NUMERISCHE MATHEMATIK

für das Unterrichtsfach Mathematik

Gabriela SCHRANZ-KIRLINGER

Technische Universität Wien

Institut für Analysis und Scientific Computing

Jänner 2011

Inhaltsverzeichnis

Kapitel 1

Fehlerbetrachtungen

Da für komplexe Problemstellungen eine exakte Lösung meist überhaupt nicht oder nur mit großem Aufwand gefunden werden kann, ist oft die einzige Alternative die Ermittlung einer **Näherungslösung** mit Hilfe eines **numerischen Verfahrens**. Für die meisten Anwendungen ist die Kenntnis einer exakten Lösung ohnehin oft nicht notwendig, oft reicht eine entsprechend genaue numerische Näherung. In dieser Vorlesung werden die wichtigsten Konzepte und Eigenschaften solcher **algorithmisch-numerischer Lösungsmethoden** diskutiert und auf Schwachstellen aufmerksam gemacht.

Durch den Einsatz von numerischen Methoden ist die zu ermittelnde Lösung normalerweise mit unvermeidlichen Fehlern behaftet. Diese Fehlerarten werden in der Numerik in vier Gruppen zusammengefasst:

Modellfehler, **Datenfehler**, **Verfahrensfehler** und **Rechenfehler**

Eine numerische Methode beinhaltet nicht nur ein Rechenverfahren, also einen **Algorithmus**, zur Ermittlung der Näherungslösung sondern auch eine zuverlässige **Schätzung für den Fehler** dieser Näherungslösung, um zu überprüfen, ob die erzielte Genauigkeit für den Anwendungszweck ausreichend ist.

Aber was heißt *Genauigkeit soll für einen bestimmten Anwendungszweck ausreichend sein*?

Beispiel 1.0.1. Landung einer Weltraumkapsel, siehe Abb. 1.1

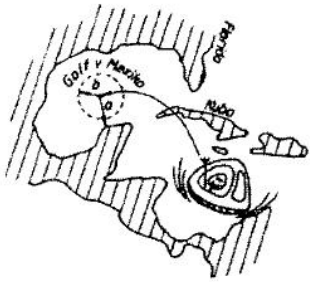


Abbildung 1.1: Landung einer Weltraumkapsel

Aus technischen Gründen kann die Kapsel nicht landen sondern nur wassern. Der beabsichtigte Punkt der Wasserung liegt mitten im Golf von Mexiko. Die geringste Entfernung von diesem Punkt zur Küste ist a, daher ist die Genauigkeitsforderung, also die **Toleanz**, bei der Berechnung des Landemanövers: $\mathit{Toleranz} = a$. Falls darüber hinaus die Kapsel z.B. innerhalb einer Stunde nach der Wasserung geborgen werden muss, und falls die Bergungsschiffe in einer Stunde einen Weg von der Länge b zurücklegen können: $\mathit{Toleranz} = \min(a, b)$.

1.1 Modellfehler

Es gibt kein mathematisches Modell eines realen Vorganges, das die Realität voll erfasst. Zu einem realen Vorgang gibt es i.a. eine ganze Schar mehr oder weniger feiner mathematischer Modelle, bei denen mehr oder weniger Aspekte dieses Vorganges erfasst werden.

Beispiel 1.1.1. Landung (Wasserung) einer Weltraumkapsel.

Meistens: Modellierung der Kapsel als *Massenpunkt*, d.h. die ganze Masse denkt man sich in einem Punkt vereinigt.

Feinere Modellierung: Modellierung der Kapsel als *starren Körper*, ermöglicht die Erfassung von Rotations- und Schlingerbewegungen.

Noch feinere Modellierung: Modellierug als *elastischer Körper*, der unter dem Einfluß von Kräften verformbar oder sogar zerstörbar ist – beim Wiedereintritt in die Erdatmosphäre ist die Kapsel aufgrund ihrer hohen Geschwindigkeit sehr starken Reibungskräften ausgesetzt und es sollte untersucht werden, ob sie diesen Belastungen standhält.

Noch feineres Modell: Berücksichtigung von *thermodynamischen Effekten*; in der Erdatmosphäre werden Reibungskräfte teilweise in Wärme umgewandelt, d.h. es ergibt sich eine starke Hitzeentwicklung und es kommt zum Verglühen des Hitzeschildes.

Ein wie feines Modell der Raumkapsel benötigt wird, hängt vom Anwendungszweck ab: z.B. bei der Berechnung der Landebahn und Ermittlung des Wasserungspunktes genügt i.a. die Modellierung als Massenpunkt. Die dabei gemachten **Modellfehler** sind die Vernachlässigung der Ausdehnung und geometrischen Form der Kapsel der Elastizitätseigenschaften, der thermodynamischen Effekte, ... und es muss untersucht werden, ob die Effekte dieser Modellfehler so klein sind, dass die Genauigkeitstoleranz, die Größen a oder b oder $\min(a, b)$, siehe Abb. 1.1, nicht gefährdet erscheint.

$\Rightarrow$ Mathematisches Modell des Landemanövers: Massenpunkt, der sich unter dem Einfluß von Kräften (Gravitation, Luftwiderstand) bewegt, die Bahn kann durch Lösen eines Systems gewöhnlicher Differentialgleichungen – eines Anfangswertproblems – ermittelt werden. Dabei müssen die Kräfte in jedem Raumpunkt gegeben sein.

$\Rightarrow$ Ein mathematisches Modell des Gravitationsfeldes und der Erdatmosphäre wird benötigt.

Modell der Gravitation: Gesetz von Newton:

$$F = \frac{Gm_1m_2}{r^2}$$

F	=	Gravitationskraft (genauer: Betrag des Kraftvektors)
G	=	Gravitationskonstante
r	=	Abstand
m_2	=	Masse der Kapsel
m_1	=	Massenelement von der Erde oder eventuell vom Mond, von der Sonne, von anderen Planeten

Die Gesamtgravitationskraft, die auf die Kapsel wirkt, ergibt sich durch Aufsummation (Integration) über alle Massenelemente.

Gravitationsmodelle: Vernachlässigung von Gravitationswirkungen des Mondes, der Sonne, der anderen Planeten und Himmelskörper, nur Gravitationswirkungen der Erde werden berücksichtigt. Vernachlässigung ungleichmäßiger Massenverteilungen im Erdinneren und der zeitabhängigen Wasserverteilung auf der Erde (Ebbe-Flut) usw.

Wieder muss versucht werden, die Auswirkungen dieser Modellfehler bei der Modellierung des Gravitationsfeldes auf die Berechnung der Bahn der Kapsel einzuschätzen. Ähnliche Betrachtungen sind bezügliche der Modellierung der Erdatmosphäre zur Berechnung der Reibungskräfte (Luftwiderstand) notwendig, usw.

Oft können Auswirkungen von Modellfehler durch **Interpretation als Datenfehler** abgeschätzt werden. Abschätzung von Datenfehlereffekten sind für zahlreiche mathematische Problemstellungen bekannt.

Aber nicht alle Modellfehler lassen sich als Datenfehler interpretieren: Z.B. dass die Raumkapsel als Massenpunkt oder als starrer Körper modelliert wurde, obwohl sie in Wirklichkeit ein elastisches Gebilde ist. Wollte man die Auswirkungen der Elastizität miterfassen, müsste man den Bereich der gewöhnlichen Differentialgleichungen verlassen. Die Elastizitätstheorie wird in der Physik durch partielle Differentialgleichungen dargestellt. Die Auswirkungen solcher Modellfehler müssten mit Konditionsabschätzungen partieller Differentialgleichungen untersucht werden – soferne man solche Abschätzungen überhaupt in der mathematischen Literatur findet. In der Praxis verzichtet man jedoch auf solche Untersuchungen, man geht davon aus, dass die Einflüsse von elastischen Verformungen auf die Bahn der Kapsel so gering sind, dass sie ohne jede praktische Relevanz sind.

1.2 Datenfehler

Daten ... Größen, die schon vor der Rechnung bekannt und verfügbar sind.

z.B. Masse der Raumkapsel, Gravitationsfeld der Erde (Gravitationskraftvektor als Funktion des Ortes), Dichte der Atmosphäre in Abhängigkeit von der Höhe zur Zeit der Landung, ...

Daten können sein: *Zahlen* (z.B. die Masse der Raumkapsel), *Vektoren* (z.B. Anfangslage (Ortsvektor) und Anfangsgeschwindigkeit der Kapsel am Beginn des Landemanövers), *Matrizen*, *Tensoren* oder auch *Funktionen* (z.B. Dichte der Atmosphäre als Funktion der Höhe = $\rho(h)$ ist eine Abbildung vom

Typ $\rho : \mathbb{R}^1 \to \mathbb{R}^1$, oder der Gravitationsvektor als Funktion des Ortes ist eine Vektorfunktion vom Typ $\mathbb{R}^3 \to \mathbb{R}^3$ bzw. wenn man den Gravitationsvektor als zeitlich veränderlich ansieht, wegen der zeitlich veränderlichen Massenverteilungen, bedingt durch Ebbe und Flut, dann ist das Gravitationsfeld vom Typ $\mathbb{R}^4 \to \mathbb{R}^3$, usw.) Also es kommen beliebige mathematische Objekte als Daten in Betracht.

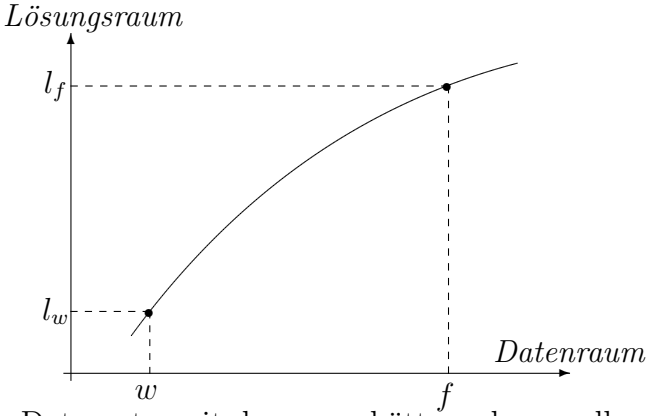
Häufige Ursachen für Datenfehler:

Messfehler, z.B. bei der Bestimmung der Masse der Raumkapsel – falls diese direkt durch Wägung ermittelt wurde oder falls sie berechnet wurde aufgrund ihrer Materialzusammensetzung, aber mit messfehlerbehafteten spezifischen Gewichten der Materialien

Modellfehler, die man als Datenfehler interpretiert

1.2.1 Kondition

Wie die Lösung eines mathematischen Problems auf Datenänderungen reagiert, wird durch die **Kondition** beschrieben, siehe Abb. 1.2.



w ... wahrer Datensatz, mit dem man hätte rechnen sollen
f ... durch Datenfehler verfälschter Datensatz
l_w ... wahre Lösung, die sich mit dem wahren Datensatz ergeben hätte
l_f ... verfälschte Lösung, die man aufgrund des verfügbaren verfälschten Datensatzes erhält

Abbildung 1.2: Datenraum → Lösungsraum, einem konkreten Datensatz (z.B. w oder f) aus dem Datenraum entspricht eine konkrete Lösung (l_w oder l_f) aus dem Lösungsraum

Gute Kondition: (siehe Abb. 1.3) falls sich die Lösung nur wenig ändert, wenn die Daten des Problems verändert werden, z.B. die Lösung ändert sich etwa so stark wie die Daten selbst

Schlechte Kondition: (siehe Abb. 1.3) falls sich die Lösung bei geringer Änderung der Daten extrem verändert, dh. *Datenfehler haben katastrophale Auswirkungen*

Man sollte sich stets vorab Informationen über die Kondition des vorliegenden Problems verschaffen, um die Auswirkungen der stets unvermeidlichen Datenfehler auf die Genauigkeit der Lösung einschätzen zu können. Auch jene Modellfehler, die sich als Datenfehler interpretieren lassen, können dann mit Konditionsabschätzungen für das gegebene Problem abgedeckt werden.

