

Höhere Mathematik für Ingenieure IV A

Sommersemester 2025

Stand April 29, 2025

Diese Notizen dienen als Begleitmaterial zur Vorlesung in diesem Semester und sind nicht als in sich geschlossenes Skript gedacht, sondern als Unterstützung zur Erstellung von eigenen Notizen während der Vorlesung. Entsprechend ist der Inhalt sehr knapp gehalten um während der Vorlesung selbstständig ergänzt zu werden und wird sich im Laufe des Semesters erweitern.

Insbesondere stellen diese Notizen keinen Anspruch auf Vollständigkeit und die Nummerierung der Aussagen und Beispiele wird nicht immer mit der Nummerierung in der Vorlesung übereinstimmen!

Ziel dieser Vorlesung ist exemplarisch Verfahren aus der Numerik kennenzulernen, mit denen wir Lösungen von Problemen annähern können, deren exakte Lösungen entweder nur sehr schwer oder gar nicht explizit berechenbar sind.

Bei Fragen oder Fehlern wenden Sie sich bitte an Christian Steinhart.

Contents

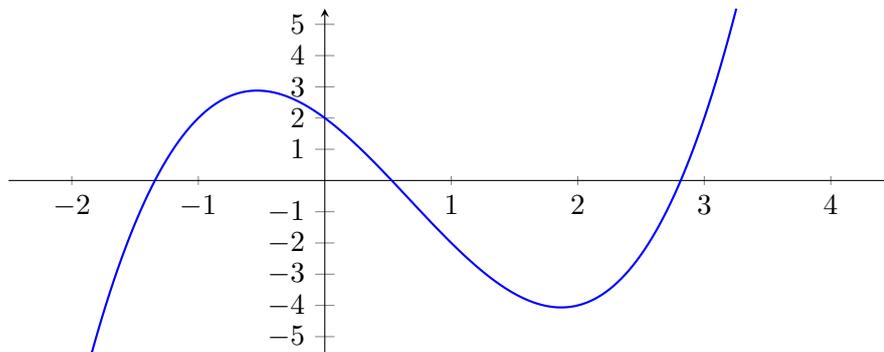
1 Banachscher Fixpunktsatz	3
2 Lineare Gleichungssysteme	12

1 Banachscher Fixpunktsatz

Als erste Motivation betrachten wir folgendes

Problem: Wir suchen eine Nullstelle für eine Funktion $f : \mathbb{R} \rightarrow \mathbb{R}$.

Im Folgenden betrachten wir hierzu das Polynom $f(x) = x^3 - 2x^2 - 3x + 2$:



Die sicherlich einfachste Methode hierzu wäre wiederholt den Zwischenwertsatz anzuwenden:

Algorithmus 1.1 (Bisektionsverfahren)

Ist f stetig und gilt für $a < b \in \mathbb{R}$ bereits $f(a) < 0 < f(b)$, dann enthält das Intervall $[a, b]$ nach dem ZWS eine Nullstelle von f . Wir wollen nun eine mögliche Nullstelle von f schrittweise (iterativ) annähern, indem wir das entsprechende Intervall in dem sich die Nullstelle enthält immer kleiner machen.

Hierzu probieren wir $c = (a + b)/2$. Wir haben nun die folgenden Fälle:

- Ist $f(c) = 0$, sind wir fertig.
- Ist $f(c) < 0$, dann hat f eine Nullstelle im Intervall $[c, b]$, wir können also unseren Anfangswert a durch c ersetzen.
- Ist $f(c) > 0$, dann hat f eine Nullstelle im Intervall $[a, c]$, wir können also unseren Anfangswert b durch c ersetzen.

Iterativ halbieren wir also die Länge des Intervalls in dem sich eine Nullstelle von f befindet bei jedem Schritt. Merken wir uns bei jedem Schritt die Intervallgrenzen erhalten wir also rekursiv die Folge:

$$\text{mit } a_0 = a, b_0 = b \quad \underline{\mathbf{x}}^{(n)} = \begin{pmatrix} a_n \\ b_n \end{pmatrix}$$

$$\text{und für } c_n := (a_n + b_n)/2 \quad \underline{\mathbf{x}}^{(n+1)} := \phi(\underline{\mathbf{x}}^{(n)}) := \begin{cases} \begin{pmatrix} a_n \\ c_n \end{pmatrix} & , \text{ falls } f(c_n) \geq 0 \\ \begin{pmatrix} c_n \\ b_n \end{pmatrix} & , \text{ falls } f(c_n) < 0 \end{cases}$$

Sind wir z.B. nur an einer Nullstelle $f(c) \approx 0$ interessiert, erhalten wir als Pseudo-Code:

Algorithm 1 Bisektionsverfahren

Input: Stetige Funktion f , $a < b$ mit $f(a) < 0 < f(b)$, Abbruchbedingung $\varepsilon > 0$

