $x^{(k+1)} \geq \hat{x}$. Da nach Voraussetzung und Induktionsvoraussetzung sowohl $f(x^{(k)})$ als auch $f'(x^{(k)})$ nichtnegativ sind, gilt wegen der Definition des Newton Verfahrens,

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

dass $x^{(k+1)} \leq x^{(k)}$. Damit ist die Induktionsbehauptung (5.8) vollständig bewiesen. □

5.2.1 Das Sekantenverfahren

Will man beim Newton-Verfahren (5.7) f' nicht auswerten, so ersetzt man die Ableitung $f'(x^{(k)})$ durch einen Differenzenquotienten, etwa

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

Auf diese Weise erhält man für $k \geq 1$ die Iterationsvorschrift des *Sekantenverfahrens*:

$$\begin{aligned} x^{(k+1)} &= x^{(k)} - \frac{x^{(k)} - x^{(k-1)}}{f(x^{(k)}) - f(x^{(k-1)})} f(x^{(k)} \\ &= \frac{x^{(k-1)} f(x^{(k)}) - x^{(k)} f(x^{(k-1)})}{f(x^{(k)}) - f(x^{(k-1)})} \; . \end{aligned} \tag{5.9}$$

Der Name Sekantenverfahren beruht auf der folgenden geometrischen Interpretation:

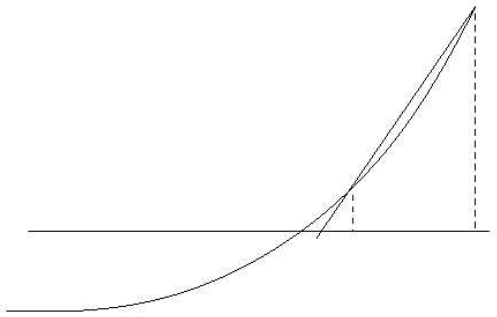

Abbildung 5.1: Geometrische Interpretation des Sekantenverfahrens

Satz 5.8. *f sei zweimal stetig differenzierbar in $[a,b]$ und es sei $f(\hat{x}) = 0$ für ein $\hat{x} \in (a,b)$ mit $f'(\hat{x}) \neq 0$ und $f''(\hat{x}) \neq 0$. Dann konvergiert das Sekantenverfahren lokal gegen $\hat{x}$ mit exakter Konvergenzordnung*

$$p = \frac{1}{2}(1+\sqrt{5}) = 1.61803\cdots$$

5.3 Das Newton–Verfahren in $\mathbb{R}^n$

Das Newton–Verfahren (5.7) lässt sich unmittelbar zur Nullstellenbestimmung einer Funktion $F : \mathcal{D}(F) \subseteq \mathbb{R}^n \to \mathbb{R}^n$ übertragen:

$$x^{(k+1)} = x^{(k)} - \nabla F(x^{(k)})^{-1} F(x^{(k)}) \; . \tag{5.10}$$

Hierbei ist $\nabla F(x)$ die *Jacobi-Matrix*

$$\nabla F(x) = \left[\frac{\partial F_i}{\partial x_j}\right]_{ij} \in \mathbb{R}^{n\times n} \; .$$

Dabei bezeichnet i die Zeilen und j die Spaltenposition.

Um die Wohldefiniertheit dieses Verfahrens sicherzustellen, müssen wir nun fordern, dass $\nabla F(\hat{x})$ nicht singulär ist; dies entspricht der Verallgemeinerung der Bedingung im eindimensionalen Fall, wo wir gefordert haben, dass $f'(\hat{x}) \neq 0$. Die Rekursion (5.10) ist äquivalent zu

$$F(x^{(k)}) + \nabla F(x^{(k)})(x^{(k+1)} - x^{(k)}) = 0 \; . \tag{5.11}$$

Also ist $x^{(k+1)}$ eine Nullstelle der Linearisierung von F um $x^{(k)}$. In (5.11) wird die ursprüngliche *nichtlineare* Gleichung

$$F(x) = 0$$

durch die lokale *Linearisierung*

$$F(x^{(k)}) + \nabla F(x^{(k)})(x - x^{(k)}) = 0$$

ersetzt.

Für die Implementierung des Newton–Verfahrens verwendet man zumeist die äquivalente Formulierung (5.11) anstelle von (5.10).

Algorithmus 5.9. *Newton–Verfahren in $\mathbb{R}^n$:*

- `Wähle` $x^{(0)} \in \mathcal{D}(F)$
- `for` $k = 1, 2, \cdots$
  `löse das lineare Gleichungssystem`

  $$\nabla F(x^{(k)}) h^{(k)} = -F(x^{(k)})$$

  `Ersetze` $x^{(k+1)} = x^{(k)} + h^{(k)}$ `und überprüfe` $x^{(k+1)} \in \mathcal{D}(F)$
  `end for`

In $\mathbb{R}^n$ verwendet man jenen linearen Gleichungssystemlöser, der die speziellen Eigenschaften der Jacobi–Matrizen am besten nutzt.

Wir beweisen nun einen Konvergenzsatz für Algorithmus 5.9.

Satz 5.10. *$\|\cdot\|$ und $|||\cdot|||$ seien ein Paar einer verträglichen Vektor- bzw. Matrixnorm. $F : \mathcal{D}(F) \subseteq \mathbb{R}^n \to \mathbb{R}^n$ habe eine Nullstelle $\hat{x}$ und sei stetig differenzierbar in einer Kugel $\mathcal{U} \subseteq \mathcal{D}(F)$ um $\hat{x}$. $\nabla F(x)$ sei invertierbar für alle $x \in \mathcal{U}$ und es existiert ein $L > 0$, sodass für alle $x, y \in \mathcal{U}$ gilt:*

$$\left|\left|\left|\nabla F(x)^{-1}(\nabla F(y) - \nabla F(x))\right|\right|\right| \leq L\|y - x\| \,. \tag{5.12}$$

Dann gilt: Ist $x^{(0)} \in \mathcal{U}$ mit

$$\rho := \|\hat{x} - x^{(0)}\| < 2/L \,,$$

dann liegen alle Iterierten $x^{(k)}$ von (5.10) in der Kugel $\mathcal{U}_\rho(\hat{x})$ mit Radius ρ um $\hat{x}$ und konvergieren quadratisch gegen $\hat{x}$,

$$\|\hat{x} - x^{(k+1)}\| \leq \frac{L}{2}\|\hat{x} - x^{(k)}\|^2 \,, \quad k = 0, 1, 2, \cdots \tag{5.13}$$

Beweis. Da $\hat{x}$ eine Nullstelle von F ist, gilt

$$\begin{aligned} x^{(k+1)} - \hat{x} &= x^{(k)} - \nabla F(x^{(k)})^{-1} F(x^{(k)}) - \hat{x} \\ &= x^{(k)} - \hat{x} - \nabla F(x^{(k)})^{-1}(F(x^{(k)}) - F(\hat{x})) \\ &= \nabla F(x^{(k)})^{-1} \left(F(\hat{x}) - F(x^{(k)}) - \nabla F(x^{(k)})(\hat{x} - x^{(k)}) \right) . \end{aligned}$$