Beispiel 1.2.1. Lineares Gleichungssystem mit zwei Gleichungen und zwei Unbekannten

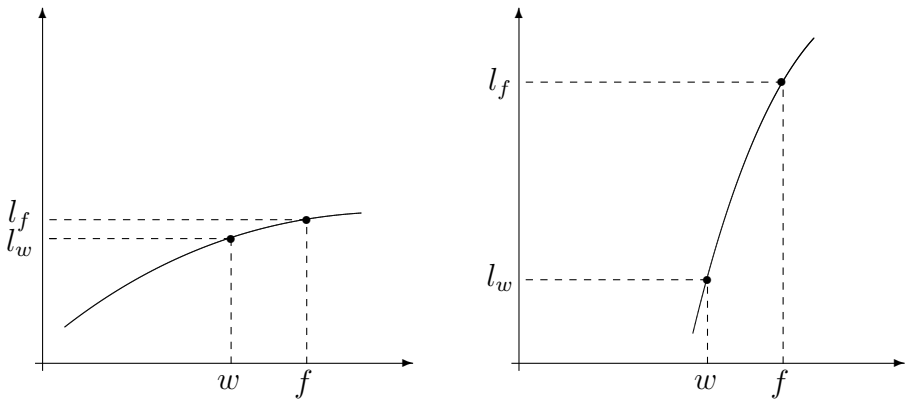


Abbildung 1.3: Gute Kondition Schlechte Kondition

a)

$$\begin{aligned} 1.253672417x_1 &+ 1.247798111x_2 &= 3.654199872 \\ -2.672344812x_1 &+ 2.695328007x_2 &= 2.479981003 \end{aligned}$$

Lösung:

$$\begin{aligned} x_1 &= 1.006128817\ldots \\ x_2 &= 1.917653108\ldots \end{aligned}$$

verfälschtes System:

$$\begin{aligned} 1.253672000x_1 &+ 1.247798000x_2 &= 3.654199000 \\ -2.672344000x_1 &+ 2.695328000x_2 &= 2.479981000 \end{aligned}$$

verfälschte Lösung:

$$\begin{aligned} x_1 &= \underline{1.006128}871\ldots \\ x_2 &= \underline{1.917652}862\ldots \end{aligned}$$

Verfälschung der Lösung in derselben Größenordnung wie die Verfälschung der Daten $\Longrightarrow$ gut konditioniertes Problem!

b)

$$\begin{aligned} 1.743681226x_1 &- 0.5287326143x_2 &= 2.666771987 \\ 4.359203065x_1 &- 1.321302803x_2 &= 6.667195145 \end{aligned}$$

Lösung:

$$\begin{aligned} x_1 &= 1.682330907\ldots \\ x_2 &= 0.5043710646\ldots \end{aligned}$$

verfälschtes System:

$$\begin{aligned} 1.743681000x_1 &- 0.5287326000x_2 &= 2.666771000 \\ 4.359203000x_1 &- 1.321302000x_2 &= 6.667195000 \end{aligned}$$

verfälschte Lösung:

$$\begin{aligned} x_1 &= \underline{1.682}09869\ldots \\ x_2 &= \underline{0.503}6052756\ldots \end{aligned}$$

Verfälschung der Lösung um Faktor 10000 größer wie die Verfälschung der Daten. $\Longrightarrow$ sehr schlecht konditioniertes Problem!

Zwei verschiedene Probleme aus derselben Problemklasse (lineares Gleichungssystem mit zwei Gleichungen und zwei Unbakannten) können je nach Datensatz sehr verschieden konditioniert sein.

In der mathematischen Literatur finden sich zahlreiche Konditionsabschätzungen für verschiedene Problemklassen.

Beispiel 1.2.2. Anfangswertprobleme gewöhnlicher Differentialgleichungen:

Ungestörtes Problem:

$$\begin{aligned} y'(t) &= f(t, y(t)) \qquad f : [0,T] \times \mathbb{R} \to \mathbb{R} \\ y(0) &= y_0 \qquad y : [0,T] \to \mathbb{R},\ y_0 \in \mathbb{R} \end{aligned}$$

Problem mit verfälschten Daten:

$$\begin{aligned} z'(t) &= f(t, z(t)) + \Delta(t, z(t)) \\ z(0) &= y_0 + \Delta z_0 \end{aligned}$$

Es gelte:

1. $f\ldots$ Lipschitzstetig mit Lipschitzkonstante $L \in \mathbb{R}^+$, $\|f(t,y_1) - f(t,y_2)\| \leq L\|y_1 - y_2\|$
2. und für die Störungen: $\|\Delta(t, z(t))\| \leq \Delta$ und $\|\Delta z_0\| \leq \Delta_0$

Dann gilt die Konditionsabschätzung (ohne Beweis):

$$\|y(t) - z(t)\| \leq e^{Lt}\Delta_0 + \frac{e^{Lt} - 1}{L}\Delta$$

Kritischer Fall $L \gg 0$, falls die Konditionsabschätzung realistisch ist, liegt dann schlechte Kondition des Anfangswertproblems vor:

$$\text{z.B.} \quad L = 100 \quad t = 1 \quad e^{Lt} \approx 2.7 * 10^{43}!$$

Mit solchen Konditionsabschätzungen aus der mathematischen Literatur lässt sich die Kondition eines gegebenen Problems aus der entsprechenden Klasse abschätzen. Wenn man z.B. für ein gegebenes Anfangswertproblem Schranken Δ_0 und Δ für die Datenstörungen hat und L kennt, kommt man mit obiger Abschätzung zu einer zahlenmässigen Abschätzung für $\|y(t) - z(t)\|$. Etwa bei dem Landemanöver der Raumkapsel benötigt man Schranken Δ_0 für die Messgenauigkeiten bezüglich des Anfangszustandes (Anfangslage und Anfangsgeschwindigkeit), weiters wird eine Schranke Δ für die Ungenauigkeiten der rechten Seite $f(t, y)$ der Differentialgleichung benötigt, also in diesem Fall für die Modellfehler bei der Modellierung des Gravitationsfeldes und der Erdatmosphäre (Reibungskräfte).

Konditionsabschätzungen für konkrete Problemklassen werden wir später kennenlernen.

1.3 Verfahrensfehler

Die meisten mathematischen Probleme sind nicht exakt lösbar!

Beispiel 1.3.1. Integration einer Funktion $f(t)$

Hauptsatz der Differential und Integralrechnung:
$f : [a, b] \to \mathbb{R}$ *stetig,* F *Stammfunktion von* f

$$\forall a, b \in \mathbb{R} \qquad \int_a^b f(t)dt \;=\; F(b) - F(a) \tag{1.1}$$

Damit die Aufgabenstellung $\int_a^b f(t)dt$ überhaupt einen Sinn hat, muss $f(t)$ etwa im Sinn von Riemann integrierbar sein. Damit das Integral gemäß (1.1) berechnet werden kann, muss f eine Stammfunktion F besitzen. Die meisten in den Anwendungen auftretenden Funktionen sind integrierbar und besitzen eine Stammfunktion. Z.B. sind alle auf $[a, b]$ stetigen Funktionen integrierbar und besitzen eine Stammfunktion. Dennoch kann man in der Praxis sehr oft Integrale nicht gemäß (1.1) berechnen. Der entscheidende Punkt ist, dass sich sehr viele Funktionen *nicht geschlossen darstellen* lassen, d.h. man muss unterscheiden zwischen

1. Existenz von Funktionen im mathematischen Sinn und
2. der Tatsache, dass sich gewisse Funktionen als Formelausdruck darstellen lassen, d.h. auf endliche Weise aus den 4 Grundrechenarten $+ \; - \; * \; \div$, aus $\sqrt[n]{}$ und aus den elementaren Funktionen sin, cos, tan, exp, arcsin, arccos, arctan, ln, ... aufgebaut sind.

Z.B. ist die Menge aller stetigen Funktionen auf dem Intervall $[0, 1]$ viel umfassender als die Menge der auf $[0, 1]$ stetigen, geschlossen darstellbaren Funktionen. Sehr oft liegt die Situation vor, dass der Integrand $f(t)$ geschlossen darstellbar ist, dass die Stammfunktion zwar existiert, aber nicht geschlossen darstellbar ist, z.B.

$$\frac{\exp(t)}{t}, \quad \frac{\sin(t)}{t}, \quad \frac{1}{\ln(t)}$$

sind solche Beispiele. Man kann beweisen, dass die Stammfunktionen zu diesen Integralen nicht geschlossen darstellbar sind. Für solche Integrale scheidet der Weg über (1.1) zur Berechnung aus und man ist auf *numerische Näherungsverfahren* angewiesen. In anderen Fällen ist es einfach bequemer, das Integral über ein Näherungsverfahren zur berechnen, und zwar dann, wenn die Stammfunktion geschlossen dargestellt werden kann, die Berechnung der Stammfunktion aber sehr mühsam ist.

Analoge Situation bezüglich Differentialgleichungsproblemen (gewöhnliche und partielle Differentialgleichungen, Anfangs-Randwertaufgaben), Integralgleichungen, Integrodifferentialgleichungen, ... fast immer können die Lösungen nicht geschlossen dargestellt werden und müssen mit numerischen Näherungsverfahren berechnet werden

$$\Longrightarrow \quad \textbf{Verfahrensfehler} \;:=\; \text{Numerische Näherung} - \text{exakte Lösung} \tag{1.2}$$

Beispiel 1.3.2. Numerische Differentiation

Wenn f geschlossen darstellbar ist, ist auch f' geschlossen darstellbar und kann mit Hilfe von Ableitungsregeln berechnet werden. So gesehen könnte man auf numerische Differentiation völlig verzichten. Aber numerische Differentiation ist oft bequemer. Z.B. Newtonverfahren zur Lösung von $F(x) = 0$, wobei F eine hochdimensionale Vektorfunktion ist; für dieses Verfahren benötigt man die Jacobimatrix

$$\begin{pmatrix} \frac{\partial F_1}{\partial x_1} & \frac{\partial F_1}{\partial x_2} & \cdots & \frac{\partial F_1}{\partial x_n} \\ \vdots & & & \vdots \\ \frac{\partial F_n}{\partial x_1} & \frac{\partial F_n}{\partial x_2} & \cdots & \frac{\partial F_n}{\partial x_n} \end{pmatrix}$$

deren geschlossene Darstellung sehr mühsam sein kann.

Numerische Näherungsformel:

$$\begin{aligned} \textit{Differenzenquotient}: \qquad & f'(x) \;\approx\; \frac{f(x+h)-f(x)}{h} \\ \textit{Verfahrensfehler}: \qquad & \underbrace{\frac{f(x+h)-f(x)}{h}}_{\text{num. Näherungsausdruck}} - \underbrace{f'(x)}_{\text{exakter Wert}} \end{aligned} \tag{1.3}$$

Taylorreihenentwicklung:

$$\begin{aligned} &\frac{f(x+h)-f(x)}{h} - f'(x) \;= \\ &= \frac{1}{h}\left[\left(f(x) + hf'(x) + \frac{h^2}{2}f''(\theta)\right) - f(x)\right] - f'(x) \;= \\ &= \frac{h}{2}f''(\theta) \qquad \theta \in (x, x+h) \end{aligned} \tag{1.4}$$

Verfahrensfehlerschranke (a-priori Schranke)

$$\left|\frac{f(x+h)-f(x)}{h} - f'(x)\right| \;\leq\; \frac{M_2}{2}h = \frac{M_2}{2}h^1 \tag{1.5}$$

M_2 ... Schranke für f'' in einer geeigneten Umgebung von x

Die a-priori Schranke (1.5) wird nicht zahlenmäßig ausgewertet: Wenn man $f'(x)$ nicht kennt sondern numerisch berechnet, kennt man erst recht nicht f'' in einer Umgebung von x und daher kennt man

auch nicht M_2. $\frac{M_2}{2}h$ kann also nicht zahlenmässig berechnet werden und daher nicht zur konkreten Abschätzung verwendet werden. Die Bedeutung von (1.5) liegt in der mathematischen Information, (1.5) ist ein *Konvergenzresultat:*

Falls f zweimal stetig differenzierbar ist konvergiert für $h \to 0$ der numerische Näherungsausdruck $\frac{f(x+h)-f(x)}{h}$ gegen $f'(x)$.