Output: Approximation einer Nullstelle

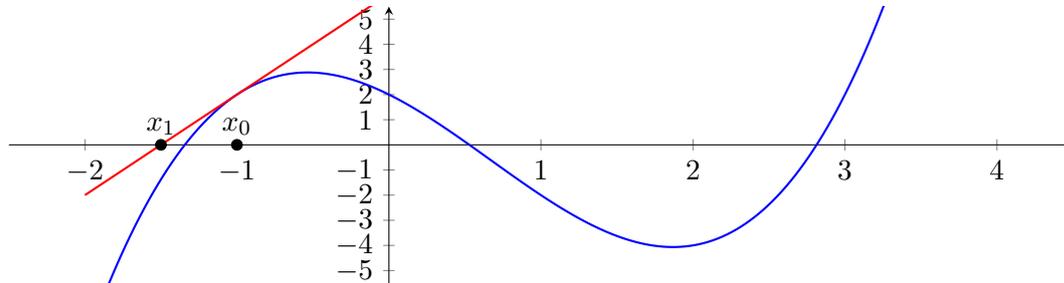
```
1:  $c \leftarrow (a + b)/2$  ▷ setze  $c$  auf den Mittelwert
2: while  $|f(c)| > \varepsilon$  do ▷ solange  $f(c)$  "weg" von der Null ist
3:   if  $f(c) > 0$  then
4:      $b \leftarrow c$ 
5:   else
6:      $a \leftarrow c$ 
7:   end if
8:    $c \leftarrow (a + b)/2$ 
9: end while
10: return  $c$  ▷ gib unsere Näherung als Antwort
```

Alternativ können wir als Abbruchbedingung in Zeile 2 auch den Abstand zur Nullstelle, d.h. die Größe des Intervalls $b - a < \varepsilon$ festlegen, je nachdem ob wir an der Nullstelle selbst interessiert sind oder ein c mit $f(c)$ nahe 0 suchen. Ist anfangs $f(a) > 0 > f(b)$, vertauschen sich entsprechend die Zeilen 4 und 6 im Algorithmus.

Aufgabe: Skizzieren Sie oben für die Startwerte $a = -2, b = 0$ die ersten drei Näherungen c_0, c_1 und c_2 des Algorithmus.

Algorithmus 1.2 (Newtonverfahren)

Wir “raten” einen Startwert x_0 nahe einer Nullstelle und betrachten als Annäherung von f bei x_0 die lineare Funktion $g(x) = f(x_0) + f'(x_0)(x - x_0)$:



Da g als Annäherung von f dient, wäre also die Nullstelle x_1 von g eine vermutlich bessere Wahl:

$$0 = g(x_1) = f(x_0) + f'(x_0)(x_1 - x_0) \iff x_1 = x_0 - \frac{f(x_0)}{f'(x_0)} =: \phi(x_0)$$

Dies wiederholen wir nun und erhalten iterativ die Folge $x_{n+1} := \phi(x_n) = x_n - \frac{f(x_n)}{f'(x_n)}$.

Frage: Konvergiert die Folge x_n (gegen eine Nullstelle und überhaupt)? Welche der beiden Verfahren sind schneller?

Definition 1.3

Ein *Iterationsverfahren* ist eine durch eine *Verfahrensfunktion* $\phi : D \rightarrow D$ und einen Startpunkt $\underline{\mathbf{x}}^{(0)} \in D$ gegebene Folge $(\underline{\mathbf{x}}^{(n)})_{n \in \mathbb{N}_0} \subset D$ mit

$$\underline{\mathbf{x}}^{(n)} := \phi^n(\underline{\mathbf{x}}^{(0)}) = \phi(\underline{\mathbf{x}}^{(n-1)}).$$

Wir sagen, dass das Iterationsverfahren *konvergiert*, falls die Folge $\underline{\mathbf{x}}^{(n)}$ konvergiert.

Definition 1.4

Sei V ein normierter Vektorraum (über \mathbb{R} oder \mathbb{C}).

- a) Eine Folge $(\underline{\mathbf{x}}^{(n)})_{n \in \mathbb{N}_0} \subseteq V$ heißt *konvergent*, falls ein *Grenzwert* $\underline{\mathbf{x}} \in V$ existiert sodass gilt:
Für alle $\varepsilon > 0$ existiert ein $N = N(\varepsilon) \in \mathbb{N}$ mit

$$\|\underline{\mathbf{x}}^{(n)} - \underline{\mathbf{x}}\| < \varepsilon \text{ für alle } n \geq N.$$

- b) Eine Folge $(\underline{\mathbf{x}}^{(n)})_{n \in \mathbb{N}_0} \subseteq V$ heißt *Cauchy-Folge*, falls:
Für jedes $\varepsilon > 0$ existiert ein $N = N(\varepsilon) \in \mathbb{N}$ mit

$$\|\underline{\mathbf{x}}^{(n)} - \underline{\mathbf{x}}^{(k)}\| < \varepsilon \text{ für alle } n, k \geq N.$$

Bemerkung 1.5

Konvergente Folgen sind nach der Dreiecksungleichung immer Cauchy-Folgen, aber nicht jede Cauchy-Folge konvergiert (in V).