Aus dem Mittelwertsatz folgt

$$\begin{aligned} & x^{(k+1)} - \hat{x} \\ &= \nabla F(x^{(k)})^{-1} \left(\int_0^1 \nabla F(x^{(k)} + t(\hat{x} - x^{(k)}))(\hat{x} - x^{(k)})\, dt \; - \nabla F(x^{(k)})(\hat{x} - x^{(k)}) \right) \\ &= \int_0^1 \nabla F(x^{(k)})^{-1} \left((\nabla F(x^{(k)} + t(\hat{x} - x^{(k)})) - \nabla F(x^{(k)}))(\hat{x} - x^{(k)}) \right) dt . \end{aligned}$$

Folglich gilt

$$\begin{aligned} & \|x^{(k+1)} - \hat{x}\| \\ &\le \int_0^1 \left| \left\| \nabla F(x^{(k)})^{-1} (\nabla F(x^{(k)} + t(\hat{x} - x^{(k)})) - \nabla F(x^{(k)})) \right\| \right| \|\hat{x} - x^{(k)}\|\, dt \\ &\le L \|x^{(k)} - \hat{x}\|^2 \int_0^1 t\, dt \\ &= \frac{L}{2} \|x^{(k)} - \hat{x}\|^2 , \end{aligned}$$

womit die quadratische Konvergenz (5.13) nachgewiesen ist. Gilt $\|x^{(k)} - \hat{x}\| \le \rho < 2/L$, dann gilt auch

$$\|x^{(k+1)} - \hat{x}\| \le \left(\frac{L}{2} \|x^{(k)} - \hat{x}\| \right) \|x^{(k)} - \hat{x}\| \le \rho \frac{L}{2} \|x^{(k)} - \hat{x}\| < \|x^{(k)} - \hat{x}\| < \rho .$$

Damit haben wir gezeigt, dass $\|x^{(k)} - \hat{x}\|$ monoton fallend ist und $x^{(k)} \in \mathcal{U}_\rho(\hat{x})$. Damit ist insbesondere die Folge $(\|x^{(k)} - \hat{x}\|)_{k \in \mathbb{N}}$ konvergent. Bezeichnen wir den Grenzwert mit ε, so erfüllt er wegen (5.13) die Ungleichung

$$0 \le \varepsilon \le \frac{L}{2} \varepsilon^2 \le \rho \frac{L}{2} \varepsilon .$$

Da $\rho \frac{L}{2} < 1$, muss $\varepsilon = 0$ sein. Also konvergiert $x^{(k)}$ gegen $\hat{x}$. □

Bemerkung 5.11.

1. In der Kugel $\mathcal{U}_\rho(\hat{x})$ kann es nur eine Nullstelle von F geben: ist nämlich $\tilde{x}$ eine zweite Nullstelle von F, dann konvergiert die Iteration (5.10) mit $x^{(0)} = \tilde{x}$ gegen $\tilde{x}$.

2. (5.12) ist eine Art Lipschitz–Bedingung an $\nabla F(\cdot)$. Die Konstante ist dabei unabhängig von möglichen Transformationen

$$\tilde{F}(x) := AF(x) \text{ mit nichtsingulärem } A \in \mathbb{R}^{n\times n} .$$

3. Für $n = 1$ ist (5.12) erfüllt, falls ∇F Lipschitz-stetig ist. Dies ist eine deutlich schwächere Voraussetzung als in Satz 5.6, wo wir gefordert haben, dass $f \in C^3(a, b)$.

Kapitel 6

Splines

Historisch wurde zunächst die Polynominterpolation zur Approximation skalarer Funktionen verwendet. Diese Art der Interpolation führt aber zu starken Oszillationen. Die Alternative ist die gesuchte Funktion durch stückweise zusammengesetzte Funktionen zu approximieren, die sogenannten *Splines*.

6.1 Treppenfunktionen

Gegeben sei ein reelles Intervall $[a, b]$ und ein Gitter

$$\Delta = \{a = x_0 < x_1 < \ldots < x_l = b\} \ . \tag{6.1}$$

Wir definieren

$$h := \max_{i=1,\ldots,l} h_i \, , \quad h_i = x_i - x_{i-1} \ .$$

Unter einer *Treppenfunktion* versteht man eine von rechts stetige Funktion s mit der Eigenschaft

$$s(x) = s_i \, , \quad x_{i-1} \leq x < x_i \, , \quad i = 1, \ldots, l \ .$$

Diese Treppenfunktionen bilden einen linearen Raum $S_{0,\Delta}$ der dimension l. Als Basisfunktion verwendet man üblicherweise die characteristischen Funktionen χ_i der l Teilintervalle. Es gilt somit

$$s(x) = \sum_{i=1}^{l} s_i \chi_i(x) \ .$$

Satz 6.1. *Sei* $f : [a, b] \to \mathbb{R}$. *Dann ist* $s(x) = \sum_{i=1}^{l} s_i \chi_i(x)$ *mit*

$$s_i = \frac{1}{h_i} \int_{x_{i-1}}^{x_i} f(x)\, dx\,, \quad i = 1, \ldots, l \tag{6.2}$$

Dann gilt: Das Funktional

$$\rho \in S_{0,\Delta} \to \int_a^b (f(x) - \rho(x))^2\, dx$$

ist minimal for s.

Beweis. Sei

$$\rho(x) = \sum_{i=1}^{l} \rho_i \chi_i(x) \in S_{0,\Delta}$$

beliebig. Dann gilt

$$\int_a^b (f(x) - \rho(x))^2\, dx \;=\; \sum_{i=1}^{l} \int_{x_{i-1}}^{x_i} (f(x) - \rho_i)^2\, dx\;.$$

Damit gilt, dass $\{s_i : i = 1, \ldots, l\}$, die Koeffizienten der minimierenden Funktion s das quadratische Funktional

$$s_i \to \int_{x_{i-1}}^{x_i} f^2(x)\, dx - 2\rho_i \int_{x_{i-1}}^{x_i} f(x)\, dx + h_i \rho_i^2$$

minimieren. Damit erfüllen sie auch die Optimalitätsbedingung,

$$\int_{x_{i-1}}^{x_i} f(x)\, dx = h_i s_i\,,$$

und damit die Behauptung. □

6.2 Lineare Splines

Definition 6.2. Ein *Spline vom Grad* n ist eine Funktion $s \in C^{n-1}[a, b]$, die auf jedem Intervall $[x_{i-1}, x_i)$ ein Polynom vom Grad n ist. Für den Raum der Splines vom Grad n schreiben wir $S_{n,\Delta}$.

