Numerische Mathematik I

Prof. Dr. Christof Büskens

AG Optimierung & Optimale Steuerung
Zentrum für Technomathematik
Universität Bremen
28334 Bremen, Germany

Vorlesungsskript
Sommersemester 2004

(Unkorrigierte Fassung)

Vorwort

Die vorliegende Ausarbeitung entstand während meiner Tätigkeit am Zentrum für Technomathematik der Universität Bremen. Sie entstand im Rahmen einer Vorlesung, die ich im Sommersemster 2004 gehalten habe. An dieser Stelle möchte ich mich bei allen Teilnehmerinnen und Teilnehmern für ihr reges Interesse und ihre aktive Mitarbeit bedanken.

Bremen, Juli 2004 Christof Büskens

Inhaltsverzeichnis

Kapitel 1

Einleitung

1.1 Einführung

Gegenstand der numerischen Mathematik (oder einfach *Numerik*) oder auch praktischen Mathematik ist die näherungsweise Lösung mathematischer Probleme durch Zahlenwerte. Die Lösungsberechnung erfolgt dabei durch einen *Algorithmus*, d.h. durch eine Folge von elementaren Anweisungen und Rechenoperationen, die sich auf einem Computer ausführen lassen. Ein solcher Algorithmus stützt sich oft auf Ergebnisse der reinen Mathematik und reflektiert mathematische Eigenschaften des Problems. Die zu behandelnden Probleme stammen oft aus den Ingenieur– und Naturwissenschaften.

Beispiel 1.1. *Als ein erstes praktisches Beispiel sei der Landeanflug eines Verkehrsflugzeuges bei Scherwinden benannt, bei dem es zu 2–3 Unfällen pro Jahr kommt (bereits* > 500 *Tote), vgl. Abbildung 1.1.*

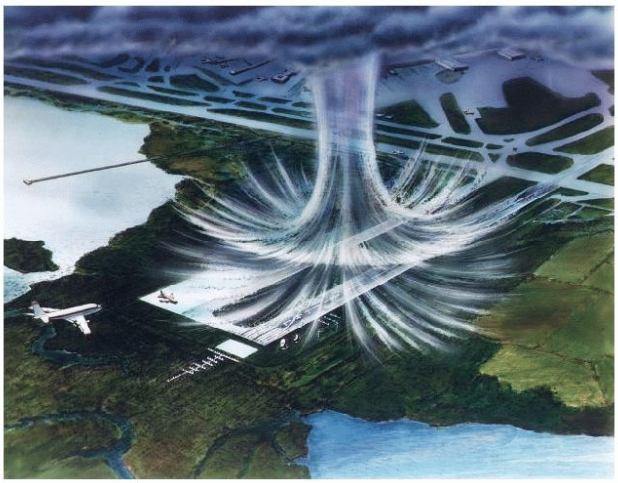


Abbildung 1.1: Scherwinde beim Landeanflug.

Aufgrund der Fallwinde wäre eine sichere Vorgehensweise, den Landeanflug abzubrechen, was aber ist hierzu die sicherste Vorgehensweise? Ein sehr sicherer Weg ist die während des Durchfluges durch den Scherwind angenommene minmale Höhe zu maximieren, vgl. Abbildung 1.2; wie aber kann das erreicht werden?

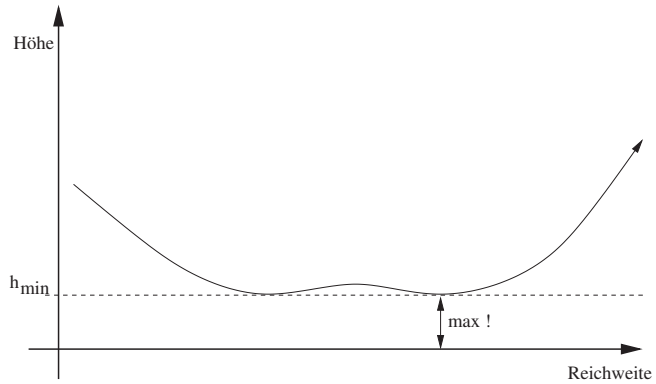


Abbildung 1.2: Maximierung der minimalen Höhe.

Da die physikalischen Vorgänge sehr gut bekannt sind kann zunächst ein sehr realitätsnahes mathematisches Modell erstellt werden.
Die Mathematik kommt dann intensiv bei der Lösung des Problems zur Anwendung. Hierzu muß zunächst eine theoretische Aufarbeitung der zu verwendenden Lösungsmechanismen vorgenommen werden, bzw. neu entwickelt werden. Für unser Beispiel greifen wir auf die sogenannte Variationsrechnung *bzw.* Optimale Steuerung *zurück. Die hierzu angebotenen Lösungsmethoden sind jedoch nicht mehr analytisch auf unser Flugmodell anwendbar und wir werden eine numerische Lösung auf einem Computer bemühen müssen. Zur Anwendung kommen numerische Verfahren für Differentialgleichungen oder lineare und nichtlineare Gleichungssystemlöser.*

Ziel der Veranstaltung ist die Einführung in verschiedenen Gebiete der numerischen Mathematik, wie z.B.:

- Lineare Gleichungssysteme,
- Interpolation,
- numerische Integration,
- nichtlineare Gleichungssysteme,
- Numerik der Differentialgleichungen.

Klassischerweise werden in den Vorlesungen Numerische Mathematik 1 und Numerische Mathematik 2 einfache Vorkenntnisse vermittelt, während in den Vorlesungen Numerische Mathematik 3, 4 (die häufig auch anders genannt werden) Spezialisierungen, Vertiefungen und Erweiterungen behandelt werden. Wichtig für alle Vorlesungen zur Numerik sind immer hinreichende Programmierkenntnisse!

1.2 Literatur

In der Numerik gibt es eine Fülle ausgezeichneter Bücher, die die verschiedenen angesprochenen Thematiken umfangreich beleuchten und darüberhinaus ergänzenden Stoff vermitteln. Stellvertretend für andere seien die nachfolgenden Bücher erwähnt:

- Deuflhard/Hohmann: Numerische Mathematik I, Verlag Walter de Gruyter
- Hämmerlin/Hoffmann: Numerische Mathematik, Springer Verlag
- Schwarz: Numerische Mathematik, Teubner Verlag
- Stoer: Numerische Mathematik I, Springer Verlag
- Stoer, Bulirsch: Numerische Mathematik II, Springer Verlag
- Werner: Numerische Mathematik, Vieweg Verlag
- u.v.a.

Es sei erneut erwähnt, dass es sich bei allen Büchern um ausgezeichnete Zusammenstellungen zur Numerik handelt. Das hier zusammengestellte Skript orientiert sich an mehrerern Büchern und es ist daher keines im besonderen Maße hervorzuheben.

1.3 Ein kurzer geschichtlicher Rückblick

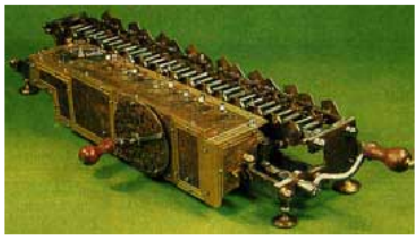
Ausgangspunkt für numerische Fragestellungen war eine Belebung der Mathematik durch konkrete Fragestellungen aus den Anwendungen. Nicht nur die Existenz, sondern auch die Bestimmung der Lösung, z.B. wie bei der Vorhersage von Himmelserscheinungen, traten in das Zentrum mathematischer Fragestellungen.

Ein aus historischer Sicht vorläufiger Höhepunkt der Numerik im weitesten Sinne wurde von Leonhard Euler (1707/Basel–1781/Petersburg) geschaffen. Euler

untersuchte günstige Verteilungen von Masten auf Segelschiffen. Für diese Arbeiten erhielt er den Preis der Pariser Akademie der Wissenschaften im Alter von nur 20 Jahren; und dies bevor er je den Ozean sah.

In diese Epoche fällt auch der erste Entwurf einer Rechenmaschine (1672), von Gottfried Wilhelm Leibniz, die er bereits ein Jahr später der Royal Society in London vorführte und die alle vier Grundrechenarten bewältigte.

Gottfried Wilhelm Leibniz:
Erfinder: Differential– und Integralrechnung
Infinitesimalrechnung, Rechenmaschine (1672)



Genie: Psychologe, Ingenieur, Mediziner
Nachlass: 75.000 Seiten

Voltaire: *Leibniz hat Könige gelehrt, Weise erleuchtet, weiser als sie, wusste er um den Zweifel.*

Abbildung 1.3: Gottfried Wilhelm Leibniz und die erste Rechenmaschine.

Die Zeit für die Numerik war jedoch noch nicht reif und kritisch betrachtet, könnte man sagen, dass die Numerik über viele Jahrzehnte hinweg nicht der Durchbruch gelang. Die angewendeten Beweistechniken waren bis ca. 1900 motiviert durch die praktischen Anwendungen/Lösungen meist konstruktiv, doch aus numerischer Sicht nicht brauchbar. Als Folge ist daher (nicht überraschend) zu verzeichnen, dass einer rein logischen Vorgehensweise der Vorzug gegeben wurde. Der berühmte Mathematiker Jacobi äußerte sogar:

'Die Mathematik dient einzig und alleine der Ehre des menschlichen Geistes.'

Heute wissen wir, dass diese Aussage nicht richtig ist!

Der eigentliche Aufstieg/Durchbruch der numerischen Mathematik gelang dann mit dem Aufkommen (moderner) Rechenanlagen. Während der Anstieg der Rechengeschwindigkeit bis ca. 1940 um lediglich den Faktor 10 (durch Tricks) gelang, liegt er heute bei 10^{15} (Stand 2004) oder höher.

1.4 Was ist Numerik?

Wir erinnern an das Scherwindbeispiel und stellen einige Dinge fest:
Die Anwendung mathematischer Lösungsmethoden auf realistische Aufgabenstellungen der Praxis erfordert fast immer den Einsatz eines Rechners. Die Anforderungen an die Numerik sind dabei vielschichtig:

- Entwicklung von Verfahren zur Konstruktion von Lösungen, meist *Näherungslösungen* mathematischer Aufgabenstellungen
- Effiziente Implementierung auf Rechenanlagen
- Auswahl geeigneter Verfahren
- Aussagen über Güte der Approximation

In diesem Zusammenhang ist die Kette

Problemstellung $\longrightarrow$ Physikalisches Modell $\longrightarrow$ Mathematisches Modell $\longrightarrow$ Mathematische/numerische Lösung $\longrightarrow$ Diskussion der Ergebnisse

wichtig, die i.A. mehrfach durchlaufen werden muß.

Aus praktischer Sicht hat man hierbei insbesondere das schwierige Problem, dass mathematische Modell möglichst gut an die Realität anzupassen:

$$\text{Modell} \approx \text{Realität}$$

In dieser Vorlesung werden wir uns genau mit den oben genannten Punkten beschäftigen.

1.5 Motivationsbeispiel

Wir wollen nachfolgend ein konkretes Problem betrachten und gleichzeitig auf eine besondere Problematik aufmerksam machen.

Beispiel 1.2. *Es soll das Integral*

$$I_n = \int_0^1 \frac{x^n}{x+5}\, dx, \qquad n \in \mathbb{N} \cup \{0\} = \mathbb{N}_0$$

für $i = 0, 1, 2, \ldots, 20$ berechnet werden. Wir stellen fest:

- *Elementare Integration versagt*
  (dennoch: analytische Methoden stets zuerst versuchen!)

- *numerische Quadraturverfahren nicht geeignet, da zu aufwendig für das spezielle Problem (⟶ Auswahl geeigneter Verfahren)*
- *Lösung: Kombination von analytischer Vorarbeit und numerischer Durchführung*

Für die Zahlen I_n kann schnell eine Rekursionsvorschrift angegeben werden:

$$I_0 = \int_0^1 \frac{dx}{x+5} = [ln|x+5|]_0^1 = \ln\frac{6}{5} \approx 0.182321556...$$

$$I_1 = \int_0^1 \frac{x}{x+5}dx = \int_0^1 \left(\frac{x+5}{x+5} - \frac{5}{x+5}\right) dx = 1 - 5\ I_0$$

Allgemeiner erhalten wir für $n \to n+1$:

$$I_{n+1} = \int_0^1 \frac{x^n}{x+5}(x+5-5)dx = \int_0^1 x^n\ dx - 5\int_0^1 \frac{x^n}{x+5}dx$$

$$I_{n+1} = \frac{1}{n+1} - 5I_n \tag{1.1}$$

Ausgehend von $I_0 = \ln\frac{6}{5}$ können wir somit theoretisch *alle Werte I_n berechnen. In der rechentechnischen Realisierung erhalten wir jedoch bereits nach wenigen Schritten unbrauchbare Ergebnisse. Mit einem Taschenrechner ergibt sich dann etwa (taschenrechnerspezifisch) folgendes Bild:*

I_1	=	0.08839...	I_{11}	=	0.01377...	
I_2	=	0.05803...	I_{12}	=	0.01445...	*(Widerspruch zur Monotonie)*
⋮			⋮			
I_{10}	=	0.01542...	I_{14}	=	0.04814...	
			I_{15}	=	−0.17404...	*(Widerspruch zum Vorzeichen)*

Wir wollen die Güte der berechneten Lösung etwas genauer analysieren und stellen fest:

1. *Für $x \in [0,1] : \frac{x^n}{x+5} \geq 0 \Longrightarrow \forall n : I_n \geq 0.$*
2. *Für $x \in]0,1[: x^{n+1} < x^n \Longrightarrow \frac{x^{n+1}}{x+5} < \frac{x^n}{x+5} \Longrightarrow I_{n+1} < I_n.$*

Somit ist (I_n), $n \in \mathbb{N}_0$ streng monton fallend und wegen $I_n \geq 0$ nach unten beschränkt, also konvergent. Wegen

$$I_n < \int_0^1 x^{n-1}dx = \frac{1}{n}, \qquad \forall n \geq 1$$

folgt

$$\lim_{n\to\infty} I_n = 0,$$

was sich leider mit unseren numerischen Erfahrungen nicht deckt.

Wir wagen einen weiteren Versuch und gehen das Problem von hinten an (Rückwärtsrekursion):
Es ist

$$I_{10}+5I_9 = \frac{1}{10} \quad \Longrightarrow \quad I_{10} = \frac{1}{10} - 5I_9 < \frac{1}{10} - 5I_{10} \quad \Longrightarrow \quad I_{10} < \frac{1}{60}$$

$$I_9 = \frac{1}{50} - \frac{1}{5}I_{10} > \frac{1}{50} - \frac{1}{5}I_9 \quad \Longrightarrow \quad I_9 > \frac{1}{60}$$

Werten wir nun die Rückwärtsrekursion

$$I_{n-1} = \frac{1}{5n} - \frac{1}{5}I_n \tag{1.2}$$

aus, so erhalten wir:
$I_9 = \frac{1}{60}$
$I_8 = 0.01888...$
$\vdots$
$I_3 = 0.04313...$
$I_2 = 0.05803...$
$I_1 = 0.08839...$
$I_0 = 0.182321556...$ *alle Stellen richtig!*

Bei der Vorwärtsberechnung in (1.1) wird ein Fehler, den wir in I_n z.B. durch Rundung erhalten haben, mit dem Faktor 5 multipliziert und geht so verstärkt in I_{n+1} ein. In der Rückwärtsberechnung (1.2) hingegen, reduziert sich der Fehler um den Faktor $\frac{1}{5}$, so dass die Genauigkeit der Lösung mit jedem weiteren Schritt wächst.

Eine detailliertere Klärung der Situation werden wir später angeben.

1.6 Vorläufiges Fazit

Viele Probleme der Mathematik lassen sich nicht analytisch lösen (es gibt keine explizite Darstellung der Lösung) oder nur sehr schwer lösen (z.B. zu komplex), während eine Lösungsberechnung mit numerischen Verfahren jedoch häufig möglich ist.

Bei numerischen Verfahren können weitere Fehler auftreten, die bei der Analyse der berechneten Lösung zu beachten sind. Hierbei bedeutet Fehler nicht, daß

man etwas falsch gemacht hat. Vielmehr können unvermeidbare Abweichungen vom *exakten*, d.h. realitätsgenauen Ergebnis, auftreten.

Beispiel 1.3. *Wir kommen zurück auf unser Scherwindproblem, die Lösung des Problems kann nur noch* numerisch *berechnet werden, da eine analytische Lösung nicht existiert. Ihr Lösungsansatz führt auf ein Anfangswertproblem mit einer gewöhnlichen Differentialgleichung. Es können verschiedenen Fehler auftreten:*
Modellfehler: *Die Modellierung ist ungenau, z.B. sind die Windbedingungen nicht beliebig genau modellierbar.*
Datenfehler: *Parameter des DGL–Systems oder Anfangswerte sind nur ungenau angebbar.*
Verfahrensfehler: *Das numerische Verfahren zur Lösung der DGL berechnet nur eine genäherte Lösung.*
Rundungsfehler: *Der Computer kann nicht mit beliebig vielen Nachkommastellen rechnen (ein Computer hat nur endlich vielen Speicher), z.B. wird π in der Regel mit nur 16 oder 32 Nachkommastellen berücksichtigt.*

Verfahrensfehler und Rundungsfehler sind Fehler, die aufgrund des Lösungsansatzes durch Numerik auftreten, sie sind innerhalb dieser Vorlesung genauer zu analysieren.

Bemerkung 1.4. *Numerik ist somit nicht nur die Entwicklung von Algorithmen oder Verfahren. Auch die Analyse und die Effizienz der Verfahren sind wesentliche Bestandteile.*

Kapitel 2

Fehleranalyse

Wie bereits festgestellt, können bei der Anwendung mathematischer/numerischer Methoden Fehler z.B. bei der Modellbildung, bei den Eingabeparametern, bei der Approximation oder durch Rundung auftreten. Die beiden letztgenannten wollen wir genauer untersuchen. Hierzu ist es erforderlich die Struktur der Zahlendarstellung auf einem Computer zu untersuchen.

2.1 Maschinenzahlen

Die mit einer bestimmten Codierung darstellbaren Zahlen bezeichnen wir als Menge der *Maschinenzahlen*. Gebräuchlichste Codierungsform ist dabei die sogenannte Gleitpunkt-Darstellung: Die Zahl

$$x = V_M(d_1 p^{-1} + d_2 p^{-2} + \ldots + d_l p^{-l}) \cdot p^E \text{ mit } E = V_E(e_1 p^{n-1} + e_2 p^{n-2} + \ldots + e_{n-1} p + e_n)$$

wird codiert durch

$$\underbrace{V_M d_1 d_2 \ldots d_l}_{\text{Mantisse}} \,|\, \underbrace{V_E e_1 e_2 \ldots e_n}_{\text{Exponent}}$$

Dabei ist

$p \in \mathbb{N}$, $p > 1$, p fest		Basis
$d_\lambda \in \{0, 1, \ldots, p-1\}$,	$\lambda = 1, \ldots, l$	Ziffern der Mantisse
$e_v \in \{0, 1, \ldots, p-1\}$,	$v = 1, \ldots, n$	Ziffern des Exponenten
$V_M, V_E \in \{+, -\}$		Vorzeichen der Mantisse bzw. des Exponenten

Beispiel 2.1. $p = 10,\ l = 4,\ n = 3$

$$\begin{array}{lcll} x = 4711 & \rightarrow & +4711| + 004 & \rightarrow \ 0.4711 \cdot 10^4 \\ x = -17.5 & \rightarrow & -1750| + 002 & \rightarrow \ -0.1750 \cdot 10^2 \\ x = 0.008008 & \rightarrow & +8008| - 002 & \rightarrow \ 0.8008 \cdot 10^{-2} \end{array}$$

Bemerkung 2.2. *Fordert man $d_1 \neq 0$ für $x \neq 0$ (normalisierte Gleitpunkt-Darstellung), dann ist für jede Maschinenzahl $x \neq 0$ die Gleitpunktdarstellung eindeutig; lediglich beim Exponenten 0 bleibt V_E unbestimmt.*

Definition 2.3 (Gleitpunktzahl). *Sei*

$p > 1,\ p \in \mathbb{N}$	*Basis; z. B.:* $p = 2, 10, 16$
$D, E \in \mathbb{Z}$	*Mantisse, Exponent*
$l > 0.$	

Dann lautet die Menge der l-stelligen, normalisierten Gleitpunktzahlen zur Basis p:

$$\mathbb{G} = \{Dp^{E-l} | D = 0 \vee p^{l-1} \leq |D| < p^l\}$$

Beispiel 2.4. *4-stellige, normalisierte Zahlen zur Basis 10:*

$$\mathbb{G} = \{D \cdot 10^{E-4} | D = 0 \ \vee \ 10^3 \leq |D| < 10^4\}$$

Die übliche Schreibweise lautet:

$$Dp^{E-l} = \pm \underbrace{d_1 d_2 \ldots}_{|E| \text{ Stellen}} , \underbrace{\ldots d_{l-1} d_l}_{l-|E| \text{ Stellen}} \underbrace{0 \ldots 0}_{\substack{\text{vor Komma} \\ \text{nach Komma}}}$$

Bezeichnungen:
$\mathbb{M}(p, l, n)$ oder $\mathbb{M}$: Menge der Maschinenzahlen
Offensichtlich ist diese Menge endlich und somit gilt: $\mathbb{M} \overset{\subset}{\neq} \mathbb{R}$.

Bemerkung 2.5.

1. *Die Wahl der Basis p wird durch die Rechnerkonstruktion bestimmt.*

Dualsystem:	$p = 2$	*mit Ziffern*	$0, 1$
Hexadezimalsystem:	$p = 16$	*mit Ziffern*	$0, 1, \ldots, 9, A, \ldots, F$

2. *Der Exponent ist in der Praxis durch den Speicherplatz eingeschränkt.*

Beispiel 2.6. *Typische Situation auf PC's für PASCAL, FORTRAN: real*4:*
1 Real–Zahl benötigt 4 Byte Speicherplatz:
1 Byte für Vorzeichen, Exponent
3 Byte für Mantisse
Es stellt sich als nächstes die Frage, welche Zahlen sich überhaupt damit darstellen lassen (1 Byte = 8 Bit, 1 Bit ist entweder 0 oder 1).

i) *Stellenzahl:*
3 Byte = 24 Bit für die Mantisse, also 24-stellige Dualzahlen:

$$2^{24} = 10^l \Longrightarrow l = \log_{10} 2^{24} = 24 \log_{10} 2 \approx 24 \cdot 0.3010 \approx 7.2\ldots$$

d.h. 7-stellige Mantisse im 10er System

ii) *Exponentenbereich*
2 Bit für die beiden Vorzeichen, d.h. 6 Bit für den Exponenten: $\alpha \leq E \leq \beta$ *mit* $\beta = -\alpha = 63 = \underline{1} \cdot 2^5 + \underline{1} \cdot 2^4 + \underline{1} \cdot 2^3 + \underline{1} \cdot 2^2 + \underline{1} \cdot 2^1 + \cdot \underline{1} \cdot 2^0$

Allgemein erkennen wir, dass es E_1 und E_2 gibt mit

$$\mathbb{M} = \{g \in \mathbb{G} | E_1 \leq E \leq E_2\}$$

Offensichtlich gibt es eine kleinste positive Maschinenzahl x_{min} und eine größte Maschinenzahl x_{max}.

Beispiel 2.7. *Aufgrund der Fülle verschiedenener Parametersetzungen gab es 1983 einen Standardisierungsversuch (IEEE) für* $p = 2$*:*

einfache Genauigkeit (32 Bits):	$l = 23$*; 8 Bit für E*
doppelte Genauigkeit (64 Bits):	$l = 52$*; 11 Bit für E*
Register (80 Bits)	$l = 64$*; 15 Bit für E*

Bemerkung 2.8. *Integer–Zahlen werden in vergleichbarer Weise codiert.*

2.2 Maschinenzahlen auf der Zahlengerade

Wie festgestellt, existieren in $\mathbb{M}$ eine kleinste positive Maschinenzahl $x_{\min}$ und eine größte Maschinenzahl $x_{\max}$. Die Maschinenzahlen dazwischen sind jedoch nicht gleichmäßig verteilt:
Verteilung auf der Zahlengeraden:

a) Innerhalb jeden Intervalls $[p^{k-1}, p^k)$ liegen die Maschinenzahlen in gleichen Abständen:

$$a \cdot p^k \text{ mit } a = p^{-l} = .\underbrace{00\ldots01}_{l} \cdot p^0$$

hierbei bezeichnet a die Einheit der letzten Mantissenstelle beim Exponenten 0.

b) Die Maschinenzahlen sind nicht auf $\mathbb{R} \cap [x_{\min}, x_{\max}]$ gleichabständig verteilt. Der relative Abstand

$$\frac{x_{i+1} - x_i}{x_i}, \qquad x_i \neq 0$$

zweier aufeinanderfolgender Maschinenzahlen variiert höchstens um einen Faktor ρ.

Beispiel 2.9. $\mathbb{M}(2, 3, 2)$

Verfügbare positive Mantissen	*Verfügbare Exponenten*
+100 = 1/2	*00 = 0*
+101 = 5/8	*± 01 =± 1*
+110 = 3/4	*± 10 =± 2*
+111 = 7/8	*± 11 =± 3*

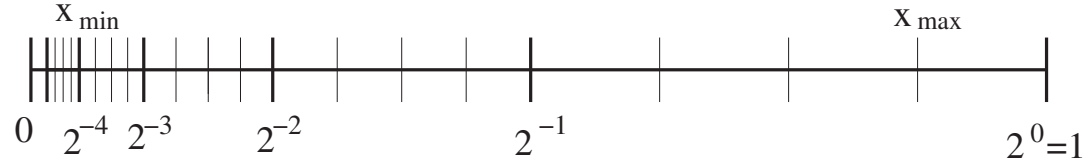


Abbildung 2.1: Beispiel zur Verteilung der Maschinenzahlen auf dem Computer.

2.3 Rundung

Definition 2.10 (Rundung). *Eine* korrekte Rundung *ist die Abbildung*

$$rd : \mathbb{R} \to \mathbb{M},$$

die jedem $r \in \mathbb{R}$ das nächstgelegene $x \in \mathbb{M}$ zuordnet:

$$|rd(r) - r| \leq |x - r| \qquad \forall x \in \mathbb{M}.$$

Bemerkung 2.11. *Die Definition ist bis auf Ausnahmen eindeutig; dort muß eine Zusatzbedingung zur eindeutigen Behandlung eines Umschlagpunktes $\bar{x}$ angegeben werden.*

Ausführung der korrekten Rundung:
Bei der technischen Realisierung der korrekten Rundung muß r nicht exakt bekannt sein, z.B. $r = \sqrt{2}, \pi$. Offenbar reicht es, die $(l+1)$ste Mantisse zu kennen:[1]

$$r = \pm.d_1 d_2 \ldots d_l d_{l+1} \ldots p^E \qquad \text{(normalisiert)}$$

$$r' := \begin{cases} .d_1 d_2 \ldots d_{l-1} d_l & \text{falls } 0 \leq d_{l+1} < p/2 \\ .d_1 d_2 \ldots d_{l-1}(d_l + 1) & \text{falls } p/2 \leq d_{l+1}, \\ \quad \overleftarrow{\quad\quad\quad} \\ \quad \text{(evtl. Übertrag)} \end{cases} \qquad \text{(Aufrunden in } \overline{x})$$

[1] Eine Realisierung der korrekten Rundung läßt sich jedoch erst mit zwei Schutzziffern realisieren.

$$rd(r) := \text{ sign}(r) \cdot r' \cdot p^E$$

Beispiel 2.12. $l = 4, p = 10$

$$rd(0.142842_{10^2}) = 0.1428_{10^2}$$
$$rd(0.14285_{10^0}) = 0.1429_{10^0}$$
$$rd(0.14997_{10^{-1}}) = 0.1500_{10^{-1}}$$

Schranken für den absoluten und den relativen Fehler:
Sei $r = \pm.d_1 d_2 \dots d_l d_{l+1} \dots p^E \qquad (d_1 \neq 0)$.

$$|rd(r) - r| \leq \frac{a}{2} p^E = \frac{1}{2} p^{E-l} \quad (a = p^{-l}) \qquad \text{absoluter Fehler} \tag{2.1}$$

$$\frac{|rd(r) - r|}{|r|} \leq \frac{p}{2} p^{-l} \qquad \text{relativer Fehler} \tag{2.2}$$

$$|r| \overset{\uparrow}{\geq} 0.1 p^E$$

Wir erhalten

Satz 2.13. *Für alle* $r \in \mathbb{R} \cap ([-x_{max}, -x_{min}] \cup [x_{min}, x_{max}])$ *gibt es ein* $\varepsilon \in \mathbb{R}$, $|\varepsilon| \leq eps$,

$$eps := \frac{p}{2} p^{-l}, \qquad \textit{(relative Maschinengenauigkeit)}, \tag{2.3}$$

so dass

$$rd(r) = r(1 + \varepsilon) \tag{2.4}$$

gilt.

Es gibt einige Besonderheiten zu berücksichtigen:

- Zwischenresultate können $\mathbb{M}$ verlassen: z. B. bei $c = \sqrt{a^2 + b^2}$
- Exponentenunterlauf: Für $r \approx 0,$ d.h. $r \in]-x_{\min}, x_{\min}[$ gilt:

  $$rd(r) := 0.$$

  Für $r \in]-x_{\min}, x_{\min}[\backslash \{0\}$ gilt: Der relative Fehler ist stets 1, während der absolute Fehler klein, nämlich $< x_{\min}$ ist. In diesem Fall sollte eine Warnung "underflow" gesetzt werden; eventuell können massive Probleme auftreten, z.B. bei $c = \frac{1}{r}$.

- Exponentenüberlauf: Für $|r| > x_{\max}$ gilt:

$$rd(r) := sign(r) \cdot x_{\max},$$

gleichzeitig sollte eine Warnung "exponential overflow" gesetzt werden; eventuell können auch hier massive Probleme auftreten, z.B. bei $c = r^2$, $r \to \infty$. Hier ist der relative Fehler durch 1 beschränkt, während der absolute Fehler beliebig groß werden kann.

Bemerkung 2.14. *Die Fehlerschranken (2.1) und (2.2) gelten nur solang kein Exponentenunter- bzw. überlauf vorkommt.*

Bemerkung 2.15. *Es gibt verschiedene Vereinbarungen an Umschlagpunkte, wie z.B Aufrunden, Abrunden oder Abschneiden. Stets muß die Rundungsvorschrift jedoch eine idempotente ($c \in \mathbb{M}$, $\Rightarrow rd(c) = c$) und monotone ($c_1 \leq c_2 \Rightarrow rd(c_1) \leq rd(c_2)$) Abbildung sein.*

Sondercodierungen: Für $0, +\infty, -\infty, \infty$, NAN (Not-A-Number;) hat man oft Sondercodierungen mit speziellen Rechenregeln, z. B.: $x = 0, \frac{1}{x} = NAN; \infty - \infty = NAN; \ldots$

Beispiel 2.16. *Wir wollen nachfolgend die Rekursionsformel aus Beispiel 1.2 genauer untersuchen:*

$$I_n + 5I_{n-1} = \frac{1}{n}\ ,\ I_0 = ln\frac{6}{5} \notin \mathbb{M}$$

Auswirkungen des Eingabefehlers *in* I_0 :

$$\hat{I}_0 = I_0 + \Delta I_0$$

$$\Rightarrow \quad I_n + \Delta I_n = \hat{I}_n = \frac{1}{n} - 5\hat{I}_{n-1} = \frac{1}{n} - 5(I_{n-1} + \Delta I_{n-1}) = I_n - 5\Delta I_{n-1}$$

$$\Rightarrow \quad \Delta I_n = -5\Delta I_{n-1} = (-5)^2 \Delta I_{n-2} = \ldots = (-5)^n \Delta I_0$$

Ist beispielsweise $\Delta I_0 = 10^{-10}$*, dann ist bereits* $\Delta I_{15} = (-5)^{15} 10^{-10} \approx -3.05\ldots$
Hierbei ist $(-5)^i$ *oszillierend und stark anwachsend.*
Umgekehrt tritt bei der Rückwärtsrekursion Fehlerdämpfung auf:

$$\Delta I_0 = \frac{1}{(-5)^n} \Delta I_n$$

Mit beispielsweise $\Delta I_{15} = 1$ *folgt*

$$\Rightarrow \Delta I_0 = -3.27\ldots \cdot 10^{-11}.$$

2.4 Gleitpunkt-Arithmetik

Sind $x, y \in \mathbb{M}$, so braucht $x \cdot y$ mit $\cdot \in \{+, -, \times, /\}$ nicht aus $\mathbb{M}$ zu sein.

Beispiel 2.17. $\mathbb{M}(10, 5, 2)$, *mit* $x = .25684_{10^1}$, *und* $y = .32791_{10^{-2}}$*:*

$$\left.\begin{array}{rcl} x + y & = & .25716791_{10^1} \\ x \times y & = & .842204044_{10^{-2}} \\ x/y & = & .7832637004\ldots_{10^3} \end{array}\right\} \notin \mathbb{M}$$

Gleitpunkt-Operationen $\oplus, \ominus, \otimes, \oslash$:
Die korrekte Rundung lautet

$$x \odot y := rd(x \cdot y), \cdot \in \{+, -, \times, /\},$$

wobei wegen (2.4) gilt:

$$\begin{aligned} x \oplus y &= (x + y)(1 + \alpha) \\ x \ominus y &= (x - y)(1 + \beta) \\ x \otimes y &= (x \times y)(1 + \gamma) \\ x \oslash y &= (x/y)(1 + \delta) \end{aligned}$$

mit $|\alpha|, |\beta|, |\gamma|, |\delta| \leq eps$ (Maschinengenauigkeit).

Bemerkung 2.18. *Es ist anzumerken, dass* α*,* β*,* γ*,* δ *von* x *und* y *abhängen, nicht jedoch von ihrer Schranke (eps ist a priori bekannt).* α*,* β*,* γ*,* δ *sind die relativen Fehler der Gleitpunkt-Operationen.*

Bemerkung 2.19. *Nur die Kommutativität der Addition und der Multiplikation bleiben auch bei der Gleitpunkt-Arithmetik erhalten. Assoziativ- und Distributivgesetze gelten nicht mehr!*

Beispiel 2.20. $\mathbb{M}(10, 8, 2)$, $a = .23371258_{10^{-4}}$, $b = .33678429_{10^2}$, $c = -.33677811_{10^2}$.

$$\begin{aligned} a \oplus (b \oplus c) = a \oplus .61800000_{10^{-3}} &= .64137126_{10^{-3}} \\ (a \oplus b) \oplus c = .33678452_{10^2} \oplus c &= .64100000_{10^{-3}} \\ a + b + c = &= .641371258_{10^{-3}} \end{aligned}$$

Bemerkung 2.21. *Die korrekt rundende Gleitpunkt-Arithmetik kann technisch realisiert werden, wenn das Rechenwerk über mindestens zwei Stellen mehr verfügt als die Mantissenlänge* l*. Es gibt jedoch nur wenige Anlagen, die korrekt runden!*

2.5 Fehlerfortpflanzung, Kondition

Wir hatten festgestellt, dass bei der Berechnung mit dem Computer verschiedene Fehler auftreten können. Betrachten wir für $D \subset \mathbb{R}^n$ und $f : D \to \mathbb{R}^m$ das Problem der Berechnung

$$y = f(x), \quad x \in D,$$

so bezeichnen wir mit x die *Eingabedaten*, während y *Ausgabe–* oder *Resultatdaten* genannt werden. Die Genauigkeit der Berechnung von $y = f(x)$ wird durch die *Fehlertypen*

- Fehler in den Eingabedaten,
- Abbrechfehler oder Diskretsierungsfehler,
- Rundungsfehler während der Rechnungen.

begrenzt. Anstelle einer exakten Rechnung $y = f(x)$ wird man daher eine Approximation $\tilde{f}(\tilde{x})$, mit

$\tilde{x}$: Approximation für x, z.B. durch $\tilde{x} = rd(x)$
$\tilde{f}$: Approximation für f
berechnen, vgl. Abbildung 2.2.