Definition 1.6

Ein normierter Vektorraum heißt *vollständig* bzw. *Banachraum*, wenn jede Cauchy-Folge in V auch konvergiert.

Bemerkung 1.7

Endlich-dimensionale \mathbb{R} -Vektorräume sind immer vollständig.

Erinnerung 1.8

Sei V ein normierter Raum und $\phi : D \rightarrow V$ mit $D \subseteq V$ eine Abbildung.

- a) ϕ heißt (L) -Lipschitz-stetig (auf D), falls eine Lipschitz-Konstante $L > 0$ existiert sodass für alle $x, y \in D$ gilt:

$$\|\phi(x) - \phi(y)\| \leq L\|x - y\|.$$

- b) ϕ heißt *Kontraktion*, falls ϕ Lipschitz-stetig mit Lipschitz-Konstante $L < 1$ ist.

Beispiel 1.9

Sei $\phi : \mathbb{R} \rightarrow \mathbb{R}$ differenzierbar, dann gilt nach dem Mittelwertsatz $\frac{\phi(b) - \phi(a)}{b - a} = \phi'(\zeta)$ für ein $\zeta \in [a, b]$.

Ist also $|\phi'|$ beschränkt durch L , dann ist ϕ bereits L -Lipschitz-stetig.

Satz 1.10 (Banachscher Fixpunktsatz)

Sei $D \subseteq V$ eine abgeschlossene Teilmenge eines Banachraums V und $\phi : D \rightarrow D$ eine Kontraktion mit Lipschitz-Konstante $L < 1$. Dann gilt:

- a) ϕ hat einen eindeutigen Fixpunkt $\underline{x}^* \in D$.
- b) Für **jeden** Startwert $\underline{x}^{(0)} \in D$ konvergiert das Iterationsverfahren $\underline{x}^{(n)} := \phi^n(\underline{x}^{(0)})$ gegen \underline{x}^* .
- c) Wir erhalten als Fehlerabschätzungen für alle $n \in \mathbb{N}$:

$$\begin{aligned} \|\underline{x}^{(n)} - \underline{x}^*\| &\leq \frac{L^n}{1 - L} \cdot \|\underline{x}^{(1)} - \underline{x}^{(0)}\| \\ \text{und} \quad \|\underline{x}^{(n)} - \underline{x}^*\| &\leq \frac{1}{1 - L} \cdot \|\underline{x}^{(n+1)} - \underline{x}^{(n)}\| \end{aligned}$$

Beweis: • $\underline{x}^{(n)} := \phi^n(\underline{x}^{(0)})$ ist eine Cauchy-Folge:

$$\begin{aligned} \|\underline{x}^{(n+1)} - \underline{x}^{(n)}\| &= \|\phi(\underline{x}^{(n)}) - \phi(\underline{x}^{(n-1)})\| \\ &\leq L\|\underline{x}^{(n)} - \underline{x}^{(n-1)}\| \leq \dots \leq L^n\|\underline{x}^{(1)} - \underline{x}^{(0)}\| \\ \Rightarrow \|\underline{x}^{(n+k)} - \underline{x}^{(n)}\| &\leq \left(\sum_{k=0}^{\infty} L^k\right)\|\underline{x}^{(n+1)} - \underline{x}^{(n)}\| \leq \frac{L^n}{1 - L}\|\underline{x}^{(1)} - \underline{x}^{(0)}\| \xrightarrow{n \rightarrow \infty} 0 \end{aligned}$$

mit $\underline{x}^* := \lim \underline{x}^{(n)}$ folgt damit direkt c).

- \underline{x}^* ist FP:

$$0 \leq \|\phi(\underline{x}^*) - \underline{x}^*\| \leq \|\phi(\underline{x}^*) - \phi(\underline{x}^{(n)})\| + \|\underline{x}^* - \underline{x}^{(n)}\| \xrightarrow{n \rightarrow \infty} L \cdot 0 + 0$$