Neben den Treppenfunktionen sind die stückweise linearen Funktionen, die sogenannten *linearen Splines* ($n = 1$), und die *kubischen Splines* ($n = 3$) wichtig. Bevor wir uns im Detail mit diesen Funktionen beschäftigen studieren wir einige allgemeine Resultate:

Bemerkung 6.3.
1. Der Raum $S_{n,\Delta}$ ist ein linearer Vektorraum, der die Polynome vom Grad n enthält.
2. Ist $s \in S_{n,\Delta}$, dann ist für $0 \leq k < n$, $s^k \in S_{n-k,\Delta}$ (die k-te Ableitung).
3. Die n-te Ableitung ist zwischen den Knoten konstant.

Proposition 6.4. *$S_{n,\Delta}$ ist ein $(n + l)$-dimensionaler Raum.*

Beweis. Sei $\hat{\chi}_i$ eine beliebige n-te Stammfunktion der charakteristischen Funktion χ_i. Aus der Definition ergibt sich, dass $\hat{\chi}_i \in S_{n,\Delta}$. Wie wir schon festgestellt haben, sind die Polynome vom Grad kleiner gleich n, und damit alle Monome x^j, $j = 0, 1, \ldots, n-1$, in $S_{n,\Delta}$ enthalten. Wir zeigen nun, dass

$$\{\hat{\chi}_i : i = 1, \ldots, l\} \cup \{x^j : j = 0, \ldots, n-1\}$$

eine Basis des $S_{n,\Delta}$ bildet.

Sei $s \in S_{n,\Delta}$ beliebig, so is $s^n \in S_{0,\Delta}$ und kann daher geschrieben werden als $s^n = \sum_{i=1}^{l} s_i \chi_i$. Folglich stimmen s und $\sum_{i=1}^{l} s_i \hat{\chi}_i$ bis auf ein Polynom p vom Grad $n-1$ überein, d.h.:

$$s = p + \sum_{i=1}^{l} s_i \hat{\chi}_i \text{ in } [a, b] \,. \tag{6.3}$$

Zum Nachweis der linearen Unabhängigkeit verwenden wir (6.3), und zeigen, dass für ein $s = 0$ auch $s_i = 0$ und $p = 0$ gilt.

Aus (6.3) folgt aus $s = 0$ somit auch

$$0 = s^n = \sum_{i=1}^{l} s_i \chi_i \text{ in } [a, b] \,,$$

woraus aus den Basiseigenschaften von χ_i auch folgt, dass $s_i = 0$ für alle $i = 1, \ldots, l$. Folglich folgt aus (6.3) und unserer Annahme $s = 0$, dass $s = p = 0$, und somit die Behauptung. □

Bemerkung 6.5.

- Für lineare Splines ergeben sich genau $(l+1)$ Freiheitsgrade.
- Anstatt (6.3) verwendet man in der Numerik die sogenannte *nodale Basis der Hutfunktionen* Λ_i, $i = 0, 1, \ldots, l$.
- Die Hutfunktionen erfüllen

$$\Lambda_i(x_j) = \delta_{ij}\,, \quad i, j = 0, \ldots, l\,. \tag{6.4}$$

(6.4) erlaubt eine einfache Bestimmung der Koeffizienten des interpolierenden linearen Splines:

Satz 6.6. *Seien $y_0, \ldots, y_l$ vorgegebene Werte. Dann ist $s = \sum_{i=0}^{l} y_i \Lambda_i \in S_{1,\Lambda}$ der eindeutig bestimmte lineare Spline mit*

$$s(x_i) = y_i\,, \quad i = 0, \ldots, l\,.$$

Beweis. Wegen (6.4) erfüllt $s = \sum_{i=0}^{l} y_i \Lambda_i$ die Interpolationseigenschaft an den Knoten. Die Eindeutigkeit folgt nun daraus, dass eine lineare Funktion auf $[x_{i-1}, x_i]$ durch die beiden Randwerte in diesem Intervall eindeutig bestimmt ist. □

Im Abschluß zu den linearen Splines beschäftigen wir uns noch mit der Berechnung des approximierenden linearen Splines: Dazu verwenden wir die *Gramsche Matrix* der linearen Splines

$$G = \left[\int_a^b \Lambda_i(x)\Lambda_j(x)\,dx\right]_{i,j=0,1,\ldots,l}\,. \tag{6.5}$$

Diese Matrix hat folgende Eigenschaften:

1. Die Dimension der Matrix ist $(l+1)\times(l+1)$ (also Knoten zum Quadrat).
2. Die Matrix ist regulär, da die Λ_i linear unabhängig sind.
3. Da für $|i-j| > 2$ $\int_a^b \Lambda_i(x)\Lambda_j(x)\,dx = 0$ gilt, ist G eine symmetrische Tridiagonalmatrix.
4. Die Diagonal- und Nebendiagonaleinträge ergeben sich wie folgt:

Diagonale: Für $i = j \in \{1, \ldots, l\}$ gilt:

$$\int_{x_{i-1}}^{x_i} \Lambda_i^2(x)\, dx = \int_0^{h_i} \frac{t^2}{h_i^2}\, dt = \frac{h_i}{3}\,.$$

Analog sieht man, dass für $i = j \in \{0, \ldots, l-1\}$

$$\int_{x_i}^{x_{i+1}} \Lambda_i^2(x)\, dx = \frac{h_{i+1}}{3}\,.$$

Zusammengefaßt gilt also für die Diagonale von G

$$g_{i,i} = \begin{cases} \frac{h_1}{3} & i = 0\,, \\ \frac{h_i + h_{i+1}}{3} & i = 1, \ldots, l-1\,, \\ \frac{h_l}{3} & i = l\,. \end{cases}$$

Nebendiagonalen: Wegen der Symmetrie reicht es, die untere Nebendiagonale zu bestimmen, also ist $j = i-1$, $i = 1, \ldots, l$:

$$\begin{aligned} g_{i,i-1} &= \int_{x_{i-1}}^{x_i} \Lambda_i(x)\Lambda_{i-1}(x)\, dx \\ &= \int_0^{h_i} \frac{t}{h_i}\frac{h_i - t}{h_i}\, dt \\ &= \frac{1}{h_i^2}\left(\frac{h_i^3}{2} - \frac{h_i^3}{3}\right) \\ &= \frac{h_i}{6}\,. \end{aligned}$$

Zusammengefasst:

$$G := \frac{1}{6}\begin{bmatrix} 2h_1 & h_1 & & & 0 \\ h_1 & 2(h_1+h_2) & h_2 & & \\ & h_2 & \ddots & \ddots & \\ & & \ddots & 2(h_{l-1}+h_l) & h_l \\ 0 & & & h_l & 2h_l \end{bmatrix}$$

Das folgende allgemeine Resultat gibt Auskunft, wie der best approximierende Spline mit der Gramschen Matrix berechnet werden kann.