Besseres Verfahren:

$$\textit{Zentraler Differenzenquotient}: \qquad f'(x) \approx \frac{f(x+h)-f(x-h)}{2h} \tag{1.6}$$

$$\textit{Verfahrensfehler}: \qquad \frac{f(x+h)-f(x-h)}{2h} - f'(x) \tag{1.7}$$

Taylorreihenentwicklung:

$$\begin{aligned}
&\frac{f(x+h)-f(x-h)}{2h} - f'(x) \; = \\
&= \frac{1}{2h}\Big[\Big(f(x) + hf'(x) + \frac{h^2}{2}f''(x) + \frac{h^3}{6}f'''(\theta_1)\Big) - \\
&\quad - \Big(f(x) - hf'(x) + \frac{h^2}{2}f''(x) - \frac{h^3}{6}f'''(\theta_2)\Big)\Big] - f'(x) \\
&= \frac{h^2}{12}f'''(\theta_1) + \frac{h^2}{12}f'''(\theta_2) \qquad \theta_1 \in (x, x+h), \theta_2 \in (x-h, x)
\end{aligned} \tag{1.8}$$

$\Longrightarrow$ Verfahrensfehlerschranke (a-priori)

$$\left|\frac{f(x+h)-f(x-h)}{2h} - f'(x)\right| \; \leq \; \left(\frac{M_3}{12} + \frac{M_3}{12}\right)h^2 = \frac{M_3}{6}h^2 \tag{1.9}$$

(1.9) ist wieder ein Konvergenzuresultat, für $h \to 0$ geht der Verfahrensfehler wie h^2 gegen Null, d.h. bei Halbierung der Schrittweite geht die Fehlerschranke auf $\frac{1}{4}$ zurück. Das zeigt die Überlegenheit des zentralen Differenzenquotienten für kleine h-Werte, natürlich nur wenn f dreimal stetig differenzierbar ist.

Man bezeichnet die Hochzahl von h in den Darstellungen (1.5) und (1.9) als die *Ordnung des Verfahrens.*

Längere Taylorreihenentwicklung in (1.8) – wieder für hinreichend oft differenzierbares f:

$$\begin{aligned}
&\frac{f(x+h)-f(x-h)}{2h} - f'(x) \; = \\
&= \frac{1}{2h}\Big[\Big(f(x) + hf'(x) + \frac{h^2}{2}f''(x) + \frac{h^3}{6}f'''(x) + \frac{h^4}{24}f^{IV}(x) + \frac{h^5}{120}f^{V}(\theta_1)\Big) - \\
&\quad - \Big(f(x) - hf'(x) + \frac{h^2}{2}f''(x) - \frac{h^3}{6}f'''(x) + \frac{h^4}{24}f^{IV}(x) - \frac{h^5}{120}f^{V}(\theta_2)\Big)\Big] - f'(x) = \\
&= \frac{h^2}{6}f'''(x) + \frac{h^4}{240}f^{V}(\theta_1) + \frac{h^4}{240}f^{V}(\theta_2).
\end{aligned} \tag{1.10}$$

$\Rightarrow$ **Verfahrensfehlerdarstellung** (**asymptotische Entwicklung**)

$$\begin{aligned} \frac{f(x+h)-f(x-h)}{2h} - f'(x) &= \frac{h^2}{6} f'''(x) + R \\ \text{mit} \quad |R| &\leq h^4 \frac{M_5}{120} \end{aligned} \tag{1.11}$$

Zwei Anwendungen asymptotischer Entwicklungen:

(i) Zahlenmäßige Fehlerschätzungen (**a-posteriori Schätzung**) für den Verfahrensfehler

(ii) Konstruktion besserer Verfahren

Bei der numerischen Differentiation ist alles sehr einfach und transparent, aber durchaus typisch. Auch bei viel komplizierteren numerischen Algorithmen treten ähnliche Erscheinungen bezüglich des Verfahrensfehlers auf wie *Konvergenz*, *Konvergenz verschiedener Ordnung*, *a-priori Schranken*, *a-posteriori Schätzungen*, *asymptotische Entwicklungen*, *Konstruktion genauerer Verfahren aufgrund von asymptotischen Entwicklungen des Verfahrensfehlers*, ... nur ist alles viel schwerer zu beweisen. Der einzig untypische Punkt bei der numerischen Differentiation ist, dass beim Grenzübergang $h \to 0$ (Fehler $\to 0$) keine Steigerung des Rechenaufwandes erfolgt. Um einen Differenzenquotienten zu berechnen muss man für jeden Wert von h den selben Rechenaufwand leisten. Das ist bei anderen Situationen (numerische Integration, Differentialgleichungsalgorithmen, ...) anders, wo man den kleiner und kleiner werdenden Verfahrensfehler i.a. mit einer entsprechenden Rechenaufwandssteigerung erkaufen muss.

Beispiel 1.3.3. Numerische Integration mit der Trapezregel (siehe Abb. 1.4)

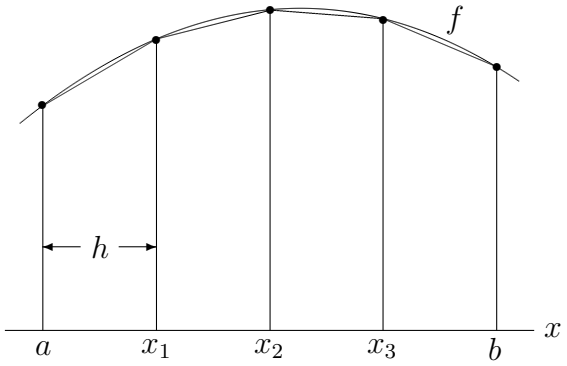


Abbildung 1.4: Trapezregel

Als Näherung für $\int_a^b f(x)dx$ wird die Summe der Flächeninhalte der eingeschriebenen Trapeze genommen. Die Fläche eines Trapezes ist $\frac{h}{2}[f(x_i) + f(x_{i+1})]$.

Näherungswert:

$$T_h(f) = h\left[\frac{1}{2}f(a) + f(x_1) + \ldots + f(x_{n-1}) + \frac{1}{2}f(b)\right] \tag{1.12}$$

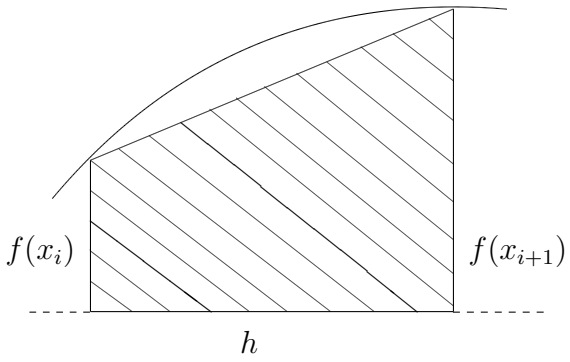


Abbildung 1.5: Fläche eines Trapezes

$$x_i = a + ih,\ i = 0, 1, \ldots, n; \quad \frac{b-a}{n} = h,\ (x_0 = a,\ x_n = b)$$

Verfahrensfehler der Trapezregel:

$$T_h(f) - \int_a^b f(x)dx \tag{1.13}$$

Zunächst: Fehler bei einem Trapez, siehe Abb. 1.6

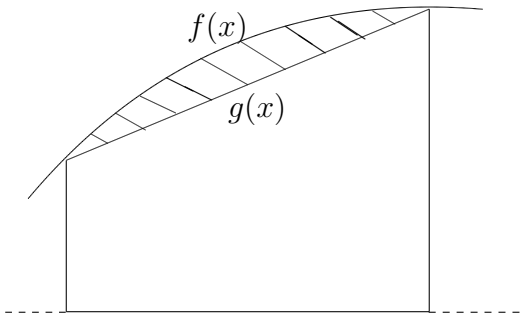


Abbildung 1.6: Fehler bei einem Trapez

Es gilt (vgl. Kapitel über den Interpolationsfehler, siehe (4.23)):

$$f(x) - g(x) \ = \ \frac{f''(\theta(x))}{2}(x - x_i)(x - x_{i+1})$$

für die von x abhängige Größe $\theta(x)$ gilt $x_i \leq \theta(x) \leq x_{i+1}$ und $g(x)$ die Gerade durch die Punkte $(x_i, f(x_i))$ und $(x_{i+1}, f(x_{i+1}))$. Also Fehler bei einem Trapez:

$$\begin{aligned} &|\int_{x_i}^{x_{i+1}} \frac{f''(\theta(x))}{2}(x - x_i)(x - x_{x+1})dx| \ \leq \\ &\quad \leq \ \frac{M_2}{2}\,|\int_{x_i}^{x_{i+1}} (x - x_i)(x - x_{i+1})dx| \end{aligned} \tag{1.14}$$

M_2 ... Schranke für $f''(x)$, $x \in [x_i, x_{i+1}]$, $x_{i+1} - x_i = h$ und $\zeta = x - x_i - \frac{h}{2}$

$$\begin{aligned}\int_{x_i}^{x_{i+1}} (x-x_i)(x-x_{i+1})dx &= \int_{-\frac{h}{2}}^{\frac{h}{2}} \Big(\xi + \frac{h}{2})(\xi - \frac{h}{2}\Big)d\xi = \\ &= \int_{-\frac{h}{2}}^{\frac{h}{2}} \Big(\xi^2 - \frac{h^2}{4}\Big)d\xi = \Big(\frac{\xi^3}{3} - \xi\frac{h^2}{4}\Big)\Big|_{\frac{-h}{2}}^{\frac{h}{2}} = \\ &= \Big(\frac{\frac{h^3}{8}}{3} - \frac{h^3}{8}\Big) - \Big(\frac{-\frac{h^3}{8}}{3} + \frac{h^3}{8}\Big) = -\frac{h^3}{6}\end{aligned}$$

daraus folgt

$$\frac{M_2}{2} \,|\int_{x_i}^{x_{i+1}} (x-x_i)(x-x_{i+1})| \;\leq\; \frac{M_2}{12}h^3$$

Aufsummation über alle $n = \frac{b-a}{h}$ Intervalle:

$$|T_h(f) - \int_a^b f(x)dx| \;\leq\; \frac{M_2}{12}\frac{b-a}{h}h^3 = \frac{M_2(b-a)}{12}h^2 \tag{1.15}$$

Bemerkungen:

1. Der Verfahrensfehler der Trapezregel konvergiert für $h \to 0$ wie h^2 gegen Null. Für kleiner werdendes h hat man jedoch im Intervall mehr Gitterpunkte ($n = \frac{b-a}{h}$), man muss mehr Funktionswerte $f(x_i)$ berechnen. Die wachsende Genauigkeit für $h \to 0$ wird also durch höheren Rechenaufwand erkauft.
2. Der Verfahrensfehler der Trapezregel besitzt auch eine asymptotische Entwicklung (wie der Verfahrensfehler der numerischen Differentiation) und zwar eine *Entwicklung nach geraden h-Potenzen* (wie der zentrale Differenzenquotient $\frac{f(x+h)-f(x-h)}{2h}$), siehe Kapitel über Numerische Quadratur (Euler-Maclaurinsche Summenformel, Romberg-Integration).