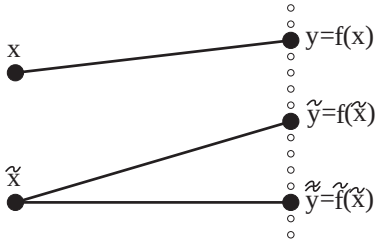


Abbildung 2.2: Fehlereinflüsse.

Wir befassen uns zunächst damit, wie sich Fehler in x auf das Ergebnis $y = f(x)$ auswirken. Sei $\tilde{x}$ eine Näherung von x und sei

$\triangle x = \tilde{x} - x$: der *absolute Fehler*,
$\frac{\tilde{x}_i - x_i}{x_i}$: der *relative Fehler*, $(i = 1, \ldots, n)$,
$\tilde{y} = f(\tilde{x})$: der Näherungswert für $y = f(x)$.

Die Funktion f sei eine C^1–Funktion. Die Taylor–Entwicklung erster Ordnung liefert für den absoluten Fehler $\triangle y = \tilde{y} - y$ die Approximation

$$(2.5) \quad \triangle y_i = f_i(x + \triangle x) - f_i(x) \approx \frac{\partial f_i(x)}{\partial x} \triangle x = \sum_{j=1}^{n} \frac{\partial f_i(x)}{\partial x_j} \triangle x_j, \quad i = 1, \ldots, m.$$

Für den relativen Fehler erhält man dann

$$\frac{\triangle y_i}{y_i} \approx \sum_{j=1}^{n} \left(\frac{\partial f_i(x)}{\partial x_j} \frac{x_j}{f_i(x)} \right) \cdot \left(\frac{\triangle x_j}{x_j} \right), \quad x_j, y_i \neq 0. \tag{2.6}$$

Definition 2.22 (Kondition).

1. *Die Zahlen*

$$k_{ij}(x) = \left| \frac{\partial f_i(x)}{\partial x_j} \frac{x_j}{f_i(x)} \right| \tag{2.7}$$

   heißen Verstärkungsfaktoren *bzw. (relative)* Konditionszahlen.

2. *Das Problem 'Berechne $y = f(x)$' heißt* gut konditioniert*, falls alle $k_{ij}(x)$ die Größenordnung 1 haben. Andernfalls heißt das Problem* schlecht konditioniert.

Zuerst untersuchen wir damit die *arithmetischen Operationen* $+, -, *, /$.

Multiplikation: $y = f(x_1, x_2) = x_1 * x_2$
Es gilt $k_{11}(x) = k_{12}(x) = 1$: gutartig.

Division: $y = f(x_1, x_2) = x_1/x_2$
Es gilt $k_{11}(x) = k_{12}(x) = 1$: gutartig.

Addition, Subtraktion: $y = f(x_1, x_2) = x_1 + x_2$
Es gilt

$$k_{11}(x) = \left| \frac{x_1}{x_1 + x_2} \right|, \quad k_{12}(x) = \left| \frac{x_2}{x_1 + x_2} \right|$$

Das Problem ist schlecht konditioniert, falls $x_1 \approx -x_2$. Daher ist die Subtraktion nahezu gleichgroßer Zahlen mit gleichen Vorzeichen schlecht konditioniert.
Dieses Phänomen heißt *Auslöschung.*

Beispiel 2.23.

$$1.31 - 1.25 = 0.06$$
$$1.32 - 1.24 = 0.08 \qquad \textit{(Störrechnung)}$$

Es gilt:

$$\begin{aligned} x &= (1.31, -1.25) \\ y &= x_1 + x_2 = 0.06 \\ \triangle x &= (0.01, 0.01) \\ \left| \frac{\triangle x_i}{x_i} \right| &\leq 0.008, \quad \textit{d.h. relativer Eingabefehler ca. } 0.8\% \\ k_{1,i}(x) &\leq 22, \quad i = 1, 2 \end{aligned}$$

Der relative Fehler im Ergebnis ist ca. 40 mal (Summe) größer als der relative Fehler in den Daten.

Wurzel: $y = f(x_1) = \sqrt{x_1}$, $x_1 > 0$
Es gilt $k(x) = 1/2$: gutartig. **(Übung)**

Bemerkung 2.24. *Bei einigen Problemen kann die Auslöschung durch geeignete Umformulierung vermieden werden, vgl. die nachfolgenden Beispiele.*

2.6 Algorithmen

Ein *Algorithmus* zur Berechnung der Lösung $y = f(x)$ eines Problems ist eine Sequenz von endlich vielen 'elementaren Operationen' $(+, -, *, /, cos(x), \sqrt{x}, \ldots)$. Es gibt i.A. mehrere Anordnungen der Rechenschritte, welche zum gleichen Ergebnis $y = f(x)$ führen. In jedem Rechenschritt fallen Rundungsfehler an. Dabei kann der Fall auftreten, daß bei der Lösung eines an sich gut konditionierten Problems eine ungünstige Anordnung der Rechenschritte zum Aufschaukeln der Rundungsfehler führt. Der zugehörige Algorithmus ist numersich instabil.

Beispiel 2.25. *Gesucht ist die betragskleinere Lösung von* $x^2 + 2px - q = 0$ *mit* $p >> q$.
Die exakte Lösung ist gegeben durch

$$y = f(p, q) = -p + \sqrt{p^2 + q} \tag{2.8}$$

Für die Konditionszahlen gilt: **(Übung)**

$$k_p(p, q) < 1, \qquad k_q(p, q) < 1, \quad \textit{(wegen } p >> q\textit{)} \tag{2.9}$$

also ist die Aufgabe gut konditioniert für $q > 0$.
(Wäre jedoch etwa $q \approx -p^2$*, wäre das Problem schlecht konditioniert.)*

Beispiel 2.26. *Wir betrachten erneut die Aufgabenstellung aus Beispiel 2.25 und untersuchen zwei Algorithmen:*
Algorithmus 1: $y = f(p, q) = -p + \sqrt{p^2 + q}$

$$\begin{aligned} s &:= p * p \\ t &:= s + q \\ u &:= \sqrt{t} \\ y &:= -p + u \end{aligned}$$

Für $p >> q$ tritt in $y := -p + u$ Auslöschung auf, der Algorithmus ist in diesem Fall schlecht konditioniert, obgleich das Problem gut konditioniert ist.
Algorithmus 2: $y = f(p,q) = \frac{q}{p+\sqrt{p^2+q}}$

$$\begin{aligned} s &:= p * p \\ t &:= s + q \\ u &:= \sqrt{t} \\ v &:= p + u \\ y &:= q/v \end{aligned}$$

Algorithmus 2 ist für $p >> q$ gutartig.

Zahlenwerte: $p = 6.0002$, $q = 0.01$ *und einer Mantissenlänge von* 5 *erhalten wir*
Algorithmus 1: 0.0008
Algorithmus 2: 0.00083326
exakte Lösung: 0.00083325 *(gerundet auf Mantissenlänge)*
(Übung: Nachrechnen)

Kapitel 3

Lineare Gleichungssysteme

3.1 Einführung und Aufgabenstellung

Algorithmen zur Lösung linearer Gleichungssysteme bilden die Basis für viele Anwendungen der Numerik.

Aufgabenstellung: Sei A eine (m,n)–Matrix und sei $b \in \mathbb{R}^m$. Gesucht ist ein Vektor $x \in \mathbb{R}^n$, welcher das lineare Gleichungssystem (LGS)

$$Ax = b \tag{3.1}$$

löst.
In der Numerik unterscheidet man in *direkte Methoden* zur Lösung von $Ax = b$, bei der eine Lösung x in *endlich* vielen Schritten berechnet wird und *indirekte Methoden*, bei denen eine Näherungslösung x von $Ax = b$ iterativ bestimmt wird. In diesem Kapitel werden wir uns mit den direkten Methoden auseinandersetzen, die indirekten Methoden sind dann Bestandteil der Numerik II.

Bemerkung 3.1. *Abhängig vom "Aussehen" der Koeffizientenmatrix A unterscheidet man in*

kleine	*- große*	*Systeme*
symmetrische	*- nichtsymmetrische*	*Matrizen*
mit	*- ohne*	*Bandstruktur*
schwach	*- vollbesetzte*	*Matrizen*

Danach richtet sich auch die Auswahl der Verfahren.

Beispiel 3.2. *Wir betrachten das Gleichstromnetzwerk in Abbildung 3.1.*
Nach den Kirchhoffschen *Gesetzen müssen sich zunächst an allen Knoten*

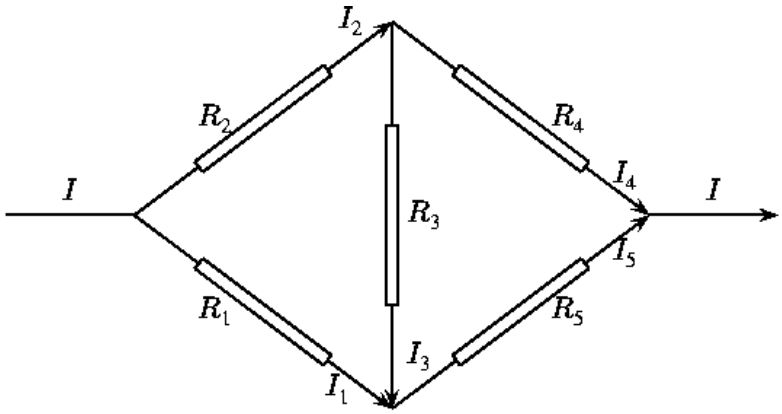


Abbildung 3.1: Gleichstromnetzwerk.

die eingehenden und die ausgehenden Ströme zu Null ergänzen. Für Anfang und Ende erhalten wir

$$I_1 + I_2 = I = I_4 + I_5,$$

für Oben und Unten

$$I_2 = I_3 + I_4 \qquad \textit{und} \qquad I_1 + I_3 = I_5.$$

Darüberhinaus müssen sich die Spannungen in den beiden Dreiecken zu Null summieren, nach dem Ohm*schen Gesetz ($U = R \cdot I$) führt dies für die bekannten Widerstände R_i auf die beiden Gleichungen*

$$R_2I_2 + R_3I_3 - R_1I_1 = 0 \qquad \textit{und} \qquad R_3I_3 + R_5I_5 - R_4I_4 = 0$$

Wir erhalten somit nach Umsortierung ein lineares Gleichungssystem der Form $Ax = b$:

$$\begin{array}{rcrcrcrcrcl}
I_1 & + & I_2 & & & & & & & = & I \\
 & & & & & & I_4 & + & I_5 & = & I \\
 & & I_2 & - & I_3 & - & I_4 & & & = & 0 \\
I_1 & & & + & I_3 & & & - & I_5 & = & 0 \\
-R_1I_1 & + & R_2I_2 & + & R_3I_3 & & & & & = & 0 \\
 & & & & -R_3I_3 & + & R_4I_4 & - & R_5I_5 & = & 0
\end{array}$$

Während man das letzte Beispiel sicher noch von Hand lösen kann, werden größere Probleme am Computer gelöst, vgl. Abbildung 3.2.
Bei der Lösung linearer Gleichungssysteme sind verschiedene Fälle möglich:

1. $m = n$: $rang(A) = n$, d.h. $Ax = b$ ist eindeutig lösbar. Da A invertierbar ist folgt $x = A^{-1}b$. Für numerische Rechnungen ist diese Darstellung jedoch nicht geeignet: auch die Cramersche Regel ist für $n \geq 3$ numerisch nicht brauchbar.

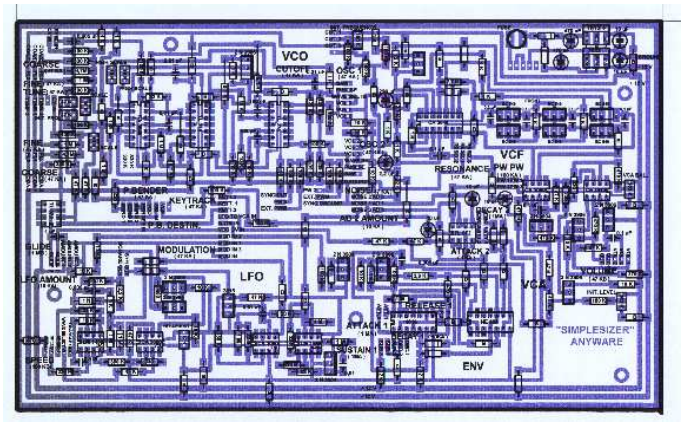


Abbildung 3.2: Eine eher kleine Platine.

2. $m > n$: Das LGS $Ax = b$ heißt *überbestimmt* und hat im allgemeinen keine Lösung. Stattdessen wird ein Ersatzproblem gelöst, vgl. Lineare Ausgleichsrechnung.

3. $m < n$: Das LGS $Ax = b$ heißt *unterbestimmt*. Wenn eine Lösung existiert, dann hat der Lösungsraum die Dimension $n - rang(A)$. Anwendungen findet man etwa in der Linearen Optimierung.

3.2 *LR*–Zerlegung und Gauß–Elimination

3.2.1 Idee der Gauß–Elimination/LR–Zerlegung

Sei $A = (a_{i,k})$ eine (n, n)–Matrix und $b \in \mathbb{R}^n$. Zu lösen sei das LGS

$$Ax = b \tag{3.2}$$

Das *Gauß'sche–Eliminationsverfahren* zur Lösung von LGS haben sie bereits im Rahmen ihres bisherigen Studiums kennengelernt. Es ist ein recht anschauliches Verfahren, das sich zudem leicht implementieren läßt.
Wir betrachten zunächst den vereinfachten Fall, daß die Matrix A in oberer Dreiecksform vorliegt, d.h.

$$A = R = \begin{pmatrix} r_{1,1} & r_{1,2} & \cdots & r_{1,n} \\ 0 & r_{2,2} & \cdots & r_{2,n} \\ \vdots & \ddots & \ddots & \vdots \\ 0 & \cdots & 0 & r_{n,n} \end{pmatrix} \tag{3.3}$$

Man spricht dann von einem *gestaffelten Gleichungssystem*, der Grund ist leicht ersichtlich. In diesem Fall kann für $r_{i,i} \neq 0$ leicht eine Lösung von $Rx = c$ (hier

mit $c := b$) mittels der rekursiven Vorschrift

$$\begin{aligned} x_n &= \frac{c_n}{r_{n,n}}, \\ x_{n-1} &= \frac{c_{n-1} - r_{n-1,n}x_n}{r_{n-1,n-1}}, \\ &\vdots \\ x_1 &= \frac{c_1 - r_{1,2}x_2 - \ldots - r_{1,n}x_n}{r_{1,1}}, \end{aligned} \tag{3.4}$$

beschrieben werden. In kompakter Form ergibt sich

$$x_i = \frac{1}{r_{i,i}} \left(c_i - \sum_{j=i+1}^{n} r_{i,j}x_j \right), \quad i = n, n-1, \ldots, 1. \tag{3.5}$$

Idee der Gauß-Elimination: Forme ein allgemeines Gleichungssystem $Ax = b$ in ein Gleichungssystem $Rx = c$ um, so dass die Matrix R in oberer Dreiecksform vorliegt.

3.2.2 Frobeniusmatrizen

Zur Durchführung der Gauß-Elimination werden sogenannte *Frobeniusmatrizen* bzw. *Elementarmatrizen* L_j benötigt:

$$L_j = \begin{pmatrix} 1 & & & & & \\ & \ddots & & & 0 & \\ & & \ddots & & & \\ & & & 1 & & \\ & & & -l_{j+1,j} & \ddots & \\ 0 & & & \vdots & & \ddots \\ & & & -l_{n,j} & & 1 \end{pmatrix} \leftarrow j\text{-te Zeile}$$

$$\uparrow$$

$$j\text{-te Spalte}$$

Für diese Elementarmatrizen gelten die nachfolgend aufgeführten Rechenregeln. Sei hierzu die Matrix

$$A = \begin{pmatrix} a_1 \\ \vdots \\ a_n \end{pmatrix} \text{ mit } a_i = (a_{i1}, \ldots, a_{in}).$$

gegeben. Für die Multiplikation einer Elementarmatrix mit einer Matrix gilt:

$$L_j A = \begin{pmatrix} a_1 \\ \vdots \\ a_j \\ a_{j+1} - l_{j+1,j} a_j \\ \vdots \\ a_n - l_{n,j} a_j \end{pmatrix}. \tag{3.6}$$

Die Inverse einer Elementarmatrix läßt sich einfach angeben:

$$L_j^{-1} = \begin{pmatrix} 1 & & & & \\ & \ddots & & & \\ & & 1 & & \\ & & l_{j+1,j} & \ddots & \\ & & \vdots & & \ddots \\ & & l_{n,j} & & 1 \end{pmatrix}. \tag{3.7}$$

Für die Multiplikation zweier invertierter Elementarmatrizen gilt die interessante Beziehung:

$$L_j^{-1} \cdot L_k^{-1} = \begin{pmatrix} 1 & & & & & & & \\ & \ddots & & & & & & \\ & & 1 & & & & & \\ & & l_{j+1,j} & \ddots & & & & \\ & & \vdots & & \ddots & & & \\ & & \vdots & & & 1 & & \\ & & \vdots & & & l_{k+1,k} & \ddots & \\ & & \vdots & & & \vdots & & \ddots \\ & & l_{n,j} & & & l_{n,k} & & 1 \end{pmatrix} \begin{matrix} \\ \\ \\ \leftarrow j \\ \\ \\ \leftarrow k \\ \\ \\ \end{matrix} \tag{3.8}$$

$$\begin{matrix} & & \uparrow & & \uparrow & & \\ & & j & & k & & \end{matrix}$$

3.2.3 Gauß–Elimination/LR–Zerlegung ohne Pivoting

Sei die Matrix

$$A := \begin{pmatrix} a_{1,1}^{(1)} & \cdots & a_{1,n}^{(1)} \\ \vdots & & \vdots \\ a_{n,1}^{(1)} & \cdots & a_{n,n}^{(1)} \end{pmatrix}$$

gegeben.

1. Schritt: Sei $a_{1,1}^{(1)} \neq 0$

$$L_1 A = \begin{pmatrix} a_{1,1}^{(1)} & \cdots & \cdots & a_{1,n}^{(1)} \\ 0 & a_{2,2}^{(2)} & \cdots & a_{2,n}^{(2)} \\ \vdots & \vdots & & \vdots \\ 0 & a_{n,2}^{(2)} & \cdots & a_{n,n}^{(2)} \end{pmatrix} \qquad \text{(vgl. (3.6) mit } j = 1)$$

mit

$$\begin{aligned} l_{i1} &:= \frac{a_{i1}^{(1)}}{a_{11}^{(1)}}, & i &= 2, \ldots, n, \\ a_{ik}^{(2)} &= a_{ik}^{(1)} - l_{i1} a_{1k}^{(1)}, & i, k &= 2, \ldots, n. \end{aligned}$$

In Worten: Subtrahiere von der i-ten Zeile der Matrix A das l_{i1}-fache der 1. Zeile, $i = 2, ..., n$.
Allgemein sei nun unsere Ausgangssituation vor dem j-ten Schritt ($j \geq 2$) bekannt.
Ausgangsmatrix vor dem j-ten Schritt ($j \geq 2$):

$$L_{j-1} \ldots L_1 A = \begin{pmatrix} a_{11}^{(1)} & & & & a_{1n}^{(1)} \\ & \ddots & & & \vdots \\ & & a_{jj}^{(j)} & \cdots & a_{jn}^{(j)} \\ 0 & & \vdots & & \vdots \\ & & a_{nj}^{(j)} & \cdots & a_{nn}^{(j)} \end{pmatrix}.$$

Mit diesem Wissen können wir den j-ten Schritt in Angriff nehmen:

j-ter Schritt ($j \geq 2$): Sei $a_{jj}^{(j)} \neq 0$

$$L_j L_{j-1} \dots L_1 A = \begin{pmatrix} a_{1,1}^{(1)} & \cdots & \cdots & \cdots & \cdots & a_{1,n}^{(1)} \\ & \ddots & & & & \vdots \\ & & a_{j,j}^{(j)} & \cdots & \cdots & a_{j,n}^{(j)} \\ & & 0 & a_{j+1,j+1}^{(j+1)} & \cdots & a_{j+1,n}^{(j+1)} \\ & & \vdots & \vdots & & \vdots \\ & & 0 & a_{n,j+1}^{(j+1)} & \cdots & a_{n,n}^{(j+1)} \end{pmatrix}$$

mit

$$\begin{aligned} l_{ij} &= \frac{a_{ij}^{(j)}}{a_{jj}^{(j)}}, & i &= j+1, \dots, n, \\ a_{ik}^{(j+1)} &= a_{ik}^{(j)} - l_{ij} a_{jk}^{(j)}, & i, k &= j+1, \dots, n. \end{aligned}$$

Nach $n-1$ Schritten erhalten wir dann das gewünschte Resultat:

$$L_{n-1} \dots L_1 A = \begin{pmatrix} a_{11}^{(1)} & \cdots & \cdots & a_{1n}^{(1)} \\ & \ddots & & \vdots \\ & & \ddots & \vdots \\ 0 & & & a_{nn}^{(n)} \end{pmatrix} =: R = (r_{ik}) \tag{3.9}$$

mit $r_{ii} = a_{ii}^{(i)} \neq 0$.

Wendet man die Matrizen L_j direkt auf die erweiterte Matrix (A, b) an, so ergibt sich

$$L_{n-1} \dots L_1 (A, b) = (R, c).$$

Das LGS $Ax = b$ ist dann äquivalent zu $Rx = c$ und kann gemäß (3.5) gelöst werden (Gauß-Elimination).

Aus (3.9) folgt mittels der Formeln (3.7)(3.8) die LR-Zerlegung der Matrix A:

$$A = L_1^{-1} \dots L_{n-1}^{-1} R =: LR \tag{3.10}$$

$$L = \begin{pmatrix} 1 & & & & 0 \\ l_{2,1} & \ddots & & & \\ \vdots & \ddots & \ddots & & \\ \vdots & & \ddots & \ddots & \\ l_{n,1} & \cdots & \cdots & l_{n,n-1} & 1 \end{pmatrix} \text{ linke Dreiecksmatrix}$$

Bei gegebener LR-Zerlegung $A = LR$ ist das LGS $Ax = b$ äquivalent zu den beiden leicht auflösbaren LGS

$$Lc = b, \quad Rx = c.$$

Insbesondere folgt noch aus (3.10)

$$\det(A) = \det(L)\det(R) = \prod_{j=1}^{n} r_{jj}.$$

Bei unseren bisherigen Überlegungen hatten wir stets $a_{jj}^{(j)} \neq 0$ voraussetzen müssen und es stellt sich die Frage, wann dies gesichert anzunehmen ist.

Problem: Wann gilt $a_{jj}^{(j)} \neq 0$?

Satz 3.3. *Sei A eine (n,n)-Matrix, deren Hauptabschnittsmatrizen A_j regulär sind. Dann gibt es eine eindeutige Zerlegung*

$A = LR$,
L linke Dreiecksmatrix mit $l_{jj} = 1, j = 1, \ldots n$,
R reguläre rechte Dreiecksmatrix.

Beweis: Vgl. Satz 3.9

3.2.4 Permutationsmatrizen

Zur Behandlung des Falles $a_{jj}^{(j)} = 0$ für ein j benötigen wir sogenannte *Permutationsmatrizen*. Hierzu sei

$$e_i = \begin{pmatrix} 0 \\ \vdots \\ 0 \\ 1 \\ 0 \\ \vdots \\ 0 \end{pmatrix} \leftarrow i \qquad \text{der } i-te \text{ kanonische Einheitsvektor}$$

Eine Matrix P heißt Permutationsmatrix, wenn eine Permutation $(i_1, \ldots, i_n)$ von $(1, \ldots, n)$ existiert mit

$$P = \begin{pmatrix} e_{i_1}^T \\ \vdots \\ e_{i_n}^T \end{pmatrix}.$$

Insbesondere haben Permutationsmatrizen die Eigenschaften: $P^2 = I$, also $P^{-1} = P$.

3.2.5 Gauß–Elimination/LR–Zerlegung mit Pivoting

Wir gehen davon aus, dass wir vielleicht bereits einige Schritte zur Gauß–Elimination bzw. LR–Zerlegung durchgeführt haben und befinden uns im j–ten $j \geq 1$ Schritt.

j-ter Schritt ($j \geq 1$): Die Ausgangsmatrix sei

$$A^{(j)} := \begin{pmatrix} a_{11}^{(1)} & & & & a_{1n}^{(1)} \\ & \ddots & & & \vdots \\ & & a_{jj}^{(j)} & \cdots & a_{jn}^{(j)} \\ 0 & & \vdots & & \vdots \\ & & a_{nj}^{(j)} & \cdots & a_{nn}^{(j)} \end{pmatrix}, \quad A^{(1)} := A \tag{3.11}$$

Wir führen eine Spaltenpivot-Suche durch: Wähle eine Zeile r mit

$$|a_{rj}^{(j)}| = \max_{i \geq j} |a_{ij}^{(j)}|.$$

Wir haben verschiedenen Fälle zu unterscheiden:

1. Fall: $a_{rj}^{(j)} = 0$: A ist singulär, setze $L_j = I$.

2. Fall: $a_{rj}^{(j)} \neq 0$: Vertausche die j-te Zeile mit der r-ten Zeile in $A^{(j)}$. Dies entspricht

einer Multiplikation von links mit der Permutationsmatrix

$$P_j = \begin{pmatrix} 1 & & & & & & & & & & & \\ & \ddots & & & & & & & & & & \\ & & 1 & & & & & & & & & \\ \cdots & \cdots & \cdots & 0 & \cdots & \cdots & \cdots & 1 & \cdots & \cdots & \cdots & \leftarrow j \\ & & & & 1 & & & & & & & \\ & & & & & \ddots & & & & & & \\ & & & & & & 1 & & & & & \\ \cdots & \cdots & \cdots & 1 & \cdots & \cdots & \cdots & 0 & \cdots & \cdots & \cdots & \leftarrow r \\ & & & & & & & & 1 & & & \\ & & & & & & & & & \ddots & & \\ & & & & & & & & & & 1 & \end{pmatrix}.$$

Damit erhalten wir im ungünstigsten Fall (in jedem Schritt ist eine Vertauschung von Zeilen notwendig) nach $n-1$ Schritten

$$L_{n-1}P_{n-1}\dots L_2P_2L_1P_1A = R. \tag{3.12}$$

Dies führt zu einer LR-Zerlegung mittels des folgenden Hilfssatzes.

Hilfssatz 3.4. *Sei $j < k$. Die Permutationsmatrix P_k vertausche die Zeilen k und $r \geq k$. Dann gilt*

$$P_kL_j = L'_jP_k,$$

wobei L'_j aus L_j dadurch hervorgeht, dass man in der j-ten Spalte das k-te und r-te Element vertauscht.

Beweis:

$$L_j = \begin{pmatrix} 1 & & & & & & \\ & \ddots & & & & & \\ & & 1 & & & & \leftarrow j \\ & & \vdots & \ddots & & & \\ & & -l_{kj} & & 1 & & 0 & \leftarrow k \\ & & \vdots & & & \ddots & & \\ & & -l_{rj} & & & & \ddots & \leftarrow r \\ & & \vdots & & 0 & & 1 & \end{pmatrix}$$

$$P_k L_j = \begin{pmatrix} 1 & & & & & & & \\ & \ddots & & & & & & \\ & & 1 & & & & & \\ & & \vdots & \ddots & & & & \\ & & -l_{rj} & & 0 & \cdots & 1 & \\ & & \vdots & & \vdots & & \vdots & \\ & & -l_{kj} & & 1 & \cdots & 0 & \\ & & \vdots & & & & & \ddots \end{pmatrix} \begin{matrix} \\ \\ \leftarrow j \\ \\ \leftarrow k \\ \\ \leftarrow r \\ \\ \end{matrix}$$

$$\begin{matrix} & & & \uparrow & & \uparrow & \\ & & & k & & r & \end{matrix}$$

$$\Rightarrow P_k L_j P_k = L'_j, \qquad P_k^2 = I$$

$$\Rightarrow P_k L_j = L'_j P_k.$$

$\diamond$

Die Anwendung des Hilfssatzes auf (3.12) zeigen wir der Einfachheit halber für $n = 4$:

$$\begin{aligned} & L_3 P_3 L_2 P_2 L_1 P_1 A && = R \\ \Leftrightarrow \quad & L_3 P_3 L_2 L'_1 P_2 P_1 A && = R \\ \Leftrightarrow \quad & L_3 L'_2 L''_1 \underbrace{P_3 P_2 P_1}_{=:P(\text{Permutationsmatrix})} A && = R \\ \Leftrightarrow \quad & PA = LR, \quad L := L''^{-1}_1 L'^{-1}_2 L_3^{-1}. \end{aligned}$$

Die Anwendung der obigen Operationen auf die erweiterte Matrix (A, b) führt auf die Matrix (R, c). R ist regulär, wenn A regulär ist, und das LGS $Rx = c$ kann gemäß (3.5) gelöst werden.
Zusammenfassend erhalten wir das folgende Resultat:

Satz 3.5 (LR–Zerlegung und Gauß-Elimination). *Zu jeder (n, n)-Matrix A gibt es eine Permutationsmatrix P, eine linke Dreiecksmatrix L und eine rechte Dreiecksmatrix R, so dass*

$$PA = LR, \quad l_{jj} = 1 \quad \text{für } j = 1, \ldots n.$$

Ist A regulär, so ist auch R regulär und die Gauß-Elimation liefert die eindeutige Lösung von $Ax = b$.

Bemerkung 3.6. *Bei der praktischen Durchführung der Gauß-Elimination kann man die wesentlichen Elemente von L, d.h. $l_{i,k},\ i \geq k+1,\ k \leq j-1$, auf den Null-Elementen der Matrix $A^{(j)}$ in 3.11 abspeichern.*

Beispiel 3.7.

$$\begin{pmatrix} 3 & 1 & 6 \\ 2 & 1 & 3 \\ 1 & 1 & 1 \end{pmatrix} \begin{pmatrix} x_1 \\ x_2 \\ x_3 \end{pmatrix} = \begin{pmatrix} 2 \\ 7 \\ 4 \end{pmatrix}$$

Die Pivot–Elemente in den erweiterten Matrizen werden durch einen Unterstrich markiert.

1. Schritt:

$$\left(\begin{array}{ccc|c} \underline{3} & 1 & 6 & 2 \\ 2 & 1 & 3 & 7 \\ 1 & 1 & 1 & 4 \end{array}\right)$$

Anwendung von L_1:

$$\left(\begin{array}{ccc|c} 3 & 1 & 6 & 2 \\ \boxed{2/3} & 1/3 & -1 & 17/3 \\ \boxed{1/3} & \underline{2/3} & -1 & 10/3 \end{array}\right)$$

2. Schritt:

Vertausche Zeile 2 und 3

$$\left(\begin{array}{ccc|c} 3 & 1 & 6 & 2 \\ \boxed{1/3} & \underline{2/3} & -1 & 10/3 \\ \boxed{2/3} & 1/3 & -1 & 17/3 \end{array}\right)$$

Anwendung von L_2 :

$$\left(\begin{array}{ccc|c} 3 & 1 & 6 & 2 \\ \boxed{1/3} & 2/3 & -1 & 10/3 \\ \boxed{2/3} & \boxed{1/2} & -1/2 & 4 \end{array}\right)$$

$$\Rightarrow \quad L = \begin{pmatrix} 1 & 0 & 0 \\ 1/3 & 1 & 0 \\ 2/3 & 1/2 & 1 \end{pmatrix}, \quad R = \begin{pmatrix} 3 & 1 & 6 \\ 0 & 2/3 & -1 \\ 0 & 0 & -1/2 \end{pmatrix}$$

$$\Rightarrow \quad P = \begin{pmatrix} 1 & 0 & 0 \\ 0 & 0 & 1 \\ 0 & 1 & 0 \end{pmatrix}, \quad PA = \begin{pmatrix} 3 & 1 & 6 \\ 1 & 1 & 1 \\ 2 & 1 & 3 \end{pmatrix}$$

Somit gilt $PA = LR$

Wir erhalten das gestaffelte Gleichungssystem $Rx = c$:

$$\begin{pmatrix} 3 & 1 & 6 \\ 0 & 2/3 & -1 \\ 0 & 0 & -1/2 \end{pmatrix} \begin{pmatrix} x_1 \\ x_2 \\ x_3 \end{pmatrix} = \begin{pmatrix} 2 \\ 10/3 \\ 4 \end{pmatrix}, \quad \begin{array}{l} x_1 = 19 \\ x_2 = -7 \\ x_3 = -8 \end{array}$$

Das Gauß–Verfahren zur Lösung von $Ax = b$ untergliedert sich somit in drei wesentliche Schritte:
Gauß-Elimination:

1. $PA = LR$
   $p = (p_1, \ldots, p_n)$ Permutationsvektor
2. $Lc = Pb$
   Vorwärtseinsetzen: $i = 1, \ldots, n$:
   $$c_i = b_{p_i} - \sum_{k=1}^{i-1} l_{ik} c_k$$
3. $Rx = c$
   Rückwärtseinsetzen: $i = n, n-1, \ldots, 1$:
   $$x_i = \frac{1}{r_{ii}}(c_i - \sum_{k=i+1}^{n} r_{ik} x_k)$$

3.2.6 Aufwandsbestimmung

Ein wichtiger Aspekt bei der Analyse numerischer Verfahren ist es zu untersuchen, wie lange diese Verfahren in der Regel benötigen, um zum gewünschten Ergebnis zu gelangen. Da sich die Rechenzeiten von Computer zu Computer unterscheiden, orientiert man sich nicht an der *Rechenzeit*, sondern an der *Anzahl der Rechenoperationen*, die ein Algorithmus benötigt.
Das vorgestellte Gauß–Verfahren liefert nach endlich vielen Schritten ein Ergebnis, wobei die Anzahl der elementaren Rechenoperationen von der Dimension n der Matrix A abhängt. Multiplikationen und Divisionen sind sogenannte *wesentliche Rechenoperationen*. Die Auswertung einer wesentlichen Rechenoperation war im Allgemeinen noch vor einigen Jahren deutlich 'teurer' als eine Addition oder Subtraktion (rechnerintern wird nicht in Addition und Subtraktion unterschieden). Die Unterschiede verschmelzen jedoch mehr und mehr mit moderenen Rechnerarchitekturen.
Zur *Aufwandsbestimmung* zählen wir die Rechenoperationen einfach ab. Zuvor erinnern wir uns an:

$$\sum_{i=1}^{n} i = \frac{1}{2}(n+1)n$$

und

$$\sum_{i=1}^{n} i^2 = \frac{1}{6} n(n+1)(2n+1).$$

Anzahl der Operationen: (ohne Additionen)

1. $PA = LR$
   $[(n-1) + (n-1)^2] + [(n-2) + (n-2)^2] + \ldots + [1 + 1^2]$
   $= \sum_{j=1}^{n-1} [(n-j) + (n-j)^2] = \frac{1}{2}n(n-1) + \frac{1}{6}n(n-1)(2n-1)$
   $= \frac{1}{3}(n^3 - n)$
2. $Lc = Pb$
   $1 + 2 + \ldots + (n-1) = \frac{1}{2}(n^2 - n)$
3. $Rx = c$
   $1 + 2 + \ldots + n = \frac{1}{2}(n^2 + n)$

Gesamtaufwand: $\frac{1}{3}n^3 + n^2 - \frac{1}{3}n$ Multiplikationen. (Additionen: $\frac{1}{3}n^3 + \frac{1}{2}n^2 - \frac{5}{6}n$)

Bemerkung 3.8. *Der Aufwand und damit die Rechenzeit steigt mit der dritten Potenz der Zahl der Unbekannten an:* $\mathcal{O}(n^3)$.

3.2.7 Algorithmus

Wir formulieren abschließend den Algorithmus.