Satz 6.7. *Sei $f \in X$, wobei X ein Hilbertraum ist, und $\{\phi_i : i \in \mathbb{N}\}$ eine Basis von X_n. $\mathcal{G}$ bezeichne die zu dieser Basis gehörige Gramsche Matrix. Dann ist $f_n := \sum_{i=1}^{n} x_i \phi_i \in X_n$ genau dann die beste Approximation an f aus X_n, wenn*

$$Gx = b \text{ für } x = \begin{pmatrix} x_1 \\ \vdots \\ x_n \end{pmatrix} \text{ und } \begin{pmatrix} \int_a^b \phi_1(x) f(x)\, dx \\ \vdots \\ \int_a^b \phi_n(x) f(x)\, dx \end{pmatrix} .$$

6.3 Kubische Splines

Kubische Splines, das sind die Elemente von $S_{3,\Delta}$, werden häufig in der Computergraphik verwendet.

Wir fassen einige allgemeine Resultate, die wir vorher kennengelernt haben zusammen:

1. Ein kubischer Spline ist zweimal stetig differenzierbar.

2. Wir wissen aus Bemerkung 6.3, dass wir zur eindeutigen Bestimmung eines kubischen Splines $(l + 3)$ Bedingungen brauchen.

3. Ist $s \in S_{3,\Delta}$, dann ist $s'' \in S_{1,\Delta}$. Damit kann man s'' mit Hilfe von Hutfunktionen darstellen:

$$s'' = \sum_{i=0}^{l} \gamma_i \Lambda_i \,, \tag{6.6}$$

   wobei $\gamma_i = s''(x_i)\,, i = 0, \ldots, l\,,$ gilt. Man nennt γ_i die *Momente* des kubischen Splines.

Im folgenden bestimmen wir die Bedingungen, die zu einer eindeutigen Berechnung des kubischen Splines führen.

Zuerst verwenden wir eine Darstellung für beliebige C^2-Funktion ρ in

einem Intervall $[x_{i-1}, x_i]$, die dann auf Splines angewendet wird:

$$\begin{aligned} & \rho(x) - \rho(x_i) \\ = & \int_{x_i}^{x} \rho'(t) \cdot 1 \, dt \\ \underbrace{=}_{\text{partielle Integration}} & \rho'(t)(t-x)|_{t=x_i}^{x} - \int_{x_i}^{x} \rho''(t)(t-x) \, dt \\ = & -\rho'(x_i)(x_i - x) - \int_{x_i}^{x} \rho''(t)(t-x) \, dt \, . \end{aligned} \tag{6.7}$$

Weiters stellen wir fest, dass für $t \in [x_{i-1}, x_i]$

$$\begin{aligned} s''(t) &= \gamma_{i-1}\Lambda_{i-1}(t) + \gamma_i \Lambda_i(t) \\ &= \gamma_{i-1}\frac{x_i - t}{x_i - x_{i-1}} + \gamma_i \frac{t - x_{i-1}}{x_i - x_{i-1}} \\ &= -\frac{\gamma_{i-1}}{h_i}(t - x_i) + \frac{\gamma_i}{h_i}(t - x_{i-1}) \\ &= \frac{\gamma_i - \gamma_{i-1}}{h_i}(t - x_i) + \gamma_i \end{aligned} \tag{6.8}$$

gilt, woraus sich für $x \in [x_{i-1}, x_i)$ ergibt

$$\begin{aligned} & s(x) - s(x_i) + s'(x_i)(x_i - x) \\ \underbrace{=}_{(6.7)} & -\int_{x_i}^{x} s''(t)(t-x) \, dt \\ \underbrace{=}_{(6.8)} & -\frac{\gamma_i - \gamma_{i-1}}{h_i} \int_{x_i}^{x} (t - x_i)(t-x) \, dt - \gamma_i \int_{x_i}^{x} t - x \, dt \\ \underbrace{=}_{\text{partielle Integration}} & \frac{\gamma_i - \gamma_{i-1}}{2h_i} \int_{x_i}^{x} (t - x_i)^2 \, dt \\ & + \frac{\gamma_i - \gamma_{i-1}}{2h_i}(t - x_i)^2(t-x)\Big|_{t=x_i}^{x} \\ & - \frac{\gamma_i}{2}(t-x)^2\Big|_{t=x_i}^{x} \\ = & \frac{\gamma_i - \gamma_{i-1}}{h_i}\frac{(x - x_i)^3}{6} + \gamma_i \frac{(x - x_i)^2}{2} \, . \end{aligned} \tag{6.9}$$

Mit der Abkürzung

$$s_i = s(x_i) \text{ und } s'_i = s'(x_i) \text{ für } i = 0, \ldots, l$$

erhalten wir aud (6.9) somit eine kompakte Identität für den kubischen Spline: Für $x \in [x_{i-1}, x_i]$ und $i = 1, \ldots, l$ gilt

$$s(x) = s_i + s'_i(x - x_i) + \gamma_i \frac{(x - x_i)^2}{2} + \frac{\gamma_i - \gamma_{i-1}}{h_i} \frac{(x - x_i)^3}{6} \,. \tag{6.10}$$

Insbesondere gilt somit für $i = 1, \ldots, l$

$$\begin{aligned} s_{i-1} &= s_i - s'_i h_i + \frac{\gamma_i h_i^2}{2} - \frac{(\gamma_i - \gamma_{i-1}) h_i^2}{6} \\ &= s_i - s'_i h_i + \frac{h_i^2}{6}(2\gamma_i + \gamma_{i-1}) \,, \\ s'_{i-1} &= s'_i - \frac{h_i}{2}(\gamma_{i-1} + \gamma_i) \,. \end{aligned} \tag{6.11}$$

Durch Kombination dieser Gleichungen erhalten wir für $i = 1, \ldots, l-1$

$$\begin{aligned} &\frac{s_{i+1} - s_i}{h_{i+1}} - \frac{s_i - s_{i-1}}{h_i} \\ =& s'_{i+1} - \gamma_i \frac{h_{i+1}}{6} - \gamma_{i+1} \frac{h_{i+1}}{3} - s'_i + \gamma_{i-1} \frac{h_i}{6} + \gamma_i \frac{h_i}{3} \\ =& (\gamma_i + \gamma_{i+1}) \frac{h_{i+1}}{2} - \gamma_i \frac{h_{i+1}}{6} - \gamma_{i+1} \frac{h_{i+1}}{3} + \gamma_{i-1} \frac{h_i}{6} + \gamma_i \frac{h_i}{3} \\ =& \frac{1}{6}(h_{i+1}\gamma_{i+1} + 2\gamma_i(h_i + h_{i+1}) + \gamma_{i-1} h_i) \,. \end{aligned}$$