Konkretes **Zahlenbeispiel** zur Illustration:

$$\int_0^1 e^x dx = e - 1 = 1.718281828\ldots$$

$h = \frac{1}{4}$ liefert die Näherung

$$\begin{aligned}T_h(f) &= h\Big[\frac{1}{2}f(0) + f(x_1) + f(x_2) + f(x_3) + \frac{1}{2}f(1)\Big] = \\ &= \frac{1}{4}\Big[\frac{1}{2}e^0 + e^{\frac{1}{4}} + e^{\frac{1}{2}} + e^{\frac{3}{4}} + \frac{1}{2}e^1\Big] = \\ &= 1.727221905\ldots\end{aligned}$$

Fehler: $T_h(f) - \int_0^1 e^x dx = 8.94\ldots 10^{-3}$

$h = \frac{1}{8}$ ergibt

$$T_h(f) = 1.720518592\ldots$$

Fehler: $T_h(f) - \int_0^1 e^x dx = 2.23\ldots 10^{-3}$ also Rückgang des Fehlers auf $\frac{1}{4}$ des Fehlers für $h = \frac{1}{4}$

Zusammenfassende Diskussion:

A-priori Schranken für den Verfahrensfehler:

a) Dfferenzenquotient

$$\left| \frac{f(x+h) - f(x)}{h} - f'(x) \right| \leq \frac{M_2}{2} h$$

(siehe (1.5))

b) Zentraler Differenzenquotient

$$|\frac{f(x+h) - f(x-h)}{2h} - f'(x)| \leq \frac{M_3}{6} h^2$$

(siehe (1.9))

c) Trapezregel

$$|T_h(f) - \int_a^b f(x)dx| \leq \frac{M_2(b-a)}{12} h^2$$

(siehe (1.15))

d)

$$\|\eta_\nu - y(t_\nu)\| \leq \frac{e^{Lt_\nu} - 1}{L} \frac{M_2}{2} h$$

ist eine Verfahrensfehlerschranke für das explizite **Eulerverfahren** (*Polygonzugmethode*) zur Lösung von Anfangswertproblemen

$$\begin{aligned} y'(t) &= f(t, y(t)) \\ y(0) &= y_0 \end{aligned}$$

wobei

$t \in [0, T]$		... Integrationsintervall
$y(t)$		... gesuchte (exakte) Lösung
$t_\nu = \nu h$	$\nu = 0, 1, \ldots, n$	... Gitterpunkte
$h = \frac{T}{n}$		... Schrittweite
η_ν		... numerische Näherung für $y(t_\nu)$
M_2		... Schranke für $y''(t)$ für $t \in [0, T]$
L		... Lipschitzkonstante von f bezüglich Argument y

Bemerkungen:

1) a)-d) sind theoretische, mathematische Aussagen, die für konkrete mathematische Probleme **nicht** zahlenmässig ausgewertet werden. Bei a) und b) wäre so eine explizite Auswertung der Fehlerschranken in der Praxis gar nicht möglich: Um M_2 oder M_3 zu ermitteln, müsste man f'' oder f''' in einer geeigneten Umgebung von x kennen. Wenn man aber f formelmäßig zweimal oder dreimal differenziert, kennt man auch f' als Formelausdruck und eine numerische näherungsweise Berechnung von $f'(x)$ wäre nicht notwendig, siehe S. 12 nach (1.5). Ebenso lässt sich d) in konkreten Fällen nicht zahlenmäßig auswerten, denn man wird kaum a-priori eine Schranke M_2 für y'' zur Verfügung haben, ohne die Lösung $y(t)$ selbst zu kennen. Bezüglich c) liegt eine etwas andere Situation vor: Die Differentiation ist i.a. viel einfacher wie die Bestimmung von Stammfunktionen und auch in Fällen formelmanipulativ möglich, wo sich die Stammfunktion

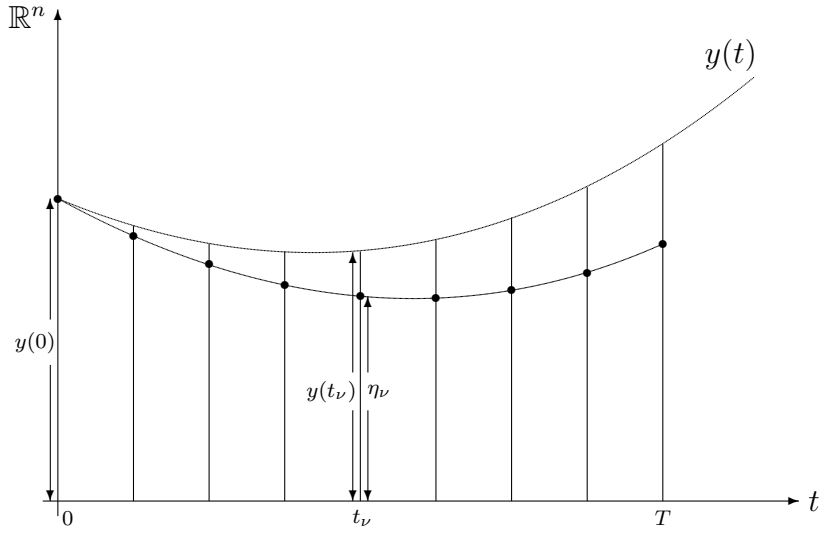


Abbildung 1.7: Polygonzugmethode

nicht geschlossen darstellen lässt und man zu numerischer Integration gezwungen ist. Es erscheint daher vertretbar zuerst den Integranden f zweimal zu differenzieren und M_2 und damit die a-priori Schranke als Ausdruck in h zu ermitteln, das Integral selbst aber durch numerische Integration mit der Trapezregel näherungsweise zu berechnen. Dennoch geschieht dies in der Praxis eher selten.

2) Bedeutung der a-priori Schranken: Mathematische Rechtfertigung für die betrachteten Näherungsverfahren; die Konstruktion von numerischen Methoden beruht zunächst nur auf heuristischen Überlegungen, erst a-priori Schranken weisen die entsprechenden Verfahren als vernünftig aus. Die stets auftretenden Faktoren h oder h^2 oder allgemein h^p stellen die *Konvergenz* der Verfahren sicher (siehe *Konvergenzresultat* auf S. 13), d.h. für $h \to 0$ geht die Fehlerschranke (und damit auch der tatsächliche Verfahrensfehler) wie h (bzw. wie h^p) gegen Null. p bezeichnet man als die *Ordnung* des Verfahrens. Bei unseren Beispielen sind a) und d) von der Ordnung 1 und b) und c) von der Ordnung 2. Die a-priori Schranke stellt sicher, dass man durch entsprechende Wahl der Schrittweite h den Fehler beliebig klein machen kann und dass mit $h \to 0$ der Fehler umso rascher gegen Null geht, je höher die Ordnung des Verfahrens ist.

A-priori Schranken ermöglichen auch Effizienzbetrachtungen für Software-Entwicklungen:

Beispiel 1.3.4. Wir vergleichen zwei Verfahren, eines der Ordnung 1 und eines der Ordnung 2 und nehmen an, dass bei dem Verfahren der Ordnung 2 pro Schritt ein doppelt so hoher Aufwand nötig ist wie bei dem Verfahren der Ordnung 1.

So eine Situation wäre z.B. gegeben, wenn man Anfangswertprobleme gewöhnlicher Differentialgleichungen mit dem Eulerverfahren ($p = 1$) oder mit der sogenannten *Methode von Heun* ($p = 2$) (ein *Runge-Kutta-Verfahren* mit 2 Funktionsauswertungen pro Schritt) lösen wollte.

Die Bilder Abb. 1.8 und Abb. 1.9 bestätigen die bekannte Faustformel: **Verfahren hoher Ordnung lohnen sich nur bei strengen Genauigkeitsniveau**.

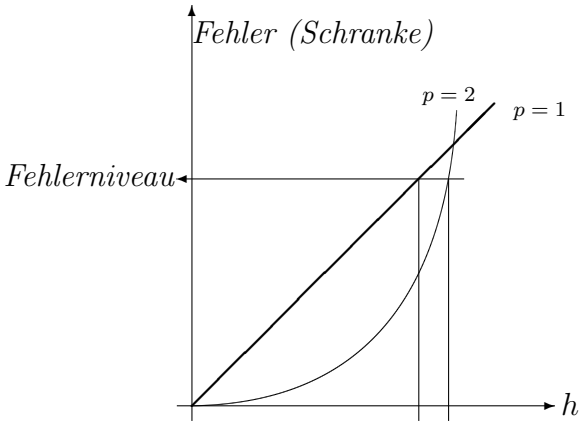


Abbildung 1.8: Bei ungenauem Fehlerniveau ist die Ersparnis bei der Anzahl der Schritte so gering, dass sich das Verfahren mit $p = 1$ sogar als effizienter erweist. Bei dem Verfahren mit $p = 2$ hat man etwa 80% der Schritte im Vergleich mit $p = 1$, aber pro Schritt einen doppelt so hohen Aufwand.

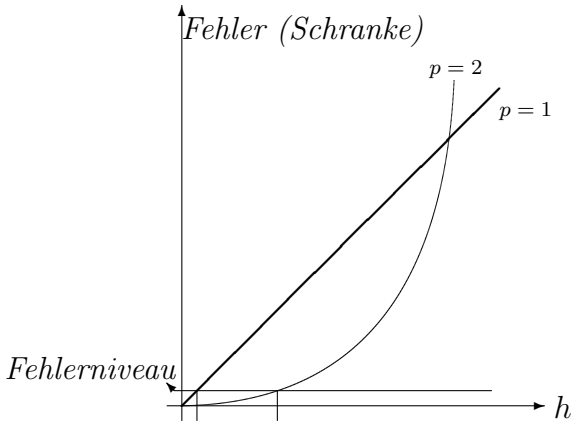


Abbildung 1.9: Bei hoher Genauigkeit ist das Verfahren mit $p = 2$ viel effizienter, etwa nur 10% der Schritte verglichen mit $p = 1$ aber pro Schritt nur der doppelte Aufwand.

3) A-priori Schranken vom Typ a)-d) sind in der numerischen Mathematik für sehr viele Näherungsverfahren bekannt. Trotzdem gibt es noch zahlreiche mathematische Problemklassen und zugehörige Näherungsverfahren, für die es bisher noch nicht gelungen ist, solche Schranken zu beweisen. (Beispiel aus dem Bereich der Anfangswertprobleme gewöhnlicher Differentialgleichungen: für große Klassen nichtlinearer steifer Differentialgleichungen und numerischen Näherungsverfahren, von denen man aus der Praxis weiß, dass sie effizient funktionieren, fehlen entsprechende Konvergenzresultate.)

   Es gibt in der Numerik sogar noch kritischere Situationen: Es gibt Klassen mathematischer Probleme, für die noch keine effizienten numerischen Näherungsverfahren bekannt sind (Beispiel aus dem Bereich der partiellen Differentialgleichungen: Hier gibt es Rand-Anfangswertprobleme mit sogenannten schockwellenbehafteten Lösungen, bei denen die naheliegenden numerischen Näherungsverfahren zu falschen Schockwellengeschwindigkeiten führen. Erst auf so feinen Gittern, die aus Gründen der Rechenzeit nicht vertretbar sind, würde sich die korrekte Schockwellengeschwindigkeit ergeben. Praktische Probleme dieser Art ergeben sich bei der mathematisch-

physikalischen Modellierung von Klopfvorgängen in den Zylindern von Verbrennungsmotoren.)

4) Typischerweise steigt für kleiner werdende Schrittweiten h der Aufwand der numerischen Näherungsverfahren. D.h. es liegt meist eine Situation wie bei c) und d) vor. Die Fälle a) und b) sind extrem untypische Ausnahmen, da bei der Berechnung von Differenzenquotienten die Anzahl der Funktionsauswertungen unabhängig von h ist, siehe Bemerkung auf S. 14. Bei diesen beiden Fällen scheint das Verfahren b) ($p = 2$) gegenüber a) ($p = 1$) keinen Vorteil aufzuweisen, da man mit beiden Verfahren durch entsprechende Wahl von der Schrittweite h ein beliebiges Genauigkeitsniveau des Verfahrensfehlers erreichen kann und darüber hinaus bei beiden Verfahren zwei Funktionsauswertungen benötigt, bei a)$f(x+h), f(x)$ und bei b)$f(x+h), f(x-h)$ benötigt. Wenn man jedoch auch Rundungsfehler einkalkuliert, erweist sich b) als überlegen, man kann beim Verfahren der Ordnung 2 mit größeren Schrittweiten ein sehr gutes Niveau des Verfahrensfehlers erreichen und diese größeren Schrittweiten sind noch nicht so problematisch für das Rundungsfehlerniveau, siehe Kap. 1.4.

5) Bei älteren Aussagen über a-priori Schranken vom Typ

$$\underbrace{\text{Ausdruck}(M_i, L, \ldots)}_{\text{Ausdruck in problemcharakterisierenden Parametern}} h^p$$

stand die Ordnung p im Vordergrund der Betrachtungen und die Tatsache, dass so eine Schranke gleichmäßig für $h \to 0$ gilt, dass also hier Gitter mit beliebig vielen Gitterpunkten abgedeckt sind, war die nichttriviale mathematische Aussage. Bei moderneren Betrachtungen diskutiert man auch eingehender die Struktur des Ausdruckes in den problemcharakterisierenden Parametern, der als Faktor neben h^p in der a-priori Schranke auftritt.