Programm: $PA = LR$
für $j = 1, \ldots, n$:
  $p_j = j$
für $j = 1, \ldots, n-1$:
  Pivotsuche:
  $\max = |a_{jj}|, \quad r = j$
  für $i = j+1, \ldots, n$:
    falls $|a_{ij}| > \max$:
      $\max = |a_{ij}|, \quad r = i$
  falls $\max = 0$: STOP A singulär

  Zeilentausch:
  falls $r > j$:
    für $k = 1, \ldots, n$:
      $h_r = a_{jk}, \; a_{jk} = a_{rk}, \; a_{rk} = h_r$
    $h_i = p_j, \; p_j = p_r, \; p_r = h_i$
  Transformation:
    für $i = j+1, \ldots, n$:
      $a_{ij} = a_{ij}/a_{jj}$
      für $k = j+1, \ldots, n$:
        $a_{ik} = a_{ik} - a_{ij}a_{jk}$

3.3 Matrizen mit speziellen Eigenschaften

Besitzen Matrizen spezielle Eigenschaften, so kann es sich lohnen diese Eigenschaften gewinnbringend bei der Implementierung zu berücksichtigen.

3.3.1 Diagonaldominante Matrizen: Diagonalstrategie

Zunächst geben wir Bedingungen an, die die Durchführung der Gauß-Elimination ohne Pivotsuche ermöglichen (Diagonalstrategie).

Satz 3.9. *Sei A eine (n,n)-Matrix, deren Hauptabschnittsmatrizen A_j regulär sind. Dann gibt es eine eindeutige Zerlegung*

$$A = LR$$
$$L: \text{ linke Dreiecksmatrix mit } l_{jj} = 1,\ j = 1, \ldots, n,$$
$$R: \text{ reguläre rechte Dreiecksmatrix.}$$

Beweis: Der Beweis wird durch Induktion über n geführt.
IA: Für $n = 1$ ist die Beh. trivial.
IV: Die Beh. sei richtig für $n - 1$.
IS: Für eine (n,n)-Matrix ist die folgende Zerlegung zu zeigen.

$$A = \left(\begin{array}{c|c} A_{n-1} & c \\ \hline a^T & a_{nn} \end{array}\right) = \left(\begin{array}{c|c} L_{n-1} & 0 \\ \hline l^T & 1 \end{array}\right) \left(\begin{array}{c|c} R_{n-1} & r \\ \hline 0 & r_{nn} \end{array}\right).$$

Nach der Induktionsvoraussetzung gibt es eine Zerlegung

$$A_{n-1} = L_{n-1} R_{n-1}.$$

Für die gesuchten $l, r \in \mathbb{R}^{n-1}, r_{nn} \in \mathbb{R}$ erhält man die Gleichungen

$$c = L_{n-1} r \tag{3.13}$$
$$l^T R_{n-1} = a^T \qquad \Rightarrow R_{n-1}^T l = a \tag{3.14}$$
$$l^T r + r_{nn} = a_{nn}. \tag{3.15}$$

Diese Gleichungen sind eindeutig auflösbar, da nach Voraussetzung L_{n-1}, R_{n-1} regulär sind. ◇

Mit

$$D = diag(r_{jj}) = diag(a_{jj}^{(j)})$$

erhält man somit die Zerlegung

$$\boxed{\text{A=LDR}}, \quad l_{jj} = 1, \quad r_{jj} = 1.$$

Die Regularität der Hauptabschnittsmatrizen von A kann mit einer einfachen Bedingung für die Elemente a_{ij} von A nachgeprüft werden.

Definition 3.10 (Diagonaldominanz). *Die Matrix A heißt* diagonaldominant*, wenn*

$$|a_{ii}| > \sum_{\substack{k=1 \\ k\neq i}}^{n} |a_{ik}|, \quad (i = 1, \ldots, n).$$

Satz 3.11. *Bei einer diagonaldominanten Matrix A sind alle Hauptabschnittsmatrizen regulär, also existiert die LR-Zerlegung $A = LR$.*

Beweis: Für die j-te Abschnittsmatrix A_j gelte

$$A_j x = 0 \qquad \text{für ein } x \in \mathbb{R}^j.$$

Zu zeigen ist dann $x = 0$. Wäre

$$|x_r| = \max_{1\leq i\leq j} |x_i| > 0,$$

so betrachten wir die r-te Gleichung

$$\sum_{i=1}^{j} a_{ri} x_i = 0.$$

Zusammen mit

$$|a_{rr}| > \sum_{\substack{k=1 \\ k\neq r}}^{j} |a_{rk}|$$

ergibt sich hieraus ein Widerspruch:

$$\begin{aligned} |a_{rr}||x_r| &= |\sum_{\substack{k=1 \\ k\neq r}}^{j} a_{rk}x_k| \\ &\leq \sum_{k\neq r} |a_{rk}||x_k| \\ &\leq \sum_{k\neq r} |a_{rk}||x_r| < |a_{rr}||x_r|. \end{aligned}$$

◇

Beispiel 3.12. *Die bei der Berechnung von Spline-Funktionen (vgl. Kapitel zur Interpolation) auftretende tridiagonale Matrix*

$$\begin{pmatrix} 4 & 1 & & & 0 \\ 1 & 4 & 1 & & \\ & \ddots & \ddots & \ddots & \\ & & 1 & 4 & 1 \\ 0 & & & 1 & 4 \end{pmatrix}$$

ist diagonal dominant und damit LR-zerlegbar.

Bemerkung 3.13. *Spezielle Matrizen, die das Kriterium in Satz 3.9 erfüllen, sind die positiv definiten Matrizen (vgl. nächsten Abschnitt).*

3.3.2 Positiv definite Matrizen: Cholesky–Verfahren

Definition 3.14. *Eine (n,n)–Matrix A heißt* symmetrisch*, falls $A = A^T$ gilt.*

Definition 3.15. *Eine symmetrische (n,n)–Matrix A heißt* positiv definit*, falls*

$$x^T A x > 0, \qquad \text{für alle} \quad x \in \mathbb{R}^n,\, x \neq 0 \tag{3.16}$$

gilt.

Die positive Definitheit scheint sehr einschränkend zu sein, dennoch ist sie in vielen Anwendungen erfüllt.

Bemerkung 3.16. *Für positiv definite Matrizen kann eine LR–Zerlegung ohne Pivoting durchgeführt werden.*

Satz 3.17. *Sei A positiv definit.*

1. *Alle Hauptabschnitt-Matrizen von A sind positiv definit und regulär. Insbesondere ist A regulär.*

2. *Es gibt genau eine linke Dreiecksmatrix L mit $l_{ii} > 0, i = 1, \ldots, n$, so dass gilt*

$$\boxed{A = LL^T}$$

(Beachte: $l_{ii} = 1$ wird nicht gefordert)

Beweis:
zu 1: Übung
zu 2: Nach Satz 3.9 gibt es genau eine Zerlegung

$$\begin{aligned} A &= UV \\ U &= (u_{ik}) \;:\; \text{linke Dreiecksmatrix, } u_{ii} = 1, \\ V &= (v_{ik}) \;:\; \text{reguläre rechte Dreiecksmatrix} \end{aligned}$$

Sei

$$D = \begin{pmatrix} v_{11} & & 0 \\ & \ddots & \\ 0 & & v_{nn} \end{pmatrix}, \quad v_{ii} \neq 0.$$

Setze

$$\begin{aligned} & R &=& D^{-1}V: \text{ rechte Dreiecksmatrix, } r_{ii}=1 \\ \Rightarrow \quad & A &=& UDR, \quad A=A^T=R^TD^TU^T=R^TDU^T. \end{aligned}$$

Wegen der Eindeutigkeit der Zerlegung folgt:

$$R^T=U, \text{ d.h. } A=UDU^T=R^TDR.$$

Behauptung: D ist positiv definit, d.h. $v_{ii}>0$.
Für alle $x\neq 0$ gilt:

$$\begin{aligned} & 0 &<& x^TAx=x^TR^TDRx=(Rx)^TDRx \\ \Rightarrow\ & 0 &<& y^TDy \quad \text{für alle } y\neq 0, \text{ da } R \text{ regulär}, \\ \Rightarrow\ & D & & \text{positiv definit.} \end{aligned}$$

Mit

$$D^{1/2}:=\begin{pmatrix} \sqrt{v_{11}} & & 0 \\ & \ddots & \\ 0 & & \sqrt{v_{nn}} \end{pmatrix}, \quad L:=UD^{1/2}$$

gilt

$$\boxed{A=LL^T}$$

◇

Bemerkung 3.18. *Ist L eine linke untere Dreiecksmatrix, so ist L^T eine rechte obere Dreiecksmatrix, d.h. für positiv definite Matrizen existiert eine LR–Zerlegung mit $R=L^T$. (Achtung: Hier sind die Diagonalelemente von L nicht normiert.)*

Bemerkung 3.19. *Offensichtlich reicht es aufgrund von Satz 3.9 für eine Cholesky–Zerlegung $A=LL^T$ die Matrix L zu bestimmen.*

Leider ist das Cholesky–Verfahren nicht so anschaulich wie die Gauß–Elimination. Zur Bestimmung der Komponenten von L geht man induktiv Spaltenweise vor: Sei $L=(l_{i,j})$ die linke untere (n,n)–Dreiecksmatrix mit $A=LL^T$, die nach Satz 3.9 existiert und eindeutig ist. In Komponentenschreibweise ergibt sich

$$A=\begin{pmatrix} a_{1,1} & \cdots & a_{1,n} \\ \vdots & & \vdots \\ a_{n,1} & \cdots & a_{n,n} \end{pmatrix}=\begin{pmatrix} l_{1,1} & & 0 \\ \vdots & \ddots & \\ l_{n,1} & \cdots & l_{n,n} \end{pmatrix}\begin{pmatrix} l_{1,1} & \cdots & l_{n,1} \\ & \ddots & \vdots \\ 0 & & l_{n,n} \end{pmatrix}. \tag{3.17}$$

Offensichtlich gilt

$$a_{1,1}=l_{1,1}\cdot l_{1,1}, \quad \text{also} \quad l_{1,1}=\sqrt{a_{1,1}}, \tag{3.18}$$

d.h. $l_{1,1}$ läßt sich einfach berechnen. Ebenso gilt

$$a_{i,1} = l_{i,1} \cdot l_{1,1}, \quad \text{also} \quad l_{i,1} = \frac{a_{i,1}}{l_{1,1}}, \quad i = 2, \ldots, n, \tag{3.19}$$

womit die erste Spalte von L bekannt ist (Dieses war der Induktionsanfang). Seien also die $l_{i,j}$, für $j \leq k-1$ bekannt (Induktionsvoraussetzung). Wir möchten als nächstes die Elemente der k-ten Spalte berechnen. Aus (3.17) ergibt sich

$$a_{k,k} = l_{k,1}^2 + \ldots + l_{k,k}^2, \tag{3.20}$$

und somit aufgrund der Eindeutigkeit von L

$$l_{k,k} = \sqrt{a_{k,k} - \sum_{j=1}^{k-1} l_{k,j}^2}\,. \tag{3.21}$$

Ebenso ergibt sich aus (3.17)

$$a_{i,k} = \sum_{j=1}^{k} l_{i,j} l_{k,j} \tag{3.22}$$

und damit

$$l_{i,k} = \frac{1}{l_{k,k}} \left(a_{i,k} - \sum_{j=1}^{k-1} l_{i,j} l_{k,j} \right), \qquad i \geq k+1. \tag{3.23}$$

Auf diesem Wege können wir die vollständige Matrix L bestimmen. Der Zusammenhang mit dem Ausgangsproblem ist durch

$$Ax = LL^T x = Lc = b$$

gegeben. Eine Abschätzung des Aufwandes ergibt, daß außer n Quadratwurzeln noch

$$\boxed{\frac{1}{6} n^3 + \mathcal{O}(n^2)} \tag{3.24}$$

Rechenoperationen durchgeführt werden müssen.

Bemerkung 3.20. *Auch wenn Gauß– und Cholesky–Verfahren beide die Ordnung $O(n^3)$ besitzen, so ist das Cholesky–Verfahren für große n dennoch etwa doppelt so schnell wie das Gauß–Verfahren, man vergleiche die jeweiligen Faktoren vor n^3.*

Die Lösung des LGS $Ax = b$ nach der Metholde von Cholesky erfolgt in den drei Schritten

(3.25)

1.	A	$=$	LL^T	:	Cholesky–Zerlegung
2.	Lc	$=$	b	:	Vorwärtseinsetzen
3.	L^Tx	$=$	c	:	Rückwärtseinsetzen

Bei positiv definiten Matrizen A sind die Hauptdiagonalelemente $a_{ii} = e_i^T A e_i > 0$ d.h. positiv. Darüberhinaus kann man leicht zeigen, dass diagonal-dominante Matrizen (vgl. Definition 3.10) mit $a_{ii} > 0$, d.h.

$$a_{ii} > \sum_{k \neq i} |a_{ik}| \qquad (i = 1, \ldots, n),$$

positiv definit sind.
Für eine positiv definite Matrix A ist die Reduktion der quadratischen Form $x^T Ax$ auf eine Summe von Quadraten (im Körper der reellen Zahlen) möglich:

$$\begin{aligned} x^T Ax &= x^T LL^T x = (L^T x)^T (L^T x) \\ &= \sum_{j=1}^{n} (\sum_{k=j}^{n} l_{kj} x_k)^2. \end{aligned}$$

Zusätzlich ergibt sich für die Hauptabschnittsmatrizen (Hauptmenoren):

$$\det A = \prod_{j=1}^{n} l_{jj}^2 = \prod_{j=1}^{n} a_{jj}^{(j)} > 0, \quad \det \begin{pmatrix} a_{11} & \cdots & a_{1k} \\ \vdots & & \vdots \\ a_{k1} & \cdots & a_{kk} \end{pmatrix} = \prod_{j=1}^{k} l_{jj}^2 > 0.$$

Folgerung 3.21. *Eine symmetrische Matrix A ist genau dann positiv definit, wenn*

$$\det \begin{pmatrix} a_{11} & \cdots & a_{1k} \\ \vdots & & \vdots \\ a_{k1} & \cdots & a_{kk} \end{pmatrix} > 0 \text{ für } k = 1, \ldots, n.$$

Beispiel 3.22. *Die bei der Diskretisierung von Randwertproblemen für Differentialgleichungen auftretende Matrix*

$$A_n = \left.\begin{pmatrix} 2 & -1 & & & 0 \\ -1 & 2 & -1 & & \\ & \ddots & \ddots & \ddots & \\ & & -1 & 2 & -1 \\ 0 & & & -1 & 2 \end{pmatrix}\right\} n$$

ist positiv definit, denn mittels der Rekursion

$$\det A_{n+1} = 2 \det A_n - \det A_{n-1}$$

erkennt man $\det A_n = n + 1 > 0$.

3.3.3 Bandmatrizen: Bandausnutzende Verfahren

In vielen Anwendungen spielen Bandmatrizen eine wichtige Rolle.

Definition 3.23. *Unter der Bandbreite einer Matrix A versteht man die kleinste natürliche Zahl $m < n$, so dass gilt*

$$a_{ik} = 0 \qquad \textit{für alle } i \textit{ und } k \textit{ mit } |i - k| > m.$$

Beispiel 3.24. *Die Matrix*

$$A = \begin{pmatrix} * & * & * & & & & \\ * & * & * & * & & & \\ * & * & * & * & * & & \\ & \ddots & \ddots & \ddots & \ddots & \ddots & \\ & & * & * & * & * & * \\ & & & * & * & * & * \\ & & & & * & * & * \end{pmatrix}$$

hat die Bandbreite $m = 2$.

Ein Blick auf den Gauß–Algorithmus zeigt leicht die folgende Aussage:

Satz 3.25. *Besetzt die Matrix A mit der Bandbreite m eine LR-Zerlegung $A = LR$, so haben L und R ebenfalls die Bandbreite m, denn es gilt*

$$\begin{aligned} l_{ik} &= 0 \quad \textit{für } i, k \textit{ mit } i - k > m, \\ r_{ik} &= 0 \quad \textit{für } i, k \textit{ mit } k - i > m. \end{aligned}$$

Korollar 3.26. *Die Linksdreiecksmatrix L der Cholesky-Zerlegung $A = LL^T$ einer positiv definiten Bandmatrix mit der Bandbreite m besitzt ebenfalls Bandbreite m.*

Besonders einfach zu behandelnde LGS erhält man für tridiagonale Matrizen A der Bandbreite $m = 1$. Zu lösen sei das LGS $Ax = d$ mit

$$A = \begin{pmatrix} a_1 & b_1 & & & \\ c_2 & a_2 & b_2 & & \\ & \ddots & \ddots & \ddots & \\ & & c_{n-1} & a_{n-1} & b_{n-1} \\ & & & c_n & a_n \end{pmatrix}, \quad d = \begin{pmatrix} d_1 \\ \vdots \\ d_n \end{pmatrix}$$

Es existiere die LR-Zerlegung $A = LR$. Nach Satz 3.25 sind L und R bidiagonal und können in der Form angesetzt werden

$$L = \begin{pmatrix} 1 & & & & \\ l_2 & 1 & & & \\ & l_3 & 1 & & \\ & & \ddots & \ddots & \\ & & & l_n & 1 \end{pmatrix}, \quad R = \begin{pmatrix} m_1 & r_1 & & & \\ & m_2 & r_2 & & \\ & & & & \\ & & & m_{n-1} & r_{n-1} \\ & & & & m_n \end{pmatrix}.$$

Die Ausmultiplikation $A = LR$ führt auf die Beziehung $r_i = b_i$, $i = 1, \ldots, n$ und den folgenden Algorithmus zur Lösung von $Ax = d$:

(3.26)

$\underline{A = LR}$
$m_1 = a_1$
für $i = 2, \ldots, n$:
$l_i = c_i / m_{i-1}$
$m_i = a_i - l_i \cdot b_{i-1}$
$\underline{Ly = d}$
$y_1 = d_1$
für $i = 2, \ldots, n$:
$y_i = d_i - l_i \cdot y_{i-1}$
$\underline{Rx = y}$
$x_n = y_n / m_n$
für $i = n-1, n-2, \ldots, 1$:
$x_i = (y_i - b_i \cdot x_{i+1}) / m_i$

3.4 Fehleranalyse und Fehlerbehandlung

3.4.1 Fehlerabschätzungen

Wie in der Einleitung ausgeführt können Computer nicht alle reellen Zahlen darstellen, daher werden die meisten Zahlen intern gerundet. Als Konsequenz ergeben sich Rundungsfehler. Selbst wenn Eingabedaten und das Ergebnis eines Algorithmus frei von Rundungsfehlern wären, können Zwischenergebnisse gerundet worden sein. Aus diesem Grund wird in der Regel nicht die Lösung x des Gleichungssystems $Ax = b$ berechnet, sondern die Lösung $\tilde{x}$ eines 'benachbarten' oder 'gestörten' Gleichungssystems

$$(A + \triangle A)\tilde{x} = b + \triangle b \tag{3.27}$$

$\triangle b$: Fehler im Vektor b (Residuum),
$\triangle A$: Fehler in der Matrix A,
$\triangle x := \tilde{x} - x$: Fehler der Näherungslösung

Um die nachfolgende Analyse durchzuführen benötigen wir den Begriff der *zugeordneten Matrixnorm*. Wir erinnern zunächst an verschiedene Normen für Vektoren $x \in \mathbb{R}^n$. In dieser Vorlesung verwenden wir üblicherweise

$$\|x\|_2 = \sqrt{x^T x} = \sqrt{\sum_{i=1}^{n} x_i^2}, \quad \text{(euklidische Norm oder 2–Norm)}$$

welche wir meistens einfach mit $\|.\|$ bezeichnen. Weitere Normen sind die

$$\|x\|_1 = \sum_{i=1}^{n} |x_i|, \quad \text{(1–Norm)}$$

oder die

$$\|x\|_\infty = \max_{i=1,\ldots,n} |x_i|, \quad \text{(Maximumsnorm oder } \infty\text{–Norm)}.$$

Für alle Vektornormen kann man eine zugeordnete Matrixnorm definieren.

Definition 3.27. *Sei A eine (n, n)–Matrix und $\|.\|_p$ eine Vektornorm im $\mathbb{R}^n$. Die Zahl*

$$\|A\|_p := \max_{\|x\|_p = 1} \|Ax\|_p$$

heißt die der Vektor–Norm $\|.\|_p$ zugeordnete Matrixnorm.

Bemerkung 3.28. *Wir bezeichnen nachfolgend die Vektornormen und die ihnen zugeordneten Matrixnormen mit dem gleichen Symbol.*

Sei nachfolgend $A = (a_{i,j})$.

Beispiel 3.29.

$$\|A\|_1 := \max_{j=1,\dots,n} \sum_{i=1}^{n} |a_{i,j}|, \qquad \textit{(Spaltensummennorm)}$$

Beispiel 3.30.

$$\|A\|_\infty := \max_{i=1,\dots,n} \sum_{j=1}^{n} |a_{i,j}|, \qquad \textit{(Zeilensummennorm)}$$

Beispiel 3.31.

$$\|A\|_2 := \sqrt{\rho(A^T A)}, \qquad \textit{(Spektralnorm)}$$

wobei $\rho(B)$ den Betrag des betragsgrößten Eigenwert einer symmetrischen Matrix B bezeichnet.

Definition 3.32. *Als* Kondition von A *bzgl. einer Matrixnorm $\|.\|_p$ bezeichnen wir die Zahl*

$$cond_p(A) := \|A\|_p \|A^{-1}\|_p \,.$$

Satz 3.33 (Fehleranalyse). *Sei x die eindeutige Lösung von $Ax = b$, und $\triangle A$, $\triangle b$ Störungen von A, b mit*

$$q = cond(A)\frac{\|\triangle A\|}{\|A\|} < 1. \tag{3.28}$$

Dann ist auch das gestörte System

$$(A + \triangle A)(x + \triangle x) = b + \triangle b \tag{3.29}$$

eindeutig lösbar und es gilt

$$\boxed{\frac{\|\triangle x\|}{\|x\|} \leq \frac{cond(A)}{1-q}\left(\frac{\|\triangle A\|}{\|A\|} + \frac{\|\triangle b\|}{\|b\|}\right)} \tag{3.30}$$

Beweis: Sei $x + \triangle x$ Lösung von (3.29). Nach Ausmultiplizieren ergibt sich

$$Ax + A\triangle x + \triangle A x + \triangle A \triangle x - b - \triangle b = 0$$

und weiter wegen $Ax - b = 0$

$$A\triangle x = \triangle b - \triangle A x - \triangle A \triangle x$$

und somit

$$\triangle x = -A^{-1}\left(-\triangle b + \triangle A x + \triangle A \triangle x\right).$$

Für *verträgliche Matrixnormen* folgt hieraus (Dreiecksungleichung)

$$\begin{aligned} \|\triangle x\| &\leq \|A^{-1}\| \cdot \| - \triangle b + \triangle A x + \triangle A \triangle x\| \\ &\leq \|A^{-1}\| \left(\|\triangle b\| + \|\triangle A\| \cdot \|x\| + \|\triangle A\| \cdot \|\triangle x\|\right) \end{aligned}$$

und weiter

$$\left(1 - \|A^{-1}\| \cdot \|\triangle A\|\right) \|\triangle x\| \leq \|A^{-1}\| \left(\|\triangle b\| + \|\triangle A\| \cdot \|x\|\right)$$

Aufgrund der Voraussetzung (3.28) mit $q < 1$ folgt

$$\|\triangle x\| \leq \frac{\|A^{-1}\|}{1-q} \left(\|\triangle b\| + \|\triangle A\| \cdot \|x\|\right)$$

als absolutem Fehler. Es ergibt sich weiter

$$\frac{\|\triangle x\|}{\|x\|} \leq \frac{\|A^{-1}\|}{1-q} \left(\frac{\|\triangle b\|}{\|x\|} + \|\triangle A\|\right). \tag{3.31}$$

Aus $\|b\| = \|Ax\| \leq \|A\| \cdot \|x\|$ folgt

$$\|x\| \geq \frac{\|b\|}{\|A\|}$$

und somit aus (3.31)

$$\frac{\|\triangle x\|}{\|x\|} \leq \frac{\|A^{-1}\| \cdot \|A\|}{1-q} \left(\frac{\|\triangle b\|}{\|b\|} + \frac{\|\triangle A\|}{\|A\|}\right)$$

und hieraus die Behauptung. ◇

Die Zahl $cond(A)$ hat also die Bedeutung eines Verstärkungsfaktors und mißt die Empfindlichkeit der Lösung x gegenüber Störungen in A und b. Das LGS $Ax = b$ heißt schlecht konditioniert, wenn $cond(A) >> 1$.

Beispiel 3.34. *Auswirkung schlechter Kondition:*

$$A = \begin{pmatrix} 1 & 1 \\ 1 & 0.99 \end{pmatrix}, \quad b = \begin{pmatrix} 1 \\ 1 \end{pmatrix}, \quad x = \begin{pmatrix} 1 \\ 0 \end{pmatrix}$$

$$\Delta A = \begin{pmatrix} 0.01 & 0.01 \\ 0 & 0 \end{pmatrix}, \quad \Delta b = \begin{pmatrix} 0 \\ 0 \end{pmatrix}, \quad x + \Delta x = \begin{pmatrix} 200/101 \\ -100/101 \end{pmatrix}$$

Obwohl der Fehler in A bei 1% liegt, haben $x, x+\Delta x$ nichts mehr miteinander zu tun.
Erklärung:

$$||A||_\infty = 2, \quad A^{-1} = \begin{pmatrix} -99 & 100 \\ 100 & -100 \end{pmatrix}, \quad ||A^{-1}||_\infty = 200, \quad cond_\infty(A) = 400!$$

Geometrisch: Die Zeilenvektoren a_1, a_2 von A haben beinahe die gleiche Richtung.

3.4.2 Skalierung

Die Kondition eines Problems kann ggf. durch Skalierung der Matrix A verbessert werden. Unter Skalierung versteht man den Übergang

$$A \to DA, \quad D = \begin{pmatrix} d_1 & & 0 \\ & \ddots & \\ 0 & & d_n \end{pmatrix}, \qquad d_i \neq 0,$$

d.h. die i-te Zeile von A wird mit d_i multipliziert. Die optimale Wahl einer Diagonalmatrix D, welche $cond(DA)$ möglichst klein macht, erhält man durch den folgenden Satz (ohne Beweis):

Satz 3.35 (Van der Sluis). *Für $A = (a_{ik})$ sei*

$$\sum_{k=1}^{n} |a_{ik}| = 1, \quad i = 1, \ldots, n \quad (\textit{insbesondere } ||A||_\infty = 1).$$

Dann gilt für jede Diagonalmatrix D mit $\det D \neq 0$

$$cond_\infty(DA) \geq cond_\infty(A).$$

Folgerung 3.36. *Für eine beliebige reguläre Matrix $A = (a_{ik})$ ist mit der Skalierung*

$$D = diag(d_i), \quad d_i := \left(\sum_{k=1}^{n} |a_{ik}| \right)^{-1}$$

die Kondition $cond_\infty(DA)$ möglichst klein.

3.4.3 Iterative Nachverbesserung

Unabhängig von einer schlechten Kondition der Matrix A liefern numerische Verfahren zur Lösung linearer Gleichungssysteme nicht die exakte Lösung. Wir erhalten dann lediglich eine Näherungslösung, die unseren Anforderungen aber möglicherweise nicht hinreichend gerecht wird. Mit einem kleinen Trick läßt sich die berechnete Näherungslösung aber dennoch weiter verbessern:

x sei exakte Lösung von $Ax = b$
$\tilde{x}$ sei irgendeine Näherungslösung, z.B. aus Gauß-Algorithmus.

Verbesserung in drei Schritten:

1. Berechne $r := b - A\tilde{x}$ *"Residuum"*
2. Bestimme Δx aus $A\Delta x = r$ *"Korrektur"*
3. Berechne $x' := \tilde{x} + \Delta x$

Begründung:

$$\begin{aligned} x' &= \tilde{x} + \Delta x = \tilde{x} + A^{-1}r = \tilde{x} + A^{-1}(b - A\tilde{x}) \\ &= \tilde{x} + x - \tilde{x} = x \end{aligned}$$

In der praktischen Anwendung/Implementierung bewirken Rundungsfehler, dass i.A. $x' \neq x$ ist. Die Verbesserung kann wiederholt werden, solange Δx nur mit einer Stelle korrekt berechnet wird. In diesem Fall ist x' besser als $\tilde{x}$.

Bei der algorithmischen Durchführung sind einige Dinge zu beachten:

- Wurde $\tilde{x}$ durch den Gauß-Algorithmus gewonnen, so erfüllt $\tilde{x}$ das Gleichungssystem meist sehr gut, d. h.

  $$b \approx A\tilde{x} \quad \Rightarrow \quad r = b - A\tilde{x} \quad \text{auslöschungsgefährdet!}$$

  Um dem entgegenzuwirken berechnet man im Schritt 1. das Residuum mit doppelter Genauigkeit.

- Bei der Auflösung von $A \cdot \Delta x = r$ benutze man die bereits berechnete $LR - Zerlegung$.

- Rundungsfehler im 3. Schritt begrenzen i.A. die erreichbare Genauigkeit.

Beispiel 3.37. *Sechs Dezimalstellen; unterstrichene Stellen sind falsch*

$$\begin{bmatrix} 0.566012 & 0.765456 \\ 0.389953 & 0.527611 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} = \begin{bmatrix} 0.395102 \\ 0.272744 \end{bmatrix}$$

$$\begin{array}{rrcl} \textit{exakt}: & x_1 & = & -\ 2.20227459\ldots \\ & x_2 & = & \ \ 2.14462470\ldots \\ \textit{Gauß} & \tilde{x}_1 & = & -\ 2.19\underline{453} \\ & \tilde{x}_2 & = & \ \ 2.13\underline{889} \end{array}$$

1. *Nachverbesserung*	2. *Nachverbesserung*
$x_1' = -2.20\ 22\underline{6}$	$x_1'' = -2.20227$
$x_2' = \ \ 2.1446\underline{1}$	$x_2'' = \ \ 2.14462$

Bemerkung 3.38. *Meist reicht nur ein Schritt, um das Resultat deutlich zu verbessern.*

3.5 Die QR-Zerlegung einer Matrix, das Verfahren von Householder

3.5.1 Einleitung und Motivation

Sei A eine (n, n)-Matrix (reell, nicht notwenig regulär).
Bei der LR-Zerlegung (ohne Pivotsuche) hatten wir das Ergebnis:

$$\boxed{A = LR}$$

L: linke Dreiecksmatrix
R: rechte Dreiecksmatrix.

Bei der QR-Zerlegung suchen wir hingegen eine Zerlegung der Form:

$$\boxed{A = QR}$$

Q: orthogonal, d. h. $Q^TQ = I$,
R: rechte Dreiecksmatrix.

Motivation zur QR-Zerlegung:
Zur Lösung des LGS $Ax = b$ erzeugt man bei der LR-Zerlegung und Gauß-Elimination eine Sequenz

$(A, b) = (A^{(1)}, b^{(1)}) \to \ldots \to (A^{(j)}, b^{(j)}) \to \ldots \to (A^{(n)}, b^{(n)}) = (R, c)$
$(A^{(j+1)}, b^{(j+1)}) = L_j(A^{(j)}, b^{(j)})$.

Sei $\varepsilon^{(j)}$ der Rundungsfehler bei der Berechnung von $(A^{(j)}, b^{(j)})$. Für irgendeine Vektornorm $\|x\|$ gilt nach Satz 3.33 die Abschätzung

$$\frac{\|\Delta x\|}{\|x\|} \leq \sum_{j=1}^{n} \varepsilon^{(j)} cond(A^{(j)}).$$

Die Gauß-Elimination ist daher nicht gutartig, falls

$$cond(A^{(j)}) >> cond(A^{(1)}) = cond(A).$$

Idee: Wähle Matrix Q_j mit Übergang

$$(A^{(j+1)}, b^{(j+1)}) = Q_j(A^{(j)}, b^{(j)}), \quad cond(A^{(j+1)}) = cond(A^{(j)}).$$

Dazu beschränken wir uns auf die euklidische Norm

$$\|x\| = \|x\|_2 = (x^T x)^{1/2}, \qquad \|A\| = \|A\|_2$$

und notieren eine später zu benutzende Hilfsaussage:

Hilfssatz 3.39. *Sei Q orthogonal, dann gilt:*

(i) $\|Q\|_2 = 1$

(ii) $\|QA\|_2 = \|A\|_2$ *für alle* A

(iii) Wenn A regulär ist, gilt $cond_2(QA) = cond_2(A)$.

Beweis: Übung

3.5.2 Householdermatrizen

Sei $w \in \mathbb{R}^n$ mit $w^T w = 1$ und sei die *Householdermatrix* Q definiert durch

$$Q := I - 2ww^T, \qquad ww^T = (w_i \cdot w_k).$$

Dann hat die so konstruierte Matrix Q folgende Eigenschaften:
Q ist symmetrisch:

$$Q^T = I - 2(ww^T)^T = I - 2ww^T = Q$$

Q ist orthogonal wegen $w^T w = 1$:

$$\begin{aligned} Q^T Q &= (I - 2ww^T)(I - 2ww^T) \\ &= I - 2ww^T - 2ww^T + 4ww^T ww^T = I \end{aligned}$$

Für $x \in \mathbb{R}^n$ bedeutet

$$Qx = (I - 2ww^T)x = x - 2(w^T x)w$$

eine Spiegelung an der Hyperebene

$$H = \{z \in \mathbb{R}^n | w^T z = 0\} :$$

$x = y + z$ mit $w^T z = 0$, (y aus dem orthogonalen Koplement)

$= \alpha w + z$

$\Rightarrow w^T x = \alpha w^T w + w^T z = \alpha$

$\Rightarrow Qx = x - 2(w^T x)w = x - 2\alpha w = \alpha w + z - 2\alpha w = -\alpha w + z = -y + z$

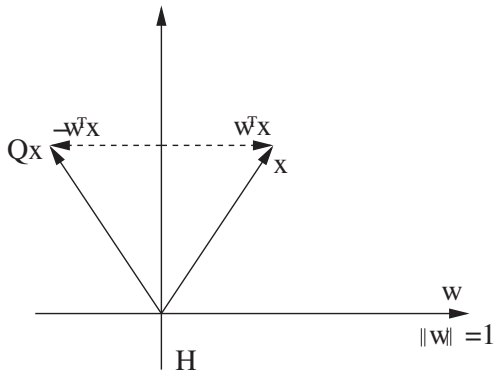


Abbildung 3.3: Spiegelung an Hyperebene.

Problem: Sei $x = (x_1, \ldots, x_n)^T \neq 0$ vorgegeben. Bestimme $w \in \mathbb{R}^n$, $w^T w = 1$, mit

$$Qx = ke_1, \qquad k \in \mathbb{R}.$$

In diesem Fall ist Q eine spezielle Spiegelung an einer Hyperebene, vgl. die nachfolgende Abbildung:

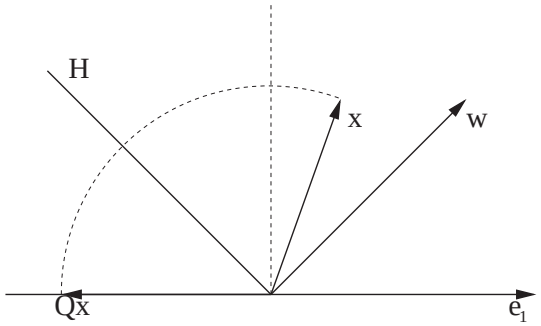


Abbildung 3.4: Spiegelung.