In Matrixschreibweise lautet das System:

$$\frac{1}{6}\underbrace{\begin{bmatrix} h_1 & 2(h_1+h_2) & h_2 & & & 0 \\ & h_2 & 2(h_2+h_3) & \ddots & & \\ & & \ddots & \ddots & h_{l-1} & \\ & & & h_{l-1} & 2(h_{l-1}+h_l) & h_l \end{bmatrix}}_{\in\mathbb{R}^{(l-1)\times(l+1)}} \begin{bmatrix} \gamma_0 \\ \gamma_1 \\ \vdots \\ \gamma_{l-1} \\ \gamma_l \end{bmatrix}$$

$$= -\underbrace{\begin{bmatrix} -h_1^{-1} & h_1^{-1}+h_2^{-1} & -h_2^{-1} & & & 0 \\ & -h_2^{-1} & h_2^{-1}+h_3^{-1} & \ddots & & \\ & & \ddots & \ddots & -h_{l-1}^{-1} & \\ & & & -h_{l-1}^{-1} & h_{l-1}^{-1}+h_l^{-1} & -h_l^{-1} \end{bmatrix}}_{\in\mathbb{R}^{(l-1)\times(l+1)}} \begin{bmatrix} s_0 \\ s_1 \\ \vdots \\ s_{l-1} \\ s_l \end{bmatrix} . \tag{6.12}$$

Erfüllen umgekehrt die Momente γ_i und die Funktionswerte $s_i = s(x_i)$ die Gleichung (6.12), dann definieren diese Werte den interpolierenden kubischen Spline. Von dem sind die Ableitungen wegen (6.10) gegeben durch

$$s_i' = s'(x_i) = \frac{s_i - s_{i-1}}{h_i} + \gamma_{i-1}\frac{h_i}{6} + \gamma_i\frac{h_i^2}{3} .$$

Die beiden Matrizen in (6.12) haben die Dimension $(l-1)\times(l+1)$, und daher sind die Gleichungssysteme unterbestimmt. Man muß also zusätzliche Bedingungen aufnehmen. Bei *natürlichen kubischen Splines* fordert man zusätzlich, daß

$$s''(a) = s''(b) = 0 \tag{6.13}$$

gilt. Dies bedeutet aber wegen (6.6) gerade, dass

$$\gamma_0 = s''(a) = 0 \text{ and } \gamma_l = s''(b) = 0 . \tag{6.14}$$

Damit vereinfacht sich das Gleichungessystem (6.12) zu

$$\underbrace{\frac{1}{6}\begin{bmatrix} 2(h_1+h_2) & h_2 & & 0 \\ h_2 & 2(h_2+h_3) & \ddots & \\ & \ddots & \ddots & h_{l-1} \\ & & h_{l-1} & 2(h_{l-1}+h_l) \end{bmatrix}}_{:=\mathcal{G}\in\mathbb{R}^{(l-1)\times(l-1)}} \begin{bmatrix} \gamma_1 \\ \vdots \\ \gamma_{l-1} \end{bmatrix} = \begin{bmatrix} d_1 \\ \vdots \\ d_{l-1} \end{bmatrix}, \tag{6.15}$$

wobei

$$d_i = \frac{s_{i+1}-s_i}{h_{i+1}} - \frac{s_i - s_{i-1}}{h_i} = \frac{s_{i+1}}{h_{i+1}} - s_i\left(\frac{1}{h_i}+\frac{1}{h_{i+1}}\right) + \frac{s_{i-1}}{h_{i-1}}, \qquad i = 1,\dots,l-1\,. \tag{6.16}$$

Die Matrix $\mathcal{G}$ in (6.15) ist die Gramsche Matrix der Hutfunktionen $\{\Lambda_1,\dots,\Lambda_{l-1}\}$. Sie ist somit positiv definit und damit hat Gleichung (6.15) eine eindeutige Lösung. Wir fassen das Gesagte in einem Satz zusammen:

Satz 6.8. *Seien $y_0,\dots,y_l$ vorgegebene Daten, dann gibt es genau einen natürlichen Spline $s \in S_{3,\Delta}$ mit*

$$s(x_i) = y_i\,, \qquad i = 0,\dots,l\,.$$

Beachte, dass die Konstruktion es Splines eindeutig ist, d.h. die Lösung von (6.15) ist eindeutig. Die Splines sind aber gerade so, dass nur eine Basisfunktion ungleich Null ist.

Beispiel 6.9. Auf einem äquidistanten Gitter mit Gitterweite h. Wir bestimmen den natürlichen kubischen Spline s, der für ein festes $j = 0,1,\dots,l$ die Lagrange-Interpolationsaufgabe

$$s(x_i) = \delta_{ij}\,, \quad i = 0,\dots,l\,, \tag{6.17}$$

löst. Das Gleichungssystem (6.15), (6.16) hat die Gestalt

$$\begin{aligned}
&\begin{bmatrix} 4 & 1 & & 0 \\ 1 & 4 & \ddots & \\ & \ddots & \ddots & 1 \\ 0 & & 1 & 4 \end{bmatrix} \begin{bmatrix} \gamma_1 \\ \vdots \\ \gamma_{l-1} \end{bmatrix} \\
&= s_{i+1} - 2s_i + s_{i-1} \\
&= \frac{6}{h^2}(\delta_{i+1,j} - 2\delta_{i,j} + \delta_{i-1,j}) \\
&= \frac{6}{h^2}(e_{j+1} - 2e_j + e_{j-1})\,,
\end{aligned} \tag{6.18}$$

wobei $e_j, j = 1, \ldots, l-1$ die kartesischen Basisvektoren in $\mathbb{R}^{l-1}$ sind. Formal setzt man $e_0 = e_l = 0$.

Kapitel 7

Numerische Quadratur

Gegenstand dieses Kapitels ist die numerische Approximation bestimmter Integrale

$$I[f] = \int_a^b f(x)\,dx\,, \tag{7.1}$$

die nicht in geschlossener Form durch Angabe einer Stammfunktion integriert werden können.

7.1 Trapezregel

Die einfachsten Approximationsformeln sind die *Mittelpunktsformel*

$$\int_a^b f(x)\,dx \approx (b-a) f\left(\frac{a+b}{2}\right) \tag{7.2}$$

und die *Trapezformel*

$$\int_a^b f(x)\,dx \approx \frac{b-a}{2} f(a) + \frac{b-a}{2} f(b)\,. \tag{7.3}$$

Natürlich gilt bei (7.2) und (7.3) in der Regel keine Gleichheit. In der Praxis zerlegt man das Intervall $[a,b]$ in n (gleich große) Teilintervalle und wendet (7.2) und (7.3) auf jedes Teilintervall an. Bei der Mittelpunktsformel ergibt sich somit eine Riemann'sche Zwischensumme, während die Trapezformel auf

die *(zusammengesetzte) Trapezregel* (oder Trapezsumme) führt:

$$\begin{aligned} a = x_0 < x_1 < x_2 < \cdots < x_n = b\,, \\ x_i = a + ih\,, h = \frac{b-a}{n}\,, \\ T_n[f] := \sum_{i=1}^{n} \frac{x_i - x_{i-1}}{2}(f(x_i) + f(x_{i-1})) \\ = \frac{h}{2}f(a) + h\sum_{i=1}^{n-1} f(x_i) + \frac{h}{2}f(b)\,. \end{aligned} \tag{7.4}$$

Im folgenden betrachten wir effizientere Methoden zur Approximation von (7.1), Zur Approximation von $I[f]$ verwenden wir Ausdrücke der Form

$$Q[f] = \sum_{i=0}^{m} \hat{w}_i f(x_i)$$

mit *Knoten* $\{x_i : i = 0, \ldots, m\}$ und *Gewichten* $\{\hat{w}_i : i = 0, \ldots, m\}$. Unter dem zugehörigen *zusammengesetzten Quadraturverfahren* $Q_n[f]$ verstehen wir dann die Unterteilung von $[a, b]$ in n gleich große Teilintervalle, auf die jeweils eine Quadraturformel angewendet wird.