- Eindeutigkeit: Angenommen $\underline{\mathbf{y}} \neq 0$ ist ein weiterer FP, dann ist

$$0 < \|\underline{\mathbf{x}}^* - \underline{\mathbf{y}}\| = \|\phi(\underline{\mathbf{x}}^*) - \phi(\underline{\mathbf{y}})\| \leq L\|\underline{\mathbf{x}}^* - \underline{\mathbf{y}}\| < \|\underline{\mathbf{x}}^* - \underline{\mathbf{y}}\|$$

was der gewünschte Widerspruch ist. □

Beispiel 1.11

Wir suchen für die Funktion

$$f : \mathbb{R}_{>0} \rightarrow \mathbb{R}, \quad x \mapsto \sqrt{x} + 3$$

einen Fixpunkt. Man beachte, dass wegen $f'(x) = \frac{1}{2\sqrt{x}}$ die Funktion nicht auf gesamt $\mathbb{R}_{>0}$ eine Kontraktion ist. Schränken wir uns aber ein, erhalten wir wegen $f(x) > 3$ für alle $x > 0$ für $D := [3, \infty)$ bereits $f(D) \subseteq D$ und $f'(x) \leq \frac{1}{2\sqrt{3}} =: L$ für alle $x \in D$. Also können wir Satz 1.10 verwenden und erhalten als FP z.B. $x^* := \lim f^n(3)$. Als Vergleich kann man den Fixpunkt auch algebraisch berechnen (und die auftretende Wurzel numerisch annähern/dem Taschenrechner geben).

Eine Diskussion der auftretenden Fehlerabschätzungen findet sich bei den Programmschnipseln.

Zurück zu Newton:

Setzen wir der Einfachheit halber $a(x) = \frac{1}{f'(x)}$, hatten wir als Iterationsvorschrift

$$\phi(x) = x - a(x)f(x).$$

Wegen $a(x) \neq 0$, ist $f(x) = 0 \iff \phi(x) = x$. Um 1.10 anzuwenden, brauchen wir $\phi'(x) \leq L < 1$ auf D und $\phi(D) \subseteq D$. Probieren wir hier $D = [x_0 - r, x_0 + r]$ für einen Radius $r > 0$ ergibt das:

$$\phi'(x) = 1 - (af)'(x) \leq L \quad \forall x \in D$$

und wegen $f(D) \subset D \iff |\phi(x) - x_0| \stackrel{!}{\leq} r$:

$$\begin{aligned} |\phi(x) - x_0| &= |\phi(x) - \phi(x_0) + \phi(x_0) - x_0| \leq |\phi(x) - \phi(x_0)| + |\phi(x_0) - x_0| \\ &\leq L \cdot r + |a(x_0)f(x_0)| \stackrel{!}{\leq} r \\ \leadsto |f(x_0)| &\leq \frac{(1-L)r}{|a(x_0)|} \Rightarrow f(D) \subset D \end{aligned}$$

Zusammengefasst erfüllt erhalten wir also:

Satz 1.12

Für $x_0 \in \mathbb{R}$, $r > 0$ und $I := [x_0 - r, x_0 + r]$ seien $f, a : I \rightarrow \mathbb{R}$ zwei differenzierbare Funktionen mit $a(x) \neq 0$ für alle $x \in I$. Existiert zusätzlich ein $L < 1$ mit

- a) Für alle $x \in I$ gilt: $|1 - (af)'(x)| \leq L$
- b) $|f(x_0)| \leq \frac{(1-L)r}{|a(x_0)|}$

dann konvergiert das (modifizierte) Newton-Verfahren mit der Iterationsvorschrift

$$x_{n+1} = \phi(x_n) := x_n - a(x_n)f(x_n)$$

mit der Fehlerabschätzung

$$|x^* - x_n| \leq \frac{L^n}{1-L} |a(x_0)f(x_0)|$$

Bemerkung 1.13

- a) Die Bedingungen von Satz 1.12 sind umso leichter nachzuprüfen, je näher x_0 an einer Nullstelle von f liegt.

- b) Wir können z.B. $a(x) \equiv \frac{1}{f'(x_0)}$ setzen, um die Bedingungen von Satz 1.12 leichter zu prüfen (auf Kosten von etwas langsamerer Konvergenz).

Etwas einfacher ist die Situation, wenn wir konvexe/konkave Funktionen anschauen:

Satz 1.14

Sei $f : [a, b] \rightarrow \mathbb{R}$ eine konvexe Funktion, d.h. $f''(x) > 0$ für alle $x \in [a, b]$.

- a) Ist $f(a) < 0 < f(b)$, dann konvergiert das Newton-Verfahren mit Startwert $x_0 = b$.
- b) Ist $f(a) > 0 > f(b)$, dann konvergiert das Newton-Verfahren mit Startwert $x_0 = a$.

Beweis: Da f konvex ist, liegt die Tangente immer unterhalb des Graphen. Im Fall a) ist also $f(x_n) > 0$ und damit x_n monoton fallend und von unten beschränkt. Für die Folge gilt dann wegen

$$x_{n+1} = x_n - \frac{f(x_n)}{f'(x_n)},$$

da (x_n) eine Cauchy-Folge ist bereits $\lim \frac{f(x_n)}{f'(x_n)} = 0$ und da $f'(x)$ beschränkt ist direkt $\lim f(x_n) = f(x^*) = 0$. \square

Für konkave Funktionen erhält man die entsprechende Aussage mit Startwert mit negativem Funktionswert $f(x_0) < 0$.