Analytische Berechnung von Q: (Für $Qx = ke_1$)

$$\begin{array}{rrcl} \Rightarrow & |k| & = & \|Qx\| = \|x\|, \quad k = \pm\|x\|. \\ & Qx & = & (I - 2ww^T)x = x - 2w(w^Tx) = ke_1 \\ \Rightarrow & w & = & \frac{x-ke_1}{2(w^Tx)} = c(x - ke_1) \quad (w \text{ ist Vielfaches vom Vektor } x - ke_1) \\ \overset{\|w\|=1}{\Rightarrow} & w & = & \frac{x-ke_1}{\|x-ke_1\|} \end{array}$$

An dieser Stelle ist lediglich das Vorzeichen von $k = \pm\|x\|$ noch unbekannt. Aus Stabilitätsgründen (Vermeidung von Auslöschung) wählen wir k in geeigneter Weise. Es ist

$$\|x - ke_1\| = \left((x_1 - k)^2 + x_2^2 + \ldots + x_n^2\right)^{1/2}.$$

Keine Auslöschung tritt auf für

$$\begin{array}{rl} & k = -sign(x_1)\|x\|, \quad (x_1 - k)^2 = (|x_1| + \|x\|)^2. \\ \Rightarrow & \|x - ke_1\|^2 = \|x\|^2 + 2\|x\|\,|x_1| + \|x\|^2 = 2\|x\|\,(\|x\| + |x_1|) \end{array}$$

Insgesamt erhalten wir

(3.32)

$$\boxed{\begin{array}{rcl} Q & = & I - 2ww^T = I - 2\frac{(x-ke_1)(x-ke_1)^T}{\|x-ke_1\|^2} \\ & = & I - \beta uu^T \\ k & = & -sign(x_1)\|x\|, \quad \beta = \frac{1}{\|x\|(|x_1|+\|x\|)} \\ u & := & x - ke_1 = \begin{pmatrix} sign(x_1)(|x_1| + \|x\|) \\ x_2 \\ \vdots \\ x_n \end{pmatrix} \end{array}}$$

Householder-Transformation

3.5.3 QR–Zerlegung/Verfahren von Householder

Zur Zerlegung der Matrix A bilden wir die Sequenz

$$\begin{array}{l} A = A^{(1)} \to A^{(2)} \to \ldots \to A^{(n)} = R, \\ A^{(j+1)} = Q_j A^{(j)},\ Q_j \text{ orthogonal.} \end{array}$$

j-ter Schritt ($j \geq 1$): Sei

$$A^{(j)} = \left(\begin{array}{ccc|ccc} * & \cdots & * & * & \cdots & * \\ & \ddots & \vdots & \vdots & & \vdots \\ 0 & & * & * & \cdots & * \\ \hline & & & a_{jj}^{(j)} & \cdots & a_{jn}^{(j)} \\ & 0 & & \vdots & & \vdots \\ & & & a_{nj}^{(j)} & \cdots & a_{nn}^{(j)} \end{array}\right) \begin{array}{l} \left.\begin{array}{l} \\ \\ \\ \end{array}\right\} j-1 \\ \left.\begin{array}{l} \\ \\ \\ \end{array}\right\} n-j+1 \end{array}$$

$$x := (a_{jj}^{(j)}, \ldots, a_{nj}^{(j)})^T \in \mathbb{R}^{n-j+1}.$$

1. Fall: $x = 0$: A ist singulär (Beweis!), setzt $Q_j = I$
2. Fall: $x \neq 0$: Bestimme nach (3.32) die orthogonale $(n-j+1, n-j+1)-$ Matrix $\tilde{Q}_j$ mit

$$\tilde{Q}_j \begin{pmatrix} a_{jj}^{(j)} \\ \vdots \\ a_{nj}^{(j)} \end{pmatrix} = k \begin{pmatrix} 1 \\ 0 \\ \vdots \\ 0 \end{pmatrix} \in \mathbb{R}^{n+1-j}.$$

Setzen wir nun jeweils

$$Q_j = \begin{pmatrix} I_{j-1} & 0 \\ 0 & \tilde{Q}_j \end{pmatrix} \qquad \in \mathbb{R}^{n \times n}, \quad \text{orthogonal, symmetrisch}$$

so erhalten wir nach n Schritten

$$R := A^{(n)} = Q_{n-1} Q_{n-2} \ldots Q_1 A. \tag{3.33}$$

Definieren wir die orthogonale Matrix

$$Q := (Q_{n-1} \cdots Q_1)^{-1} = Q_1 \cdots Q_{n-1}, \text{ (da } Q_j \text{ orthogonal, symmetrisch)}$$
$$\Rightarrow \; A = QR$$

Satz 3.40 (QR–Zerlegung). *Zu jeder (n,n)-Matrix A existiert eine orthogonale (n,n)-Matrix Q und eine rechte Dreiecksmatrix R mit*

$$A = QR.$$

Ist A regulär, so ist R regulär.

Bei einer regulären Matrix A bildet man zur Lösung des LGS $Ax = b$ analog zu (3.33) den Ausdruck

$$c := b^{(n)} = Q_{n-1} \cdots Q_1 b$$

und löst dann das gestaffelte LGS $Rx = c$.
Anzahl der Operationen: $\boxed{\approx \frac{2}{3}n^3}$ Jedoch sind keine zusätzlichen Permutationsmatrizen notwendig.

3.5.4 Erweiterungen

Die QR-Zerlegung kann unmittelbar auf nichtquadratische (m, n)-Matrizen A $(m > n)$ erweitert werden. Hier bildet man eine Sequenz

$$\begin{aligned} A^{(j+1)} &= Q_j A^{(j)} \qquad (j \geq 1),\ A^{(1)} = A \\ & \quad Q_j \text{ orthogonale } (m,m)\text{-Matrix.} \end{aligned}$$

Wegen $m > n$ erhält man nach n Schritten

(3.34)
$$\boxed{\begin{aligned} A^{(n+1)} &= \tilde{Q}A = \underbrace{\begin{pmatrix} R \\ \hline 0 \end{pmatrix}}_{n} \begin{matrix} \}n \\ \}m-n \end{matrix} \\ \tilde{Q} &= Q_n \dots Q_1 \quad \text{orthogonal,} \\ R &= \begin{pmatrix} r_{11} & \cdots & r_{1n} \\ & \ddots & \vdots \\ 0 & & r_{nn} \end{pmatrix} \quad \text{obere Dreiecksmatrix} \end{aligned}}.$$

Praktische Durchführung mit (3.32)

$$A^{(j)} = \underbrace{\left(\begin{array}{c|c} * & * \\ \hline 0 & \tilde{A}^{(j)} \end{array}\right)}_{n} \begin{matrix} \}j-1 \\ \}m-j+1 \end{matrix}, \quad A^{(1)} = A$$

$$Q_j = \underbrace{\begin{pmatrix} I_{j-1} & 0 \\ 0 & \tilde{Q}_j \end{pmatrix}}_{m} \begin{matrix} \}j-1 \\ \}m-j+1 \end{matrix}$$

$$\tilde{Q}_j = I - \beta_j u_j u_j^T, \quad j = 1, \dots, n,$$

wobei nach (3.32) gilt:

$$\begin{aligned} x_j &= (a_{jj}^{(j)}, \ldots, a_{mj}^{(j)})^T \in \mathbb{R}^{m-j+1}, \\ k_j &= -sign((x_j)_1)\|x_j\|, \\ \beta_j &= \frac{1}{\|x_j\|(|x_{j1}|+\|x_j\|)}, \\ u_j &= x_j - k_j e_j. \\ \tilde{Q}_j \tilde{A}^{(j)} &= \tilde{A}^{(j)} - u_j s_j^T \\ s_j^T &= \beta_j u_j^T \tilde{A}^{(j)}. \end{aligned}$$

$$\text{d.h. } (u_j s_j^T)_{i,k} = a_{ij}\beta_j \sum_{l=j}^{m} a_{lj} a_{lk}$$

Programm QR(A,d)
Die u_j stehen spaltenweise im linken Teil von A, $R\backslash diag(R)$ steht im rechten Teil von A, $diag(R)$ steht auf $d = (d_1, \ldots, d_n)$.

für $j = 1, \ldots, n$:

$$xnorm = \left(\sum_{i=j}^{m} a_{ij}^2 \right)^{1/2}$$

falls $xnorm$=0: $STOP$

$d_j = -sign(a_{jj}) \cdot xnorm$

$beta = 1/(xnorm(|a_{jj}| + xnorm))$

$a_{jj} = a_{jj} - d_j$

für $k = j+1, \ldots, n$:

$$s = beta \cdot \sum_{i=j}^{m} a_{ij} a_{ik}$$

für $i = j, \ldots, m$:

$$a_{ik} = a_{ik} - a_{ij} \cdot s.$$

3.6 Lineare Ausgleichsrechnung, diskrete Approximation

3.6.1 Normalgleichung

Ausgleichrechnungen sind für viele praktische Zwecke besonders wichtig.

Beispiel 3.41. *Wir haben bei einem Experiment für die Eingabewerte $t_1, \ldots, t_m$ Messwerte $s_1, \ldots, s_m$ erhalten. Aufgrund theoretischer Überlegungen (etwa physikalische Gesetze) kennt man eine Funktion $f(t)$, für die $f(t_i) = s_i$ gelten soll.*

Die Funktion f hängt aber in der Regel von unbekannten Parametern $x_1, \ldots, x_n$ ab; wir schreiben $f(t; x)$ für $x = (x_1, \ldots, x_n)^T$ um dies zu betonen. Beispielsweise könnte f durch eine Parabel

$$f(t; x) = x_1 + x_2 t + x_3 t^2 \tag{3.35}$$

gegeben sein. In der Regel hat man mehr Messwerte als Parameter ($m > n$) und es liegen Messfehler vor, so dass der naheliegende Versuch durch Lösen des m dimensionalen (i.A. nichtlinearen) Gleichungssystems

$$\begin{aligned} f(t_1; x) &= s_1 \\ &\vdots \\ f(t_m; x) &= s_m \end{aligned} \tag{3.36}$$

den n dimensionalen Lösungsvektor zu bestimmen scheitern muß.

In vielen praktischen Fällen ist das Gleichungssystem (3.36) linear, so z.B. auch für die obige Parabel und wir erhalten aus (3.36) ein überbestimmtes (i.A. nicht lösbares) LGS

$$\boxed{Cx = s},$$

mit $C \in \mathbb{R}^{m \times n}$, $m \geq n$ und $s \in \mathbb{R}^m$.

Beispiel 3.42. *Für unser Beispiel einer Parabel in (3.35) erhalten wir etwa*

$$C = \begin{pmatrix} 1 & t_1 & t_1^2 \\ \vdots & \vdots & \vdots \\ 1 & t_m & t_m^2 \end{pmatrix} \quad \textit{und} \quad s = \begin{pmatrix} s_1 \\ \vdots \\ s_m \end{pmatrix}.$$

Anstelle also den (vermutlich vergeblichen Versuch) zu unternehmen, eine exakte Lösung x des Systems $Cx = s$ zu finden, begnügen wir uns damit, ein x zu finden, so daß Cx 'möglichst nahe' bei s liegt. Als Ersatzproblem betrachtet man das Optimierungsproblem

$$\min_{x \in \mathbb{R}^n} \|s - Cx\|_2^2 \tag{3.37}$$

Um eine Lösung von (3.37) zu bestimmen, bildet man die erste Ableitung, die in einem Minimum zwangsweise gleich Null sein muß.

$$0 = \frac{\partial \|s - Cx\|_2^2}{\partial x} = \frac{\partial \left[(s - Cx)^T (s - Cx)\right]}{\partial x} = 2C^T Cx - 2C^T s.$$

Dies ergibt die *Normalgleichung*

$$\boxed{C^T C x = C^T s}, \tag{3.38}$$

die nach Definition von $A := C^T C$ und $b := C^T s$ dem LGS

$$Ax = b$$

entspricht.
Für eine beliebige Matrix C ist die Lösung von (3.38) nicht eindeutig und es gilt:

Satz 3.43. *Das lineare Ausgleichsproblem (3.37) besitzt mindestens eine Lösung x_0, d.h. $C^T C x_0 = C^T s$.*

Beweisidee:

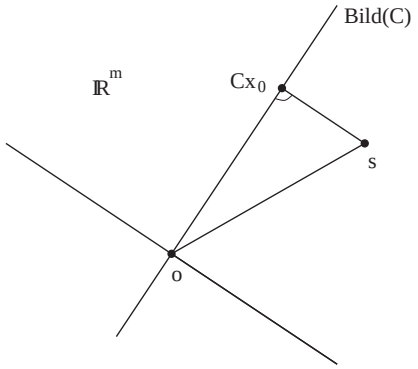


Abbildung 3.5: Lösung im Unterraum.

Nach linearer Algebra gilt die Zerlegung $\mathbb{R}^m = Bild(C) \oplus Kern(C^T)$. Daher kann $s \in \mathbb{R}^m$ zerlegt werden in

$$s = y + r, \quad y = Cx_0, \quad C^T r = 0,$$

und es folgt

$$C^T s = C^T y = C^T C x_0,$$

d.h. x_0 ist eine Lösung des linearen Ausgleichsproblems. Wegen $Kern(C^T C) = Kern(C)$ prüft man nun leicht nach, dass

$$x_0 + Kern(C)$$

die Gesamtheit der Lösungen ist:

$$\begin{aligned} & C^T C(x_0 + Kern(C)) &&= C^T s \\ \Leftrightarrow \quad & C^T C(x_0 + w) &&= C^T s \text{ mit } Cw = 0 \\ \Leftrightarrow \quad & C^T C x_0 &&= C^T s \end{aligned}$$

$\diamond$

3.6.2 Numerische Lösung

Von nun an sei rang $(C) = n < m$: Die Matrix C^TC ist dann positiv definit, und die Normalgleichung $C^TCx = C^Ts$ kann im Prinzip mit dem Cholesky–Verfahren gelöst werden. Dieses Verfahren hat jedoch zwei Nachteile:

(1) C^TC ist schwierig auszurechnen, z. B.:

$$C = \begin{pmatrix} 1 & 1 \\ \varepsilon & 0 \\ 0 & \varepsilon \end{pmatrix}, \qquad C^TC = \begin{pmatrix} 1+\varepsilon^2 & 1 \\ 1 & 1+\varepsilon^2 \end{pmatrix}.$$

Mit $\varepsilon = \frac{1}{2}\sqrt{eps}$ ist auf der Maschine

$$rd(C^TC) = \begin{pmatrix} 1 & 1 \\ 1 & 1 \end{pmatrix} \text{ singulär.}$$

(2) Die Kondition und damit die Empfindlichkeit gegenüber Störungen in C, y beträgt

$$cond(C^TC).$$

Beide Nachteile können mit der Householder–Transformation vermieden werden. Nach (3.34) gibt es eine QR-Zerlegung mit

$$QC = \underbrace{\begin{pmatrix} R \\ \hline 0 \end{pmatrix}}_{n} \begin{matrix} \} \ n \\ \} \ m-n \end{matrix}$$

R obere reguläre (n, n)-Dreiecksmatrix
Q orthogonale (m, m)-Matrix

Mit

$$Qs = \begin{pmatrix} h_1 \\ h_2 \end{pmatrix}, h_1 \in \mathbb{R}^n, \ h_2 \in \mathbb{R}^{m-n}$$

berechnet man

$$\begin{aligned} \|s - Cx\|_2^2 &= \|Q(s - Cx)\|_2^2 \\ &= \|h_1 - Rx\|_2^2 + \|h_2\|_2^2 \, . \end{aligned}$$

Dieser Ausdruck wird minimal für $x \in \mathbb{R}^n$ mit $Rx = h_1$. Die Lösung des linearen Ausgleichsproblems ist also

$$\boxed{x = R^{-1}h_1, \ \|s - Cx\|_2 = \|h_2\|_2} \, . \tag{3.39}$$

Die Kondition bei Anwendung der Householder-Transformation beträgt i.W. $cond_2(R)$; (vergleiche Stoer I, §4.8.3).

3.6.3 Diskrete Approximation

Als eine Anwendung der linearen Ausgleichsrechnung betrachten wir die diskrete Approximation.
Zu $n+1$ Basisfunktionen

$$f_0(t), f_1(t), \ldots, f_n(t)$$

und $m \geq n+2$ Meßpunkten

$$(t_i, s_i), \quad i = 1, \ldots, m$$

wird eine Linearkombination

$$f(t) = \sum_{k=0}^{n} x_k f_k(t)$$

gesucht, die die Werte s_i in den Punkten t_i möglichst gut annähert. Dies führt (vergleichbar zu linearen Ausgleichsrechnung) auf das Optimierungsproblem

$$\min_{x \in \mathbb{R}^{n+1}} \sum_{i=1}^{m} \left(s_i - \sum_{k=0}^{n} x_k f_k(t_i) \right)^2$$

also zu einem Problem der Form (3.37) mit

$$C = \begin{pmatrix} f_0(t_1) & \cdots & f_n(t_1) \\ \vdots & & \vdots \\ f_0(t_m) & \cdots & f_n(t_m) \end{pmatrix}$$

und $x = (x_0, \ldots, x_n)^T$, $s = (s_1, \ldots, s_m)^T$. Für die Basisfunktionen $f_k(t) = t^k, \quad k = 0, \ldots, n$, ist

$$C = \begin{pmatrix} 1 & t_1 & t_1^2 & \ldots & t_1^n \\ \vdots & \vdots & \vdots & & \vdots \\ 1 & t_m & t_m^2 & \ldots & t_m^n \end{pmatrix}$$

die Vandermonde–Matrix, für die gilt:

$$rang(C) = n+1, \quad \text{falls } t_i \neq t_j \quad \text{für } i \neq j.$$

Also ist $C^T C$ positiv definit und die Lösunge x ist eindeutig bestimmt.

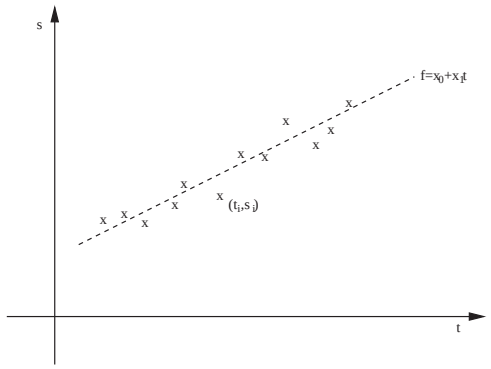


Abbildung 3.6: Ausgleichsgrade.

Beispiel 3.44. *Der Fall $n = 1$: (Ausgleichsgerade)*
Die Normalgleichungen für das Problem

$$\min_{x_0,x_1} \sum_{i=1}^{m} (s_i - (x_0 + x_1 t_i))^2$$

lauten

$$x_0 m + x_1 \sum_{i=1}^{m} t_i = \sum_{i=1}^{m} s_i$$

$$x_0 \sum_{i=1}^{m} t_i + x_1 \sum_{i=1}^{m} t_i^2 = \sum_{i=1}^{m} s_i t_i$$

und können explizit nach x_0, x_1 aufgelöst werden. In der Statistik spricht man dabei von Regressionsrechnung.

Beispiel 3.45. *Der Fall $n = 2$: (Ausgleichsparabel)*
Zu den Meßpunkten:

i	*1*	*2*	*3*	*4*	*5*	*6*	*7*
t_i	*0.04*	*0.32*	*0.51*	*0.73*	*1.03*	*1.42*	*1.60*
s_i	*2.63*	*1.18*	*1.16*	*1.54*	*2.65*	*5.41*	*7.67*

erhält man die Ausgleichsparabel (Schwarz, S.288)

$$\begin{aligned} f(t) &= x_0 + x_1 t + x_2 t^2, \\ x_0 &= 2.749198 \\ x_1 &= -5.954657 \\ x_2 &= 5.607247 \end{aligned}$$

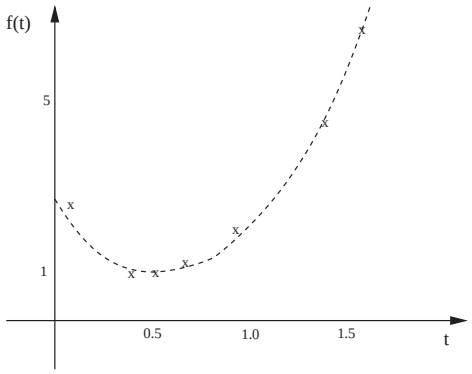


Abbildung 3.7: Ausgleichsparabel.

Kapitel 4

Nichtlineare Gleichungen und Gleichungssysteme

4.1 Einführung und Aufgabenstellung

Die Berechnung von Nullstellen nichtlinearer Gleichungssysteme bildet eine natürliche Erweiterung der LGS aus dem vorhergehenden Kapitel. Nichtlineare Gleichungen und Gleichungssysteme müssen in vielen Anwendungen der Mathematik gelöst werden. Typischerweise werden die Lösungen nichtlinearer Gleichungen über die Nullstellen einer Funktion $f : \mathbb{R}^n \to \mathbb{R}^n$ definiert, für die dann ein $x^* \in \mathbb{R}^n$ mit

$$f(x^*) = 0 \tag{4.1}$$

gesucht wird.

Beispiel 4.1.

1. *Polynome:*

   $$f(x) = a_0 + a_1 x + a_2 x^2 + \ldots + a_n x^n = 0, \quad x \in \mathbb{R}$$

   Z.B. sind Eigenwerte von Matrizen Nullstellen des charakteristischen Polynoms.

2. *Bei der Berechnung der Schwingungen eines Balkens tritt das Problem*

   $$f(x) = x - \tan(x) = 0$$

   auf. In den Intervallen $((k - 1/2)\pi, (k + 1/2)\pi)$ *liegen Nullstellen.*

3. *Bei der Lösung nichtlinearer Optimierungsprobleme.*

4. *Gleichgewichtslösungen chemischer Prozesse.*

4.2 Grundlagen

4.2.1 Fixpunkte

Fixpunktgleichungen lassen sich an vielen Stellen leichter analysieren als die Nullstellen nichtlinearer Funktionen, tatsächlich sind aber beide Problemklassen ineinander überführbar.
Sei $D \subset \mathbb{R}^n$ und $g : D \to \mathbb{R}^n$. Gesucht sind die Lösungen $x \in D$ der Gleichung

$$x = g(x). \tag{4.2}$$

Ein Punkt $x^* \in D$ heißt *Fixpunkt von* g, wenn $x^* = g(x^*)$ gilt.

Bemerkung 4.2. *Durch Definition von* $f(x) := x - g(x)$ *wird eine Fixpunktgleichung in eine Nullstellenberechnung überführt. Ist umgekehrt* $A(x)$ *eine reguläre* (n, n)*–Matrix (z.B. die Einheitsmatrix),* $x \in D$*, dann ist die Nullstellenbestimmung*

$$f(x) = 0$$

äquivalent zur Fixpunktgleichung

$$x = g(x) := x + A(x)f(x).$$

Für gegebene Startwerte $x^0, x^1, \ldots, x^s$, $s \geq 0$ werden Fixpunkte mit *Iterationsverfahren* der Form

$$\boxed{x^{k+1} = \psi(x^k, x^{k-1}, \ldots, x^{k-s}), \quad k \geq s} \tag{4.3}$$

bestimmt.
ψ heißt *Iterationsfunktion* und hängt von g ab. Oft kann $\psi = g$ und $s = 0$ gewählt werden, so daß die Iteration dann lautet

$$\boxed{x^{k+1} = g(x^k), \quad k = 1, 2, 3 \ldots, \quad \text{für gegebenes } x^0 \in D.} \tag{4.4}$$

Es stellen sich die folgenden Fragen:

1. Wie findet man eine passende Iterationsfunktion?
2. Wie findet man passende Anfangspunkte?
3. Wann konvergiert die Folge gegen einen Fixpunkt?
4. Wie schnell konvergiert die Folge?

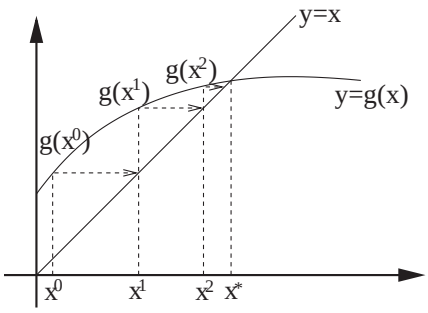


Abbildung 4.1: Graphische Darstellung eines Fixpunktes.

4.2.2 Konvergenz

Der Begriff der Konvergenzordnung erlaubt es, iterative Verfahren auf ihre Konvergenzgeschwindigkeit hin zu untersuchen.

Iterationsverfahren liefern eine Folge $\{x^k\} \subset \mathbb{R}^n$ approximativer Lösungen, die gegen die exakte Lösung x^* konvergieren. Die Konvergenzordnung gibt an, wie schnell der Fehler $\|x^k - x^*\|$ gegen Null konvergiert.

Definition 4.3 (Konvergenzordnung). *Sei $\|.\|$ eine Norm für $\mathbb{R}^n$ und sei $\{x^k\} \subset \mathbb{R}^n$ eine aus einem Iterationsverfahren entstandene Folge mit*

$$x^* = \lim_{k\to\infty} x^k.$$

1. *Existiert eine Konstante $c \in (0,1)$, so daß*

$$\frac{\|x^{k+1} - x^*\|}{\|x^k - x^*\|} \leq c, \qquad \forall \quad k = 0, 1, 2, \ldots \tag{4.5}$$

   gilt, so heißt $\{x^k\}$ linear konvergent*. Das zugehörige Iterationsverfahren wird* linear konvergent *oder* konvergent von der Ordnung 1 *genannt.*

2. *Existieren Konstanten $c > 0$ und $p > 1$, so daß*

$$\frac{\|x^{k+1} - x^*\|}{\|x^k - x^*\|^p} \leq c, \qquad \forall \quad k = 0, 1, 2, \ldots \tag{4.6}$$

   gilt, so heißt $\{x^k\}$ konvergent von Grade p*. Das zugehörige Iterationsverfahren wird* konvergent von der Ordnung p *genannt. Im Sonderfall $p = 2$ spricht man auch von* quadratischer Konvergenz*.*

Bemerkung 4.4. *Aus der aus (4.6) hergeleiteten Schreibweise*

$$\|x^{k+1} - x^*\| \leq c\|x^k - x^*\|^p, \qquad \forall \quad k = 0, 1, 2, \ldots$$

ergibt sich für den Fehler $e_k := \|x^k - x^*\|$ *die Beziehung*

$$e_{k+1} \leq c e_k^p,$$

d.h. es ist

$$e_{k+1} = \mathcal{O}(e_k^p).$$

Dies verdeutlich nochmals den Begriff der Ordnung eines Verfahrens.

Bemerkung 4.5. *Im Sonderfall* $p = 1$ *(lineare Konvergenz) erhält man aus (4.4) die Abschätzung*

$$e_{k+m} \leq c^m e_k, \qquad m > 0. \tag{4.7}$$

$\{x_k\}$ mit einem unteren Index k bezeichnet.

Beispiel 4.6. *Für* $0 < q < 1$ *sei* x_k *der Abschnitt der geometrischen Reihe*

$$\begin{aligned} x_k &= \sum_{i=0}^{k} q^i \ , \\ \textit{mit}\quad x^* &= \lim_{k\to\infty} x_k = \tfrac{1}{1-q} \ , \\ &\lim_{k\to\infty} \tfrac{e_{k+1}}{e_k} = q \end{aligned}$$

$$\Rightarrow \quad p = 1 : \textit{lineare Konvergenz.}$$

4.3 Nichtlineare Gleichungen

In diesem Abschnitt betrachten wir den Sonderfall $D = \mathbb{R}$.

Hinweis: Zur Vermeidung von Mißverständnissen werden reelle Folgen $\{x_k\}$ mit einem unteren Index k bezeichnet.

4.3.1 Bisektionsverfahren

Wir suchen eine Nullstelle $x^* \in \mathbb{R}$ einer reellen stetigen Funktion $f : \mathbb{R} \to \mathbb{R}$, $f(x^*) = 0$. Das Bisektionsverfahren wird auch *Intervallhalbierungsverfahren* genannt. Der Name wird sofort aus dem Algorithmus ersichtlich.

Algorithmus 4.7. (Bisektionsverfahren) *Gegeben sei eine stetige Funktion* $f : \mathbb{R} \to \mathbb{R}$ *und Werte* $a < b$ *mit* $f(a) \cdot f(b) < 0$ *(d.h.* $f(a)$ *und* $f(b)$ *haben unterschiedliches Vorzeichen und damit existiert eine Nullstelle von* $f(x)$ *im Intervall* (a, b)*). Die gewünschte Genauigkeit sei durch ein* $\varepsilon > 0$ *gegeben.*

1. *Setze* $k = 0$ *(Zählindex) und* $a_0 = a$, $b_0 = b$.

2. *Setze* $x_k = a_k + (b_k - a_k)/2$ *(Intervallhalbierung).*

3. *Ist* $f(x_k) = 0$ *oder* $(b_k - a_k)/2 < \varepsilon$ *beende den Algorithmus.*

4. *Ist* $f(x_k)f(a_k) < 0$, *dann setze* $a_{k+1} = a_k$, $b_{k+1} = x_k$.
   Ist $f(x_k)f(a_k) > 0$, *dann setze* $a_{k+1} = x_k$, $b_{k+1} = b_k$.
   Setze $k = k + 1$ *und gehe zu Schritt 2.*

Man kann die Punkte a_k und b_k als Intervallgrenzen der Intervalle $[a_k, b_k]$ verstehen, mit denen die Nullstelle durch immer weitere Halbierung eingeschachtelt wird. Daher stammt der Name *Bisektion* (=Zweiteilung).
Die Auswahlbedingung der neuen Werte a_{k+1} und b_{k+1} stellt sicher, daß $f(a_{k+1})$ und $f(b_{k+1})$ unterschiedliches Vorzeichen haben, daher muß sich immer eine Nullstelle zwischen den Werten befinden (daher auch die Voraussetzung der Stetigkeit).
Vorteilig beim Bisektionsverfahren ist:

- Es funktioniert für allgemeine stetige Funktionen.
- Es liefert immer ein Ergebnis (globale Konvergenz), wenn man geeignete Startwerte finden kann.
- Die Anzahl der Schritte bis zur gewünschten Genauigkeit hängt nur von a und b ab, aber nicht von f.

Leider konvergiert das Verfahren (vgl. auch Beispiel) nur sehr langsam und wird daher in der Praxis so gut wie nie eingesetzt.

Bemerkung 4.8. *Wegen*

$$|b_{k+1} - a_{k+1}| \leq \frac{1}{2}|b_k - a_k|$$

folgt aus (4.7) lineare Konvergenz.

Beispiel 4.9. *Für* $f(x) = x - tan(x)$, *berechnet man mit dem Bisektionsverfahren eine Nullstelle von* f*:*

$$\begin{array}{rclrcl} a & = & 2, & b & = & 4.6 \\ f(a) & \approx & 4.18, & f(b) & \approx & -4.26 \end{array}$$

Damit

$$\begin{array}{rcrcrcl} & & x_5 & = & 4.47812 & , & f(x_5) = 2.87 \cdot 10^{-1} \\ & & x_{20} & = & 4.493410 & , & f(x_{20}) = -1.51 \cdot 10^{-5} \\ x^* & = & x_{100} & = & 4.49340946 & , & f(x_{100}) = -1.72294616 \cdot 10^{-10}. \end{array}$$

4.3.2 Newton–Verfahren

Wir betrachten ein weiteres Verfahren zur Nullstellenbestimmung einer gegebenen Funktion f. Im Gegensatz zum Bisektionsverfahren benötigen wir nicht nur die Funktion f sondern auch ihre erste Ableitung.

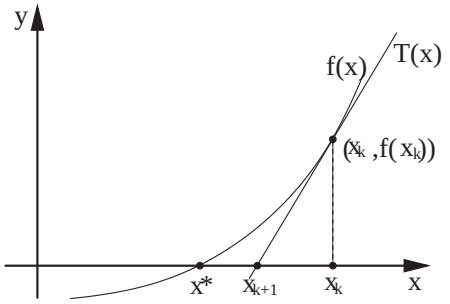


Abbildung 4.2: Zur Motivation des Newton–Verfahrens.

Sei x_k eine Näherung für x^*. Im Punkt $(x_k, f(x_k))$ wird eine Tangente

$$T(x) = f(x_k) + f'(x_k)(x - x_k)$$

an die Kurve $y = f(x)$ konstruiert und x_{k+1} als Nullstelle von T gewählt. Also

$$f(x_k) + f'(x_k)(x_{k+1} - x_k) = 0 \quad \Longleftrightarrow \quad x_{k+1} = x_k - \frac{f(x_k)}{f'(x_k)}$$

Eine Lösung existiert nur für $f'(x_k) \neq 0$. (Was bedeutet das anschaulich?)

Algorithmus 4.10. (Newton–Verfahren) *Gegeben sei eine Funktion $f : \mathbb{R} \to \mathbb{R}$, ihre Ableitung f, ein Anfangswert x_0 und eine gewünscht Genauigkeit $\varepsilon > 0$.*

1. *Berechne $x_{k+1} = x_k - \frac{f(x_k)}{f'(x_k)}$.*
2. *Ist $|x_{k+1} - x_k| < \varepsilon$, beende den Algorithmus,*
   sonst setze $k = k + 1$ gehe zu 1.

Bemerkung 4.11. *Das Newton–Verfahren ist eine Fixpunktiteration.*

Satz 4.12 (Konvergenz des Newton–Verfahrens). *Ist f zwei mal stetig differenzierbar, $f'(x) \neq 0$ und x_0 hinreichend nahe bei x^*, dann konvergiert das Newton–Verfahren quadratisch.*

Beweis: Später.

Beispiel 4.13. *Es sei* $f(x) = x^2 - 2$ *(d.h. Berechnung von* $x^* = \sqrt{2} \approx 1.414213562373$*) mit* $f'(x) = 2x$*. Die Iterationsvorschrift des Newton–Verfahrens ergibt hier*

$$x_{k+1} = x_k - \frac{x_k^2 - 2}{2x_k} = \frac{1}{2}x_k + \frac{1}{x_k}$$

Wir starten mit $x_0 = 1$ *und erhalten*

k	x_k	*Anzahl korrekter Dezimalstellen*
0	1.0	1
1	1.5	1
2	1.417	3
3	1.414216	6
4	1.414213562	10

Die schnelle Konvergenz belegt das theoretische Resultat einer quadratischen Konvergenz.

Anmerkung: *Newton hatte bereits 1669 ein Verfahren zur Berechnung einer Wurzel einer kubischen Gleichung entwickelt, das auf einen iterativen Linearisierungsprozeß hinausläuft. Er veröffentlichte sein Verfahren als Mittel zur Lösung der Keplerschen Gleichung:*

$$E = e \cdot \sin(E) + \frac{2\pi}{U}t$$

zur Bahnbestimmung von Planeten. Gesucht ist die 'exzentrische Anomalie' E bei einer Umlaufzeit U, einer Zeit (in Tagen) t seit dem Periheldurchgang und einer numerischen Exzentrizität e der Bahnellipse. Joseph Raphson brachte um 1690 die Newtonschen Überlegungen für Polynome auf eine Form, die der heutigen Darstellung näher kommt. Man spricht deshalb häufig vom Newton–Raphson–Verfahren.
Es ist unmittelbar einzusehen, daß das Newton–Verfahren nicht immer konvergiert (vgl. Abbildung 4.3).

Bemerkung 4.14. *Wesentlicher Nachteil des Newton–Verfahrens ist die Abhängigkeit von der Ableitung. Die Ableitung kann zwar auch numerisch berechnet werden, ist dann jedoch häufig anfällig gegenüber Rundungsfehlern.*

4.3.3 Sekanten–Verfahren

Zur Vermeidung von Ableitungen betrachten wir das Sekanten–Verfahren, hier ergibt sich die neue Näherung x_{k+1} nicht nur aus x_k sondern auch aus x_{k-1}.

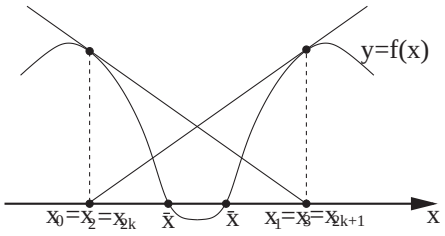


Abbildung 4.3: Keine Konvergenz beim Newton–Verfahren.