Qualitative Merkmale einer Quadraturformel bzw. eines zusammengesetzten Quadraturverfahrens sind der *Exaktheitsgrad* und die *Konvergenzordnung*.

Definition 7.1. Sei Π_m den Vektorraum aller Polynome vom Grad $\leq m$.

1. Eine Quadraturformel $Q[f]$ hat *Exaktheitsgrad* q, falls

$$Q[p] = I[p]\,, \quad \forall p \in \Pi_q\,.$$

2. Ein zusammengesetztes Quadraturverfahren konvergiert gegen $I[f]$ mit der Ordnung s, falls

$$|Q_n[f] - I[f]| = O(n^{-s})\,, \quad n \to \infty\,.$$

Beispiel 7.2. Sei $\hat{w} = 1$, dann hat die Trapezformel Exaktheitsgrad $q = 1$, und die zusammengesetzte Trapezformel konvergiert mit Ordnung $s = 2$ für alle $f \in C^2[a, b]$.

7.2 Polynominterpolation

Polynominterpolation ist die Aufgabe, bei vorgegebenen Knoten $x_0 < x_1 < \cdots < x_m$ und Werten $y_0, y_1, \ldots, y_m$ ein Polynom $p \in \Pi_m$ zu finden mit

$$p(x_i) = y_i\,, \quad i = 0, \ldots, m\,. \tag{7.5}$$

Definition 7.3. Wir bezeichnen mit

$$w(x) := \prod_{i=0}^{m} (x - x_i) \in \Pi_{m+1}$$

das zu den Knoten $\{x_i\}$ gehörige *Knotenpolynom*. Die Polynome

$$l_i(x) := \frac{w(x)}{(x - x_i)w'(x_i)} = \prod_{j=0, j\neq i}^{m} \frac{x - x_j}{x_i - x_j} \in \Pi_m$$

werden *Lagrange-Grundpolynome* genannt.

Für die Lagrange-Grundpolynome gilt, dass

$$l_i(x_j) = \delta_{ij}\,. \tag{7.6}$$

Satz 7.4. *Die Interpolationsaufgabe (7.5) hat genau eine Lösung p, nämlich*

$$p(x) = \sum_{i=0}^{m} y_i l_i(x)\,.$$

Beweis. Wegen (7.6) gilt $p(x_j) = \sum_{i=0}^{m} y_i l_i(x_j) = y_j$, also (7.5). Damit ist die Existenz einer Lösung gesichert. Seien $p, q \in \Pi_m$ zwei Lösungen der Interpolationsaufgabe (7.5), dann folgt

$$(p - q)(x_i) = 0\,, \quad i = 0, \ldots, m\,,$$

d.h., das Polynom $p - q \in \Pi_m$ hat $m + 1$ Nullstellen. Daraus folgt $p = q$. □

Beispiel 7.5. Die Funktion $f(x)$ wird durch das Polynom

$$p(x) = f(a) - \frac{f(b) - f(a)}{b - a}(x - a)$$

in den Punkten $(a, f(a))$ und $(b, f(b))$ interpoliert.

7.3 Newton-Cotes-Formeln

Mit Hilfe der Polynominterpolation lassen sich leicht Quadraturformeln für $I[f]$ mit beliebigem Exaktheitsgrad q angeben.

Seien $x_0 < x_1 < \cdots < x_m$ vorgegebene Knoten in $[a, b]$ und sei

$$\hat{w}_i := \int_a^b l_i(x)\,dx\,, \tag{7.7}$$

dann gilt:

Proposition 7.6. *Die* Newton-Cotes *Quadraturformel*

$$Q[f] = \sum_{i=0}^{m} \hat{w}_i f(x_i)$$

hat mindestens den Exaktheitsgrad $q = m$.

Beweis. Sei $p \in \Pi_m$. Wegen der Eindeutigkeit des Interpolationspolynoms gilt daher

$$p(x) = \sum_{i=0}^{m} p(x_i) l_i(x)$$

(vgl. Satz 7.4). Daraus folgt mit

$$I[p] = \int_a^b p(x)\,dx = \sum_{i=0}^{m} p(x_i) \int_a^b l_i(x)\,dx = \sum_{i=0}^{m} \hat{w}_i p(x_i) = Q[f]$$

die Behauptung. □

Beispiel 7.7. Wir beschränken uns auf den Fall äquidistanter Knoten $a = x_0 < x_1 < \cdots < x_m = b$ und $\hat{w} = 1$. In diesem Fall spricht man von *Newton-Cotes-Formeln*. Im Fall $m = 1$ erhält man speziell die Trapezregel (7.3). Für $m = 2$ ergibt sich die **Simpson-Formel**

$$\int_a^b f(x)\,dx \approx \frac{b-a}{6}\left(f(a) + 4f\left(\frac{a+b}{2}\right) + f(b)\right)\,. \tag{7.8}$$

Wegen Proposition 7.6 hat die Simpson-Regel mindestens Exaktheitsgrad 2. Tatsächlich hat sie sogar 3.