1.4 Rechen- bzw. Rundungsfehler

Oft ergeben sich Probleme bei der numerischen Berechnung für kleine h-Werte. Für $h = 10^{-4}$ ist der zentrale Differenzenquotient der Funktion $f(x) = e^x$

$$\frac{f(x+h) - f(x-h)}{2h} = \frac{e^{1+10^{-4}} - e^{1-10^{-4}}}{2 \cdot 10^{-4}}$$

Der kritische Punkt ist hier die Differenzbildung $e^{1+10^{-4}} - e^{1-10^{-4}}$, z.B. auf einem 10-stelligen Taschenrechner ergibt sich:

$e^{1+10^{-4}}$	=	2.718553670	$\ldots *)$
$e^{1-10^{-4}}$	=	2.718010014	$\ldots *)$
$e^{1+10^{-4}} - e^{1-10^{-4}}$	=	0.000543656	$\ldots *)$

$*)$ in der 10-stelligen Arithmetik verlorene Stellen

Aufgrund der verlorenen Stellen von $e^{1+10^{-4}}$ und $e^{1-10^{-4}}$ ist das Ergebnis der Substraktion nur mehr 6-stellig! Die letzten 4 Stellen, die man für eine 10-stellige Arithmetik benötigen würde, sind verloren!

Diese **Auslöschung** liegt vor, bei der Differenzbildung von annähernd gleich großen verfälschten Zahlen.

$$\begin{array}{rl} 3.67253748913 & (\diamond) \\ -3.67231866741 & \\ \hline 0.00021882172 & \end{array} \qquad \begin{array}{rl} 3.6725 & (\square) \\ -3.6723 & \\ \hline 0.0002 & \leftarrow \text{Ergebnis exakt} \end{array}$$

Vergleich von der exakten Rechnung ($\diamond$) mit ($\square$), wo schon vor der Differenzbildung auf die 5 stellige Arithmetik gerundet wurde: bei ($\square$) hat man nur mehr eine (!) Stelle.

Die Differenzbildung etwa gleicher Zahlen ist harmlos, wenn man exakte Größen die in der gegebenen Arithmetik exakt dargestellt werden können zur Verfügung hat. Hat man aber z.B. durch Rundung verfälschte Zahlen, ergibt sich aufgrund der Auslöschung ein katastrophaler Genauigkeitsverlust. Es wird zwar bei der Differenzbildung kein neuer Fehler generiert, aber die Fehlerfortpflanzung bezüglich der vorhergehenden Rundung der beiden Operanden ist i.a. katastrophal, und zwar wird dadurch nicht das absolute sondern das relative Fehlerniveau betroffen.

Um Rundungsfehler systematisch untersuchen und abschätzen zu können, benötigen wir einige Informationen über *Computerarithmetik*.

1.4.1 Computerarithmetik

Basis der Zahlendarstellung beim menschlichen Rechnen ist 10. Die Stelle, wo eine Ziffer steht, gibt an, mit welcher 10-er Potenz die Ziffer zu multiplizieren ist,[1)]. z.B.

$$365.22 \;=\; 3 \cdot 10^2 + 6 \cdot 10^1 + 5 \cdot 10^0 + 2 \cdot 10^{-1} + 2 \cdot 10^{-2}$$

Im Computer werden i.a. andere Basisgrößen als 10 verwendet, meist 2, 8, 16.

Z.B. *Binärarithmetik (Basis 2):* mögliche Ziffern 0, 1

$$\underset{\substack{\uparrow \\ \text{Binärpunkt}}}{11001.11} \;=\; 1 \cdot 2^4 + 1 \cdot 2^3 + 0 \cdot 2^2 + 0 \cdot 2^1 + 1 \cdot 2^0 + 1 \cdot 2^{-1} + 1 \cdot 2^{-2} \;=\; \underset{\substack{\uparrow \\ \text{Dezimalpunkt}}}{25.75}$$

Oktalarithmetik (Basis 8): mögliche Ziffern 0, 1, 2, 3, 4, 5, 6, 7

Hexadezimalarithmetik (Basis 16): mögliche Ziffern 0, 1, 2, 3, 4, 5, 6, 7, 8, 9, a, b, c, d, e, f (die Buchstaben a - f stehen für die Zahlen 10 - 15)

Gleitkommadarstellung: Schreibweise mit Exponenten, z.B.

$$365.22 \;=\; 3.6522 \cdot 10^2 \;=\; \underset{\substack{\uparrow \\ \textit{normalisierte}\ \text{Gleitkommadarstellung}}}{0.36522 \cdot 10^3}$$

[1)] Diese sehr vorteilhafte und intelligente Art, Zahlen zu codieren erscheint uns heute ganz selbstverständlich; trotzdem waren früher ungeeignetere Systeme zur Codierung von Zahlen in Verwendung, z.B. schrieben die Römer 365 in der Form CCCLXV. Das System der römischen Zahlen unterscheidet sich von unserem System nicht nur durch die Ziffernsymbolik sondern ganz grundsätzlich. Um den Vorteil unseres Systems zu erkennen, überlege man etwa, wie kompliziert bei den Römern Multiplikationsalgorithmen zu formulieren wären. Außerdem erscheint die Basis 10 für das menschliche Rechnen ein idealer Kompromiss. Binärarithmetik wäre für den Menschen ungeeignet, da die Zahlen lang und unübersichtlich wären, wenn auch das kleine Einmaleins bei der Binärarithmetik sehr einfach zu lernen wäre: $0 \times 0 = 0$, $0 \times 1 = 0$, $1 \times 0 = 0$, $1 \times 1 = 1$

Normalisierte Gleitkommadarstellung: Konvention, die führende nicht verschwindende Stelle unmittelbar hinter dem Dezimalpunkt oder hinter dem Binärpunkt, ... zu schreiben.[2)] (Die Ziffer 0, wo es keine nichtverschwindenden Stellen gibt, hat eine Sonderdarstellung.)

$$-0.0027653 \;=\; -0.27653 \cdot 10^{-2}$$

wird codiert als

Vorzeichen der Mantisse	Mantisse	Vorzeichen des Exponenten	Exponent
$-$	27653	$-$	2

Als *Mantisse* bezeichnet man die Ziffernstellen einer Gleitkommazahl.

Die tatsächliche interne Zahlendarstellung ist bei den verschiedenen Rechnern noch etwas anders realisiert. Häufig gelingt es, das Vorzeichen des Exponenten zu ersparen. Bei einer Binärarithmetik wäre es redundant, die führende Stelle, die 1 sein muss, zu codieren, was daher meist unterbleibt (*verstecktes Bit*).

Eine **Computerarithmetik** ist also durch folgende Parameter festgelegt:

- b ... Basis
- l ... Anzahl der Mantissenstellen, also Anzahl der Stellen nach dem Komma
- e_1 ... kleinster Exponent
- e_2 ... größter Exponent

Schreibweise für die Menge der **Maschinenzahlen**:

$\mathbb{M}(b, l, e_1, e_2)$ ... Symbol für die Menge aller *normalisierten Gleitkommazahlen* zur Basis b mit l Mantissenstellen und einem Exponenten E mit $e_1 \leq E \leq e_2$. Auf jedem Computer stehen *nur endlich viele Maschinenzahlen* zur Verfügung.

Beispiel 1.4.1. Artifizielles Beispiel einer Maschinenarithmetik $\mathbb{M}(2,\, 3,\, -3,\, +3)$

verfügbare Mantissen			verfügbare Exponenten
100	=	$1 \cdot \frac{1}{2}$	$-3, -2, -1, 0, 1, 2, 3$
101	=	$1 \cdot \frac{1}{2} + 1 \cdot \frac{1}{8}$	
110	=	$1 \cdot \frac{1}{2} + 1 \cdot \frac{1}{4}$	
111	=	$1 \cdot \frac{1}{2} + 1 \cdot \frac{1}{4} + 1 \cdot \frac{1}{8}$	

Z.B. $-101\ -\ 2$ ist die Darstellung von

$$\underbrace{-0.625}_{-\frac{5}{8}} \cdot \underbrace{2^{-2}}_{\frac{1}{4}} \;=\; -\frac{5}{32} \;=\; -0.15625$$

Darstellung aller positven Maschinezahlen dieses artifiziellen Beispiels, siehe Abb. 1.10.

Größte Maschinenzahl:

$$+111\ +\ 3 \quad \text{entspricht} \quad (\frac{1}{2} + \frac{1}{4} + \frac{1}{8}) \cdot 2^3 \;=\; 7$$

[2)] Oft versteht man unter Normalisierung auch die Konvention, den Dezimalpunkt unmittelbar hinter der führenden Stelle zu schreiben, also z.B. $-2.7653 \cdot 10^{-3}$ statt $-0.27653 \cdot 10^{-2}$, etwa im Taschenrechner.

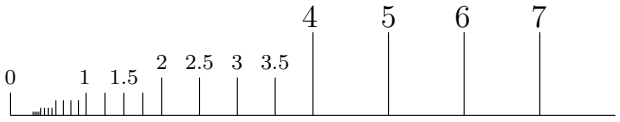


Abbildung 1.10: Positive Maschinenzahlen von $\mathbb{M}(2,\ 3,\ -3,\ +3)$

Kleinste positive Maschinenzahl:

$$+100\ -\ 3 \quad \text{entspricht} \quad \frac{1}{2}\cdot 2^{-3}\ =\ 2^{-4}$$

Vergrößerung der Umgebung der Null, siehe Abb. 1.11.

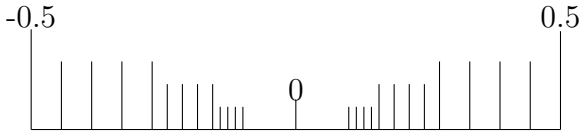


Abbildung 1.11: Umgebung der Null in $\mathbb{M}(2,\ 3,\ -3,\ +3)$

Unmittelbare Umgebung der Null: Es ergibt sich ein Loch weil nur normalisierte Zahlen zugelassen sind. Ausfüllen dieses Lochs wäre möglich, wenn sogenannte *subnormale* Zahlen zugelassen würden, d.h. Mantissen, wo auch die führende Stelle 0 sein darf, z.B. $+\,001\ -\ 3$ entspricht $2\cdot 2^{-3}\cdot 2^{-3} = 2^{-6}$. Heutzutage gibt es in vielen Standardarithmetiken (z.B. im IEEE-Standard) subnormale Zahlen.

Typische Taschenrechnerarithmetik: $\mathbb{M}(10,\ 10,\ -98,\ +100)$[3)] also Basis 10, 10 Mantissenstellen, Exponentialbereich von -98 bis $+100$.

IEEE Standard (*single* Format): $\mathbb{M}(2,\ 24,\ -125,\ 128)$

IEEE Standard (*double* Format): $\mathbb{M}(2,\ 53,\ -1021,\ 1024)$

Bemerkungen.

Eine Variable vom Typ *single* wird nach IEEE 754-Standard in 4 Bytes = 32 Bits wie folgt gespeichert:

Ein Bit speichert das Vorzeichen.

Die Mantisse für eine normalisierte Gleitpunktzahl benötigt 23 Bit bei 24 Stellen, da die erste Ziffer 1 ist und nicht gespeichert werden muss.

Der Exponent wird in den restlichen 8 Bits gepeichert.

Um Rundungsfehleranalysen durchzuführen, muss noch klar sein, ob vom Rechner *gerundet* oder *abgeschnitten* wird.

[3)] Tatsächlich werden am Taschenrechner typischerweise 10-stellige Dezimalzahlen so dargestellt, dass im Gegensatz zu normalisierten Zahlen die führende Stelle vor dem Komma steht. Der 2-stellige Exponent läuft von -99 bis $+99$. Da aber $\mathbb{M}$ immer das Symbol für normalisierte Maschinenzahlen ist mit führender Stelle hinter dem Komma muss in der Schreibweise $\mathbb{M}(10, 10, -98, +100)$ zum Ausgleich der Exponent von -98 bis $+100$ laufen.