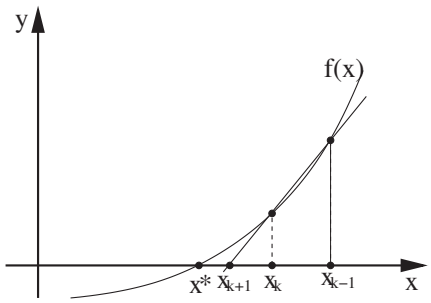


Abbildung 4.4: Zur Motivation des Sekanten–Verfahrens.

Die Sekantenmethode ist eine Vereinfachung des Newton–Verfahrens, wobei die Tangente durch die Sekante der letzten beiden Punkte ersetzt wird. Die Steigung der Sekante ergibt sich zu

$$\frac{f(x_k) - f(x_{k-1})}{x_k - x_{k-1}} \approx f'(x_k). \tag{4.8}$$

Für das Sekanten–Verfahren wird $f'(x_k)$ einfach durch (4.8) ersetzt.

Algorithmus 4.15. (Sekanten–Verfahren) *Gegeben sei eine Funktion $f : \mathbb{R} \to \mathbb{R}$, Anfangswerte x_0, x_1 und eine gewünscht Genauigkeit $\varepsilon > 0$. Setze $k = 1$.*

1. *Berechne $x_{k+1} = x_k - \frac{f(x_k)(x_k - x_{k-1})}{f(x_k) - f(x_{k-1})}$.*
2. *Ist $|x_{k+1} - x_k| < \varepsilon$, beende den Algorithmus, sonst setze $k = k + 1$ gehe zu 1.*

Bemerkung 4.16. *Das Sekanten–Verfahren ist eine Fixpunktiteration.*

Satz 4.17 (Konvergenz des Sekanten–Verfahrens). *Ist f zwei mal stetig differenzierbar, $f'(x^*) \neq 0$ und x_0, x_1 hinreichend nahe bei x^*, dann konvergiert das Sekanten–Verfahren mit der Ordnung $p = \frac{1}{2}(1 + \sqrt{5}) = 1.618\ldots$.*

Beweis: Später.

Beispiel 4.18. *Wir betrachten erneut $f(x) = x^2 - 2$. Die Iterationsvorschrift des Sekanten–Verfahrens ergibt hier*

$$x_{k+1} = x_k - \frac{x_k^2 - 2}{x_k + x_{k-1}}$$

Wir starten mit $x_0 = 1$, $x_1 = 2$ und erhalten

k	x_k	*Anzahl korrekter Dezimalstellen*
0	1.0	1
1	2.0	0
2	1.3	1
3	1.43	2
4	1.414	4
5	1.414211	6
6	1.4142135627	10

Das Sekanten–Verfahren startet zwar in diesem Beispiel recht langsam aufgrund der schlechten Startschätzung für x_1, konvergiert aber später entsprechend schnell.

4.4 Konvergenz von Iterationsverfahren

4.4.1 Kontraktion

Sei $D \subset \mathbb{R}^n$ und $g : D \to \mathbb{R}^n$. Wir untersuchen die Frage, wann die Fixpunktiteration

$$x^{k+1} = g(x^k), \quad k \geq 0, \quad x^0 \in D \text{ gegeben}, \tag{4.9}$$

wohl definiert ist und gegen einen Fixpunkt $\overline{x} \in D$, konvergiert.

Definition 4.19. *Die Abbildung $g : D \to \mathbb{R}^n$ heißt kontrahierend in D, falls es eine Zahl $0 \leq q < 1$ gibt mit*

$$\|g(x) - g(y)\| \leq q\|x - y\| \qquad \forall x, y \in D.$$

Für differenzierbare Abbildungen g kann ein einfaches Kriterium für die Kontraktion mit der Ableitung

$$g'(x) = \begin{pmatrix} \frac{\partial g_1}{\partial x_1} & \cdots & \frac{\partial g_1}{\partial x_n} \\ \vdots & & \vdots \\ \frac{\partial g_n}{\partial x_1} & \cdots & \frac{\partial g_n}{\partial x_n} \end{pmatrix}$$

gegeben werden. D heißt konvex, wenn für $x, y \in D$ gilt

$$\alpha x + (1-\alpha)y \in D \qquad \forall \alpha \in [0,1].$$

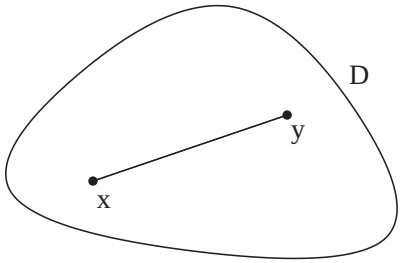


Abbildung 4.5: Konvexes Gebiet.

Satz 4.20. *Sei D konvex, $g : D \to \mathbb{R}^n$ differenzierbar und sei*

$$\sup_{x \in D} \| \, g'(x) \, \|_\infty \leq q < 1.$$

Dann ist g kontrahierend in D.

Beweis: Für zwei beliebige Punkte $x, y \in D$ betrachten wir $\varphi : [0,1] \to \mathbb{R}^n$:

$$\begin{aligned} \varphi(\lambda) &:= g(\lambda x + (1-\lambda)y), \quad \lambda \in [0,1], \\ \varphi(1) &= g(x), \quad \varphi(0) = g(y), \\ \varphi'(\lambda) &= g'(\lambda x + (1-\lambda)y)(x-y). \end{aligned}$$

Aus dem Mittelwertsatz folgt:

$$| \, \varphi_i(1) - \varphi_i(0) | \, \leq \max_{0 \leq \lambda \leq 1} | \, \varphi_i'(\lambda) \, | \, , \qquad i = 1, \ldots, n.$$

$$\begin{aligned} \Rightarrow \quad \|g(x) - g(y)\|_\infty &= \|\varphi(1) - \varphi(0)\|_\infty \\ &\leq \max_{0 \leq \lambda \leq 1} \|\varphi'(\lambda)\|_\infty \\ &= \max_{0 \leq \lambda \leq 1} \|g'(\lambda x + (1-\lambda)y)(x-y)\|_\infty \\ &\leq \sup_{z \in D} \|g'(z)\|_\infty \|x - y\|_\infty. \end{aligned}$$

◇

Für $n = 1$ ist $D = [a,b]$ konvex und $g \in C^1[a,b]$ kontrahierend, falls

$$\max_{a \leq x \leq b} |g'(x)| = q < 1.$$

(Vgl. Graphik zum Fixpunkt).

4.4.2 Fixpunktsatz von Banach

Satz 4.21 (Fixpunktsatz von Banach). *Sei D abgeschlossen und $g : D \to \mathbb{R}^n$ kontrahierend in D mit $g(D) \subseteq D$. Dann konvergiert die Folge*

$$x^{k+1} = g(x^k),\ k = 0, 1, 2, \ldots, \qquad x^0 \in D \text{ beliebig,}$$

gegen den eindeutig bestimmten Fixpunkt $\overline{x}$ von g in D und es gilt:

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

(ii) $\|\overline{x} - x^k\| \leq q\|\overline{x} - x^{k-1}\|$.

Beweis: Wir zeigen zunächst, dass $\{x^k\}$ eine Cauchy-Folge ist. Für $k \geq 1$ gilt

$$\|x^{k+1} - x^k\| = \|g(x^k) - g(x^{k-1})\| \leq q\|x^k - x^{k-1}\|$$

$$\leq q^2\|x^{k-1} - x^{k-2}\| \leq q^k\|x^1 - x^0\|$$

und damit für $j > l$

$$\begin{aligned} \|x^j - x^l\| &= \|\sum_{k=l}^{j-1}(x^{k+1} - x^k)\| \leq \sum_{k=l}^{j-1}\|x^{k+1} - x^k\| \\ &\leq \sum_{k=l}^{j-1} q^k\|x^1 - x^0\| \\ &\leq q^l\tfrac{1}{1-q}\|x^1 - x^0\| \underset{l\to\infty}{\longrightarrow} 0. \end{aligned} \tag{4.10}$$

Die x^k bilden damit eine Cauchy-Folge. Sei $\overline{x} \in D$ der Grenzwert der Folge x^k. Dann gilt

$$g(\overline{x}) \underset{k\to\infty}{\longleftarrow} g(x^k) = x^{k+1} \underset{k\to\infty}{\longrightarrow} \overline{x},$$

$$\Rightarrow \quad g(\overline{x}) = \overline{x}.$$

In (4.10) ergibt der Grenzwert für $j \to \infty$:

$$\|\overline{x} - x^l\| \leq \frac{q^l}{1-q}\|x^1 - x^0\|.$$

Mit $l = 1$ folgt hieraus nach Ersetzen $x^0 \to x^{k-1}$

$$\|\overline{x} - x^k\| \leq \frac{q}{1-q}\|x^k - x^{k-1}\|.$$

Damit ist (i) gezeigt; (ii) folgt aus der Kontraktionseigenschaft:

$$\|\overline{x} - x^k\| = \|g(\overline{x}) - g(x^{k-1})\| \leq q\|\overline{x} - x^{k-1}\|.$$

Zu zeigen bleibt noch die Eindeutigkeit von $\overline{x}$.
Seien $\overline{x}, x^*$ Fixpunkte von g:

$$\|\overline{x} - x^*\| = \|g(\overline{x}) - g(x^*)\| \leq q\|\overline{x} - x^*\|\,, \quad q < 1,$$
$$\Rightarrow \quad \|\overline{x} - x^*\| = 0.$$

$\diamond$

Bemerkung 4.22. *Wegen Teil (ii) konvergiert $\{x^k\}$ linear gegen $\overline{x}$.*

Bemerkung 4.23. *Die Schwierigkeiten bei der Anwendung des Kontraktionssatzes auf ein konkretes Problem bestehen darin:*

(a) man finde eine zugehörige kontrahierende Funktion $g : D \to \mathbb{R}^n$,

(b) man prüfe $g(D) \subseteq D$.

Beispiel 4.24. *Gesucht ist die Lösung $\overline{x}$ der Gleichung*

$$x = e^{-x} =: g(x), \quad x \in \mathbb{R}.$$

Auf das Intervall $D = [0.5, 0.69]$ trifft die Voraussetzung $g(D) \subset D$ zu. Als Kontraktionszahl q dient nach Satz 4.20 die Zahl

$$\max_{x \in D} |g'(x)| = e^{-0.5} = 0.606531 < 1.$$

Zum Startwert $x^{(0)} = 0.55 \in D$ berechnet man die Iterierten:

k	$x^{(k)}$	k	$x^{(k)}$	k	$x^{(k)}$
0	0.55000000	10	0.56708394	20	0.56714309
1	0.57694981	11	0.56717695	21	0.56714340
2	0.56160877	12	0.56712420	22	0.56714323
3	0.57029086	13	0.56715412	23	0.56714332
4	0.56536097	14	0.56713715	24	0.56714327

Mit der a priori Fehlerabschätzung $\|\overline{x} - x^{(k)}\| \leq \frac{q^k}{1-q}\|x^{(1)} - x^{(0)}\|$ kann die Anzahl k der Iteration geschätzt werden, die nötig sind, damit z. B. $\|\overline{x} - x^{(k)}\| \leq \varepsilon = 10^{-6}$ gilt. Man erhält

$$k \geq \log\left(\frac{\varepsilon(1-q)}{\|x^1 - x^0\|}\right) / \log q = 22.3,$$

eine gegenüber der Tabelle leichte Überschätzung. Für den Wert $x^{(12)}$ erhält man die a posteriori Fehlerschranke

$$\|\overline{x} - x^{(12)}\| \leq \frac{q}{1-q}\|x^{(12)} - x^{(11)}\| = 8.3 \cdot 10^{-5}.$$

Beispiel 4.25. *Erneut greifen wir das Beispiel* $f(x) = x - \tan(x)$ *auf.*
Die Nullstelle $\overline{x}$ *wird in* $D = [\pi, \frac{3}{2}\pi]$ *gesucht. Die Funktion* $g(x) = \tan x$ *ist nicht kontrahierend wegen*

$$g'(x) = \frac{1}{\cos^2 x} \geq 1.$$

Umformulierung:

$$x = \tan x = \tan(x - \pi) \Leftrightarrow \arctan x = x - \pi.$$

Setze nun

$$g(x) = \pi + \arctan x, \qquad D = [\pi, \frac{3}{2}\pi].$$

Offenbar gilt $g(D) \subseteq D$ *und*

$$q := \max_{x \in D} |g'(x)| = \frac{1}{1 + \pi^2} \approx 0.092 < 1.$$

g *ist also kontrahierend in* D *nach (4.20).*
Für $\overline{x} = 4.4934094$ *ist* $g'(\overline{x}) = 0.04719$.

k	x^k	$\frac{q}{1-q}\lvert x^k - x^{k-1}\rvert$	$\lvert\overline{x} - x^k\rvert$	$\frac{\lvert\overline{x}-x^k\rvert}{\lvert\overline{x}-x^{k-1}\rvert}$
0	*3.14159265*	–	–	–
1	*4.40421991*	*0.1351*	*0.0892*	–
2	*4.48911945*	*0.008918*	*0.0043*	*0.0672*
3	*4.49320683*	*0.0004280*	*0.0002*	*0.0481*
4	*4.4933999*	–	–	*0.0472*

4.4.3 Konvergenzsätze

Satz 4.26 (Lokaler Konvergenzsatz). *Sei* $g : \mathbb{R}^n \longrightarrow \mathbb{R}^n$ *mit* $g(\overline{x}) = \overline{x}$. *Ist* g *in einer Umgebung von* $\overline{x}$ *stetig differenzierbar und* $\|g'(\overline{x})\|_\infty < 1$, *dann gibt es eine Umgebung* D *von* $\overline{x}$, *so dass das Iterationsverfahren*

$$x^{k+1} = g(x^k), \qquad x^0 \in D$$

gegen $\overline{x}$ *konvergiert.*

Beweis: Sei D eine Kugel mit Radius r um $\overline{x}$ mit $\|g'(x)\|_\infty \leq q < 1$ für $x \in D$. Für $x \in D$ gilt

$$\|g(x) - \overline{x}\|_\infty = \|g(x) - g(\overline{x})\|_\infty \leq q\|x - \overline{x}\|_\infty \leq r$$

$$\Rightarrow \quad g(x) \in D.$$

Damit ist g kontrahierend in D und es gilt $g(D) \subseteq D$. Mit dem Fixpunktsatz 4.21 folgt die Behauptung. $\diamond$

Als Anwendung erhält man im Falle $n = 1$ einfache Kriterien dafür, dass die Fixpunkt-Iteration ein Verfahren p-ter Ordnung ist.

Satz 4.27. *Sei $g : \mathbb{R} \to \mathbb{R}$ eine C^p-Funktion mit $p \in \mathbb{N}_+$. Sei $\overline{x}$ ein Fixpunkt von g mit*

$$\begin{array}{ll} (a) & |g'(\overline{x})| < 1 \quad \text{für } p = 1, \\ (b) & g^{(i)}(\overline{x}) = 0 \quad (i = 1, \ldots, p-1) \text{ für } p > 1. \end{array}$$

Dann gibt es ein Intervall

$$I = [\overline{x} - \delta, \overline{x} + \delta], \quad \delta > 0,$$

so dass für alle $x_0 \in I$ die Iteration $x_{k+1} = g(x_k), k = 0, 1, 2, \ldots$ konvergent vom Grade p ist mit

$$\lim_{k\to\infty} \frac{|x_{k+1} - \overline{x}|}{|x_k - \overline{x}|^p} = \frac{1}{p!}\, g^{(p)}(\overline{x}).$$

Beweis: Aus den Vor. (a),(b) folgt insbesondere $|g'(\overline{x})| < 1$. Der lokale Konvergenzsatz 4.26 sichert dann die (mindestens) lineare Konvergenz der Folge $x_{k+1} = g(x_k)$ für alle $x_0 \in I = [\overline{x} - \delta, \overline{x} + \delta],\ \delta > 0$ geeignet.
Die Taylor-Entwicklung ergibt mit Vor. (b) und $\overline{x} = g(\overline{x})$:

$$x_{k+1} = \overline{x} + \frac{1}{p!}\, g^{(p)}(\overline{x})(x_k - \overline{x})^p + o(|x_k - \overline{x}|^p).$$

Hieraus folgt die Behauptung

$$\lim_{k\to\infty} \frac{|x_{k+1} - \overline{x}|}{|x_k - \overline{x}|^p} = \frac{1}{p!}\, g^{(p)}(\overline{x})$$

$\diamond$

4.4.4 Konvergenz des Newton–Verfahrens

Als Anwendung betrachten wir das Newton-Verfahren

$$x_{k+1} = g(x_k) = x_k - \frac{f(x_k)}{f'(x_k)}.$$

Ist f eine C^3-Funktion, so ist g eine C^2-Funktion.

1. Fall: $\overline{x}$ ist einfache Nullstelle von f, d.h. $f'(\overline{x}) \neq 0$: man berechnet

$$\begin{array}{rcl} g'(x) & = & \frac{f(x)f''(x)}{f'(x)^2}, \quad g'(\overline{x}) = 0, \\ g''(\overline{x}) & = & \frac{f''(\overline{x})}{f'(\overline{x})}. \end{array}$$

Also ist das Newton-Verfahren (mindestens) quadratisch konvergent mit der asymptotischen Fehlerkonstanten

$$c = \frac{1}{2}\,\frac{f''(\overline{x})}{f'(\overline{x})}.$$

2. Fall: $\overline{x}$ sei m-fache Nullstelle von f, d.h. $f^{(i)}(\overline{x}) = 0$ für $i = 0, \ldots, m-1$:

$$f(x) = (x - \overline{x})^m f_0(x), \quad f_0(\overline{x}) \neq 0$$

$$\Rightarrow \quad g'(\overline{x}) = 1 - \frac{1}{m}.$$

Für $m > 1$ ist daher $g'(\overline{x}) \neq 0$ und das Newton-Verfahren ist nur linear konvergent.
Für das modifizierte Newton-Verfahren

$$x_{k+1} = g(x_k) := x_k - m\frac{f(x_k)}{f'(x_k)}$$

gilt jedoch $g'(\overline{x}) = 0$, also hat man quadratische Konvergenz.

4.5 Das Newton–Verfahren im $\mathbb{R}^n$

4.5.1 Herleitung des Newton–Verfahrens

Gegeben sei eine C^1-Funktion $f : D \to \mathbb{R}^n$. Gesucht ist eine Nullstelle $\overline{x} \in D$ von f. Das Newton-Verfahren zur Berechnung von $\overline{x}$ ist die folgende Fixpunktiteration:

$$\boxed{x^{k+1} = x^k - (f'(x^k))^{-1} f(x^k), \quad k \geq 0, \quad x^0 \in D \text{ gegeben.}} \tag{4.11}$$

Das Newton-Verfahren im $\mathbb{R}^n$ läßt sich auf verschiedene Weise erklären:

(1) Verallgemeinerung des Newton–Verfahrens mittels Taylor-Entwicklung:
Es gilt

$$0 = f(\overline{x}) = f(x^k) + f'(x^k)(\overline{x} - x^k) + o(\|\overline{x} - x^k\|).$$

Vernachlässigt man $o(\|\overline{x} - x^k\|)$ und ersetzt den unbekannten Punkt $\overline{x}$ durch x^{k+1}, so erhält man

$$0 = f(x^k) + f'(x^k)(x^{k+1} - x^k)$$

und daraus (4.11).

(2) Anwendung des lokalen Konvergenzsatzes 4.26:
$f(\overline{x}) = 0$ gilt genau dann, wenn $\overline{x}$ Fixpunkt von

$$g(x) := x + A(x)f(x)$$

ist mit einer geeignet zu wählenden regulären (n, n) C^1-Matrix $A(x)$. Nach Satz 4.26 ist g kontrahierend, falls $\|g'(\overline{x})\|_\infty < 1$ ist. Wegen $f(\overline{x}) = 0$ gilt

$$g'(\overline{x}) = I + A(\overline{x})f'(\overline{x}).$$

Wählen wir nun

$$A(\overline{x}) = -(f'(\overline{x}))^{-1}$$

so ist $g'(\overline{x}) = 0$. Da $\overline{x}$ unbekannt ist setzen wir

$$A(x) = -(f'(x))^{-1}$$

d.h.

$$g(x) = x - (f'(x))^{-1}f(x).$$

Die Fixpunktiteration $x^{k+1} = g(x^k)$ ergibt gerade (4.11). Satz 4.26 sichert wegen $g'(\overline{x}) = 0$ die lokale Konvergenz.

Beispiel 4.28. *Gesucht ist die Lösung des Systems*

$$f(x) = \begin{pmatrix} f_1(x) \\ f_2(x) \end{pmatrix} = \begin{pmatrix} x_1^2 + x_2^2 - 1 \\ x_1 \end{pmatrix} = 0$$

Die Ableitung $f'(x)$ ist gegeben durch

$$f'(x) = \begin{pmatrix} 2x_1 & 2x_2 \\ 1 & 0 \end{pmatrix},$$

wodurch sich die Inverse zu

$$(f'(x))^{-1} = \begin{pmatrix} 0 & 1 \\ \frac{1}{2x_2} & -\frac{x_1}{x_2} \end{pmatrix}$$

ergibt. Die Iterationsvorschrift lautet somit

$$x^{k+1} = x^k - \begin{pmatrix} 0 & 1 \\ \frac{1}{2x_2^k} & -\frac{x_1^k}{x_2^k} \end{pmatrix} \begin{pmatrix} (x_1^k)^2 + (x_2^k)^2 - 1 \\ x_1^k \end{pmatrix}.$$

Mit $x^0 = (1,1)$ ergibt sich

k	x_1^k	x_2^k	*Max. Anzahl korrekter Dezimalstellen*
0	1	$\underline{1}$	0
1	$\underline{0}$	$\underline{1}.5$	1
2	$\underline{0}$	$\underline{1.0}8$	2
3	$\underline{0}$	$\underline{1.00}3$	3
4	$\underline{0}$	$\underline{1.00000}5$	6

4.5.2 Praktische Realisierung

Bemerkung 4.29. *Praktisch benutzt man in höheren Dimensionen das Newton-Verfahren in der Form*

$$f'(x^k)(x^{k+1} - x^k) = -f(x^k).$$

So muß anstatt der Invertierung von $f'(x^k)$ (n^3 Operation) nur noch ein LGS gelöst werden ($\frac{1}{3}n^3$ Operationen).

Beispiel 4.30. $f : \mathbb{R}^2 \to \mathbb{R}^2$,

$$f(x) = \begin{pmatrix} 10^4 x_1 x_2 - 1 \\ e^{-x_1} + e^{-x_2} - 1.0001 \end{pmatrix}, \qquad f'(x) = \begin{pmatrix} 10^4 x_2 & 10^4 x_1 \\ -e^{-x_1} & -e^{-x_2} \end{pmatrix}.$$

$$\begin{aligned} x^0 &= \begin{pmatrix} 0 \\ 1 \end{pmatrix}, \ f(x^0) = \begin{pmatrix} -1 \\ 0,36 \end{pmatrix}, \\ \overline{x} &= x^{13} = \begin{pmatrix} 1.0981595 \times 10^{-5} \\ 9.10614 \end{pmatrix}. \end{aligned}$$

Die Matrix

$$f'(\overline{x}) = \begin{pmatrix} 9.1 \times 10^4 & 0.11 \\ -1 & -1.1 \times 10^{-4} \end{pmatrix}$$

hat die Kondition

$$\|f'(\overline{x})\|_\infty \cdot \|(f'(\overline{x}))^{-1}\|_\infty = \mathcal{O}(10^9).$$

Bei der Berechnung von $f_2(x)$ entsteht Auslösung; $f_2(x)$ läßt sich in folgender Gestalt besser berechnen:

$$\begin{aligned} e^{-x_1} + e^{-x_2} - 1.0001 &= (e^{-x_1} - 1) + (e^{-x_2} - 10^{-4}) \\ &\approx (-x_1 + (x_1)^2/2) + (e^{-x_2} - 10^{-4}). \end{aligned}$$

4.5.3 Newton–Kantorovich

Zum Nachweis der lokalen quadratischen Konvergenz des Newton-Verfahrens (4.11) benötigen wir den folgenden Hilfssatz:

Hilfssatz 4.31. *Sei $D_0 \subset D$ konvex. Es gebe $\gamma > 0$ mit*

$$\|f'(x) - f'(y)\| \leq \gamma \|x - y\| \qquad \text{für } x, y \in D_0.$$

Dann gilt

$$\|f(x) - f(y) - f'(y)(x - y)\| \leq \frac{\gamma}{2}\|x - y\|^2 \qquad \forall\ x, y \in D_0$$

Beweis: Definiere die differenzierbare Funktion $\varphi : [0, 1] \to \mathbb{R}^n$ durch

$$\begin{aligned} \varphi(t) &:= f(y + t(x - y)), \qquad x, y \in D_0, \\ \varphi'(t) &= f'(y + t(x - y))(x - y) \in \mathbb{R}^n. \end{aligned}$$

Mit der Voraussetzung folgt

$$\begin{aligned} \|\varphi'(t) - \varphi'(0)\| &= \|(f'(y + t(x - y)) - f'(y))(x - y)\| \\ &\leq \gamma t \|x - y\|\ \|x - y\|. \end{aligned}$$

Es ist

$$\begin{aligned} \triangle &:= f(x) - f(y) - f'(y)(x - y) \\ &= \varphi(1) - \varphi(0) - \varphi'(0) = \int_0^1 (\varphi'(t) - \varphi'(0))dt, \\ \Rightarrow \qquad \|\triangle\| &\leq \int_0^1 \|(\varphi'(t) - \varphi'(0)\|dt \\ &\leq \gamma \|x - y\|^2 \int_0^1 t\ dt = \tfrac{\gamma}{2}\|x - y\|^2. \end{aligned}$$

◇

Satz 4.32 (Newton-Kantorovich). *Es sei eine offene Menge $D \subseteq \mathbb{R}^n$ gegeben, ferner eine konvexe Menge D_0 mit $\overline{D}_0 \subseteq D$ und $f : D \to \mathbb{R}^n$ sei eine für alle $x \in D_0$ differenzierbare und für alle $x \in D$ stetige Funktion.*
Für ein $x^0 \in D_0$ gebe es positive Konstanten $r, \alpha, \beta, \gamma, h$ mit:

$$\begin{aligned} S_r(x^0) &:= \{x \mid \|x - x^0\| < r\} \subseteq D_0, \\ h &:= \alpha\beta\gamma/2 < 1, \\ r &:= \alpha/(1 - h). \end{aligned}$$

$f(x)$ habe die Eigenschaften

(a) $\|f'(x) - f'(y)\| \le \gamma \|x - y\|$ *für alle* $x, y \in D_0$
(Lipschitz-Bedingung für f'*),*

(b) $f'(x)^{-1}$ *existiert und es gilt*
$\|(f'(x))^{-1}\| \le \beta$ *für alle* $x \in D_0$,

(c) $\|(f'(x^0))^{-1} f(x^0)\| \le \alpha$.

Dann gilt

(i) ausgehend von x^0 *ist jedes*

$$x^{k+1} = x^k - (f'(x^k))^{-1} f(x^k), \quad k \ge 0$$

wohldefiniert und es gilt $x^k \in S_r(x^0)$ *für alle* $k \ge 0$.

(ii) $\overline{x} = \lim\limits_{k\to\infty} x^k$ *existiert und es gilt*

$$\overline{x} \in \overline{S_r(x^0)} \text{ und } f(\overline{x}) = 0.$$

(iii) $\|\overline{x} - x^k\| \le \alpha \frac{h^{2^k-1}}{1-h^{2^k}}$ *für alle* $k \ge 0$.

Wegen $0 < h < 1$ *ist also das Newton-Verfahren mindestens quadratisch konvergent.*

Beweis: zu (i):
Zunächst wird $x^k \in S_r(x^0)$, $k \ge 0$ induktiv gezeigt. Für $k = 1$ ist

$$x^1 = x^0 - f'(x^0)^{-1} f(x^0) \;\Rightarrow\; \|x^1 - x^0\| \le \alpha < \frac{\alpha}{1-h} = r$$

Seien $x^0, \ldots, x^k \in S_r(x^0)$:

$$\begin{aligned} \|x^{k+1} - x^k\| &= \| - f'(x^k)^{-1} f(x^k)\| \le \beta \|f(x^k)\| \\ &= \beta \|f(x^k) \underbrace{- f(x^{k-1}) - f'(x^{k-1})(x^k - x^{k-1})}_{=0 \text{ nach Def.}}\| \\ &\le \tfrac{1}{2}\, \beta\, \gamma\, \|x^k - x^{k-1}\|^2 \quad \text{nach Hilfssatz 4.31} \end{aligned}$$

Hiermit zeigen wir nun induktiv

$$\|x^{k+1} - x^k\| \le \alpha h^{2^k-1}. \tag{4.12}$$

Für $k = 0$ ist dies bereits gezeigt (s.o.). Ist die Abschätzung für $k \ge 0$ richtig, so (wegen $h = \frac{1}{2}\alpha\beta\gamma$) auch für $k+1$, denn

$$\|x^{k+1} - x^k\| \le \frac{\beta\gamma}{2} \|x^k - x^{k-1}\|^2 \le \frac{\beta\gamma}{2} \alpha^2 h^{2^k-2} = \alpha h^{2^k-1}.$$

Nun folgt mit (4.12)

$$\begin{aligned}\|x^{k+1}-x^0\| &\leq \|x^{k+1}-x^k\|+\|x^k-x^{k-1}\|+\ldots+\|x^1-x^0\| \\ &\leq \alpha(1+h+h^3+h^7+\ldots+h^{2^k-1}) < \alpha/(1-h) = r\end{aligned}$$

und daher $x^{k+1} \in S_r(x^0)$.
Zu (ii) und (iii):
$\{x^k\}$ ist eine Cauchy-Folge, denn für $m \geq n$ hat man nach (4.12)

$$\begin{aligned}\|x^{m+1}-x^n\| &\leq \|x^{m+1}-x^m\|+\|x^m-x^{m-1}\|+\ldots+\|x^{n+1}-x^n\| \\ &\leq \alpha h^{2^n-1}(1+h^{2^n}+(h^{2^n})^2+\ldots) \\ &< \tfrac{\alpha h^{2^n-1}}{1-h^{2^n}} < \varepsilon\end{aligned}$$

für genügend großes $n \geq N(\varepsilon)$, da $0 < h < 1$. Also existiert

$$\lim_{k\to\infty} x^k =: \overline{x} \in \overline{S_r(x^0)},$$

und für $m \to \infty$ ergibt sich die Abschätzung (iii). Zu zeigen ist noch $f(\overline{x}) = 0$: Mit der Dreiecksungleichung und $\|x-y\| \leq r$ folgt aus (a) für $K := \gamma r + \|f'(x^0)\|$

$$\|f(x^k)\| = \|f'(x^k)(x^{k+1}-x^k)\| \leq K\|x^{k+1}-x^k\| \to 0 \text{ für } k \to \infty$$

$$\Rightarrow\ f(\overline{x}) = 0, \quad \text{da } f \text{ stetig in } \overline{x} \in D \text{ ist. } \diamond$$

$\diamond$

Bemerkung 4.33. *Sei $\overline{x}$ eine isolierte Nullstelle von f, so dass die Vor. (a),(b) in einer Umgebung D_0 von $\overline{x}$ erfüllt sind. Für x^0 in einer genügend kleinen Umgebung E (Einzugsbereich) von $\overline{x}$ ist dann α in Vor. (c) hinreichend klein, so dass die Vor. von Satz 4.32 erfüllt sind. Dann konvergiert $\{x^k\}$ quadratisch gegen $\overline{x}$ für $x^0 \in E$.*

Bemerkung 4.34. *Für eine C^2-Funktion f ist die Lipschitz-Bedingung in Vor. (a) erfüllt.*

4.5.4 Erweiterungen

4.5.4.1 Approximation von $f'(x)$ durch Differenzen

Die Berechnung von $f'(x)$ kann sehr zeitraubend bzw. explizit nicht möglich sein. Man kann $f'(x^k)$ z.B. numerisch approximieren, indem man $f(x)$ in der Nähe von x^k geeignet auswertet.

$$\left.\frac{\partial f_i(x)}{\partial x_j}\right|_{x=x^k} \approx \frac{f_i(x^k+he_j)-f_i(x^k)}{h}, \quad h > 0 \text{ klein}$$

Bemerkung 4.35. *Diese Methode erfordert n Berechnungen von f für jeden Schritt des Newton-Verfahrens und dies ist für n groß ungünstig.*

4.5.4.2 λ-Strategie, Modifiziertes Newton-Verfahren

Durch Einführung eines konvergenzerzeugenden Faktors $0 < \lambda \leq 1$ kann der Einzugsbereich des Newton–Verfahrens vergrößert werden. Sei

$$d^k = f'(x^k)^{-1} f(x^k) \in \mathbb{R}^n.$$

Gewöhnliches Newton-Verfahren:

$$x^{k+1} = x^k - d^k.$$

Modifiziertes Newton-Verfahren:

$$\boxed{x^{k+1} = x^k - \lambda_k d^k \quad 0 < \lambda_k \leq 1} \tag{4.13}$$

Zur Bestimmung der konvergenzerzeugenden Faktoren λ_k vergleiche man STOER, §5.4.

Kapitel 5

Interpolation

5.1 Einführung und Aufgabenstellung

Häufig tritt die Situation auf, daß statt einer Funktion nur einige diskrete Daten (x_j, f_j), $j = 0, \ldots, n$ gegeben sind (z.B. Messwerte eines Experimentes). Historisch trat das Problem bei der Berechnung von zusätzlichen Funktionswerten zwischen tabellierten Werten auf (z.B. für sin, cos, log). Heute ist es ein häufig auftretendes Problem sowohl in der Mathematik als auch in vielen Anwendungen.

Gesucht ist eine Funktion $\Phi : \mathbb{R} \to \mathbb{R}$, für die die Gleichungen

$$\Phi(x_j) = f_j, \qquad j = 0, \ldots, n \tag{5.1}$$

gelten. Sind z.B. $f_j = f(x_j)$ Messdaten einer unbekannten Funktion f, so soll Φ möglichst nahe an f liegen. Zudem soll Φ leicht auswertbar sein. Ohne Einschränkung verlangen wir $x_0 < x_1 < \ldots < x_n$.

Beispiel 5.1. *(Lineare Interpolationsprobleme)*

- *Interpolation durch Polynome*

$$\Phi(x) = \sum_{j=0}^{n} a_j x^j,$$

- *Trigonometrische Interpolation (Es gilt $-1 = i^2$, $e^{ix} = \cos x + i \sin x$)*

$$\Phi(x) = \sum_{j=0}^{n} a_j e^{ijx} = a_0 + \sum_{j=1}^{n} a_j \cos(jx) + i \sum_{j=1}^{n} a_j \sin(jx)$$

- *Kubische Spline–Interpolation*

$$\Phi(x) \in C^2[x_0, x_n] \quad \textit{und} \quad \Phi(x) \in C^3[x_j, x_{j+1}]$$

Beispiel 5.2. *(Nichtlineare Interpolationsprobleme)*

- *Rationale Interpolation*

$$\Phi(x) = \frac{\sum_{j=0}^{n} a_j x^j}{\sum_{j=0}^{m} b_j x^j}$$

- *Interpolation durch Exponentialsummen*

$$\Phi(x) = \sum_{j=0}^{n} a_j e^{\lambda_j x}$$

In diesem Kapitel werden wir uns mit linearen Interpolationsproblemen beschäftigen.