Beispiel 1.15

Als direkte Anwendung des Newton-Verfahrens können beispielsweise auch Wurzeln bestimmt werden, d.h. suche Nullstelle von $f(x) = x^2 - d$ für ein $d > 0$.

- a) Wegen $f''(x) = 2 > 0$ ist f konvex und wegen $f(0) = -d < 0$ konvergiert nach Satz 1.14 das Newton-Verfahren für jeden Startwert x_0 mit $x_0^2 > d$. Wir erhalten als Iterationsvorschrift das *Heron-Verfahren*:

$$x_{n+1} = x_n - \frac{f(x_n)}{f'(x_n)} = x_n - \frac{x_n^2 - d}{2x_n} = \frac{1}{2} \left(x_n + \frac{d}{x_n} \right)$$

- b) Als Beispiel für eine Anwendung von Satz 1.12 mit $a(x) = \frac{1}{f'(x_0)}$ nehmen wir $d = 7$ und sehen leicht:

$$2^2 < 7 < 3^2 \Rightarrow 2 < \sqrt{7} < 3$$

Also probieren wir $I = [2, 3]$, d.h. $2 < x_0 := 2,5 < 3$ und $r = \frac{1}{2}$. Wir erhalten $a(x) = \frac{1}{f'(x_0)} = \frac{1}{5}$. Als Lipschitz-Konstante erhalten wir wegen $f'(x) = 2x \in [4, 6]$:

$$|1 - (af)'(x)| = |1 - \underbrace{\frac{1}{5} f'(x)}_{\in [4,6]}| \leq \frac{1}{5} =: L$$

Für Voraussetzung b) rechnen wir nach:

$$\begin{aligned} f(x_0) &= \frac{25}{4} - 7 = -\frac{3}{4} \\ \frac{(1-L)r}{a(x_0)} &= \frac{4/5 \cdot 1/2}{1/5} = \frac{4}{2} = 2 > |f(x_0)| \end{aligned}$$

Also konvergiert auch die Iterationsvorschrift $x_{n+1} = x_n - \frac{x_n^2 - 7}{5}$.

Bemerkung 1.16

Statt für reelle Funktionen geht das Newton-Verfahren (bzw. die Iterationsvorschrift) auch genauso gut für $f : \mathbb{R}^n \rightarrow \mathbb{R}^n$. Wie zuvor erhalten wir eine (hoffentlich bessere) Näherung an eine Nullstelle, indem wir f mit ihrer Jacobi-Matrix als lineare Funktion approximieren und erhalten die Vorschrift

$$\underline{\mathbf{x}}^{(n+1)} := \underline{\mathbf{x}}^{(n)} - J_f(\underline{\mathbf{x}}^{(n)})^{-1} \cdot f(\underline{\mathbf{x}}^{(n)}).$$

Ein klassisches Beispiel hierzu wäre die Berechnung von Extremalstellen einer Funktion $f : \mathbb{R}^n \rightarrow \mathbb{R}$. Nach HMI 3 benötigt man hierzu Nullstellen von $\text{grad}(f) : \mathbb{R}^n \rightarrow \mathbb{R}^n$, welche man nach obigem annähern kann.

2 Lineare Gleichungssysteme

Bei der numerischen Berechnungen ist immer mit kleineren Fehlern zu rechnen. Diese können unterschiedlichen Ursprungs sein:

- a) Der Algorithmus selber ist ungenau, z.B.:
 - Wir brechen irgendwann ab
 - Der Computer kann nur bis zu endlich vielen Nachkommastellen rechnen
- b) Die Eingabe an sich ist bereits fehlerhaft, z.B.
 - Durch vorherige Berechnungen (s. a))
 - Messfehler
 - ungenaue Eingabe z.B. durch Gleitzahlen

Wegen b) wollen wir, dass numerische Verfahren/Fragestellung kleine Störungen in der Eingabe nicht noch (sehr) verschlimmern.

Beispiel 2.1

Wir betrachten das LGS $\begin{pmatrix} N & 0 \\ 0 & 1/N \end{pmatrix} \cdot \underline{\mathbf{x}} = \underline{\mathbf{b}}$ für ein großes $N \in \mathbb{R}$. Hier sieht man leicht, dass kleine Änderungen in $\underline{\mathbf{x}}$ zu großen Veränderungen in $\underline{\mathbf{b}}$ führt und umgekehrt. Man sagt, dass das Problem ist schlecht gestellt/konditioniert ist.

Um hier die Fehlerverstärkung abschätzen zu können, verwenden wir den Begriff der Matrixnorm.