Runden einer Zahl: Man nimmt die nächstgelegene Maschinenzahl. Liegt eine Zahl genau zwischen zwei Maschinenzahlen, so lautet die übliche Konvention, dass man zur betragsgrößeren Zahl rundet. Oft wird in diesem Fall aber zur nächstgelegenen geraden Maschinenzahl gerundet (*round to even*).

Abschneiden: Man nimmt die erste Maschinenzahl auf die man trifft, wenn man sich in Richtung zur Null bewegt.

Z.B. Runden in $\mathbb{M}(10, 8 \ldots)$: $987.562438 \to 987.56244$
Abschneiden in $\mathbb{M}(10, 8, \ldots)$: $987.562438 \to 987.56243$

Rundungs- oder Abschneidefehler bei einem einzigen Rundungs- oder Abschneidevorgang:

exakte Zahl	gerundete / abgeschnittene Zahl aus $\mathbb{M}$	*absoluter Fehler*	*relativer Fehler*
x	$\tilde{x}$	$\tilde{x} - x$	$\frac{\tilde{x}-x}{x}$

Maximaler absoluter Abschneidefehler ist der Abstand von zwei benachbarten Maschinenzahlen in der Umgebung von x, siehe Abb. 1.12.

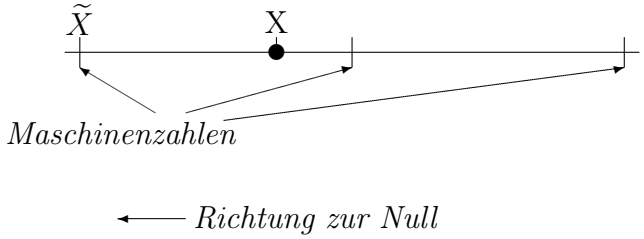


Abbildung 1.12: Abschneidefehler

Maximaler absoluter Rundungsfehler ist der halbe Abstand von zwei benachbarten Maschinenzahlen, siehe Abb. 1.13.

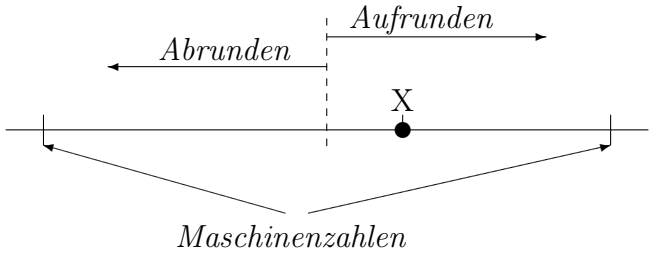


Abbildung 1.13: Rundungsfehler

Abstand von zwei normalisierten benachbarten Maschinenzahlen, mit $e_1 < e < e_2$ und d_i $i = 1, 2, \ldots l$ aus der Menge der in dieser Arithmetik zulässigen Ziffern

$$\begin{array}{rcl} & (d_1 \cdot b^{-1} + \cdots + & d_l \cdot b^{-l}) \cdot b^e \\ - & (d_1 \cdot b^{-1} + \cdots + & (d_l - 1) \cdot b^{-l}) \cdot b^e \quad = \quad b^{-l} \cdot b^e \end{array}$$

$\Rightarrow$ Schranke für absoluten Abschneidefehler von $x = (d_1 \cdot b^{-1} + \ldots) \cdot b^e$ ist $b^{-l} \cdot b^e$

$\Rightarrow$ Schranke für absoluten Rundungsfehler ist $\frac{1}{2}b^{-l}b^e$

Schranke für relativen Fehler:

$$\left|\frac{\tilde{x}-x}{x}\right| \;=\; \frac{|\tilde{x}-x|}{|x|} \;\leq\; \frac{\text{Schranke für absoluten Fehler}}{1\cdot b^{-1}\cdot b^e}$$

Da stets gilt $|x| = |d_1 b^{-1} + d_2 b^{-2} + \ldots| \cdot b^e$, ist tatsächlich $1 \cdot b^{-1} \cdot b^e$ eine untere Schranke für $|x|$

$\Rightarrow$ relativer Abschneidefehler:

$$\left|\frac{\tilde{x}-x}{x}\right| \;\leq\; \frac{b^{-l}\cdot b^e}{b^{-1}\cdot b^e} \;=\; b\cdot b^{-l} \tag{1.16}$$

$\Rightarrow$ ralativer Rundungsfehler:

$$\left|\frac{\tilde{x}-x}{x}\right| \;\leq\; \frac{1}{2}\cdot b\cdot b^{-l} \tag{1.17}$$

Eine triviale aber wichtige Identität für den relativen Fehler $\epsilon := \frac{\tilde{x}-x}{x}$ ist:

$$\tilde{x} \;=\; x(1+\epsilon) \tag{1.18}$$

Zusammengefasst: Wenn x eine exakte Zahl ist und $\tilde{x}$ die durch Abschneiden oder Runden in die Arithmetik $\mathbb{M}(b, l, \ldots)$ entstehende Größe ist, so gilt

$$\boxed{\begin{array}{rrll} & \tilde{x} & = & x(1+\epsilon) \\ \text{mit} & |\epsilon| & \leq & b\cdot b^{-l} \qquad \text{(Abschneiden)} \\ \text{bzw.} & |\epsilon| & \leq & \frac{1}{2}\cdot b\cdot b^{-l} \qquad \text{(Runden)} \end{array}} \tag{1.19}$$

(1.19) ermöglicht nun in bequemer Weise Rundungsfehlerabschätzungen. Der entscheidende Gedanke ist, dass man sich bei einem Algorithmus klar macht, was nach und nach in der Maschine wirklich aufgrund von Rundungsprozessen passiert, und dabei bei jedem Rundungsvorgang die Beziehung (1.19) benützt.

Beispiel 1.4.2. Auslöschung

Berechnung des zentralen Differenzenquotienten

$$\frac{e^{1+10^{-4}} - e^{1-10^{-4}}}{2\cdot 10^{-4}} \;=:\; D(h) \tag{1.20}$$

auf einem Taschenrechner mit der Arithmetik $\mathbb{M}(10,\, 10,\, -98,\, +100)$ und Abschneiden, d.h. der relative Abschneidefehler ϵ bei einem Abschneidevorgang erfüllt wegen (1.19) die Relation

$$|\epsilon| \;<\; 10\cdot 10^{-10} = 10^{-9} \tag{1.21}$$

Die Größen $1 + 10^{-4} = 1.0001$ und $1 - 10^{-4} = 0.9999$ sind in unserer Arithmetik exakt dargestellt. Ebenso der Nenner $2 \cdot 10^{-4}$ von (1.20). Statt den Größen

$$e^{1+10^{-4}} = \exp(1 + 10^{-4}) \quad \text{und} \quad e^{1-10^{-4}} = \exp(1 - 10^{-4})$$

entstehen im Rechner Näherungsausdrücke

$$\widetilde{\exp}(1 + 10^{-4}) \quad \text{und} \quad \widetilde{\exp}(1 - 10^{-4}),$$

die aufgrund von (1.19) den folgenden Relationen genügen:

$$\begin{aligned} \widetilde{\exp}(1 + 10^{-4}) &= \exp(1 + 10^{-4}) \cdot (1 + \epsilon_1) \\ \widetilde{\exp}(1 - 10^{-4}) &= \exp(1 - 10^{-4}) \cdot (1 + \epsilon_2) \end{aligned} \tag{1.22}$$

wobei wegen (1.21) gilt

$$|\epsilon_1| < 10^{-9} \text{ und } |\epsilon_2| < 10^{-9}$$

Dabei wird angenommen dass die Standardfunktion exp am Rechner von so guter Qualität ist, dass bei der Auswertung der Exponentialfunktion maximal ein elementarer Rundungsfehler wie bei einer Abschneideoperation gemacht wird, d.h. dass stets gilt:

$$\widetilde{\exp}(\text{Maschinenzahl}) = \exp(\text{Maschinenzahl})(1 + \epsilon) \quad \text{mit} \quad |\epsilon| < 10^{-9}.$$

Bei der Berechnung des Zählers von (1.20) wird die Differenz der beiden verfälschten Größen $\widetilde{\exp}(1 + 10^{-4})$ und $\widetilde{\exp}(1 - 10^{-4})$ exakt gebildet, d.h. bei dieser Differenzbildung wird kein neuer Abschneidefehler generiert.[4)] Im Taschenrechner entsteht also als Zähler des Quotienten in (1.20) die Größe:

$$\begin{aligned} &\widetilde{\exp}(1 + 10^{-4}) - \widetilde{\exp}(1 - 10^{-4}) = \\ &\quad = \exp(1 + 10^{-4}) \cdot (1 + \epsilon_1) - \exp(1 - 10^{-4}) \cdot (1 + \epsilon_2) \end{aligned} \tag{1.23}$$

Bei der Division wird allerdings wieder ein Abschneidefehler gemacht[5)], es entsteht also im Taschenrechner der rechenfehlerbehaftete Quotient

$$\tilde{D}(h) = \frac{e^{1+10^{-4}}(1 + \epsilon_1) - e^{1-10^{-4}}(1 + \epsilon_2)}{2 \cdot 10^{-4}}(1 + \epsilon_3) \tag{1.24}$$

wobei ϵ_3 der relative Abschneidefehler der Division ist und und natürlich auch der Beziehung $|\epsilon_3| < 10^{-9}$ genügt. Bei der nun folgenden Umformung von $\tilde{D}(h)$ werden Produkte $\epsilon_i \cdot \epsilon_k$ vernachlässigt, da die ϵ_i in der Größenordnung 10^{-9} sind und diese Produkte daher in der Größenordnung 10^{-18}.

$$\begin{aligned} \tilde{D}(h) &= \frac{e^{1+10^{-4}}(1 + \epsilon_1)(1 + \epsilon_3) - e^{1-10^{-4}}(1 + \epsilon_2)(1 + \epsilon_3)}{2 \cdot 10^{-4}} \approx \\ &\approx \underbrace{\frac{e^{1+10^{-4}} - e^{1-10^{-4}}}{2 \cdot 10^{-4}}}_{D(h) \text{ exakt}} + \underbrace{\frac{e^{1+10^{-4}}(\epsilon_1 + \epsilon_3) - e^{1-10^{-4}}(\epsilon_2 + \epsilon_3)}{2 \cdot 10^{-4}}}_{\text{Abschneidefehler}} \end{aligned} \tag{1.25}$$

[4)] Vgl. Seite 21 (□): Bei der Differenz von zwei Zahlen aus demselben Exponentialbereich wird kein neuer Runde- oder Abschneidefehler generiert.

[5)] Nur wenn zufällig die letzte Stelle des Zählers gerade wäre, wäre die Division durch 2 exakt.

woraus sich die Fehlerabschätzung[6)]

$$|\tilde{D}(h) - D(h)| \;<\; \frac{\exp(1+10^{-4})\cdot 4\cdot 10^{-9}}{2\cdot 10^{-4}} \;<\; 2.72\cdot 2\cdot 10^{-5} \tag{1.26}$$

ergibt. Ganz analog hätte man für $h = 10^{-9}$ erhalten:

$$|\tilde{D}(h) - D(h)| \;<\; \frac{\exp(1+10^{-9})\cdot 4\cdot 10^{-9}}{2\cdot 10^{-9}} \;<\; 2.72\cdot 2 \tag{1.27}$$

was deutliche zeigt, wie die Fehlerschranke für den Abschneidefehler für $h \to 0$ explodiert.

Betrachtet man den Verfahrensfehler (konvergiert wie h^2 gegen Null) und den Rechenfehler gemeinsam, ergibt sich etwa das Bild aus Abb. 1.14.