5.2 Polynominterpolation

5.2.1 Existenz und Eindeutigkeit der Polynominterpolation

Polynominterpolation ist eine einfache und effiziente Möglichkeit der Interpolation. Es bezeichne Π_n die Menge aller reellen oder komplexen Polynome vom Grade $\leq n$. Es gilt der

Satz 5.3 (Eindeutigkeit). *Zu beliebigen $n+1$ Stützstellen (x_j, f_j), $j = 0, \ldots, n$, $x_j \neq x_k$ für $j \neq k$ gibt es genau ein Polynom $P \in \Pi_n$*

$$P(x) = a_0 + a_1 x + \ldots + a_n x^n$$

mit

$$P(x_j) = f_j, \qquad j = 0, \ldots, n. \tag{5.2}$$

Beweis: Die Zerlegung von (5.2) ergibt das LGS

$$\sum_{k=0}^{n} a_k x_j^k = f_j, \qquad j = 0, \ldots, n,$$

d.h.

$$\begin{pmatrix} 1 & x_0 & \cdots & x_0^n \\ \vdots & \vdots & \ddots & \vdots \\ 1 & x_n & \cdots & x_n^n \end{pmatrix} \begin{pmatrix} a_0 \\ \vdots \\ a_n \end{pmatrix} = \begin{pmatrix} f_0 \\ \vdots \\ f_n \end{pmatrix} \tag{5.3}$$

Die Determinante der sogenannten *Vandermonde–Matrix* in (5.3) ist

$$\prod_{i=1}^{n}\left(\prod_{j=i+1}^{n}(x_i - x_j)\right)$$

und damit ungleich Null, wenn die x_i paarweise verschieden sind. Also ist die Matrix invertierbar und das Gleichungssystem eindeutig lösbar. ◇

Bemerkung 5.4. *Prinzipiell ist es somit möglich aus dem LGS die Koeffizienten zu bestimmen, jedoch mit dem Wissen eines Aufwandes von $O(n^3)$ Operationen recht teuer.*

Aus diesem Grund betrachten wir andere Techniken.

5.2.2 Interpolationsformel von Lagrange

Für die $n+1$ *Lagrange–Polynome* vom Grad n

$$L_i(x) := \prod_{\substack{j=0 \\ j\neq i}}^{n} \frac{x - x_j}{x_i - x_j}$$

gilt offensichtlich

$$L_i(x_j) = \begin{cases} 1, & \text{für } i = j \\ 0, & \text{für } i \neq j \end{cases}$$

Damit ist unser gesuchtes Polynom gegeben durch

$$\boxed{P(x) = \sum_{i=0}^{n} f_i L_i(x),} \qquad \text{da } P(x_j) = f_j.$$

Beispiel 5.5. *Wir betrachten die Daten* $(3, 68)$, $(2, 16)$, $(5, 352)$. *Die Lagrange–Polynome sind gegeben durch*

$$\begin{aligned} L_0(x) &= \frac{(x-2)(x-5)}{(3-2)(3-5)} = -\frac{1}{2}(x-2)(x-5) \\ L_1(x) &= \frac{(x-3)(x-5)}{(2-3)(2-5)} = \frac{1}{3}(x-3)(x-5) \\ L_2(x) &= \frac{(x-2)(x-3)}{(5-2)(5-3)} = \frac{1}{6}(x-2)(x-3) \end{aligned}$$

Damit erhalten wir

$$\begin{aligned} P(x) &= -68 \cdot \tfrac{1}{2}(x-2)(x-5) + 16 \cdot \tfrac{1}{3}(x-3)(x-5) + 352 \cdot \tfrac{1}{6}(x-2)(x-3) \\ &= 30x^2 - 98x + 92 \end{aligned}$$

Bemerkung 5.6. *Ein Abzählen der notwendigen Operationen ergibt den Aufwand $O(n^2)$.*

Die Lagrange–Darstellung ist praktisch, wenn man Messwerte f_i nachträglich verändern will, da diese explizit in der Polynomdarstellung auftauchen. Sie ist aber i.A. unpraktisch, wenn man Datenpunkte hinzufügen will, da man alle L_i neu berechnen muss.

5.2.3 Der Algorithmus von Aitken und Neville

Gesucht ist ein numerisch sparsamer Algorithmus zur Berechnung von $P(x)$ an einigen wenigen Stellen (z.B. nur einmalig) x.
Für $i_0, \ldots, i_k \in \{0, 1, \ldots, n\}$ sei $P_{i_0,\ldots,i_k} \in \Pi_k$ das Interpolationspolynom zu $x_{i_0}, \ldots, x_{i_k}$; insbesondere

$$P_i(x) \equiv f_i, \qquad P_{0,\ldots,n}(x) = P(x).$$

5.2.3.1 Rekursionsformel von Aitken

Es gilt die Rekursionsformel von Aitken

$$\boxed{P_{i_0,\ldots,i_k}(x) = \frac{(x-x_{i_0})P_{i_1,\ldots,i_k}(x) - (x-x_{i_k})P_{i_0,\ldots,i_{k-1}}(x)}{x_{i_k} - x_{i_0}}} \tag{5.4}$$

Beweis: Das Polynom $Q(x) \in \Pi_k$ auf der rechten Seite von (5.4) erfüllt

$$\begin{aligned} Q(x_{i_0}) &= P_{i_0,\ldots,i_{k-1}}(x_{i_0}) &&= f_{i_0} \\ Q(x_{i_k}) &= P_{i_1,\ldots,i_k}(x_{i_k}) &&= f_{i_k} \\ Q(x_{i_j}) &= f_{i_j},\ j = 1, \ldots, k-1. \end{aligned}$$

Wegen der Eindeutigkeit der Polynom-Interpolation folgt dann (5.4). ◇

5.2.3.2 Variante von Neville

Es sei x fest. Erzeuge die Werte $P_{i-k,\ldots,i}(x)$ $(k \le i)$ nach dem Schema

	$k=0$	1	2	3
x_0	$f_0 = P_0(x)$			
		$\nearrow\ P_{0,1}(x)$		
x_1	$f_1 = P_1(x)$		$\nearrow\ P_{0,1,2}(x)$	
		$\nearrow\ P_{1,2}(x)$		$\nearrow\ P_{0,1,2,3}(x)$
x_2	$f_2 = P_2(x)$		$\nearrow\ P_{1,2,3}(x)$	
		$\nearrow\ P_{2,3}(x)$		
x_3	$f_3 = P_3(x)$			

Bei Hinzunahme von weiteren Stützstellen wird das Schema einfach erweitert, ohne dass die bereits berechneten Ergebnisse ungültig werden.

Für festes x gilt mit der Bezeichnung

$$P_{i,k} := P_{i-k,\dots,i}(x) \qquad (k \le i):$$

Startwerte: $P_{i,0} := f_i, \qquad i = 0, \dots, n,$
Rekursion:

$$\begin{aligned} P_{i,k} &= \frac{(x-x_{i-k})P_{i,k-1}-(x-x_i)P_{i-1,k-1}}{x_i-x_{i-k}}, \\ &= P_{i,k-1} + \frac{P_{i,k-1}-P_{i-1,k-1}}{\frac{x-x_{i-k}}{x-x_i}-1}, \qquad k = 1, \dots, i, \\ & \qquad\qquad\qquad\qquad\qquad\qquad\qquad\qquad i = 1, 2, \dots \end{aligned} \tag{5.5}$$

(Durch die Umformulierung wird eine effiziente Berechnung ermöglicht, da die Berechnungen $x_i - x_{i-k}$ beim Hinzufügen einer Stützstelle nicht mehr ausgewertet werden müssen.) Resultat: $P_{n,n} = P(x)$.

Das obige Schema lautet dann

$$\begin{array}{c|ccccc} x_0 & f_0 = P_{0,0} & & & & \\ & & P_{1,1} & & & \\ x_1 & f_1 = P_{1,0} & & P_{2,2} & & \\ & & P_{2,1} & & \nearrow & P_{3,3} \\ x_2 & f_2 = P_{2,0} & & \nearrow & P_{3,2} & \\ & & \nearrow & P_{3,1} & & \\ x_3 & f_3 = P_{3,0} & & & & \end{array}$$

Diese Variante wird speziell für $x = 0$ bei sogenannten Extrapolationsalgorithmen angewandt.
Wird der Wert von f bis zu einer bestimmten Genauigkeit gesucht, dann werden einfach weitere Stützstellen solange hinzugefügt, bis der berechnete Wert mit der gewünschten Genauigkeit übereinstimmt.

5.2.4 Die Newton'sche Interpolationsformel, Dividierte Differenzen

Es sollen die Koeffizienten des Interpolationspolynoms $P(x)$ berechnet werden. Für $P = P_{0,\dots,n}$ macht man den Ansatz

$$\begin{aligned} P(x) &= a_0 + a_1(x - x_0) + a_2(x - x_0)(x - x_1) + \cdots \\ &\quad + a_n(x - x_0) \cdots (x - x_{n-1}). \end{aligned} \tag{5.6}$$

Die Auswertung dieses Ausdrucks für festes x geschieht mit dem Horner-Schema:

$$P(x) = [\ldots(a_n(x - x_{n-1}) + a_{n-1})(x - x_{n-2}) + \ldots + a_1](x - x_0) + a_0$$

d.h. rekursiv

$$\begin{aligned} b_n &= a_n, \\ b_i &= b_{i+1}(x - x_i) + a_i, \qquad i = n-1, \ldots, 0, \\ P(x) &= b_0. \end{aligned}$$

Für die Abschnittspolynome

$$Q_k(x) = \sum_{i=0}^{k} a_i \prod_{j=0}^{i-1}(x - x_j)$$

folgt

(a) $Q_k(x) = P_{0,1\ldots,k}(x), \qquad k = 0, \ldots, n,$

(b) a_k ist Koeffizient von x^k in $P_{0,1,\ldots,k}(x)$.

Die Koeffizienten a_k werden nicht mittels der Bedingung $P(x_j) = f_j$ berechnet, sondern mit Hilfe des Differenzenschemas:

$$\begin{array}{l|lll} x_0 & [f_0] & & \\ & & \searrow & \\ & & \nearrow & [f_0, f_1] \\ x_1 & [f_1] & & \searrow \\ & & \searrow & \nearrow & [f_0, f_1, f_2] \\ & & \nearrow & [f_1, f_2] & \\ x_2 & [f_2] & & \end{array}$$

Die hier auftretenden "Dividierten Differenzen" $[f_i, \ldots, f_k]$ sind rekursiv definiert durch

(a) $[f_i] = f_i,$

(b) $[f_i, \ldots, f_k] = \frac{[f_{i+1}, \cdots, f_k] - [f_i, \ldots, f_{k-1}]}{x_k - x_i},$

z.B.

$$\begin{aligned} [f_0, f_1] &= \tfrac{f_1 - f_0}{x_1 - x_0} \\ [f_0, f_1, f_2] &= \tfrac{[f_1, f_2] - [f_0, f_1]}{x_2 - x_0}. \end{aligned}$$

Satz 5.7. *Es gilt:*

$$P_{0,\ldots,k}(x) = \sum_{i=0}^{k} [f_0, \ldots, f_i] \prod_{j=0}^{i-1}(x - x_j).$$

Beweis: Durch Induktion bzgl. k: Für $k = 0$ ist die Behauptung offensichtlich richtig: Die Aussage gelte für $k - 1 \geq 0$.

$$P_{0,\ldots,k}(x) = P_{0,\ldots,k-1}(x) + a(x - x_0)\cdots(x - x_{k-1}).$$

Zu zeigen ist $\quad a = [f_0, \ldots, f_k]$.

Koeffizient von	x^k	in	$P_{0\ldots,k}(x)$	:	a
Koeffizient von	x^{k-1}	in	$P_{0\ldots,k-1}(x)$	:	$[f_0, \ldots, f_{k-1}]$
Koeffizient von	x^{k-1}	in	$P_{1\ldots,k}(x)$	:	$[f_1, \ldots, f_k]$

(nach Induktionsvoraussetzung)

Nach der Formel von Aitken (5.4) ist

$$P_{0,\ldots,k}(x) = \frac{(x - x_0)P_{1,\ldots,k}(x) - (x - x_k)P_{0,\ldots,k-1}(x)}{x_k - x_0}.$$

Der Koeffizient von x^k auf der rechten Seite ist

$$a = \frac{[f_1, \ldots, f_k] - [f_0, \ldots, f_{k-1}]}{x_k - x_0} = [f_0, \ldots, f_k].$$

$\uparrow$
Definition ◇

Beispiel 5.8. *(vgl. Lagrange-Form)*
Das Differenzenschema lautet

$$\begin{array}{lllll}
[f_0] = \boxed{68} & & & & \\
& \searrow\nearrow & [f_0, f_1] = \frac{16-68}{2-3} = \boxed{52} & & \\
[f_1] = 16 & & & \searrow\nearrow & [f_0, f_1, f_2] = \frac{52-112}{3-5} = \boxed{30} \\
& \searrow\nearrow & [f_1, f_2] = \frac{16-352}{2-5} = 112 & & \\
[f_2] = 352 & & & &
\end{array}$$

Damit erhalten wir das Interpolationspolynom (vgl. (5.5))

$$P(x) = P_{0,2}(x) = 68 + 52(x - 3) + 30(x - 3)(x - 2) = 30x^2 - 98x + 92$$

Satz 5.9. *Für eine beliebige Permutation $i_0, \ldots, i_n$ von $0, \ldots, n$ gilt*

$$[f_{i_0}, \ldots, f_{i_n}] = [f_0, \ldots, f_n].$$

Bemerkung 5.10. *Die Berechnung der dividierten Differenzen kann offensichtlich mit $O(n^2)$ Rechenoperationen erfolgen. Die Bestimmung des Polynoms P hingegen benötigt nur $O(n)$ Operationen, wenn man zur Auswertung das* Horner–Schema *verwendet.*

Mit dem Newton–Schema können leicht zusätzliche Datenpunkte mit in die Interpolation aufgenommen werden. Außerdem erlaubt das Horner–Schema eine effiziente Berechnung des Polynoms, wenn selbiges sehr oft für verschiedene x auszuwerten ist.

5.2.5 Interpolationsfehler

Wir betrachten den Fehler $f(x) - P(x)$ zwischen einer Funktion f und dem Interpolationspolynom P zu den Stützstellen (x_j, f_j), $f_j = f(x_j)$, $j = 0, \ldots, n$. Es gilt der

Satz 5.11. *Sei $f \in C^{n+1}[a,b]$ und $\bar{x}, x_0, \ldots, x_n \in [a,b]$. Dann gibt es ein $\xi \in [a,b]$ mit*

$$f(\bar{x}) - P(\bar{x}) = \frac{f^{(n+1)}(\xi)}{(n+1)!} L(\bar{x}), \qquad L(x) := (x-x_0)(x-x_1)\ldots(x-x_n) \tag{5.7}$$

Beweis: Betrachte für $\bar{x} \neq x_j$ die Funktion

$$F(x) := f(x) - P(x) - \frac{f(\bar{x}) - P(\bar{x})}{L(\bar{x})} L(x) \in C^{n+1}[a,b]$$

F hat die n+2 Nullstellen $\bar{x}, x_0, \ldots, x_n$ in $[a,b]$. Nach dem Satz von Rolle (Zwischen zwei Nullstellen einer Funktion $F(x)$ liegt (mindestens) eine Nullstelle ihrer Ableitung $F'(x)$) hat $F^{(n+1)}(x)$ mindestens eine Nullstelle $\xi \in [a,b]$:

$$0 = F^{(n+1)}(\xi) = f^{(n+1)}(\xi) - \frac{f(\bar{x}) - P(\bar{x})}{L(\bar{x})} (n+1)!$$

$\Rightarrow$ Behauptung. $\diamond$

Bemerkung 5.12. *Für den Interpolationsfehler erhalten wir somit die Abschätzung*

$$|f(x) - P(x)| \leq \max_{y \in [a,b]} \left| \frac{f^{(n+1)}(y)}{(n+1)!} L(x) \right|. \tag{5.8}$$

Bemerkung 5.13. *Wegen $|x - x_i| \leq b - a$ folgt sofort*

$$|f(x) - P(x)| \leq \max_{y \in [a,b]} \left| f^{(n+1)}(y) \right| \frac{(b-a)^{n+1}}{(n+1)!}. \tag{5.9}$$

Beispiel 5.14. *Es sei $f(x) = sin(x)$ auf $[0, 2\pi]$. Aus*

$$f^{(1)}(x) = \cos(x),\ f^{(2)}(x) = -\sin(x),\ f^{(3)}(x) = -\cos(x),\ \ldots$$

folgt $|f^{(k)}(y)| \leq 1$, $\forall y \in \mathbb{R}$. Mit den äquidistanten Stützstellen $x_i = 2\pi i/n$ folgt aus (5.9)

$$|f(x) - P(x)| \leq \frac{(2\pi)^{n+1}}{(n+1)!}.$$

Man sieht leicht, daß bereits für kleine n eine sehr gute Übereinstimmung der Funktionen erwartet werden kann.

5.2.6 Konvergenz

Sei

$$\Delta_m := \{a = x_0^{(m)} < x_1^{(m)} < \ldots < x_{n_m}^{(m)} = b\}, \quad m = 0, 1, \ldots,$$

eine Folge von Intervallteilungen von $[a, b]$.

$$\|\Delta_m\| = \max_i |x_{i+1}^{(m)} - x_i^{(m)}| \ .$$

$P_{\Delta_m}(x)$ bezeichne das interpolierende Polynom von f bzgl. Δ_m.

Problem: Gilt $\lim_{m\to\infty} P_{\Delta_m}(x) = f(x)$ für

$$\lim_{m\to\infty} \|\Delta_m\| = 0 \ ?$$

Antwort: i.A. nicht richtig.

Beispiel 5.15. (Beipiel von Runge)

$$f(x) = (1 + 25x^2)^{-1} \ \textit{in} \ [-1, 1]$$

$f(x)$ ist bzgl. $x \in \mathbb{C}$ keine ganze Funktion: vgl. Satz 5.17.
f wird an den Stellen $x_j = -1 + \frac{2j}{n}$, $j = 0, \ldots, n$, interpoliert durch ein Polynom

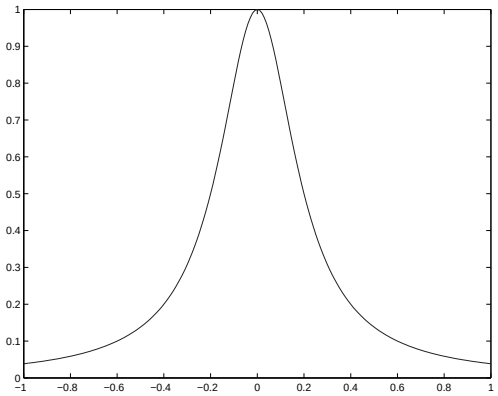


Abbildung 5.1: Beispiel von Runge.

P_n.

n	$\|f - P_n\|_\infty$
1	*0.96*
5	*0.43*
13	*1.07*
19	*8.57*

Wir wählen nun die Knoten $x_0, \ldots, x_n$ so, dass $\|L\|_\infty = \max_{-1\le x\le 1} |L(x)|$ möglichst klein wird, vgl. (5.8). Man erhält

$$L(x) = 2^{-n} T_{n+1}(x) \ , \qquad x \in [-1, 1],$$

wobei die Tschebyscheff-Polynome T_n wie folgt definiert sind:

$$T_n(x) = \cos n\theta, \qquad x = \cos\theta, \qquad 0 \le \theta \le 2\pi.$$

Es gilt $\|T_n\|_\infty \le 1$ und die Nullstellen von $L(x) = 2^{-n} T_{n+1}(x)$ sind

$$x_j = \cos \frac{(j + \frac{1}{2})\pi}{n+1} \ , \qquad j = 0, \ldots, n.$$

Bei dieser Wahl der Knoten ergibt sich die Fehlerabschätzung

n	$\|f - P_n\|_\infty$
1	*0.93*
5	*0.56*
13	*0.12*
19	*0.04*

Die Verbesserung ist erheblich, die Approximation aber immer noch unbefriedigend.

Ohne Beweis geben wir noch an:

Satz 5.16. *Zu jeder Folge $\{\Delta_m\}$ gibt es $f \in C[a, b]$, so dass $\{P_{\Delta_m}\}$ nicht gleichmäßig gegen f konvergiert.*

Satz 5.17. *Sei f eine ganze Funktion. Dann gilt $P_{\Delta_m} \to f$ gleichmäßig für alle $\{\Delta_m\}$ mit $\|\Delta_m\| \to 0$.*

Bemerkung 5.18. *$P(x) = P_{0,\ldots,n}(x)$ oszilliert i.A. stark für großes n; die Interpolation ist dann unbrauchbar. Diese Schwierigkeiten werden bei der Spline-Interpolation vermieden.*

5.3 Trigonometrische Interpolation

5.3.1 Diskrete Fouriertransformation

Gegeben seien Stützstellen (x_k, f_k), $k = 0, \ldots, n-1$ mit

$$x_k = \frac{2\pi}{n} k$$

Gesucht ist ein *Trigonometrisches Polynom*

$$P(\omega) = p(x) = \sum_{k=0}^{n-1} \beta_k \omega^k$$

mit

$$\omega := e^{ix} = cos(x) + i \cdot sin(x), \qquad i = \sqrt{-1}$$

so dass

$$p(x_k) = f_k, \qquad k = 0, 1, \ldots, n-1.$$

Bemerkung 5.19. *Ausdrücke der Form $e^{\frac{2\pi i}{n} \cdot k}$ heißen n'te Einheitswurzeln $k = 0, \ldots, n-1$. Sie liegen alle auf dem Einheitskreis.*

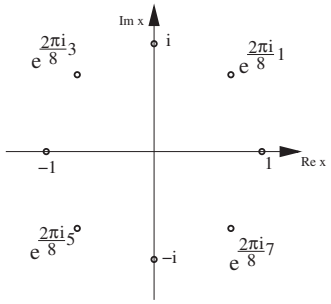


Abbildung 5.2: Einheitswurzeln, hier am Beispiel $n = 8$.

Nach Satz 5.3 ist die Interpolationsaufgabe

$$P(\omega_k) = \sum_{j=0}^{n-1} \beta_j \omega_k^j = f_k, \quad \omega_k = e^{i \cdot x_k}, \quad \text{(Fourier–Synthese)} \tag{5.10}$$

eindeutig lösbar. Die Interpolationsaufgabe (5.10) ist schreibbar als

$$f = W\beta$$

mit

$$f = \underbrace{\begin{pmatrix} f_0 \\ \vdots \\ f_{n-1} \end{pmatrix}}_{=:f} = \underbrace{\begin{pmatrix} 1 & \omega_0 & \cdots & \omega_0^{n-1} \\ \vdots & \vdots & & \vdots \\ 1 & \omega_{n-1} & \cdots & \omega_{n-1}^{n-1} \end{pmatrix}}_{=:W} \underbrace{\begin{pmatrix} \beta_0 \\ \vdots \\ \beta_{n-1} \end{pmatrix}}_{=:\beta}$$

Es gilt die Beziehung

$$W\bar{W}^T = nI, \quad \text{mit} \quad \bar{W} \text{ konjugiert komplexe Matrix zu } W, \tag{5.11}$$

da

$$(W\bar{W}^T)_{k,l} = \sum_{j=0}^{n-1} \omega_k^j \cdot \bar{\omega}_l^j = \sum_{j=0}^{n-1} e^{\frac{2\pi i}{n}\cdot k\cdot j} \cdot e^{\frac{-2\pi i}{n}\cdot l\cdot j} = \sum_{j=0}^{n-1} e^{\frac{2\pi i}{n}\cdot j(k-l)} = \sum_{j=0}^{n-1} q^j$$

mit $q = e^{\frac{2\pi i}{n}\cdot(k-l)}$. Dies ist eine geometrische Reihe und wir erhalten weiter

$$\sum_{j=0}^{n-1} q^j = \begin{cases} \frac{1-q^n}{1-q}, & \text{falls } q \neq 1 \\ n, & \text{falls } q = 1 \end{cases} \overset{q^n=1}{=} \begin{cases} 0, & \text{falls } k \neq l \\ n, & \text{falls } k = l. \end{cases}$$

Damit ist die Inverse von W aus (5.11) einfach zu berechnen:

$$W^{-1} = \frac{1}{n}\bar{W}^T \quad \Rightarrow \boxed{\beta = \tfrac{1}{n}\bar{W}^T f} \tag{5.12}$$

In Komponentenschreibweise:

$$\boxed{\beta_j = \frac{1}{n}\sum_{k=0}^{n-1} f_k \cdot e^{\frac{-2\pi i}{n}\cdot j\cdot k}} \tag{5.13}$$

Formel (5.13) wird *Diskrete Fouriertransformation der Länge n* genannt.

Bemerkung 5.20. *Der Aufwand zur Berechnung aller Komponenten beträgt offensichtlich $O(n^2)$.*

5.3.2 Trigonometrische Interpolation

Der Zusammenhang mit der Überschrift dieses Abschnittes ergibt sich durch den nachfolgenden Satz.

Satz 5.21 (Reelle Fouriertransformation). *Definiert man*

$$a_j := \frac{2}{n}\sum_{k=0}^{n-1} f_k \cdot \cos\left(\frac{2\pi kj}{n}\right), \quad b_j := \frac{2}{n}\sum_{k=0}^{n-1} f_k \cdot \sin\left(\frac{2\pi kj}{n}\right), \quad j = 0,\ldots,n-1.$$

und setzt für ungerades $n = 2m+1$

$$\Psi(x) := \frac{a_0}{2} + \sum_{j=1}^{m} \left(a_j \cos(jx) + b_j \sin(jx)\right)$$

bzw. für gerades $n = 2m$

$$\Psi(x) := \frac{a_0}{2} + \sum_{j=1}^{m-1} \left(a_j \cos(jx) + b_j \sin(jx)\right) + \frac{a_m}{2}\cos(mx)$$

so gilt

$$\Psi(x_k) = f_k, \qquad k = 0, \ldots, n-1$$

mit x_k aus (5.3.1).

Beweis: Zunächst notieren wir einige interessante Beziehungen.
Wegen $\omega_k^n = 1$ folgt für die komplexen Koeffizienten:

$$\beta_{n-j} = \frac{1}{n} \sum_{k=0}^{n-1} f_k \, \omega_k^{j-n} = \frac{1}{n} \sum_{k=0}^{n-1} f_k \, \omega_k^j. \tag{5.14}$$

Der Zusammenhang der reellen Fourierkoeffizienten mit den komplexen Koeffizienten ergibt sich wegen (5.13) zu

$$\beta_j = \frac{1}{2}(a_j - ib_j), \qquad \text{und} \qquad \beta_{n-j} = \frac{1}{2}(a_j + ib_j). \tag{5.15}$$

Hierraus folgt unmittelbar die Beziehung

$$\beta_j \, \omega_k^j + \beta_{n-j} \, \omega_k^{n-j} = a_j \cos jx_k + b_j \sin jx_k. \tag{5.16}$$

Nach diesen Vorüberlegungen wenden wir uns nun dem eigentlichen Beweis zu. Für gerades $\underline{n = 2m}$ ist

$$b_0 = 0, \qquad b_m = 0, \qquad \omega_k^m = \cos(mx_k),$$

und daher gilt (wegen $f = W\beta$) mit (5.14)–(5.16)

$$\begin{aligned} f_k = \sum_{j=0}^{n-1} \beta_j \, \omega_k^j &= \underbrace{\beta_0}_{\frac{a_0}{2}} + \sum_{j=1}^{m-1} \underbrace{(\beta_j \, \omega_k^j + \beta_{n-j} \, \omega_k^{n-j})}_{a_j \cos(jx_k) + b_j \sin(jx_k)} + \underbrace{\beta_m \, \omega_k^m}_{\frac{a_m}{2} \cos(mx)} \\ &= \psi(x_k). \end{aligned}$$

Ebenso folgt die Behauptung für ungerades $n = 2m + 1$. ◇

Bemerkung 5.22. *Der Aufwand zur Berechnung aller Komponenten beträgt $O(n^2)$.*

5.3.3 Schnelle Fourier–Transformation (FFT)

Cooley–Tukey haben eine Möglichkeit gefunden die diskrete Fouriertransformation effizienter durchzuführen. Sei hierzu $q = e^{\frac{-2\pi i}{n}}$ und n gerade, $n = 2m$, dann ist (5.13) darstellbar als

$$\beta_k = \frac{1}{n} \sum_{j=0}^{n-1} f_j \cdot q^{j \cdot k}. \tag{5.17}$$

Es gilt $q^n = 1$ und $q^{n/2} = -1$.
Idee: Spalte Summe in (5.17) nach geraden und ungeraden Indizes auf.

$$\begin{aligned}\beta_k &= \frac{1}{n}\left(\sum_{l=0}^{m-1} f_{2l}\cdot q^{(2l)k} + \sum_{l=0}^{m-1} f_{2l+1}\cdot q^{(2l+1)k}\right)\\ &= \frac{1}{2}\left(\frac{1}{n/2}\sum_{l=0}^{m-1} f_{2l}\cdot \left(q^2\right)^{lk} + q^k\frac{1}{n/2}\sum_{l=0}^{m-1} f_{2l+1}\cdot \left(q^2\right)^{lk}\right)\\ &=: \frac{1}{2}\left(g_k + q^k u_k\right)\end{aligned}$$

Wegen $q^2 = e^{\frac{2\pi i}{n/2}}$ lassen sich g_k und u_k als Fouriertransformation der Länge $\frac{n}{2}$ berechnen. Weiterhin folgt durch einfaches Nachrechnen

$$\begin{aligned} g_{k+m} &= g_k \\ u_{k+m} &= u_k \end{aligned} \tag{5.18}$$

und damit für $k = 0, 1, \ldots, m-1$

$$\beta_k = \frac{1}{2}\left(g_k + q^k u_k\right) \tag{5.19}$$

und

$$\beta_{k+m} = \frac{1}{2}\left(g_k + q^{k+m} u_k\right) = \frac{1}{2}\left(g_k - q^k u_k\right). \tag{5.20}$$

Bemerkung 5.23. *Die g_k aus (5.19) können somit bei (5.20) direkt ohne zusätzlichen Rechenaufwand wieder verwendet werden.*

Anstelle von 1 mal n^2 hat man jetzt nur noch 2 mal $(n/2)^2$, also $\frac{1}{2}n^2$ als Aufwand.

Sei nun $n = 2^p$. Die Idee ist es nun die zuvor dargestellte Berechnungsvorschrift mehrfach anzuwenden, also auch g_k und u_k gemäß (5.20) zu bestimmen. Sei M_p die Anzahl an Multiplikationen/Divisionen, die wir für die FFT benötigen. Wir vernachlässigen die Berechnung von q^k. Es gilt

$$\begin{aligned} M_0 &= 0 \\ M_{p+1} &= 2M_p + \tfrac{n}{2} + \tfrac{n}{2} \;=\; 2M_p + 2^p \end{aligned}$$

also

$$M_p = 2^{p-1} + 2\cdot 2^{p-2} + \ldots + \underbrace{2\cdot 2\cdot \ldots \cdot 2}_{p-1 \text{ mal}} = p\cdot 2^{p-1} = \frac{1}{2}\cdot p\cdot 2^p = \boxed{\frac{1}{2}\log_2 n \cdot n}$$

Damit gilt der

Satz 5.24 (Aufwand FFT). *Eine Fouriertransformation der Länge $n = 2^p$ kann mit $O(n \cdot \log_2 n)$ komplexen Multiplikationen berechnet werden.*

Bemerkung 5.25. *Die Anzahl an Additionen/Subtraktionen ergibt sich ebenfalls zu $O(n \cdot \log_2 n)$.*

Bemerkung 5.26. *Es gilt:*

$$O(n) < O(n \cdot \log_2 n) < O(n^2).$$

5.3.4 Anwendungen

Die FT findet Ihre Anwendung z.B. in der *Signalverarbeitung*. Eine wichtige Anwendung ist die *Frequenzanalyse*, hier haben die Fourierkoeffizienten der trigonometrischen Polynome (im Gegensatz zu herkömmlichen Koeffizienten) eine klare physikalische Bedeutung: Jeder der Fourierkoeffizienten a_k bzw- b_k gehört zu einer bestimmten Frequenz im Sinus bzw. Kosinus. Die Anwendbarkeit der FT ergibt sich z.B. wenn Signale durch Störungen (Rauschen) beeinflußt werden. Die FT glättet und das Originalresultat kann häufig reproduziert werden, bzw. die Hauptfrequenzen des Signals können besser identifiziert werden. Dies nutzt man dann z.B. beim *Filtern* aus. Hat man einmal ein Signal zerlegt und stehen die Fourierkoeffizienten (die eine bestimmte Frequenz beschreiben) zur Verfügung kann man durch Vergrößern oder Verkleinern der einzelnen Koeffizienten spezielle Frequenzen unterschiedlich stark betonen oder sogar ganz herausnehmen (Filtern). Weitere Anwendungen sind die Zerlegung eines Signals in verschiedenen Frequenzbereiche oder Datenkompression (Null–Setzen der Koeffizienten mit geringem Betrag). Da die Anwendungen alle in Echtzeit durchgeführt werden müssen ist eine effiziente Implementierung im Sinne der FFT offensichtlich.

5.4 Spline–Interpolation

Spline–Funktionen, die sich stückweise aus Polynomen zusammensetzen, verbinden den Vorteil einer glatten oszillationsfreien Interpolation mit denjenigen, die der Umgang mit Polynomen niedrigen Grades mit sich bringt.

5.4.1 Polynom–Splines

Sei $\Delta = \{a = x_0 < x_1 < \ldots < x_n = b\}$ eine Zerlegung des Intervalls $[a, b] \subset \mathbb{R}$ mit inneren Knoten $x_1, \ldots, x_{n-1}$ und Randknoten x_0, x_n.

Definition 5.27. *Eine Funktion $s : [a, b] \to \mathbb{R}$ heißt* Polynom-Spline vom Grad *l, $l = (0, 1, 2, \ldots)$ zur Zerlegung Δ, wenn sie folgende Eigenschaften besitzt:*

(a) $s \in C^{l-1}[a,b]$

(b) $s \in \Pi_l$ *für* $x_j \le x < x_{j+1}, \quad j = 0, 1, \ldots, n-1.$

Hierbei ist $C^{-1}[a,b]$ *der Raum der auf* $[a,b]$ *stückweise stetigen Funktionen. Die* Menge aller Polynom-Splines vom Grade l zur Zerlegung Δ *bezeichnen wir mit* $S_l(\Delta)$. *Fortan wird schlechthin von* Splines *gesprochen.*

Beispiel 5.28. (Lineare Splines): *Sind* $(n+1)$ *Punkte*

$$(x_0, y_0), (x_1, y_1), \ldots, (x_n, y_n)$$

gegeben, so stellt der Polygonzug *durch diese Punkte einen Spline* $s \in S_1(\Delta)$ *dar.*

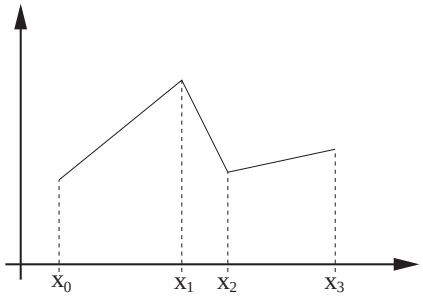


Abbildung 5.3: Polynomspline vom Grade 1.

Beispiel 5.29. (Quadratische Splines): *Bei äquidistanten Knoten* $x_j = a + jh$, $j = 0, \ldots, n$, *ist*

$$s(x) = \frac{1}{2h^2} \begin{cases} (x - x_j)^2 & , x_j \le x < x_{j+1} \\ h^2 + 2h(x - x_{j+1}) - 2(x - x_{j+1})^2, & x_{j+1} \le x < x_{j+2} \\ (x_{j+3} - x)^2 & , x_{j+2} \le x < x_{j+3} \end{cases}$$

ein quadratischer B-Spline $s \in S_2(\Delta)$.

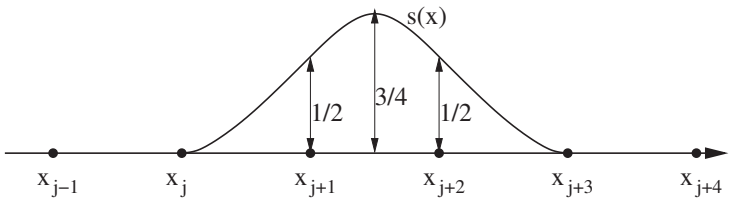


Abbildung 5.4: Polynomspline vom Grade 2.