Definition 2.2 a) Eine Norm auf einem \mathbb{K} -Vektorraum V ($\mathbb{K} = \mathbb{C}$ oder \mathbb{R}) erfüllt folgendes für alle $v, w \in V, \lambda \in \mathbb{K}$:

- $\|v\| = 0 \iff v = 0$ (Definitheit)
- $\|\lambda v\| = |\lambda| \|v\|$ (Homogenität)
- $\|v + w\| \leq \|v\| + \|w\|$ (Dreiecksungleichung/Subadditivität)

Für Matrixnormen fordern wir hier noch zusätzlich (falls A, B quadratisch sind oder die "gleiche" Norm auf allen Vektorräumen $\mathbb{K}^{n \times m}$ betrachtet wird):

- $\|A \cdot B\| \leq \|A\| \cdot \|B\|$ (Submultiplikativität)

b) Zwei Normen $\|\cdot\|_1, \|\cdot\|_2$ heißen äquivalent, falls es Konstanten $c, C \in \mathbb{R}_{>0}$ gibt mit:

$$c \cdot \|v\|_1 \leq \|v\|_2 \leq C \|v\|_1.$$

- c) Eine Matrixnorm $\|\cdot\|_M$ auf $\mathbb{R}^{n \times m}$ heißt *verträglich* mit einer (bzw. zwei) Vektornorm $\|\cdot\|_V$ auf \mathbb{R}^n und \mathbb{R}^m , falls für alle $x \in \mathbb{R}^m$ gilt:

$$\|A \cdot x\|_V \leq \|A\|_M \cdot \|x\|_V.$$

- d) Eine Norm $\|\cdot\|_V$ auf \mathbb{R}^n induziert eine Matrixnorm $\|\cdot\|_{op}$ auf $\mathbb{R}^{n \times n}$ (und genauso auch auf $\mathbb{R}^{n \times m}$) via:

$$\|A\|_{op} := \sup_{v \in \mathbb{R}^n \setminus \{0\}} \frac{\|A \cdot v\|_V}{\|v\|_V} = \max_{\|v\|_V=1} \|A \cdot v\|_V.$$

Wenn die Norm klar ist, lassen wir in diesem Fall die Indizes bei den Normen weg.

Bemerkung 2.3

- a) Auf endlich-dimensionalen Vektorräumen sind alle Normen äquivalent.
 b) Induzierte Matrixnormen sind submultiplikativ und verträglich mit der induzierten Norm.

Beispiel 2.4 a) Für die Vektornorm $\|v\|_1 := \sum_{i=1}^n |v_i|$ ist die induzierte Matrixnorm:

$$\begin{aligned} \|A\|_1 &= \max_{\|v\|_1=1} \|Av\|_1 = \max_{\|v\|_1=1} \sum_{i=1}^n \left| \sum_{j=1}^n a_{i,j} v_j \right| \\ &\leq \max_{\sum |v_j|=1} \underbrace{\sum_{j=1}^n |v_j|}_{=1} \cdot \left(\sum_{i=1}^n |a_{i,j}| \right) \\ &= \max_{j=1, \dots, n} \sum_{i=1}^n |a_{i,j}| \quad (= \text{größte Spaltensumme}) \end{aligned}$$

Man sieht leicht, dass dieses Maximum tatsächlich für ein $v = e_j$ angenommen wird.

- b) Ähnlich erhalten wir für die Maximumsnorm $\|v\|_\infty := \max_{i=1, \dots, n} |v_i|$ die induzierte Matrixnorm:

$$\|A\|_\infty = \max_{\|v\|_\infty=1} \|Av\|_\infty = \max_{i=1, \dots, n} \sum_{j=1}^n |a_{i,j}| \quad (= \text{Zeilensummennorm})$$

c) Für die euklidische Norm $\|v\|_2 := \sqrt{\sum_{i=1}^n |v_i|^2}$ erhalten wir:

$$\begin{aligned} \|A\|_2 &= \max \|A \cdot v\|_2 = \max \sqrt{\langle Av, Av \rangle} \\ &= \max \sqrt{v^T A^T A v} = \max_{\lambda \in \text{Spec}(A^T A)} \sqrt{\lambda} \end{aligned}$$

wobei wir bei der letzten Gleichung den Spektralsatz verwendet haben, d.h. $A^T A$ ist mit einer Orthogonalbasis diagonalisierbar (und hat tatsächlich nur nicht-negative Eigenwerte).