Wegen

$$b-a=\int_a^b 1\,dx = I[1]\underbrace{=}_{1\in\Pi_0} Q[1]=\sum_{i=0}^{m}\hat{w}_i\ . \tag{7.9}$$

Bemerkung 7.8. Im Fall $m=2$ erhält man so die *zusammengesetzte Simpson-Regel*: Mit $x_i=a+ih$, $i=0,\dots,2m$, und $h=\frac{b-a}{2n}$. erhält man

$$\int_a^b f(x)\,dx \sim$$
$$\frac{h}{3}\left\{f(a)+4f(x_1)+2f(x_2)+4f(x_3)+\cdots+2f(x_{2n-2})+4f(x_{2n-1})+f(b)\right\}$$

7.4 Gauß-Quadratur

Gegeben seien Knoten $x_0<x_1<\cdots<x_m$ und zugehörige Gewichte $\hat{w}_0,\hat{w}_1,\cdots,\hat{w}_m$. Wir diskutieren die Frage: Wie groß ist der maximale Exaktheitsgrad der Quadraturformel:

$$Q[f]=\sum_{i=0}^{m}\hat{w}_i f(x_i)\approx I[f]=\int_a^b f(x)\,dx\ . \tag{7.10}$$

Proposition 7.9. *Sei $\hat{w}>0$ in (a,b). Dann ist der Exaktheitsgrad der Quadraturformel $Q[\cdot]$ aus (7.10) höchstens $q=2m+1$.*

Beweis. Wähle

$$p(x)=\prod_{i=0}^{m}(x-x_i)^2\in\Pi_{2(m+1)}\ .$$

Offensichtlich ist $Q[p]=0$ und $I[p]=\int_a^b\prod_{i=0}^m(x-x_i)^2\,dx>0$, da $(x-x_i)^2$ und $\hat{w}$ jeweils positiv auf einer offenen Menge sind. □

Wir überlegen uns, dass man den Grad tatsächlich erreichen kann. Dazu leiten wir zuerst notwendige Bedingungen her. Dazu verwenden wir zuerst ein allgemeines Resultat über Orthogonalpolynome, welches wir ohne Beweis bringen:

Satz 7.10. *Sei*

$$<\phi,\psi>=\int_a^b\phi(x)\psi(x)\,dx\ .$$

Dann existiert eine eindeutige *Folge* $\{u_n : n = 0, \cdots, \infty\}$ *mit*

$$u_n(x) = \gamma_n x^n + \ldots + \gamma_0 \in \Pi_n \,,$$

mit

$$\gamma_n > 0 \text{ und } < u_n, u_m >= \delta_{m,n} \,, \quad , \quad \forall n, m \in \mathbb{N}_0 \,.$$

Insbesondere ist

$$u_0 \equiv \gamma_0 = \left(\int_a^b dx \right)^{-1/2} \,.$$

Setzt man $u_{-1} \equiv 0$ *und* $\beta_0 = 0$*, so erhält man*

$$\beta_{n+1} u_{n+1}(x) = x u_n(x) - \alpha_{n+1} u_n(x) - \beta_n u_{n-1}(x) \,, \quad n \in \mathbb{N} \,,$$

wobei

$$\alpha_{n+1} =< u_n, x u_n > \text{ und } \beta_{n+1} = \gamma_n / \gamma_{n+1} \,.$$

Wir stellen das $(m+1)$-te Orthogonalpolynom über seine Nullstellen dar:

$$u_{m+1}(x) = \prod_{i=0}^{m} (x - x_i) \in \Pi_{m+1} \,,$$

und fixieren die Nullstellen $\{x_i\}$ als Knoten für eine Quadraturformel. Die Gewichte $\hat{w}_i$ wählen wir wie in (7.7). Diese Quadraturformel wird als *(m+1)-stufige Gauß-Formel* $G[\cdot]$ bezeichnet.

Aus Proposition 7.7 folgt, dass die entsprechende Quadraturformel Q für alle $\phi \in \Pi_m$ exakt ist.

Eine vollständige Basis von Π_{2m+1} ist gegeben durch

$$\mathcal{V} := \{1, x, \ldots, x^m, u_{m+1}(x), x u_{m+1}(x), \ldots, x^m u_{m+1}(x)\}$$

und es gilt, weil u_{m+1} ein Orthogonalpolynom ist (und somit orthogonal zu x^s steht):

$$I[x^s u_{m+1}(x)] = \int_a^b x^s u_{m+1}(x)\, dx = 0 \,, \quad s = 0, \ldots, m \,.$$

Da auch $Q[x^s u_{m+1}] = 0$ gilt, integriert Q alle Elemente des Vektorraums der von $\mathcal{V}$ definiert ist, exakt. Dies ist die erste notwendige Bedingung für eine Quadraturformel vom Exaktheitsgrad $2m+1$. Weitere notwendige Bedingungen führen schlußendlich auf hinreichende Bedingungen.

Kapitel 8

Gewöhnliche Differentialgleichungen

Wir studieren numerische Verfahren zur Lösung von Systemen von gewöhnlichen Differentialgleichungen der Form

$$y' = f(t, y) \, , t \in [0, T] \text{ mit der Anfangsbedingung } y(0) = y_0 \, . \tag{8.1}$$

Dabei ist zu beachten, dass y eine vektorwertige Funktion sein kann. Wir sprechen in diesem Fall von einem *System erster Ordnung*.

8.1 Die Euler Verfahren

Das Euler-Verfahren approximiert y auf einem vorgegebenen Gitter

$$\Delta = \{0 = t_0 < t_1 < t_2 < ... < t_n\} \subseteq I$$

mittels der rekursiven Formel

$$y_{i+1} = y_i + (t_{i+1} - t_i) f(t_i, y_i) \, .$$

Beim impliziten Euler approximiert man die Lösung durch

$$y_{i+1} = y_i + (t_{i+1} - t_i) f(t_{i+1}, y_{i+1}) \, . \tag{8.2}$$

Dabei ist in jedem Schritt ein implizites Gleichungssystem zu lösen. Das Verfahren ist stabil (in einem genau zu definierendem Sinn), hat aber den Nachteil, dass es sehr langsam ist.

8.2 Runge-Kutta Verfahren

Der erhebliche Nachteil der beiden Euler-Verfahren ist ihre langsame Konvergenz (in Abhängigkeit der Zeitdiskretisierung). Schneller Konvergenz erreicht man mit Verfahren basierend auf dem Ansatz

$$y_{i+1} = y_i + h \sum_{j=1}^{s} b_j f(t_i + c_j h, n_j) \,, \quad \sum_{j=1}^{s} b_j = 1 \,, \tag{8.3}$$

mit Näherungen η_j für $y(t_i + c_j h)$; s nennt man dabei die *Stufenzahl* des Verfahrens.

Speziell bei den beiden Euler Verfahren ist jeweils $s = 1$ und $c_1 = 0$, $\eta_1 = y_i$ (explizites Euler-Verfahren), bzw., $c_1 = 1$, $\eta_1 = y_{i+1}$ (implizites Euler-Verfahren).

Da bei (8.3) jeweils ausgehend von $y_i \approx y(t_i)$ die nächste Näherung $y_{i+1} \approx y(t_{i+1})$ berechnet wird, spricht man bei Verfahren dieser Art von *Einschrittverfahren*. Im Gegensatz dazu verwenden *Mehrschrittverfahren* auch ältere Näherungen $y_{i-1}, ...$, zur Berechnung von y_{i+1}.