Beispiel 5.30. *Die Funktionen* $q_{l_j} : [a,b] \to \mathbb{R}, \quad j = 0, 1, \ldots, n-1,$

$$q_{l_j}(x) = (x - x_j)_+^l := \begin{cases} (x - x_j)^l & , \quad x \geq x_j \\ 0 & , \quad x < x_j \end{cases}$$

sind Splines vom Grade l *zu* Δ*. Man beachte, dass* q_{l_j} *keine Polynome in* $[a,b]$ *sind.*

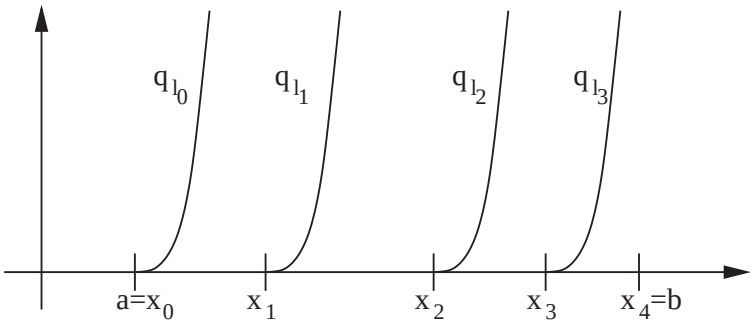


Abbildung 5.5: Splines vom Grade l.

Der Spline-Raum $S_l(\Delta)$ ist ein linearer Teilraum von $C^{l-1}(\Delta)$. Der folgende Satz gibt Auskunft über eine Basis von $S_l(\Delta)$.

Satz 5.31. *Die Menge* $S_l(\Delta)$ *ist ein linearer Raum der Dimension* $n + l$*. Die Elemente* $p_i(x) = x^i$*,* $(i = 0, \ldots, l)$*,* $q_{lj}(x) = (x - x_j)_+^l (j = 1, \ldots, n-1)$ *bilden eine Basis von* $S_l(\Delta)$*.*

Beweis: Wir haben zu zeigen, dass es zu $s \in S_l(\Delta)$ eine eindeutige Darstellung

$$s(x) = \sum_{i=0}^{l} a_i x^i + \sum_{j=1}^{n-1} b_j (x - x_j)_+^l, \qquad x \in [a,b],$$

gibt. Dies erkennt man durch Induktion bzgl. des Index j der Zerlegung Δ. Im Intervall $I_1 = [x_0, x_1]$ ist s ein Polynom $s(x) = a_0 + a_1 x + \ldots + a_l x^l$, also gilt die Darstellung

$$s(x) = \sum_{i=0}^{l} a_i x^i + \sum_{j=1}^{k-1} b_j (x - x_j)_+^l$$

für $k = 1$ auf $I_k = [x_0, x_k]$. Wir betrachten nun

$$d(x) := s(x) - \sum_{i=0}^{l} a_i x^i - \sum_{j=1}^{k-1} b_j (x - x_j)_+^l.$$

Dann ist $d \in C^{l-1}(I_{k+1})$ und $d(x) = 0$ für $x \in I_k$. Außerdem ist $d \in \Pi_l$ in $[x_k, x_{k+1}]$. Also genügt d auf $[x_k, x_{k+1}]$ der Differentialgleichung

$$\begin{aligned} d^{(l+1)}(x) &= 0, \quad x_k \le x \le x_{k+1} \\ d^{(i)}(x_k) &= 0, \quad i = 0, \ldots, l-1 \end{aligned}$$

Die Lösung dieser Anfangswertaufgabe ist

$$d(x) = b_k (x - x_k)_+^l \,, \qquad x_k \le x, \qquad b_k \in \mathbb{R}.$$

Damit ist die Behauptung für den Index $k+1$ gezeigt. Für $k = n$ bilden daher die $n + l$ linear unabhängigen Elemente

$$p_i(x) = x^i, \quad (i = 0, \ldots, l), \qquad q_{lj}(x) = (x - x_j)_+^l, \quad (j = 1, \ldots, n-1)$$

eine Basis von $S_l(\Delta)$. ◇

Wir untersuchen nun die folgende Interpolationsaufgabe:
Bestimme zu Stützwerten y_j, $j = 0, \ldots, n$ einen interpolierenden Spline $s \in S_l(\Delta)$ mit

$$s(x_j) = y_j, \qquad j = 0, \ldots, n.$$

Die Stützwerte y_j sind dabei als Funktionswerte $y_j = f(x_j)$ einer hinreichend glatten Funktion f aufzufassen. Wegen $\dim S_l(\Delta) = n + l$ können über die $n+1$ Interpolationsbedingungen hinaus noch

$$n + l - (n+1) = l - 1$$

freie Parameter bestimmt werden. Für ungerades $l = 2m - 1$ ist dies eine gerade Anzahl $2m - 2$ von Parametern, welche sich symmetrisch in den Randknoten x_0, x_n anordnen lassen.
Wir beschränken uns nachfolgend auf den in der Praxis wichtigen Fall $m = 2$ der *kubischen Splines*, d.h. Polynom–Splines mit $l = 3$, der sich besonders häufig in den Anwendungen wiederfindet.

5.4.2 Kubische Splines

5.4.2.1 Einführung und Aufgabenstellung

Nach Definition ist ein Spline $s \in C^2[x_0, x_n]$ auf jedem der Intervalle $[x_{k-1}, x_k]$, $k = 1, \ldots, n$ durch

$$s(x) = s_k(x) = a_k + b_k(x - x_{k-1}) + c_k(x - x_{k-1})^2 + d_k(x - x_{k-1})^3$$

gegeben. Neben den Interpolationsbedingungen

$$s(x_j) = f(x_j), \qquad j = 0, \ldots, n \tag{5.21}$$

erfordert die Glattheit $s \in C^2[x_0, x_n]$ (an den Nahtstellen sollen keine Knicke auftreten und die Krümmungen sollen stetig ineinander übergehen) die zusätzlichen Bedingungen

$$s_k'(x_k) = s_{k+1}'(x_k) \quad \text{und} \quad s_k''(x_k) = s_{k+1}''(x_k), \qquad k = 1, \ldots, n-1. \tag{5.22}$$

Die $2n-2$ linearen Gleichungen in (5.22) ergeben dann zusammen mit den $2n$ linearen Gleichungen aus (5.21) zusammen $4n-2$ lineare Gleichungen für die $4n$ Unbekannten a_k, b_k, c_k, d_k, $k = 1, \ldots, n$. Das Problem ist somit zwar lösbar, jedoch existieren ∞ viele Lösungen. Der Grund liegt in den Randpunkten x_0 und x_n, in denen wir keine Glattheitseigenschaften fordern mußten.
Um dennoch eine eindeutige Lösung erhalten zu können, müssen wir zwei weitere Bedingungen fordern, die üblicherweise an den Randpunkten verlangt werden. Hierzu gibt es verschiedenen Möglichkeiten:

(5.23)

(a) Natürliche Endbedingungen:

$$s''(a) = 0, \qquad s''(b) = 0$$

(b) Hermite Endbedingungen:

$$s'(a) = f'(a), \qquad s'(b) = f'(b)$$

(c) Periodische Endbedingungen:

$$s^{(i)}(a) = s^{(i)}(b), \qquad i = 0, 1, 2,$$

falls f periodisch mit

$$f^{(i)}(a) = f^{(i)}(b), \qquad i = 0, 1, 2.$$

5.4.2.2 Existenz und Eindeutigkeit

Man kann nun zeigen, dass die Interpolationsaufgabe (5.21) zusammen mit einer der Bedingungen aus (5.23) eindeutig lösbar ist. Zusätzlich erfüllt der interpolierende Spline eine Minimum-Norm-Eigenschaft bzgl. der Norm in $C^2[a, b]$.

$$\|f\|_2 := \left(\int_a^b (f''(x))^2 dx \right)^{\frac{1}{2}}, \quad f \in C^2[a, b]. \tag{5.24}$$

Für diese Norm gilt

$$\begin{aligned} \|f\|_2 = 0 &\Leftrightarrow f''(x) = 0 \qquad \text{für } x \in [a,b] \\ &\Leftrightarrow f \text{ linear in } [a,b]. \end{aligned}$$

Satz 5.32 (Existenz, Eindeutigkeit und Extremaleigenschaft von Splines). *Sei $f \in C^2[a,b]$. Dann gibt es genau einen Spline $s \in S_3(\Delta)$, der (5.21) und eine der Interpolationsbedingungen (5.23) erfüllt. Dieser interpolierende Spline genügt der Minimum-Norm-Bedingung*

$$0 \le \|f - s\|_2^2 = \|f\|_2^2 - \|s\|_2^2.$$

Beweis: Mit einem Spline $s \in S_3(\Delta)$ berechnet man für die Abweichung $d(x) = f(x) - s(x)$:

$$\begin{aligned} \|f - s\|_2^2 &= \int_a^b (f''(x) - s''(x))^2 dx \\ &= \|f\|_2^2 - \|s\|_2^2 - 2\int_a^b d''(x)s''(x)dx. \end{aligned}$$

Da nur $s \in C^2[a,b]$ gilt, müssen wir für die partielle Integration aufspalten:

$$\begin{aligned} \int_a^b d''(x)s''(x)dx &= \sum_{j=1}^n \int_{x_{j-1}}^{x_j} d''(x)s''(x)dx \\ &= \sum_{j=1}^n \left\{ [d'(x)s''(x) - d(x)s^{(3)}(x)]_{x_{j-1}}^{x_j} + \int_{x_{j-1}}^{x_j} d(x)s^{(4)}(x)dx \right\}. \end{aligned}$$

Nun ist $s^{(4)}(x) \equiv 0$ auf $[x_{j-1}, x_j]$. Die Interpolationsforderungen (5.21), (5.23) bewirken gerade, dass

$$\sum_{j=1}^n [d'(x)s''(x) - d(x)s^{(3)}(x)]_{x_{j-1}}^{x_j} = [d'(x)s''(x) - d(x)s^{(3)}(x)]_a^b = 0$$

in den Fällen (5.23)(a)-(c). Damit ist die Minimum-Norm-Bedingung

$$0 \le \|f - s\|_2^2 = \|f\|_2^2 - \|s\|_2^2$$

gezeigt, mit der sich die Eindeutigkeit von s folgendermaßen ergibt:
Ist $\tilde{s}$ ein weiterer interpolierender Spline, so kann man $f = \tilde{s}$ in der letzten Ungleichung nehmen:

$$0 \le \|\tilde{s} - s\|_2^2 = \|\tilde{s}\|_2^2 - \|s\|_2^2.$$

Durch Vertauschung von s und $\tilde{s}$ sieht man $\|s - \tilde{s}\|_2 = 0$. Also ist $s - \tilde{s}$ linear. Wegen $s(x) - \tilde{s}(x) = 0$ für $x = a$ und $x = b$ muss dann $s = \tilde{s}$ gelten.
Zum Nachweis der Existenz greifen wir auf die Basis–Darstellung in Satz 5.31 zurück. Die Interpolationsforderungen stellen ein LGS in $(n + 3)$ Unbekannten $a_0, \ldots, a_3, b_1, \ldots, b_{n-1}$ dar. In den Fällen (5.23)(a)-(c) wird das System homogen, wenn $f \equiv 0$ zu interpolieren ist. Dann ist aber $s = 0$ interpolierender Spline und nach den obigen Überlegungen auch der einzige. ◇

5.4.2.3 Geometrische und mechanische Interpretation

Die Extremaleigenschaft des kubischen Splines

$$\int_a^b (s''(x))^2 dx \leq \int_a^b (d''(x))^2 dx$$

erlaubt die folgende geometrische und mechanische Interpretation:
Die Krümmung $k(x)$ einer Kurve $y = f(x)$ in der (x,y)-Ebene ist gegeben durch

$$k(x) = \frac{f''(x)}{(1 + (f'(x))^2)^{3/2}} \ .$$

Unter der Annahme $|f'(x)| \ll 1$ wird die mittlere Gesamtkrümmung

$$\|k\|_2^2 \approx \int_a^b (f''(x))^2 dx \ .$$

Der kubische Spline minimiert also die Norm $\|k\|_2$ unter allen interpolierenden Funktionen.
Das Biegemoment $M(x)$ eines homogenen, isotropen Stabes, dessen Biegelinie durch $y = f(x)$ beschrieben wird, ist $M(x) = c_1 k(x),\ c_1 > 0$. Die Biege-Energie ist dann näherungsweise

$$E(f) = c_2 \int_a^b M(x)^2 dx \approx c_3 \int_a^b (f''(x))^2 dx.$$

Wird ein gebogener Stab durch Lager in "Interpolationspunkten" fixiert, so wird die minimale Biege-Energie durch einen kubischen Spline realisiert. Außerhalb von $[a, b]$, wo der Stab nicht fixiert ist, nimmt er die spannungsfreie "natürliche" Lage $s''(x) = 0$ an. In diesem Sinne sind die Endbedingungen $s''(a) = 0,\ s''(b) = 0$ in (5.23)(a) als "natürlich" zu versehen.

5.4.2.4 Die Berechnung von Spline-Funktionen

Zu berechnen sei die Spline-Funktion $s(x)$ mit $s(x_j) = y_j,\ j = 0, \ldots, n$, welche zusätzlich eine der Eigenschaften (a),(b),(c) hat. Wir setzen

$$\begin{aligned} h_j &:= x_j - x_{j-1}, \quad j = 1, \ldots, n, \\ M_j &:= s''(x_j) \qquad\quad j = 0, \ldots, n, \quad \text{(Momente)}. \end{aligned}$$

Da s'' linear in $[x_{j-1}, x_j]$ ist, gilt

$$s''(x) = \frac{1}{h_j}(M_j(x - x_{j-1}) + M_{j-1}(x_j - x)), \qquad x_{j-1} \le x \le x_j.$$

Durch Integration erhält man für $x \in [x_{j-1}, x_j]$

$$\begin{aligned} s'(x) &= \tfrac{1}{2h_j}(M_j(x - x_{j-1})^2 - M_{j-1}(x_j - x)^2) + a_j, \\ s(x) &= \tfrac{1}{6h_j}(M_j(x - x_{j-1})^3 + M_{j-1}(x_j - x)^3) + a_j(x - x_{j-1}) + b_j \end{aligned}$$

mit $a_j, b_j \in \mathbb{R}$. Für die Koeffizienten a_j, b_j erhält man aus $s(x_{j-1}) = y_{j-1}$, $s(x_j) = y_j$ die Gleichungen

$$\begin{aligned} M_{j-1}\frac{h_j^2}{6} + \qquad\quad b_j &= y_{j-1}, \\ M_j\frac{h_j^2}{6} + a_j h_j + \quad b_j &= y_j, \end{aligned}$$

und daraus

$$\begin{aligned} b_j &= y_{j-1} - M_{j-1}\frac{h_j^2}{6}, \\ a_j &= \frac{y_j - y_{j-1}}{h_j} - \frac{h_j}{6}(M_j - M_{j-1}). \end{aligned}$$

Damit ergibt sich

$$\begin{aligned} s'(x_j^-) &= \frac{1}{2h_j}M_j h_j^2 + a_j = \frac{y_j - y_{j-1}}{h_j} + \frac{h_j}{3}M_j + \frac{h_j}{6}M_{j-1}, \\ s'(x_j^+) &= \frac{y_{j+1} - y_j}{h_{j+1}} - \frac{h_{j+1}}{3}M_j - \frac{h_{j+1}}{6}M_{j+1}. \end{aligned}$$

Wegen $s'(x_j^-) = s'(x_j^+)$ folgt dann

$$\boxed{\mu_j M_{j-1} + M_j + \lambda_j M_{j+1} = d_j, \quad j = 1, \ldots, n-1} \tag{5.25}$$

mit

$$\begin{aligned} \mu_j &:= \frac{h_j}{2(h_j + h_{j+1})}\ , \\ \lambda_j &:= \frac{h_{j+1}}{2(h_j + h_{j+1})}\ , \qquad \mu_j + \lambda_j = \frac{1}{2}\ , \\ d_j &:= \frac{3}{h_j + h_{j+1}}\left(\frac{y_{j+1} - y_j}{h_{j+1}} - \frac{y_j - y_{j-1}}{h_j}\right)\ . \end{aligned}$$

Fall (5.23)(a): **(Natürliche Endbedingungen)**
Vorgegeben: Werte M_0, M_n

Die "natürliche" Bedingung $M_0 = M_n = 0$ ist darin enthalten. Dann stellt (5.25) ein LGS für $M_1, \ldots, M_{n-1}$ dar:

$$\begin{pmatrix} 1 & \lambda_1 & & & \\ \mu_2 & 1 & \lambda_2 & & \\ & \ddots & \ddots & \ddots & \\ & & \mu_{n-2} & 1 & \lambda_{n-2} \\ & & & \mu_{n-1} & 1 \end{pmatrix} \begin{pmatrix} M_1 \\ M_2 \\ \vdots \\ \vdots \\ M_{n-1} \end{pmatrix} = \begin{pmatrix} d_1 - \mu_1 M_0 \\ d_2 \\ \vdots \\ d_{n-2} \\ d_{n-1} - \lambda_{n-1} M_n \end{pmatrix}$$

Die Matrix ist tridiagonal und wegen $\mu_j + \lambda_j = \frac{1}{2}$ diagonal-dominant, also LR-zerlegbar nach Satz 3.11. Das LGS kann daher mit Algorithmus (3.26) gelöst werden. Bei äquidistanten Knoten x_j gilt $\mu_j = \lambda_j = \frac{1}{4}$, also ist die Matrix symmetrisch und damit positiv definit.

Fall (5.23)(b): **(Hermite Endbedingungen)**
Vorgegeben: $s'(a) = y'_0,\ s'(b) = y'_n$.

Aus der Darstellung von $s'(x)$ folgt

$$\begin{aligned} \frac{h_1}{3} M_0 + \frac{h_1}{6} M_1 &= \frac{y_1 - y_0}{h_1} - y'_0 \\ \frac{h_n}{6} M_{n-1} + \frac{h_n}{3} M_n &= y'_n - \frac{y_n - y_{n-1}}{h_n}. \end{aligned}$$

Mit

$$\begin{aligned} \lambda_0 = \frac{1}{2}\,, \quad d_0 &= \frac{3}{h_1}\left(\frac{y_1 - y_0}{h_1} - y'_0\right), \\ \mu_n = \frac{1}{2}\,, \quad d_n &= \frac{3}{h_n}\left(y'_n - \frac{y_n - y_{n-1}}{h_n}\right) \end{aligned}$$

erhalten wir ein LGS und den $(n+1)$ Unbekannten $M_0, \ldots, M_n$:

$$\begin{pmatrix} 1 & \lambda_0 & & & \\ \mu_1 & 1 & \lambda_1 & & \\ & \ddots & \ddots & \ddots & \\ & & \mu_{n-1} & 1 & \lambda_{n-1} \\ & & & \mu_n & 1 \end{pmatrix} \begin{pmatrix} M_0 \\ \vdots \\ \vdots \\ M_n \end{pmatrix} = \begin{pmatrix} d_0 \\ \vdots \\ \vdots \\ d_n \end{pmatrix}.$$

Die Matrix ist ebenfalls tridiagonal und diagonal-dominant, also LR-zerlegbar.

Beispiel 5.33. *Für die nicht äquidistanten Knoten*

j	x_j	y_j	M_j
0	*0*	*0*	*0.022181*
1	*8.2*	*0.5*	*-0.000665*
2	*14.7*	*1.0*	*-0.010253*
3	*17.0*	*1.1*	*-0.006909*
4	*21.1*	*1.2*	*-0.000613*
5	*35.0*	*1.4*	*-0.000691*
6	*54.1*	*1.5*	*-0.000040*
7	*104*	*1.6*	*-0.000014*
8	*357*	*1.7*	*0.000004*

$$y'_0 = 0.0012566,$$
$$y'_8 = 0.0001$$

liefert Algorithmus (3.26) die angegebenen Werte M_j.

Fall (5.23)(c): **(Periodische Endbedingungen)**
Hier ist

$$\begin{aligned} M_0 = M_n\ , \quad \text{da } s''(a) &= s''(b) \\ y_0 = y_n\ , \quad \text{da } s(a) &= s(b). \end{aligned}$$

Die weitere Gleichung $s'(a) = s'(b)$ ergibt eine Beziehung

$$\mu_n M_{n-1} + M_n + \lambda_n M_{n+1} = d_n,$$

wenn man setzt $h_{n+1} = h_1,\ M_{n+1} = M_1,\ y_{n+1} = y_1$. Für $M_1, \ldots, M_n$ ist das LGS zu lösen

$$\begin{pmatrix} 1 & \lambda_1 & & & \mu_1 \\ \mu_2 & 1 & \lambda_2 & & \\ & \ddots & \ddots & \ddots & \\ & & \mu_{n-1} & 1 & \lambda_{n-1} \\ \lambda_n & & & \mu_n & 1 \end{pmatrix} \begin{pmatrix} M_1 \\ \vdots \\ \\ \vdots \\ M_n \end{pmatrix} = \begin{pmatrix} d_1 \\ \vdots \\ \\ \vdots \\ d_n \end{pmatrix}.$$

Die Matrix ist nicht mehr tridiagonal, aber immer noch diagonal-dominant, also LR-zerlegbar. Bei symmetrischen Matrizen kann jedoch das CHOLESKY-Verfahren (5.3) durch eine kleine zusätzliche Betrachtung modifiziert werden, auf die wir hier jedoch nicht weiter eingehen.

5.4.2.5 Konvergenzeigenschaften

Entgegen dem Grenzverhalten von Interpolations-Polynomen konvergieren Spline-Funktionen gegen die Funktion, die sie interpolieren, bei Verfeinerung der Unterteilungen Δ. Sei

$$\Delta_m = \{a = x_0^{(m)} < x_1^{(m)} < \ldots < x_{n_m}^{(m)} = b\}$$

eine Folge von Unterteilungen des Intervalls $[a, b]$. Mit

$$\|\Delta_m\| := \max_j (x_{j+1}^{(m)} - x_j^{(m)})$$

gilt:

Satz 5.34. *Sei $f \in C^4[a, b]$ mit $L = \|f^{(4)}\|_\infty$ und sei Δ_m eine Zerlegungsfolge von $[a, b]$ mit*

$$\sup_{m,j} \frac{\|\Delta_m\|}{x_{j+1}^{(m)} - x_j^{(m)}} \leq K < +\infty.$$

Seien s_m die zu f gehörigen Spline-Funktionen mit

$$\begin{aligned} s_m(\xi) &= f(\xi) \quad \text{für } \xi \in \Delta_m, \\ s_m'(x) &= f'(x) \quad \text{für } x = a, b. \end{aligned}$$

Dann gibt es von Δ_m unabhängige Konstanten C_i (≤ 2), so dass für $x \in [a, b]$ gilt

$$|f^{(i)}(x) - s_m^{(i)}(x)| \leq C_i L K \|\Delta_m\|^{4-i}, \qquad i = 0, 1, 2, 3.$$

Beweis: Sei

$$\Delta := \Delta_m = \{a = x_0 < \ldots < x_n = b\}$$

eine feste Zerlegung. Für die Momente $M_j = s''(x_j)$ einer Spline-Funktion $s(x)$ mit $s'(x) = f'(x)$ für $x = x_0, x_n$ gilt nach (5.25) die Gleichung

$$AM = d$$

mit

$$\begin{aligned} \lambda_0 &= \mu_n = \frac{1}{2}, \\ \lambda_j &= \frac{1}{2} \frac{h_{j+1}}{h_j + h_{j+1}}, \qquad \mu_j = \frac{1}{2} - \lambda_j, \quad j = 1, \ldots, n-1 \\ d_0 &= \frac{3}{h_1} \left(\frac{y_1 - y_0}{h_1} - f'(x_0) \right), \\ d_n &= \frac{3}{h_n} \left(f'(x_n) - \frac{y_n - y_{n-1}}{h_n} \right) \\ d_j &= \frac{3}{h_j + h_{j+1}} \left(\frac{y_{j+1} - y_j}{h_{j+1}} - \frac{y_j - y_{j-1}}{h_j} \right) \quad j = 1, \ldots, n-1. \end{aligned}$$

Man zeigt leicht (Übung), dass die Matrix A die Eigenschaft hat

$$\|x\|_\infty \le 2\|Ax\|_\infty \qquad \text{für } x \in \mathbb{R}^n. \tag{5.26}$$

Für die Vektoren

$$\begin{aligned} F &:= (f''(x_0), f''(x_1), \ldots, f''(x_n))^T \\ r &:= d - AF = A(M - F) \end{aligned}$$

wird zunächst gezeigt

$$\|r\|_\infty \le \frac{3}{8} L \|\Delta\|^2. \tag{5.27}$$

Beweis von (5.27):
Die Taylor-Entwicklung von f und f'' um x_0 ergibt

$$\begin{aligned} r_0 &= d_0 - f''(x_0) - \frac{1}{2} f''(x_1) \\ &= \frac{3}{h_1} \left(\frac{f(x_1) - f(x_0)}{h_1} - f'(x_0) \right) - f''(x_0) - \frac{1}{2} f''(x_1) \\ &= \frac{h_1^2}{8}\, f^{(4)}(\tau_1) - \frac{h_1^2}{4} f^{(4)}(\tau_2) \qquad \text{mit } \tau_1, \tau_2 \in [x_0, x_1]. \\ \Rightarrow \quad & |r_0| \le \frac{3}{8} L\, \|\Delta\|^2. \end{aligned}$$

Analog erhält man für

$$r_n = d_n - \frac{1}{2} f''(x_{n-1}) - f''(x_n)$$

die Abschätzung

$$|r_n| \le \frac{3}{8} L\, \|\Delta\|^2.$$

Entsprechend ergibt sich durch die Taylor-Entwicklung um x_j für $j = 1, \ldots, n-1$:

$$\begin{aligned} r_j &= d_j - \mu_j f''(x_{j-1}) - f''(x_j) - \lambda_j f''(x_{j+1}) \\ &= \frac{1}{2(h_j + h_{j+1})} \left[\frac{h_{j+1}^3}{4} f^{(4)}(\tau_1) + \frac{h_j^3}{4} f^{(4)}(\tau_2) - \frac{h_j^3}{4} f^{(4)}(\tau_3) - \frac{h_{j+1}^3}{2} f^{(4)}(\tau_4) \right] \\ &\quad \text{mit } \tau_1, \ldots, \tau_4 \in [x_{j-1}, x_{j+1}]. \end{aligned}$$

Also

$$|r_j| \le \frac{3}{8} L\, \frac{h_j^3 + h_{j+1}^3}{h_j + h_{j+1}} \le \frac{3}{8} L\, \|\Delta\|^2.$$

Insgesamt gilt

$$\|r\|_\infty \le \frac{3}{8} L\, \|\Delta\|^2$$

und damit wegen $r = A(M - F)$ und (5.26)

$$\|M - F\|_\infty \leq 2\|r\|_\infty \leq \frac{3}{4} L \, \|\Delta\|^2. \tag{5.28}$$

Wir zeigen nun die Behauptung des Satzes für $\underline{i = 3}$:
Für $x \in [x_{j-1}, x_j]$ ist

$$\begin{aligned} s^{(3)}(x) - f^{(3)}(x) &= \frac{M_j - M_{j-1}}{h_j} - f^{(3)}(x) \\ &= \frac{M_j - f''(x_j)}{h_j} - \frac{M_{j-1} - f''(x_{j-1})}{h_j} \\ &\quad + \frac{f''(x_j) - f''(x) - (f''(x_{j-1}) - f''(x))}{h_j} - f^{(3)}(x) \end{aligned}$$

Taylor-Entwicklung um x ergibt mit der Abschätzung (5.28)

$$|s^{(3)}(x) - f^{(3)}(x)| \leq \frac{3}{2} L \, \frac{\|\Delta\|^2}{h_j} + \frac{L}{2} \frac{\|\Delta\|^2}{h_j} \leq 2LK\|\Delta\|, \tag{5.29}$$

da $\|\Delta\|/h_j \leq K$ nach Voraussetzung.

Die Behauptung für $\underline{i = 2}$ folgt so:
Für $x \in [a, b]$ gibt es $x_j = x_j(x)$ mit $|x_j(x) - x| \leq \frac{1}{2}\|\Delta\|$.
Mit

$$f''(x) - s''(x) = f''(x_j(x)) - s''(x_j(x)) + \int_{x_j(x)}^{x} (f^{(3)}(t) - s^{(3)}(t))dt$$

erhält man wegen (5.29) und $K \geq 1$:

$$\begin{aligned} |f''(x) - s''(x)| &\leq \tfrac{3}{4} L \, \|\Delta\|^2 + LK \, \|\Delta\|^2 \\ &\leq \tfrac{7}{4} LK \, \|\Delta\|^2 \end{aligned}$$

Nun zeigen wir die Behauptung für $\underline{\text{i=1}}$:
Es gilt

$$f(x_j) = s(x_j), \qquad j = 0, \ldots, n.$$

Außer $\xi_0 := a$, $\xi_{n+1} := b$ gibt es daher nach dem Satz von Rolle n Punkte $\xi_j \in (x_{j-1}, x_j)$ mit

$$f'(\xi_j) = s'(\xi_j), \qquad j = 0, \ldots, n+1.$$

Zu jedem $x \in [a, b]$ kann man also $\xi_j(x)$ wählen mit

$$|\xi_j(x) - x| \leq \|\Delta\|.$$

Damit erhält man

$$|f'(x) - s'(x)| = \left| \int_{\xi_j(x)}^{x} (f''(t) - s''(t))dt \right| \leq \frac{7}{4} LK\|\Delta\|^3$$

Anlalog zum Fall $i = 1$ ergibt sich schließlich die Behauptung für $\underline{i = 0}$:

$$|f(x) - s(x)| = \left| \int_{x_j(x)}^{x} (f'(t) - s'(t))dt \right| \leq \frac{7}{4} LK\|\Delta\|^4$$

◇

5.5 Numerische Differentiation

Häufig werden in Anwendungen Ableitungen benötigt, die nicht mehr oder nur sehr aufwendig 'von Hand' zu berechnen sind. Die Interpolation bietet eine einfache Möglichkeit auf numerischem Wege die Ableitungen zu approximieren. Anstelle die Funktion selber an einer Stelle x abzuleiten, approximieren wir die Funktion um x durch geeignete Polynome und leiten dann einfach das so erhaltene Polynom an der Stelle x ab. Hierbei nutzen wir die leichte Ableitbarkeit von Polynomen aus.

Beispiel 5.35. *Gesucht sei $f'(x_1)$ an einer Stelle x_1 für eine gegebene Funktion $f : \mathbb{R} \to \mathbb{R}$. Sei $h > 0$ eine kleine Zahl und $x_0 = x_1 - h$, $x_2 = x_1 + h$. Mittels der Interpolationsformel nach Lagrange erhalten wir ein Interpolationspolynom vom Grad 2 mit $f_i = f(x_i)$, $i = 0, 1, 2$, das nach leichter Umformung übergeht in*

(5.30)
$$f(x) \approx P(x) = \frac{f_0(x - x_1)(x - x_2)}{2h^2} - 2\frac{f_1(x - x_0)(x - x_2)}{2h^2} + \frac{f_2(x - x_0)(x - x_1)}{2h^2}$$

Einmaliges Ableiten ergibt

$$f'(x) \approx P'(x) = \frac{(f_2 - 2f_1)(x - x_0) + (f_0 + f_2)(x - x_1) + (f_0 - 2f_1)(x - x_2)}{2h^2}$$

Auswerten von $P'(x)$ an der Stelle x_1 ergibt dann:

(5.31)
$$\boxed{f'(x_1) \approx P'(x_1) = \frac{f_2 - f_0}{2h}}$$

Beispiel 5.36. *Soll auch noch die zweite Ableitung $f''(x_1)$ berechnet werden, so leiten wir (5.30) erneut ab und erhalten:*

$$\boxed{f''(x_1) \approx P''(x_1) = \frac{f_2 - 2f_1 + f_0}{h^2}} \tag{5.32}$$

Natürlich lassen sich auf diesem Wege eine Reihe weiterer Formeln herleiten. Besteht der Bedarf nach höheren Ableitungen ist das Interpolationspolynom von entsprechend höherem Grad zu wählen. Es ist jedoch zu beachten, dass wir die Interpolation durch Polynome von höherem Grade bereits als numerisch instabil festgestellt hatten, so dass damit zu rechnen ist, dass auch die numerische Differentiation dann instabil werden kann.

Kapitel 6

Integration

6.1 Einführung und Aufgabenstellung

Die Integration von Funktionen ist eine elementare mathematische Operation, die in vielen Formeln benötigt wird. Im Gegensatz zur Ableitung, die für praktisch alle mathematischen Funktionen explizit analytisch berechnet werden kann, gibt es viele Funktionen, deren Integrale man nicht explizit angeben kann. Verfahren zur numerischen Integration (man spricht auch von *Quadratur*) spielen daher eine wichtige Rolle, sowohl als eigenständige Algorithmen als auch als Basis für andere Anwendungen.

Das Problem läßt sich hierbei sehr leicht beschreiben. Für eine Funktion $f : \mathbb{R} \to \mathbb{R}$ soll das Integral

$$I := \int_a^b f(x)\, dx \tag{6.1}$$

auf einem Intervall $[a, b]$ berechnet werden.

6.2 Newton–Cotes–Formeln

Die *Stützstellen* x_i seien äquidistant und enthalten die Randpunkte des Intervalls, d.h.

$$x_i = a + i \cdot h, \quad h = \frac{b-a}{n}, \quad i = 0, \ldots, n. \tag{6.2}$$

Sei P_n das interpolierende Polynom mit
a) Grad $P_n \leq n$,

b) $P_n(x_i) = f_i := f(x_i)$, $i = 0, \ldots, n$.

Als *Approximation für I* nehmen wir den Ausdruck

$$I \approx I_n := \int_a^b P_n(x)\, dx = \sum_{i=0}^{n} a_i f(x_i). \tag{6.3}$$

In der Form von Lagrange lautet P_n:

$$P_n(x) = \sum_{i=0}^{n} f_i L_i(x), \qquad L_i(x) = \prod_{\substack{k=0 \\ k\neq i}}^{n} \frac{x - x_k}{x_i - x_k}$$

Damit folgt

$$I_n = \sum_{i=0}^{n} f_i \underbrace{\int_a^b L_i(x)\, dx}_{=:a_i}$$

Die Koeffizienten in (6.3) ergeben sich daher zu

$$a_i = \int_a^b L_i(x)\, dx = \int_a^b \prod_{\substack{k=0 \\ k\neq i}}^{n} \frac{x - x_k}{x_i - x_k}\, dx \tag{6.4}$$

Mit der Substitution $x = a + s \cdot h$, $s \in [0, n]$, $dx = h \cdot ds$ erhalten wir die *Formeln von Newton–Cotes*:

$$\boxed{\begin{aligned} I_n &= \sum_{i=0}^{n} a_i f(x_i), \quad f_i = f(a + i \cdot h) \\ a_i &= h \int_0^n \prod_{\substack{k=0 \\ k\neq i}}^{n} \frac{s - k}{i - k}\, ds =: hA_i, \quad A_i \in \mathbb{Q} \end{aligned}} \tag{6.5}$$

Beispiel 6.1. $n = 1$ *(Trapezregel): Es ist* $h = b - a$, $A_0 = A_1 = \frac{1}{2}$ *und damit*

$$I_1 = \frac{h}{2}\left(f(a) + f(b)\right)$$

Beispiel 6.2. $n = 2$ *(Simpsonregel): Es ist* $h = \frac{b-a}{2}$, $A_0 = \frac{1}{3}$, $A_1 = \frac{4}{3}$, $A_2 = \frac{1}{3}$ *und damit*

$$I_2 = \frac{h}{3}\left(f(a) + 4f(\frac{a+b}{2}) + f(b)\right)$$

Bemerkung 6.3. *Allgemein gilt:*

$$\sum_{i=0}^{n} A_i = n$$

(d.h. $\sum_{i=0}^{n} a_i = 1$*) und* $A_i = A_{n-i}$ *(Symmetrie).*

Die folgende Tabelle enthält die Koeffizienten A_i für $n \leq 4$:

n	A_0	A_1	A_2	A_3	A_4	Bezeichnung
1	$\frac{1}{2}$	$\frac{1}{2}$				Trapezregel
2	$\frac{1}{3}$	$\frac{4}{3}$	$\frac{1}{3}$			Simpson–Regel
3	$\frac{3}{8}$	$\frac{9}{8}$	$\frac{9}{8}$	$\frac{3}{8}$		Newton'sche $\frac{3}{8}$–Regel
4	$\frac{14}{45}$	$\frac{64}{45}$	$\frac{24}{45}$	$\frac{64}{45}$	$\frac{14}{45}$	Milne–Regel

Bemerkung 6.4. *Für* $n \geq 8$ *können negative Gewichte* A_i *auftreten und die Formeln werden dann aufgrund von Rundungsfehlereinfluß unbrauchbar.*

Es gilt der Fehler

$$R_n(f) = I - I_n = \int_a^b f(x)\,dx - \int_a^b P_n(x)\,dx. \tag{6.6}$$

Offensichtlich ist $R_n(f) = 0$, falls f Polynom vom Grade $\leq n$ ist. Der folgende Satz gibt eine Abschätzung für den Fehler

Satz 6.5 (Fehlerabschätzung).