Satz 2.5

Wir betrachten das LGS $A \cdot x = b$ bzw. das gestörte LGS $A \cdot \tilde{x} = \tilde{b}$ für $A \in \text{GL}_n(\mathbb{K}), b \in \mathbb{K}^n$. Wir haben also als Störung $\varepsilon := \tilde{b} - b \in \mathbb{K}^n$ und erhalten damit den Fehler $\Delta x = \tilde{x} - x = A^{-1}\varepsilon$. Seien weiterhin $\|\cdot\|$ zueinander verträgliche Matrix-/Vektornormen. Dann erhalten wir als Abschätzung für den relativen Fehler:

$$\frac{\|\Delta x\|}{\|x\|} \leq \kappa(A) \cdot \frac{\|\varepsilon\|}{\|b\|}$$

wobei $\kappa(A) := \|A\| \cdot \|A^{-1}\|$ die *Konditionszahl* von A heißt.

Die Konditionszahl von A gibt also an, inwieweit sich relative Fehler in b (im schlechtesten Fall) bei der Lösung des LGS vermehren.

Beispiel 2.6

In Bsp. 2.1 wäre für $b = \begin{pmatrix} 1 \\ 0 \end{pmatrix}, \varepsilon := \begin{pmatrix} 0 \\ \varepsilon \end{pmatrix}$ bereits $x = \begin{pmatrix} 1/N \\ 0 \end{pmatrix}$ und $\Delta x = \begin{pmatrix} 0 \\ N\varepsilon \end{pmatrix}$. Für alle Normen aus 2.4 berechnet sich leicht $\kappa(A) = N^2$ und als relativer Fehler:

$$\frac{\|\Delta x\|}{\|x\|} = N^2 \varepsilon = \kappa(A) \frac{\|\varepsilon\|}{\|b\|}.$$

Bemerkung 2.7

- Eine ähnliche Abschätzung erhält man, wenn auch die Matrix A kleine Fehler enthält.
- Wegen der Submultiplikativität $1 = \|I\| = \|A \cdot A^{-1}\| \leq \kappa(A)$ kann die Konditionszahl einer invertierbaren Matrix nie kleiner als 1 sein.

Proposition 2.8

Sei $\|\cdot\|$ eine Norm auf \mathbb{R}^n bzw. \mathbb{R}^m bzw. ihre induzierte Matrixnorm.

- a) Eine affine Abbildung $\phi : \mathbb{R}^n \rightarrow \mathbb{R}^m$, $v \mapsto A \cdot v + w$ ist Lipschitz-stetig mit Lipschitz-Konstante $\|A\|$.
- b) Sei $f : \mathbb{R}^n \rightarrow \mathbb{R}^m$ eine stetig differenzierbare Funktion mit $\|J_f(\underline{\mathbf{x}})\| \leq L$ für alle $\underline{\mathbf{x}} \in \mathbb{R}^n$. Dann ist f L -Lipschitz-stetig bezüglich $\|\cdot\|$, d.h. es gilt

$$\|f(\underline{\mathbf{v}}) - f(\underline{\mathbf{w}})\| \leq L \cdot \|\underline{\mathbf{v}} - \underline{\mathbf{w}}\|.$$

Wir wollen nun den Banachschen Fixpunktsatz 1.10 verwenden, um ein LGS $Ax = b$ schnell näherungsweise zu lösen. Ist B eine weitere leicht invertierbare Matrix, z.B. in Diagonalgestalt, dann ist

$$Ax = b \iff (A - B + B)x = b \iff Bx = (B - A)x + b \iff x = (1 - B^{-1}A)x + B^{-1}b$$

Für $S := (1 - B^{-1}A)$ ist also nach Prop. 2.8 $\phi : x \mapsto Sx + B^{-1}b$ genau dann eine Kontraktion, wenn $\|S\| < 1$ gilt und wir erhalten wie in Satz 1.10 einen Fixpunkt und eine Lösung des LGS.

Die einfachste Fall dies nachzuprüfen ist mit der folgenden Definition:

Definition 2.9

Eine Matrix $A \in \mathbb{K}^{n \times n}$ heißt *strikt diagonaldominant*, wenn in jeder Zeile der Betrag des Diagonaleintrags größer ist als die Summe der Beträge der restlichen Einträge, d.h. es gilt für alle $i \in \{1, \dots, n\}$:

$$|a_{i,i}| > \sum_{j=1, j \neq i}^n |a_{i,j}|$$

Bemerkung 2.10

Eine strikt diagonaldominante Matrix ist invertierbar.

Beweisidee: Betrachte für ein $x \in \text{Kern}(A)$ den betragsgrößten Eintrag x_i und zeige $(A \cdot x)_i > 0$ falls $x_i \neq 0$.

Satz 2.11 (Jacobi-Verfahren)

Ist $A \in \mathbb{K}^{n \times n}$ strikt diagonaldominant, dann konvergiert das Iterations-Verfahren:

$$\phi(\underline{\mathbf{x}}^{(n+1)}) := (I - D^{-1}A) \cdot \underline{\mathbf{x}}^{(n)} + D^{-1} \cdot b$$

für alle Startwerte $\underline{\mathbf{x}}^{(0)} \in \mathbb{K}^n$ gegen eine Lösung des LGS $A \cdot x = b$. Hierbei bezeichnet $D := \text{diag}(a_{1,1}, \dots, a_{n,n})$ die Diagonalmatrix mit den Diagonaleinträgen von A .