Nehmen wir nun an, dass $y_i = y(t_i)$ auf der exakten Lösungskurve liegt (lokaler Fehler) dann ergibt sich mit dem Hauptsatz der Differentialrechnung

$$\begin{aligned} y(t_{i+1}) - y_{i+1} &\underbrace{=}_{\text{Annahme } y(t_i)=y_i} y(t_{i+1}) - y(t_i) - h \sum_{j=1}^{s} b_j f(t_i + c_j h, n_j) \\ &= \int_{t_i}^{t_{i+1}} y'(t)\, dt - h \sum_{j=1}^{s} b_j f(t_i + c_j h, n_j) \\ &\underbrace{=}_{Dgl.} \int_{t_i}^{t_{i+1}} f(t, y(t))\, dt \; - h \sum_{j=1}^{s} b_j f(t_i + c_j h, n_j) \,. \end{aligned}$$

Wir sehen daher, dass der lokale Fehler klein wird, falls die Summe $h \sum_{j=1}^{s} b_j f(t_i + c_j h, n_j)$ eine gute Approximation des entsprechenden Integrals $\int_{t_i}^{t_{i+1}} f(t, y(t))\, dt$ ist.

Daher liegt es nahe, *Quadraturformeln* zur Wahl der Parameter $\{b_j\}$, $\{c_j\}$ und $\{\eta_j\}$ heranzuziehen.

Beispiel 8.1. Mit der Mittelpunktsformel ergibt sich der Ansatz

$$y_{i+1} = y_i + h f(t_i + h/2, \eta_1) \,, \tag{8.4}$$

wobei idealerweise $\eta_1 = y(t_i + h/2)$ sein sollte: Da dieser Wert nicht bekannt ist, muß eine Näherung gefunden werden. Das *Verfahren von Runge* (1895) verwendet die Näherung

$$\eta_1 = y(t_i) + \frac{h}{2} y'(t_i) \approx y_i + \frac{h}{2} f(t_i, y_i) \; .$$

Die Verwendung von Näherungen für y ist der wesentliche Unterschied (und Komplikation) zu Quadraturformeln.

Bei der Trapezformel ergibt sich

$$y_{i+1} = y_i + \frac{h}{2} f(t_i, y_i) + \frac{h}{2} f(t_{i+1}, \tilde{\eta}_1) \, ,$$

wobei nun $\tilde{\eta}_i \approx y(t_i + h)$. Geht man wie beim Verfahren vom Runge vor und ersetzt man

$$\tilde{\eta}_1 = y_i + h y'(t_i) \, ,$$

dann ergibt sich das *Verfahren von Heun*.

Die *Runge-Kutta Verfahren* beruhen auf folgender Wahl der Koeffizienten η_j:

$$\eta_j \approx y(t_i + c_j h) = y(t_i) + \int_{t_i}^{t_i + c_j h} y'(t) \, dt \; = y(t_i) + \int_{t_i}^{t_i + c_j h} f(t, y(t)) \, dt \; . \quad (8.5)$$

Zur Auswertung des Integrals bieten sich wieder Quadraturformeln an, wobei man sich üblicherweise darauf einschränkt $f(t, y)$ nur an den gleichen Knotenpunkten $f(t_i + c_j h, \eta_j)$, $j = 1, ..., s$, auszuwerten, wie sie zur Berechnung von y_{i+1} herangezogen werden. Das ergibt folgenden Ansatz:

$$\eta_j = y_i + h \sum_{k=1}^{s} a_{jk} f(t_i + c_k h, \eta_k) \, , \quad \sum_{k=1}^{s} a_{jk} = c_j \; . \quad (8.6)$$

Üblicherweise werden die Koeffizienten $\{a_{jk}, b_j, c_j\}$ in einem quadratischen Tableau zusammengefasst (das so genannte *Runge-Kutta Abc*),

$$\begin{array}{c|c} \mathrm{c} & \mathrm{A} \\ \hline & \mathrm{b}^t \end{array} \quad = \quad \begin{array}{c|ccccc} c_1 & a_{1,1} & ... & ... & ... & a_{1,s} \\ c_2 & a_{2,1} & a_{2,2} & ... & ... & ... \\ c_3 & a_{3,1} & a_{3,2} & ... & ... & ... \\ ... & ... & ... & ... & ... & ... \\ c_s & a_{s,1} & ... & ... & a_{s,s-1} & a_{s,s} \\ \hline & b_1 & b_2 & ... & b_{s-1} & b_s \end{array}$$

wobei wir $A = [a_{j,k}] \in \mathbb{R}^{s\times s}$, $b = [b_1, ..., b_s]^t \in \mathbb{R}^s$ und $c = [c_1, ..., c_s]^t \in \mathbb{R}^s$ gesetzt haben. Wir sprechen ab nun von dem Runge-Kutta Verfahren (A, b, c).

Beispiel 8.2. Für das explizite und implizite Euler Verfahren ergeben sich folgende Tableaus:

$$\begin{array}{c|c} 0 & 0 \\ \hline & 1 \end{array} \qquad \begin{array}{c|c} 1 & 1 \\ \hline & 1 \end{array}$$

Das Verfahren von Runge kann wird mit einer trivalen Gleichung ergänzt, um es in das allegmeine Schema zu zwingen:

$$\begin{aligned} \eta_0 &= y_i + h\sum_{k=0}^{1} 0 \cdot f(t_i + c_k h, \eta_k) \\ \eta_1 &= y_i + \tfrac{h}{2} f(t_i + h/2, \eta_0) \\ y_{i+1} &= y_i + h f(t_i + h/2, \eta_1)\ . \end{aligned}$$

Diese drei Gleichungen werden mit folgendem Runge-Kutta Tableau beschrieben

$$\begin{array}{c|cc} 0 & 0 & 0 \\ 1/2 & 1/2 & 0 \\ \hline & 0 & 1 \end{array}$$

Literaturverzeichnis

[1] P. Deuflhard and A. Hohmann. *Numerische Mathematik I. Eine algorithmisch orientierte Einführung*. De Gruyter, Berlin, 1993. 2., überarb. Aufl.

[2] G. H. Golub and J. M. Ortega. *Wissenschaftliches Rechnen und Differentialgleichungen*. Berliner Studienreihe zur Mathematik. Heldermann Verlag, Berlin, 1995.

[3] G. Haemmerlin and K.-H. Hoffmann. *Numerische Mathematik*. Springer Verlag, Berlin, Heidelberg, New York, fourth edition, 1994.

[4] M. Hanke. *Grundlagen der Numerischen Mathematik und des Wissenschaftlichen Rechnens*. Teubner, Stuttgart, Leipzig, Wiesbaden, 2002.

[5] N. J. Higham. *Accuracy and stability of numerical algorithms*. Society for Industrial and Applied Mathematics (SIAM), Philadelphia, PA, 1996.

[6] A. Quarteroni, R. Sacco, and F. Saleri. *Numerical Mathematics*. Springer Verlag, Berlin, 2000.

[7] H. R. Schwarz. *Numerische Mathematik*. B. G. Teubner, Stuttgart, fourth edition, 1997. With a contribution by Jörg Waldvogel.

[8] J. Stoer. *Numerische Mathematik 1*. Springer Verlag, Berlin, 1999.