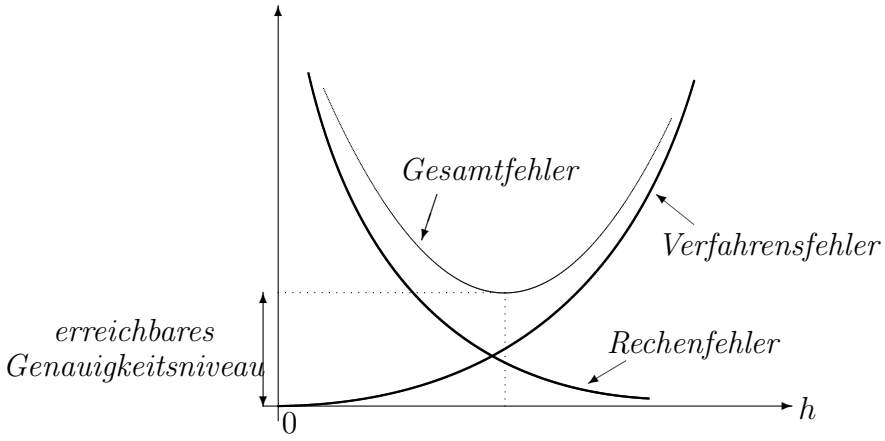


Abbildung 1.14: Gesamtfehler bei numerischer Differentiation

Diese Technik der Rundungsfehleranalysen (bei jedem Rundungs – oder Abschneideprozess einen $(1+\epsilon)$-Faktor anzuhängen) kann im Prinzip auf verschiedenste Algorithmen angewendet werden. Für umfangreiche Algorithmen mit vielen Rechenoperationen wird sie jedoch schnell unübersichtlich und führt nur in Ausnahmefällen zum Erfolg. Außerdem sind die sich ergebenden Schranken für umfangreiche Algorithmen in der Praxis meist zu pessimistisch. Die garantierte Schranke, die für alle denkbaren Datenkonstellationen des Algorithmus stimmen muss, muss immer vom ungünstigsten Fall ausgehen, z.B. bei der Betragsabschätzung von dem Fall, dass sich alle einzelnen Rundungs- oder Abschneidefehler aufaddieren oder akkumulieren, während in den meisten Fällen eine gewisse Kompensation und Auslöschung der einzelnen Rundungsfehler eintreten wird. In der Praxis bevorzugt man bei umfangreichen Algorithmen daher andere Techniken zur Rundungsfehleranalyse.

1.4.2 Rundungsfehleranalysetechniken

Statistische Schätzungen der Rundungsfehler

A-posteriori (Ab)schätzungen, die zwar nicht den Charakter einer mathematischen Aussage für alle denkbaren Anwendungsfälle des untersuchten Algorithmus haben, aber dafür in konkreten Fällen unter Einbeziehung des rundungsfehlerbehaften Resultates zu einer a-posteriori Schätzung des Rundungfehlers führen.

[6)] Eine schärfere Abschätzung wäre möglich, wenn man $e^{1+10^{-4}}\epsilon_3 - e^{1-10^{-4}}\epsilon_3$ nicht durch $2e^{1+10^{-4}}\epsilon_3$ abschätzen würde.

Experimentielle Rundungsfehlerschätzungen: Durch künstliche (mit Zufallsgenerator erzeugte) Störungen bei den einzelnen Rechenschritten eines Algorithmus wird die Rundungsfehlersensitivität des Algorithmus bezüglich der gegebenen Problemdaten experimentiell erprobt.

$(1+\epsilon)$ - Technik: Die oben in Beispiel 1.4.2 beschriebene $(1+\epsilon)$-Technik wird i.a. nur zur Analyse von kleineren Programmstücken eingesetzt etwa innerhalb eines Algorithmus soll $\sqrt{x+1}-\sqrt{x}$ berechnet werden, wobei x ein Zwischenergebnis ist, das sich aufgrund von früheren Manipulationen ergibt. Wegen

$$\sqrt{x+1}-\sqrt{x} = \frac{(\sqrt{x+1}-\sqrt{x})(\sqrt{x+1}+\sqrt{x})}{\sqrt{x+1}+\sqrt{x}} = \frac{1}{\sqrt{x+1}+\sqrt{x}}$$

hat man die Wahl, ob man direkt $\sqrt{x+1}-\sqrt{x}$ oder statt dessen $\frac{1}{\sqrt{x+1}+\sqrt{x}}$ berechnet. I.a. wird wegen der zusätzlichen Division der zweite Weg ungünstiger sein, aber für $x \gg 0$ ist wegen der Auslöschung bei $\sqrt{x+1}-\sqrt{x}$ der zweite Weg weit überlegen.

Kapitel 2

Numerische Lösung linearer Gleichungssysteme

2.1 Grundlagen aus der linearen Algebra

$m \times n$ - **Matrix** A: m Zeilen, n Spalten, d.h. $A \in \mathbb{R}^{m \times n}$

$$A = \begin{pmatrix} a_{11} & \cdots & a_{1n} \\ \vdots & & \vdots \\ a_{m1} & \cdots & a_{mn} \end{pmatrix} \tag{2.1}$$

Vektoren in $\mathbb{R}^n$: n-Tupel reeller Zahlen

$$\vec{x} = \begin{pmatrix} x_1 \\ \vdots \\ x_n \end{pmatrix} \in \mathbb{R}^n$$

Spaltenvektoren (d.h. $n \times 1$ - Matrizen)

$A\vec{x}$ ... Algebraische Schreibweise einer **linearen Abbildung** $\mathbb{R}^n \to \mathbb{R}^m$:

$$\underbrace{\begin{pmatrix} a_{11} & \cdots & a_{1n} \\ \vdots & & \vdots \\ a_{m1} & \cdots & a_{mn} \end{pmatrix}}_{A} \underbrace{\begin{pmatrix} x_1 \\ \vdots \\ x_n \end{pmatrix}}_{\vec{x}} = \underbrace{\begin{pmatrix} a_{11}x_1 & + & \cdots & + & a_{1n}x_n \\ \vdots & & & & \vdots \\ a_{m1}x_1 & + & \cdots & + & a_{mn}x_n \end{pmatrix}}_{\text{Vektor} \in \mathbb{R}^m} \tag{2.2}$$

$A\vec{x}$ ist tatsächlich eine lineare Abbildung, denn die Linearitätsaxiome sind erfüllt:

$$\begin{aligned} A(\vec{x}_1 + \vec{x}_2) &= A\vec{x}_1 + A\vec{x}_2 \qquad \forall \vec{x}_1, \vec{x}_2 \in \mathbb{R}^n \\ A(\lambda\vec{x}) &= \lambda A\vec{x} \qquad \lambda \in \mathbb{R}, \forall x \in \mathbb{R}^n \end{aligned} \tag{2.3}$$

Bedeutung der Spalten von A: Wir betrachten die Standardbasis des $\mathbb{R}^n$ (*kanonische Basis*):

$$\vec{e}_1 = \begin{pmatrix} 1 \\ 0 \\ 0 \\ \vdots \\ 0 \end{pmatrix}, \vec{e}_2 = \begin{pmatrix} 0 \\ 1 \\ 0 \\ \vdots \\ 0 \end{pmatrix}, \ldots, \vec{e}_i = \begin{pmatrix} \vdots \\ 0 \\ 1 \\ 0 \\ \vdots \end{pmatrix} \leftarrow i, \ldots, \vec{e}_n = \begin{pmatrix} 0 \\ \vdots \\ 0 \\ 0 \\ 1 \end{pmatrix}, \tag{2.4}$$

dann ist die i-te Spalte ($i = 1(1)n$) von A das Bild von $\vec{e}_i$, wegen

$$A\vec{e}_i = \begin{pmatrix} a_{11} & \cdots & a_{1i} & \cdots & a_{1n} \\ \vdots & & & & \vdots \\ a_{m1} & \cdots & a_{mi} & \cdots & a_{mn} \end{pmatrix} \begin{pmatrix} 0 \\ \vdots \\ 0 \\ 1 \\ 0 \\ \vdots \\ 0 \end{pmatrix} = \begin{pmatrix} a_{1i} \\ \vdots \\ a_{mi} \end{pmatrix} \tag{2.5}$$

Umkehrung von (2.2): *Jede* lineare Abb. $\phi : \mathbb{R}^n \to \mathbb{R}^m$ lässt sich als $\phi(\vec{x}) = A\vec{x}$ schreiben, wobei die Spalten von A die Bilder $\phi(\vec{e}_i)$ der Basisvektoren $\vec{e}_i$, $i = 1, 2, ..., n$ sind:

$$\vec{x} \text{ beliebig} \in \mathbb{R}^n \qquad \vec{x} = \sum_{i=1}^{n} x_i \vec{e}_i$$

$$\phi(\vec{x}) = \phi(\sum_{i=1}^{n} x_i \vec{e}_i) = \sum_{i=1}^{n} x_i \phi(\vec{e}_i) = \sum_{i=1}^{n} x_i \begin{pmatrix} a_{1i} \\ \vdots \\ a_{mi} \end{pmatrix} =$$

$$= \begin{pmatrix} a_{11}x_1 & + & \cdots & + & a_{1n}x_n \\ \vdots & & & & \vdots \\ a_{m1}x_1 & + & \cdots & + & a_{mn}x_n \end{pmatrix} = Ax \tag{2.6}$$

Transponierte Matrix $A^\top \in \mathbb{R}^{n \times m}$ zu $A \in \mathbb{R}^{m \times n}$,

$$A^\top = \begin{pmatrix} a_{11} & \cdots & a_{m1} \\ \vdots & & \vdots \\ a_{1n} & \cdots & a_{mn} \end{pmatrix} \text{ falls } A = \begin{pmatrix} a_{11} & \cdots & a_{1n} \\ \vdots & & \vdots \\ a_{m1} & \cdots & a_{mn} \end{pmatrix}. \tag{2.7}$$

Ein Spaltenvektor $\vec{x} \in \mathbb{R}^n$ kann durch $\vec{x}^\top = (x_1, x_2, ..., x_n)$ als Zeilenvektor geschrieben werden, d.h. aus der einspaltigen Matrix $\begin{pmatrix} x_1 \\ \vdots \\ x_n \end{pmatrix}$ wird durch Transposition die einzeilige Matrix $(x_1, \ldots, x_n)$.

Bemerkung: Die Bedeutung der linearen Algebra liegt vor allem auch in den Anwendungen: Lösung linearer Gleichungssysteme, Verständnis der Struktur linearer Abbildungen, ... Die wichtigen

Begriffe *linear abhängig* und *linear unabhängig* sind auf die Lösungstheorie linearer Gleichungssysteme zugeschnitten. Geometrische Deutung eines 3×3 linearen Gleichungssystems $A\vec{x} = \vec{b}$ ($A \in \mathbb{R}^{3\times 3}$, $\vec{b} \in \mathbb{R}^3$), den Vektor

$$\vec{b} = \begin{pmatrix} a_{11} \\ a_{21} \\ a_{31} \end{pmatrix} x_1 + \begin{pmatrix} a_{12} \\ a_{22} \\ a_{32} \end{pmatrix} x_2 + \begin{pmatrix} a_{13} \\ a_{23} \\ a_{33} \end{pmatrix} x_3$$

bezüglich der Basis

$$\begin{pmatrix} a_{11} \\ a_{21} \\ a_{31} \end{pmatrix}, \begin{pmatrix} a_{12} \\ a_{22} \\ a_{32} \end{pmatrix}, \begin{pmatrix} a_{13} \\ a_{23} \\ a_{33} \end{pmatrix}$$

also den Spaltenvektoren von A darstellen.

Dabei erkennt man sofort mögliche Entartungsfälle, etwa wenn die drei Vektoren $\vec{a}_1, \vec{a}_2, \vec{a}_3$ in einer Ebene liegen, also gar keine Basis bilden, dann lässt sich $\vec{b}$ i.a. nicht so darstellen, (außer wenn $\vec{b}$ in der Ebene liegt, wobei in diesem Fall die eindeutige Darstellung von $\vec{b}$, also die eindeutige Lösbarkeit des Gleichungssystems verloren geht). In diesem Entartungsfall ist jeder der Vektoren $\vec{a}_i$ eine Linearkombination der anderen beiden. Durch diese Überlegungen wird man in ganz natürlicher Weise zu den nachfolgenden Begriffsbildungen geführt:

Lineare Abhängigkeit, lineare Unabhängigkeit von Vektoren im $\mathbb{R}^m$

1. n Vektoren $\vec{a}_1, \ldots, \vec{a}_n$ aus dem $\mathbb{R}^m$ heißen *linear abhängig* (l.a.), falls es n Konstanten $c_1, \ldots, c_n \in \mathbb{R}$ gibt, die nicht alle gleichzeitig 0 sind, sodass die folgende Beziehung gilt:

$$c_1\vec{a}_1 + \cdots + c_n\vec{a}_n = \vec{0} \tag{2.8}$$

2. Falls jedoch aus (2.8) notwendig $c_1 = c_2 = \cdots = c_n = 0$ folgt, heißen die n Vektoren *linear unabhängig* (l.ua.).