Beispiel 2.12

Die Matrix $A := \begin{pmatrix} 3 & 2 & 0 \\ 1 & 5 & 2 \\ 4 & -1 & 7 \end{pmatrix}$ ist wegen:

$$3 > 2 + 0$$

$$5 > 1 + 2$$

$$7 > 4 + 1$$

strikt diagonaldominant. Für $D = \begin{pmatrix} 3 & 0 & 0 \\ 0 & 5 & 0 \\ 0 & 0 & 7 \end{pmatrix}$ sieht man $D^{-1} = \begin{pmatrix} 1/3 & 0 & 0 \\ 0 & 1/5 & 0 \\ 0 & 0 & 1/7 \end{pmatrix}$

und damit erhalten wir für das LGS $Ax = b$ mit $b = \begin{pmatrix} 1 \\ 2 \\ 3 \end{pmatrix}$ das Iterationsverfahren:

$$\underline{\mathbf{x}}^{(n+1)} = (I_3 - D^{-1}A)\underline{\mathbf{x}}^{(n)} + D^{-1}b = \begin{pmatrix} 0 & 2/3 & 0 \\ 1/5 & 0 & 2/5 \\ 4/7 & -1/7 & 0 \end{pmatrix} \underline{\mathbf{x}}^{(n)} + \begin{pmatrix} 1/3 \\ 2/5 \\ 3/7 \end{pmatrix}$$

Eine schnelle Probe am Computer liefert uns, dass dies tatsächlich z.B. für den Startwert $\underline{\mathbf{x}}^{(0)} = \underline{\mathbf{0}}$ gegen die Lösung des LGS konvergiert.

Beim klassischen Lösen eines LGS bringen wir durch Gaußen (in der HMI 1/LA 1) die Matrix A bzw. die erweiterte Koeffizientenmatrix $(A \mid b)$ auf obere Dreiecksgestalt. Die Gaußschritte selbst, können wir als Elementarmatrizen beschreiben, wir erhalten also $G \cdot A = R$ wobei G die entsprechenden Gaußschritte $G = G_k \cdot \dots \cdot G_1$ darstellt und R eine obere Dreiecksmatrix ist.

Bis auf eventuelle Zeilenvertauschungen sind die einzelnen G_i stets untere Dreiecksmatrizen (wir addieren/subtrahieren nur oberhalb liegende Zeilen). Führen wir alle evtl. auftretenden Zeilenvertauschungen zu Beginn durch, können wir also $G = L^{-1}P$ für eine Vertauschungsmatrix P und eine untere Dreiecksmatrix L . Wir erhalten also:

$$P \cdot A = L \cdot R$$

Für eine Vertauschungsmatrix P , eine untere Dreiecksmatrix L und eine obere Dreiecksmatrix R . Diese heißt *LR-Zerlegung von A*. Für eine gegebene *LR-Zerlegen* können wir also statt dem LGS $Ax = b$, die beiden viel einfacheren Probleme $LRx = Pb \iff Ly = Pb$ und $Rx = y$ lösen.

Beispiel 2.13

Für $A := \begin{pmatrix} 1 & 2 \\ 3 & 4 \end{pmatrix}$ erhalten wir

$$\begin{pmatrix} 1 & 2 \\ 3 & 4 \end{pmatrix} \begin{array}{l} \leftarrow_{-3} \\ \leftarrow_{+} \end{array} \hat{=} \underbrace{\begin{pmatrix} 1 & 0 \\ -3 & 1 \end{pmatrix}}_{=:G_1} \cdot A = \underbrace{\begin{pmatrix} 1 & 2 \\ 0 & -2 \end{pmatrix}}_{=:R}$$

$$\implies A = G_1^{-1}R = LR \quad \text{mit } L = \begin{pmatrix} 1 & 0 \\ 3 & 1 \end{pmatrix}$$

Das LGS $Ax = \begin{pmatrix} b_1 \\ b_2 \end{pmatrix}$ ergibt also die beiden Teilprobleme:

$$L \cdot \begin{pmatrix} y_1 \\ y_2 \end{pmatrix} = \begin{pmatrix} b_1 \\ b_2 \end{pmatrix} \Rightarrow y_1 = b_1 \text{ und } 3y_1 + y_2 = b_2 \iff y_2 = b_2 - 3y_1$$

$$R \cdot \begin{pmatrix} x_1 \\ x_2 \end{pmatrix} = \begin{pmatrix} y_1 \\ y_2 \end{pmatrix} \Rightarrow -2x_2 = y_2 \text{ und } x_1 + 2x_2 = y_1 \iff x_1 = y_1 - 2x_2$$