1. *Für* $f \in C^{n+1}[a,b]$ *gilt*

$$|R_n(f)| \leq h^{n+2} c_n \max_{[a,b]} \left|f^{(n+1)}(x)\right|$$

mit

$$c_n = \frac{1}{(n+1)!} \int_0^n \prod_{i=0}^{n} |s-i|\,ds.$$

2. *Für* n *gerade und* $f \in C^{n+2}[a,b]$ *gilt*

$$|R_n(f)| \leq h^{n+3} c_n^* \max_{[a,b]} \left|f^{(n+2)}(x)\right|, \qquad c_n^* = \frac{n}{2} c_n$$

Beweis: Wir beweisen nur den ersten Teil der Aussage: Nach der Restgliedformel der Polynominterpolation (5.7) existiert ein $\xi \in [a,b]$ mit

$$f(x) - P_n(x) = \frac{L(x)}{(n+1)!} f^{(n+1)}(\xi), \qquad L(x) = \prod_{i=0}^{n} (x - x_i).$$

Hieraus folgt

$$|R_n(f)| \leq \frac{1}{(n+1)!} \int_a^b |L(x)|\, dx \cdot \max_{[a,b]} \left| f^{(n+1)}(x) \right|. \tag{6.7}$$

Mit der Substitution $x = a + s \cdot h$, $s \in [0, n]$ ergibt sich

$$\int_a^b |L(x)|\, dx = \int_a^b \prod_{i=0}^{n} |x - x_i|\, dx = h^{n+2} \int_0^n \prod_{i=0}^{n} |s - i|\, ds.$$

Zusammen mit (6.7) erhält man die Behauptung. ◇

Da das Maximum hoher Ableitungen von f sehr schwer zu bestimmen ist, sind die Formeln zur praktischen Fehlerabschätzung i.a. unbrauchbar. Ihr Nutzen liegt in der Information, mit welcher Potenz von h der Fehler abfällt.

Beispiel 6.6. $n = 1$ *(Trapezregel):*

$$|R_1(f)| \leq \frac{h^3}{12} \max_{[a,b]} |f^{(2)}(x)|.$$

Beispiel 6.7. $n = 2$ *(Simpson–Regel):*

$$|R_2(f)| \leq \frac{h^5}{90} \max_{[a,b]} |f^{(4)}(x)|.$$

Bemerkung 6.8. *Bei geradem n gewinnt man durch den Übergang zu $n+1$ keine Potenz von h.*

Beispiel 6.9. *Es sei* $I = \int_0^1 e^x\, dx = e - 1 = 1.7182\ldots$
$I_1 = \frac{1}{2}(1 + e) = 1.8591\ldots$
$I_2 = \frac{1}{6}(1 + 4e^{1/2} + e) = 1.7189\ldots$
$I_3 = \frac{1}{8}(1 + 3e^{1/3} + 3e^{2/3} + e) = 1.7185\ldots$

6.3 Zusammengesetzte Newton–Cotes–Formeln

Wie beschrieben kann man höhere Genauigkeiten nicht durch immer höhere Polynomgrade erwirken. Es tauchen also die gleichen prinzipiellen Probleme auf, wie bei der Interpolation durch Polynome. Dies ist nicht weiter verwunderlich, da Interpolationspolynome ja den hier beschriebenen Verfahren zu Grunde liegen. Formeln höherer Genauigkeit kann man konstruieren, indem man (ähnlich wie bei der Splineinterpolation) die oben angegebenen Regeln auf Teilintervalle anwendet. Bei der Integration fällt jedoch der mühsame Weg der zusätzlichen

Glattheitsbedingungen an den Nahtstellen weg, da wir ja nicht an einer schönen Approximation der Funktion, sondern 'nur' an einer guten Approximation des Integrals interessiert sind.
Wir beschränken uns in diesem Abschnitt auf die Herleitung der zusammengesetzten Trapezregel, eine Übertragung auf die anderen Newton–Cotes Formeln verläuft analog.

6.3.1 Zusammengesetzte Trapezregel

Wir betrachten wieder äquidistante Stützstellen (6.2). Die Anwendung der Trapezregel auf das Teilintervall $[x_i, x_{i+1}]$ ergibt die Approximation

$$\int_{x_i}^{x_{i+1}} f(x)\,dx \approx \frac{h}{2}\left(f(x_i) + f(x_{i+1})\right), \qquad i = 0, 1, \ldots n-1.$$

Durch Summation erhalten wir die *zusammengesetzte Trapezregel*:
(6.8)

$$\begin{aligned}\int_a^b f(x)\,dx \approx T(h) &:= \sum_{i=0}^{n-1} \frac{h}{2}\left(f(x_i) + f(x_{i+1})\right) \\ &= h\left[\frac{f(a)}{2} + f(a+h) + \ldots + f(b-h) + \frac{f(b)}{2}\right].\end{aligned}$$

Die zusammengesetzte Trapezregel ist in Abbildung 6.1 dargestellt.

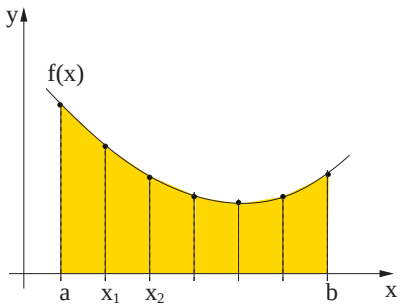


Abbildung 6.1: Zusammengesetzte Trapezregel.

Der Gesamtfehler ergibt sich aus der Summation der Fehler (bzw. aus Beispiel 6.6) zu

$$\left|T(h) - \int_a^b f(x)\,dx\right| \leq n \cdot \frac{h^3}{12} \max_{[a,b]} |f^{(2)}(x)| = (b-a) \cdot \frac{h^2}{12} \max_{[a,b]} |f^{(2)}(x)|,$$

also ein Fehler in der Größenordnung $O(h^2)$.

6.3.2 Verfeinerung der zusammengesetzten Trapezregel

Wir wollen in diesem Abschnitt einige weitere Besonderheiten aufweisen, wie aus unscheinbaren Zusammenhängen numerisches Kapital geschlagen werden kann. Eine ebenso anschauliche Approximation des Integrals (6.1), welche der Riemannschen Summe entspricht, bildet die *Mittelpunktsumme*

$$\int_a^b f(x)\,dx \approx M(h) := h\sum_{i=0}^{n-1} f\left(x_{i+\frac{1}{2}}\right), \qquad x_{i+\frac{1}{2}} := a + \left(i+\frac{1}{2}\right)h. \tag{6.9}$$

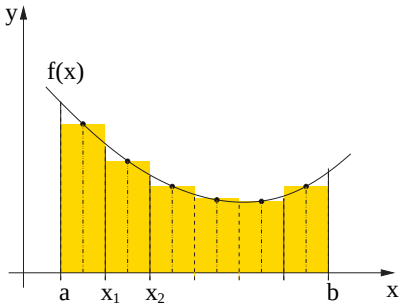


Abbildung 6.2: Zusammengesetzte Mittelpunktregel.

$M(h)$ stellt die Fläche unterhalb der Treppenkurve in Abbildung 6.2 dar und ist von gleicher Fehlerordnung $O(h^2)$ wie die zusammengesetzte Trapezregel. Aus (6.8) und (6.9) folgt unmittelbar die Relation

$$T\left(\frac{h}{2}\right) = \frac{1}{2}\left[T(h) + M(h)\right] \tag{6.10}$$

Bemerkung 6.10. *Die Relation (6.10) erlaubt eine Verbesserung der zusammengesetzten Trapezregel durch sukzessive Halbierung der Schrittlänge in der Weise, daß zu bereits berechneten Näherungen $T(h)$ auch noch $M(h)$ berechnet wird. Bei jeder Halbierung der Schrittweite h wird der Rechenaufwand (gemessen mit der Anzahl der Funktionsauswertungen) etwa verdoppelt, doch werden die schon berechneten Funktionswerte auf ökonomische Weise wieder verwendet. Die sukzessive Halbierung kann etwa dann abgebrochen werden, wenn für eine gegebene Fehlerschranke $\varepsilon > 0$ $|T(h) - M(h)| < \varepsilon$ gilt. Dann ist der Fehler $|T(\frac{h}{2}) - I|$ im allgemeinen höchstens gleich ε.*

6.4 Die Gaußsche Integrationsmethode

6.4.1 Orthogonalpolynome

Der Raum $V = C[a,b]$ mit dem inneren Produkt

$$\langle f,g\rangle = \int_a^b f(x)g(x)w(x)dx, \quad w \in C(a,b),\ w(x) > 0 \qquad \text{für } a < x < b, \tag{6.11}$$

ist ein sogenannter Prä-Hilbertraum. Wendet man das SCHMIDT'sche Orthogonalisierungsverfahren auf die Monome $u_n(x) = x^n$ an, so erhält man Orthogonalpolynome mit Höchstkoeffizient $a_n = 1$:

$$\begin{aligned} \tilde{p}_n \in \tilde{\Pi}_n &:= \{x^n + \sum_{k=0}^{n-1} a_k x^k\}\ (n = 0,1,2,\ldots), \\ \langle \tilde{p}_i, \tilde{p}_k\rangle &= 0 \text{ für } i \neq k, \qquad \langle \tilde{p}_i, \tilde{p}_i\rangle = 1 \text{ bei Orthonormalpolynomen} \end{aligned}$$

Die Polynome $\tilde{p}_0 \ldots, \tilde{p}_n$ bilden eine Basis von Π_n.
Darüberhinaus gilt der bemerkenswerte

Satz 6.11 (Nullstellensatz). *Das Orthogonalpolynom $\tilde{p}_n \in \tilde{\Pi}_n$ hat in (a,b) genau n einfache Nullstellen.*

Beweis: Seien $x_1 \ldots, x_m$ $(m \geq 0)$ die Vorzeichenwechsel von $\tilde{p}_n$ in (a,b). Wir zeigen, dass $m = n$ gilt. Mit dem Polynom

$$q(x) = \prod_{i=1}^{m}(x - x_i) \in \Pi_m$$

hat das Polynom $q \cdot \tilde{p}_n$ kontantes Vorzeichen. Für $m < n$ würde folgen

$$\langle q, \tilde{p}_n\rangle = \int_a^b q(x)\tilde{p}_n(x)w(x)dx = 0,$$

also $q \cdot \tilde{p}_n \equiv 0$ in (a,b) : Widerspruch. ◇

Beispiel 6.12. (Legendre–Polynome): *Es sei $[a,b] = [-1,1]$, $w = 1$.*
Man prüft leicht nach, dass die Legendre–Polynome

$$\tilde{L}_n(x) = \frac{n!}{(2n)!}\frac{d^n(x^2-1)^n}{dx^n} = x^n + \ldots,\ n \geq 0 \tag{6.12}$$

ein Orthogonalsystem bilden mit

$$\langle \tilde{L}_n, \tilde{L}_n\rangle = \frac{2n+1}{2}\frac{1}{(2^n n!)^2}\ .$$

Zum Beispiel ist

$$\tilde{L}_1(x) = x,\ \tilde{L}_2(x) = x^2 - \frac{1}{3},\ \tilde{L}_3(x) = x^3 - \frac{3}{5}x.$$

Beispiel 6.13. (Tschebychev–Polynome): *Es sei* $[a,b] = [-1,1]$, $w(x) = 1/\sqrt{1-x^2}$:
Die Tschebychev-Polynome $T_n \in \Pi_n$ *werden rekursiv definiert durch*

$$\begin{aligned} T_{n+1}(x) &= 2xT_n(x) - T_{n-1}(x) \quad (n = 1, 2, \ldots), \\ T_0(x) &= 1, \quad T_1(x) = x. \end{aligned} \tag{6.13}$$

Mit der Substitution

$$x = \cos\theta, \quad \theta = arcos\ x$$

gelangt man zur Darstellung

$$T_n(x) = \cos(n\theta), \quad \theta = arcos\ x,$$

denn die Rekursionsformel

$$\cos((n+1)\theta) = 2\cos\theta\cos(n\theta) - \cos((n-1)\theta)$$

entspricht gerade der Rekursion (6.13). Damit bestätigt man die Orthogonalität bzgl. $w(x) = 1/\sqrt{1-x^2}$*:*

$$\begin{aligned} \int_{-1}^{1} T_i(x)T_k(x)\tfrac{dx}{\sqrt{1-x^2}} &= \int_0^{\pi} \cos(i\theta)\cos(k\theta)\tfrac{\sin\theta}{\sin\theta}d\theta \\ &= \begin{cases} 0 & \text{für } i \neq k \\ \pi & \text{für } i = k = 0 \\ \frac{\pi}{2} & \text{für } i = k \neq 0. \end{cases} \end{aligned} \tag{6.14}$$

Die Darstellung $T_n(x) = \cos(n\theta)$ *zeigt, dass* $T_n(x)$ *die Nullstellen (Tschebychev–Abszissen)*

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

und die Extremalstellen

$$x_k^{(l)} = \cos\left(\frac{k\pi}{n}\right) \quad (k = 0, 1, \ldots, n), n \geq 1,$$

mit

$$T_n(x_k^{(l)}) = (-1)^k,$$

besitzt. Z.B. liefert die Rekursion (6.13)

$$\begin{aligned} T_2(x) &= 2x^2 - 1, \quad T_3(x) = 4x^3 - 3, \ldots, \\ T_n(x) &= 2^{n-1}\, x^n - \cdots . \end{aligned}$$

Zu normierten Polynomen $\widetilde{T}_n \in \widetilde{\Pi}_n$ gelangt man durch die Normierung

$$\widetilde{T}_n(x) = \frac{1}{2^{n-1}} T_n(x).$$

Weitere Orthogonalpolynome bzgl. anderer Gewichte $w(x)$ finden sich in M. ABRAMOVITZ, F. STEGUN: Handbook of Mathematical Functions.

6.4.2 Gaußintegration

Sei $f \in C[a,b]$ und sei $w \in C[a,b]$ eine positive Gewichtsfunktion mit $w(x) > 0$ für $x \in (a,b)$. Wir suchen eine Integrationsformel für das Integral

$$I(f) = \int_a^b w(x) f(x) dx. \tag{6.15}$$

Im folgenden benutzen wir die Bezeichnungen:

$$\begin{aligned} \Pi_j : &\quad \text{Polynome vom Grade } \leq j \qquad (j = 0, 1, 2, \ldots) \\ \widetilde{\Pi}_j : &= \{p \in \Pi_j \mid p(x) = x^j + a_{j-1}x^{j-1} + \ldots + a_0\}. \end{aligned}$$

Eine Integrationsformel für $I(f)$ der Form

$$G_n(f) = \sum_{i=1}^{n} A_i f(x_i) \tag{6.16}$$

hat die $2n$ freien Parameter A_i und x_i. Die Formeln von Newton–Cotes (6.5) mit äquidistanten Stützstellen sind exakt in Π_{n-1}. Wir wollen nun fordern, dass

$$G_n(f) = I(f) \;\text{ für alle }\; f \in \Pi_{2n-1},$$

d.h. $G_n(f)$ ist exakt in Π_{2n-1}. Dies ergibt gerade $2n$ Bedingungen für die $2n$ Parameter. Der folgende Satz zeigt, dass diese Forderung maximal ist.

Satz 6.14. *Es gibt keine Formel $G_n(f)$ des Typs (6.16), die in Π_{2n} exakt ist.*

Beweis: Annahme: $G_n(f) = \int_a^b w(x)f(x)dx$ für $f \in \Pi_{2n}$.
Mit

$$f := \prod_{i=1}^{n}(x - x_i)^2 \in \Pi_{2n}$$

erhält man einen Widerspruch wegen

$$G_n(f) = 0 \neq \int_a^b w(x)f(x)dx > 0.$$

◇

Zur Konstruktion einer in Π_{2n-1} exakten Formel $G_n(f)$ benutzt man die zur Gewichtsfunktion w gehörenden Orthogonalpolynome $\tilde{p}_n$ bzgl. des Skalarproduktes (vgl. (6.11))

$$\langle f, g\rangle = \int_a^b f(x)g(x)w(x)dx, \qquad f, g \in C[a, b].$$

Das Polynom $\tilde{p}_n \in \widetilde{\Pi}_n$ hat nach Satz 6.11 n Nullstellen $x_1, \ldots, x_n \in (a, b)$. Damit gelangen wir zum Hauptresultat dieses Abschnittes.

Satz 6.15. *Seien $x_1, \ldots, x_n$ die Nullstellen des Orthogonalpolynoms $\tilde{p}_n$ und sei*

$$L_i(x) = \prod_{\substack{k=1\\k\neq i}}^{n} \frac{x - x_k}{x_i - x_k}, \qquad i = 1, \ldots, n.$$

Dann ist die Integrationsformel

$$G_n(f) = \sum_{i=1}^{n} A_i f(x_i), \qquad A_i := \int_a^b w(x)L_i(x)^2dx$$

exakt in Π_{2n-1} und es gilt

$$A_i > 0.$$

Beweis: Nach der Interpolationsformel von Lagrange gilt

$$f(x) = \sum_{i=1}^{n} L_i(x)f(x_i) \qquad \text{für alle } f \in \Pi_{n-1}.$$

Daher ist

$$G_n(f) = \sum_{i=1}^{n} \int_a^b w(x)L_i(x)dx\, f(x_i)$$

exakt in Π_{n-1}. Ein Polynom $f \in \Pi_{2n-1}$ faktorisieren wir in

$$f = q \cdot \tilde{p}_n + r \qquad \text{mit } q, r \in \Pi_{n-1}$$

$$\begin{aligned}
\Rightarrow \quad I(f) &= \int_a^b w(x) f(x) dx = \underbrace{\int_a^b w(x) q(x) \tilde{p}_n(x) dx}_{=0,\,(\text{da } q \in \Pi_{n-1})} + \int_a^b w(x) r(x) dx \\
&= G_n(r) \qquad (\text{da } r \in \Pi_{n-1}) \\
&= G_n(r) + \underbrace{G_n(q \cdot \tilde{p}_n)}_{=0,\,(\text{da } \tilde{p}_n(x_i)=0,\ i=1,\ldots,n)} \\
&= G_n(r + q \cdot \tilde{p}_n) \\
&= G_n(f).
\end{aligned}$$

Also ist $G_n(f)$ exakt in Π_{2n-1} und die Formeln für die Gewichte A_i folgen so: Es ist $L_i^2 \in \Pi_{2n-2}$, also

$$\int_a^b w(x) L_i(x)^2 dx = G_n(L_i^2) = \sum_{k=1}^n A_k L_i(x_k)^2 = \sum_{k=1}^n A_k \delta_{ik} = A_i.$$

◇

Beispiel 6.16. *Sei* $[a, b] = [-1, 1]$, $w(x) \equiv 1$,

$$\tilde{p}_n(x) = \frac{n!}{(2n)!} \frac{d^n}{dx^n}(x^2 - 1)^n, \qquad n = 0, 1, 2, \ldots,$$

Legendre-Polynome bis auf Normierungsfaktoren:

$$\tilde{p}_1(x) = x, \quad \tilde{p}_2(x) = x^2 - \frac{1}{3}, \quad \tilde{p}_3(x) = x^3 - \frac{3}{5}x$$

n	x_1	x_2	x_3	A_1	A_2	A_3
1	*0*			*2*		
2	$-\sqrt{\frac{1}{3}}$	$+\sqrt{\frac{1}{3}}$		*1*	*1*	
3	$-\sqrt{\frac{3}{5}}$	*0*	$\sqrt{\frac{3}{5}}$	$\frac{5}{9}$	$\frac{8}{9}$	$\frac{5}{9}$

$$I = \int_{-1}^1 e^x dx = 2.350402.$$

Die Simpson-Regel liefert

$$I_2 = 2.362054.$$

Dagegen ist mit gleich vielen Funktionsauswertungen

$$G_3 = 2.350337.$$

Für den Fehler der Gauß'schen Integrationsmethode gilt die folgende Abschätzung: (vgl. Stoer I, S. 126).

Satz 6.17. *Sei $f \in C^{2n}[a,b]$ und sei $G_n(f)$ die in Satz 6.15 definierte Integrationsformel, dann gilt*

$$I(f) - G_n(f) = \frac{f^{(2n)}(\xi)}{(2n)!}\langle p_n, p_n\rangle$$

mit einem $\xi \in (a,b)$.

Die Gauß-Formeln liefern im Vergleich zu den Newton-Cotes-Formeln bzw. den Extrapolationsverfahren die genauesten Resultate (gemessen an der Zahl der Funktionsauswertungen). Im Gegensatz zu Extrapolationsverfahren können jedoch beim Übergang von einem Index n zu $n+1$ die bis dahin berechneten Funktionswerte $f(x_i)$ nicht weiter verwendet werden. Daher sind in der Praxis Extrapolationsverfahren vorzuziehen.

6.5 Integration und Extrapolation

6.5.1 Euler-Maclaurin'sche Summenformel

Durch Anwendung der Extrapolation auf die zusammengesetzte Trapezregel wollten wir nun Formeln konstruieren, deren Fehler mit einer hohen Potenz von h abfällt. Grundlage dieser Extrapolationsverfahren ist die folgende asymptotische Entwicklung von $T(h)$ nach Potenzen von h^2.

Satz 6.18 (Euler-Maclaurin'sche Summenformel). *Für $f \in C^{2m+2}[a,b]$ gilt die Entwicklung*

$$\boxed{T(h) = \tau_0 + \tau_1 h^2 + \tau_2 h^4 + \ldots + \tau_m h^{2m} + \alpha_{m+1}(h)h^{2m+2}}$$

mit

1. $\tau_0 = \int_a^b f(x)dx,$

2. $\tau_k = \frac{(-1)^{k+1}B_k}{(2k)!}(f^{(2k-1)}(b) - f^{(2k-1)}(a)), \qquad k = 1, \ldots, m$

3. $\alpha_{m+1}(h) = \frac{1}{(2m+2)!}\int_a^b f^{(2m+2)}(x) K_{2m+2}\left(\frac{x-a}{h}\right) dx$
mit $K_{2m+2} \in C[0, n]$ *und*

$$\int_a^b K_{2m+2}\left(\frac{x-a}{h}\right) dx = (-1)^m B_{m+1}(b-a).$$

Hierbei sind B_k *die* Bernoulli-Zahlen

$$B_1 = \frac{1}{6}, \quad B_2 = \frac{1}{30}, \quad B_3 = \frac{1}{42}, \ldots.$$

Beweis: Zum Beweis vergleiche man etwa Stoer 1. ◇

6.5.2 Anwendung der Extrapolation auf die Integration

Nach dem vorigen Satz gilt

$$T(h) = \underbrace{\tau_0 + \tau_1 h^2 + \ldots + \tau_m h^{2m}}_{\text{Polynom vom Grade } m \text{ in } h^2} + \alpha_{m+1}(h) h^{2m+2}.$$

Es interessiert die Größe

$$\tau_0 = \int_a^b f(x)dx = \lim_{h \to 0} T(h).$$

Idee der Extrapolation: Zu $m+1$ Schrittweiten

$$h_0 = b - a, \; h_1 = \frac{h_0}{n_1}, \ldots, h_m = \frac{h_0}{n_m}; \qquad n_i < n_{i+1} \; (n_i \in \mathbb{N})$$

bestimme man Trapezsummen

$$T_{i0} := T(h_i), \qquad i = 0, \ldots, m,$$

und dann durch Interpolation dasjenige Polynom in h^2

$$\tilde{T}_{mm}(h) := a_0 + a_1 h^2 + \ldots + a_m h^{2m} \tag{6.17}$$

mit

$$\tilde{T}_{mm}(h_i) = T(h_i), \qquad i = 0, \ldots, m.$$

Dann ist

$$\tilde{T}_{mm}(0) = a_0 \approx \tau_0 \qquad (\text{Extrapolation}).$$

Beispiel 6.19. $h_0 = b - a, \quad h_1 = \frac{b-a}{2}$. *Dann ist*

$$T_{11} := \tilde{T}_{11}(0) = L_0(0)T(h_0) + L_1(0)T(h_1)$$

mit

$$L_i(x) = \prod_{\substack{k=0 \\ k\neq i}}^{1} \frac{x - x_k}{x_i - x_k}, \quad x_i = h_i^2, \quad i = 0, 1.$$

Für $x = 0$ *erhält man*

$$L_0(0) = \frac{-h_1^2}{h_0^2 - h_1^2} = -\frac{1}{3}, \quad L_1(0) = \frac{-h_0^2}{h_1^2 - h_0^2} = \frac{4}{3}$$

und daher

$$\begin{aligned} T_{11} &= -\frac{1}{3}\underbrace{\frac{b-a}{2}\left(f(a) + f(b)\right)}_{=T(h_0)} + \frac{4}{3}\underbrace{\frac{b-a}{2}\left(\frac{f(a)}{2} + f(\frac{a+b}{2}) + \frac{f(b)}{2}\right)}_{=T(h_1)} \\ &= \frac{h_1}{3}\left(f(a) + 4f(\frac{a+b}{2}) + f(b)\right) \end{aligned}$$

und dies ist die Simpson-Regel!

Die Berechnung des Wertes $\tilde{T}_{mm}(0) = a_0$ in (6.17) erfolgt mit dem Algorithmus von Neville; vgl. (5.5). Dazu sei $1 \leq k \leq i \leq m$ und $\tilde{T}_{ik}(h)$ dasjenige Polynom in h^2 mit

$$\tilde{T}_{ik}(h_j) = T_{j0} := T(h_j) \text{ für } j = i - k,\ i - k + 1, \ldots, i.$$

Die Rekursion für $T_{ik} := \tilde{T}_{ik}(0)$ ergibt sich aus dem Algorithmus von Neville mit $x_i = h_i^2,\ x = 0$, zu

$$T_{ik} = T_{i,k-1} + \frac{T_{i,k-1} - T_{i-1,k-1}}{\left(\frac{h_{i-k}}{h_i}\right)^2 - 1}, \qquad 1 \leq k \leq i \leq m.$$

Die Berechnung des Tableaus mit den Größen T_{ik} erfolgt spaltenweise, z. B.

$$\begin{array}{c|ccccccc} h_0 & T_{00} & & & & & & \\ h_1 & T_{10} & & T_{11} & & & & \\ & & \searrow & & & & & \\ h_2 & T_{20} & \rightarrow & T_{21} & & T_{22} & & \\ & & & & & & \searrow & \\ h_3 & T_{30} & & T_{31} & & T_{32} & \rightarrow & T_{33} \end{array}$$

Beispiel 6.20. $I = \int_0^1 e^x dx = 1.718281828, \quad m = 3$

h_i	T_{i0}	T_{i1}	T_{i2}	T_{33}
1	*1.859140914*			
$\frac{1}{2}$	*1.753931092*	*1.718861151*		
$\frac{1}{4}$	*1.727221904*	*1.718318841*	*1.718282687*	
$\frac{1}{8}$	*1.720518792*	*1.718284155*	*1.718281842*	*1.718281828*

In der Praxis haben sich zwei Schrittweitenfolgen bewährt:

$$\begin{array}{lll} \underline{\text{Romberg-Folge:}} & h_0 = b-a,\ h_i = \frac{h_0}{2^i}, & i = 0, 1, \ldots \\ \underline{\text{Bulirsch-Folge:}} & h_0 = b-a,\ h_1 = \frac{h_0}{2}, & h_2 = \frac{h_0}{3},\ h_3 = \frac{h_0}{4}, \\ & h_4 = \frac{h_0}{6}, h_5 = \frac{h_0}{8}, \ldots & \end{array}$$

Bemerkung 6.21. *Die Bulirsch-Folge hat den Vorteil, dass bei ihr die Rechenarbeit für die Berechnung neuer $T(h_i)$ nicht so rasch ansteigt wie bei der Romberg-Folge. Bei der praktischen Durchführung beachte man, dass bei der Berechnung von $T(h_{i+1})$ auf die schon bei $T(h_i)$ berechneten Funktionswerte zurückgegriffen wird, vgl. etwa die Verfeinerung der zusammengesetzten Trapezregel mittels der Mittelpunktsumme.*

6.5.3 Integrationsfehler

Im folgenden soll ein Ausdruck für den Fehler

$$T_{mm} - \int_a^b f(x)dx$$

angegeben werden.

Hilfssatz 6.22. *Seien x_i, $i = 0, \ldots, m$, paarweise verschiedene Zahlen und sei*

$$L_i(x) = \prod_{\substack{k=0 \\ k \neq i}}^{m} \frac{x - x_k}{x_i - x_k}, \quad i = 0, \ldots, m.$$

Dann gilt

$$\sum_{i=0}^{m} x_i^j L_i(0) = \begin{cases} 1 & , \quad j = 0 \\ 0 & , \quad j = 1, \ldots, m \\ (-1)^m x_0 \ldots x_m & , \quad j = m+1 \end{cases}$$

Beweis: Man setze $x = 0$ in den beiden folgenden Identitäten ein:

$$\begin{aligned} x^j &\equiv \sum_{i=0}^{m} x_i^j L_i(x), \qquad j = 0, \ldots, m, \\ x^{m+1} &\equiv \sum_{i=0}^{m} x_i^{m+1} L_i(x) + (x - x_0)(x - x_1) \ldots (x - x_m). \end{aligned}$$

Die zweite Identität folgt daraus, dass die rechte Seite gleich der linken Seite in den Punkten x_i, $i = 0, \ldots, m$, ist und die Koeffizieten von x^{m+1} auf beiden Seiten übereinstimmen; vgl. auch die Restgliedformel (5.7). ◇

Die Substitution $\underline{x = h^2,\ x_i = h_i^2}$, in Hilfssatz 6.22 ergibt die Beziehung

$$\sum_{i=0}^{m} h_i^{2j} L_i(0) = \left\{ \begin{array}{lll} 1 & , & j = 0 \\ 0 & , & j = 1, \ldots, m \\ (-1)^m h_0^2\, h_1^2\, \ldots h_m^2 & , & j = m+1 \end{array} \right\}. \tag{6.18}$$

Das Polynom $\tilde{T}_{mm}(h)$ in (6.17) interpolierte die Werte $T(h_i)$, $i = 0, \ldots, m$. Also gilt nach der Formel von Lagrange

$$T_{mm} = \tilde{T}_{mm}(0) = \sum_{i=0}^{m} L_i(0) T(h_i). \tag{6.19}$$

Die asymptotische Entwicklung in Satz 6.18) von $T(h)$ ergab

$$T(h) = \int_a^b f(x)dx + \tau_1 h^2 + \ldots + \tau_m h^{2m} + \alpha_{m+1}(h) h^{2m+2},$$

$$\alpha_{m+1}(h) = \frac{1}{(2m+2)!} \int_a^b f^{(2m+2)}(x) K_{2m+2}(\frac{x-a}{h}) dx,$$

$$\int_a^b K_{2m+2}(\frac{x-a}{h}) dx = (-1)^m B_{m+1}(b-a). \tag{6.20}$$

Mit (6.18),(6.19) folgt dann

$$T_{mm} = \int_a^b f(x)dx + \frac{1}{(2m+2)!} \int_a^b f^{(2m+2)}(x) K(x) dx, \tag{6.21}$$

$$K(x) := \sum_{i=0}^{m} L_i(0) h_i^{2m+2} \, K_{2m+2}\left(\frac{x-a}{h_i}\right). \tag{6.22}$$

Man kann zeigen: Die Funktion $K(x)$ hat gleiches Vorzeichen in $[a,b]$ für die Romberg-Folge und die Bulirsch-Folge h_i. Daher gilt

$$\int_a^b f^{(2m+2)}(x) K(x) dx = f^{(2m+2)}(\xi) \int_a^b K(x) dx, \quad \xi \in [a,b],$$

und mit (6.18),(6.20) und (6.22) haben wir

$$\begin{aligned} \int_a^b K(x) dx &= \sum_{i=0}^{m} L_i(0) h_i^{2m+2} \int_a^b K_{2m+2}(\tfrac{x-a}{h_i}) dx \\ &= (-1)^m h_0^2 h_1^2 \ldots h_m^2 (-1)^m B_{m+1} (b-a) \\ &= (b-a) h_0^2 \ldots h_m^2 B_{m+1} \end{aligned}$$

Insgesamt ergibt sich dann aus (6.21) und den vorigen Beziehungen der Fehler

$$\boxed{T_{mm} - \int_a^b f(x) dx = (b-a) h_0^2 \ldots h_m^2 \tfrac{B_{m+1}}{(2m+2)!} f^{(2m+2)}(\xi)} \, . \tag{6.23}$$

Bei Interpolation mit den Schrittweiten $h_{i-k}, \ldots, h_i$ erhält man auf ähnliche Weise

$$T_{ik} - \int_a^b f(x) dx = (b-a) h_{i-k}^2 \ldots h_i^2 \frac{B_{k+1}}{(2k+2)!} f^{(2k+2)}(\xi). \tag{6.24}$$

Für $k = 0$ gewinnt man hieraus die Abschätzung für den Gesamtfehler der zusammengesetzten Trapezregel zurück wegen $B_1 = \frac{1}{6}$.

Wegen (6.24) verhält sich der Fehler von T_{im} in der $(i+1)$-ten Spalte des Tableaus wie h_{i-m}^{2m+2}, also wie der Fehler eines Verfahrens $(2m+2)$-ter Ordnung. Aus Gründen der Auslöschung geht man in der der Praxis nicht über $m = 6$ hinaus. Man beendet die Rechnung, falls das erste Mal

$$|T_{i,6} - T_{i+1,6}| \leq \epsilon \cdot s$$

erfüllt ist, wobei
ϵ : gewünschte realtive Genauigkeit,
s : grober Näherungswert von $\int_a^b |f(x)| dx$.

Literaturverzeichnis

[1] Büskens, C. *Numerische Mathematik für Naturwissenschaftler und Ingenieure*, Skript, Universität Bayreuth, Bayreuth, 2004.

[2] Cryer, C.W. *Praktische Mathematik I*, Skript, Universität Münster, Münster, 1989.

[3] Deuflhard, P., Hohmann, A. *Numerische Mathematik I*, de Gruyter, Berlin, New York, 1993.

[4] Hämmerlin, G., Hoffmann, K.H. *Numerische Mathematik*, Springer-Verlag, Berlin Heidelberg, 1991.

[5] Maurer, H. *Numerische Mathematik*, Skript, Universität Münster, Münster, 1990.

[6] Schaback, R., Werner. H. *Numerische Mathematik*, Springer-Verlag, Berlin, Heidelberg, New York, 1992.

[7] Schwarz, H.R. *Numerische Mathematik*, B.G. Teubner, Stuttgart, 1993.

[8] Stoer, J. *Einführung in die Numerische Mathematik I,* Springer-Verlag, Berlin Heidelberg, 1983.