Wir betrachten jetzt n linear unabhängige Vektoren $\vec{a}_1, \ldots, \vec{a}_n \in \mathbb{R}^m$ und die Menge aller Linearkombinationen dieser Vektoren:

$$\vec{a} = c_1\vec{a}_1 + \cdots + c_n\vec{a}_n \qquad c_1, ..., c_n \in \mathbb{R} \tag{2.9}$$

Diese Vektoren $\vec{a}$ bilden einen *linearen Unterraum* des $\mathbb{R}^m$ (denn mit $\vec{a} = c_1\vec{a}_1 + \cdots + c_n\vec{a}_n$ und mit $\bar{\vec{a}} = \bar{c}_1\vec{a}_1 + \cdots + \bar{c}_n\vec{a}_n$ liegt auch $\vec{a} + \bar{\vec{a}} = (c_1 + \bar{c}_1)\vec{a}_1 + \cdots + (c_n + \bar{c}_n)\vec{a}_n$ in dieser Menge der Linearkombinationen und ebenso $\lambda\vec{a} = (\lambda c_1)\vec{a}_1 + \cdots + (\lambda c_n)\vec{a}_n$ für $\lambda \in \mathbb{R}$). Im Spezialfall $n = m$ fällt dieser Unterraum mit dem ganzen $\mathbb{R}^m$ zusammen, $n > m$ ist nicht möglich, da dann die Vektoren nicht l.ua. sein können (im $\mathbb{R}^2$ kann es z.B. nicht drei l.ua. Vektoren geben, da drei Vektoren in einer Ebene stets l.a. sind). Jeder Vektor $\vec{a}$ aus diesen Unterraum ist durch die **Koordinaten** $c_1, \ldots, c_n \in \mathbb{R}$ *eindeutig* charakterisiert, denn indirekt angenommen

$$\begin{aligned} \vec{a} &= c_1\vec{a}_1 + \cdots + c_n\vec{a}_n \quad \text{und} \\ \vec{a} &= d_1\vec{a}_1 + \cdots + d_n\vec{a}_n \end{aligned} \tag{2.10}$$

$$\Rightarrow \vec{a} - \vec{a} = 0 = (c_1 - d_1)\vec{a}_1 + \cdots + (c_n - d_n)\vec{a}_n \tag{2.11}$$

und wegen der linearen Unabhänigkeit der Vektoren $\vec{a}_1, \ldots, \vec{a}_n$ folgt $c_1 = d_1, \ldots, c_n = d_n$.

Der Unterraum ist n-**dimensional** (bringt zum Ausdruck, dass jeder Vektor aus diesem Unterraum durch n Koordinaten eindeutig festgelegt ist, in den Fällen $n = 1, 2, 3$ deckt sich diese Sprechweise mit der üblichen anschaulichen Bedeutung des Begriffes *Dimension*) heißt

die n l.ua. Vektoren $\vec{a}_1, \ldots, \vec{a}_n$ bilden eine **Basis** dieses Unterraumes oder die Vektoren $\vec{a}_1, \ldots, \vec{a}_n$ spannen diesen Unterraum auf.

Rang einer Matrix $A \in \mathbb{R}^{m\times n}$ oder einer durch A repräsentierten linearen Abbildung: Falls der Vektor $\vec{x}$ den $\mathbb{R}^n$ durchläuft, dann durchläuft die Menge aller Bilder $A\vec{x} =: \vec{y}$ eine Teilmenge des $\mathbb{R}^m$ insbesondere einen linearen Teilraum des $\mathbb{R}^m$. Dieser Teilraum ist der **Bildraum** der betrachteten linearen Abbildung und wird als das Bild von A $Bild(A)$ bezeichnet:

$$Bild(A) \;:=\; \{\vec{y} \in \mathbb{R}^m :\; \vec{y} = A\vec{x},\, \vec{x} \in \mathbb{R}^n\} \tag{2.12}$$

die Dimension dieses Bildraumes $\dim(Bild(A))$ heißt der *Rang* von A:

$$r \;=\; Rang(A) \;=\; \dim(Bild(A)) \tag{2.13}$$

Alternative Möglichkeiten, den Rang zu definieren:

i) *Spaltenrang*: Wir betrachten die Menge aller Spaltenvektoren von A:

$$\vec{a}_1 = \begin{pmatrix} a_{11} \\ a_{21} \\ \vdots \\ a_{m1} \end{pmatrix},\; \vec{a}_2 = \begin{pmatrix} a_{12} \\ a_{22} \\ \vdots \\ a_{m2} \end{pmatrix},\; \ldots,\; \vec{a}_n = \begin{pmatrix} a_{1n} \\ a_{2n} \\ \vdots \\ a_{mn} \end{pmatrix} \tag{2.14}$$

Die maximale Anzahl r linear unabhängiger Spaltenvektoren heißt der Spaltenrang von A. Es gilt, Spaltenrang $= \dim(Bild(A))$:

$$\begin{aligned} A\vec{x} \;&=\; \begin{pmatrix} a_{11}x_1 & + & \cdots & + & a_{1n}x_n \\ & & \vdots & & \\ a_{m1}x_1 & + & \cdots & + & a_{mn}x_n \end{pmatrix} = \\ &=\; x_1 \begin{pmatrix} a_{11} \\ \vdots \\ a_{m1} \end{pmatrix} + \cdots + x_n \begin{pmatrix} a_{1n} \\ \vdots \\ a_{mn} \end{pmatrix} = \\ &=\; x_1\vec{a}_1 + \cdots + x_n\vec{a}_n \end{aligned}$$

Also wenn $\vec{x}$ den $\mathbb{R}^n$ durchläuft, durchläuft $A\vec{x}$ den durch die Spaltenvektoren $\vec{a}_1, \ldots, \vec{a}_n$ aufgespannten Unterraum des $\mathbb{R}^m$. Dieser Unterraum ist der Bildraum $Bild(A)$ und seine Dimension ist durch die Maximalzahl der l.ua. Spaltenvektoren festgelegt. □

ii) *Zeilenrang*: Wir betrachten die Menge aller Zeilenvektoren von A:

$$\begin{array}{llll} (a_{11}, & a_{12}, & \ldots, & a_{1n}) \\ (a_{21}, & a_{22}, & \ldots, & a_{2n}) \\ & \vdots & & \\ (a_{m1}, & a_{m2}, & \ldots, & a_{mn}) \end{array} \tag{2.15}$$

Die maximale Anzahl r linear unabhängiger Zeilenvektoren heißt der Zeilenrang von A. Es gilt auch jetzt wieder: Zeilenrang $= \dim(Bild(A))$

Zusammenfassung: *Die drei Möglichkeiten, den Rang zu definieren, als Zeilenrang, oder als Spaltenrang oder als* $\dim(Bild(A))$ *sind äquivalent.*

- **Kern** einer linearen Abbildung ist Menge aller $\vec{x} \in \mathbb{R}^n$, für die gilt $A\vec{x} = \vec{0}$:

$$Kern(A) \;:=\; \{\vec{x} :\, A\vec{x} = \vec{0},\, \vec{x} \in \mathbb{R}^n\} \tag{2.16}$$

Der Kern ist ein linearer Unterraum des $\mathbb{R}^n$. Seine Dimension ist wichtig im Zusammenhang mit der Lösungstheorie von linearen Gleichungsystemen $A\vec{x} = \vec{b}$, siehe Abschnitt 2.2.

2.2 Lösungstheorie für lineare Gleichungssysteme

Lineares Gleichungssystem: gegebene Daten sind $A \in \mathbb{R}^{m\times n}$, $\vec{b} \in \mathbb{R}^m$, m, n beliebig aus $\mathbb{N}$, gesucht ist ein Vektor $\vec{x} \in \mathbb{R}^n$, sodass

$$\begin{array}{ccccccccc} a_{11}x_1 & + & a_{12}x_2 & + & \cdots & + & a_{1n}x_n & = & b_1 \\ a_{21}x_1 & + & a_{22}x_2 & + & \cdots & + & a_{2n}x_n & = & b_3 \\ \vdots & & & & & & \vdots & & \vdots \\ a_{m1}x_1 & + & a_{m2}x_2 & + & \cdots & + & a_{mn}x_n & = & b_m \end{array} \tag{2.17}$$

oder in Matrix-Vektor-Notation:

$$A\vec{x} \;=\; \vec{b} \text{ mit} \tag{2.18}$$

$$A = \begin{pmatrix} a_{11} & a_{12} & \dots & a_{1n} \\ a_{21} & a_{22} & \dots & a_{2n} \\ \vdots & \vdots & & \vdots \\ a_{m1} & a_{m2} & \dots & a_{mn} \end{pmatrix} \quad \vec{x} = \begin{pmatrix} x_1 \\ x_2 \\ \vdots \\ x_n \end{pmatrix} \quad \vec{b} = \begin{pmatrix} b_1 \\ b_2 \\ \vdots \\ b_n \end{pmatrix}$$

Die Matrix $A = (a_{ij})$ heißt *Koeffizientenmatrix* des linearen Gleichungssystems. Der Spaltenvektor $\vec{b}$ wird als *Inhomogenität* oder *rechte Seite* bezeichnet. Das System heißt *homogen*, falls $\vec{b} = \vec{0}$.

Lineare Gleichungssysteme treten in sehr vielen Anwendungen auf. Manchmal führt ein einfacher mathematischer Modellierungsprozess unmittelbar zu einem linearen Gleichungssystem. Sehr häufig liegt aber auch eine etwas andere Situation vor: Zunächst führt der mathematische Modellbildungsprozess auf ein anders geartetes mathematisches Problem, und erst ein allfälliger numerischer Algorithmus zur Lösung dieses Problems resultiert schließlich in einem (oder mehreren) linearen Gleichungssystem(en).

Zwei Lösungsbegriffe:

(i) Lösung $\vec{x} \in \mathbb{R}^n$ gesucht, sodass das Gleichungssystem $A\vec{x} = \vec{b}$ erfüllt ist.

(ii) Lösung im **Ausgleichssinn** : Falls ein $\vec{x}$ das $A\vec{x} = \vec{b}$ im Sinne von (i) löst nicht existiert sind die Gleichungen *widersprüchlich*. Dann wenigstens $\vec{x}$ so bestimmen, dass $\vec{x}$ in das Gleichungssystem so gut wie möglich hineinpasst. D.h. $\vec{x}$ ist dann dadurch festgelegt, dass

$$\|A\vec{x} - \vec{b}\|_2 \quad \dots \quad \min! \tag{2.19}$$

wobei $\|\cdot\|_2$ die Euklidische Norm , also die Länge eines Vektors ist, siehe (2.49).

Eine wesentliche Rolle bei der Lösbarkeit linearer Gleichungssysteme spielt die *erweiterte Matrix:*

$$(A|\vec{b}) = \left(\begin{array}{cccc|c} a_{11} & a_{12} & \dots & a_{1n} & b_1 \\ a_{21} & a_{22} & \dots & a_{2n} & b_2 \\ \vdots & \vdots & & \vdots & \vdots \\ a_{m1} & a_{m2} & \dots & a_{mn} & b_m \end{array}\right)$$

Ein lineares inhomogenes Gleichungssystem $A\vec{x} = \vec{b}$ ist genau dann lösbar, wenn

$$Rang(A) = Rang(A|\vec{b}).$$

Für den Spezialfall $Rang(A) = m$ hat das Gleichungssystem immer eine Lösung. (Da die Spalten von A und von $(A|\vec{b})$ Elemente von $\mathbb{R}^m$ sind, gilt $m = Rang(A) \leq Rang((A|\vec{b})) \leq m$.)

Für $m = n = Rang(A)$ ist das lineare Gleichungssystem $A\vec{x} = \vec{b}$ für beliebige rechte Seiten $\vec{b} \in \mathbb{R}^m = \mathbb{R}^n$ *eindeutig lösbar*. Die l.u.a. Spalten $\vec{a}_1, \vec{a}_2, ..., \vec{a}_n$ von A bilden eine Basis des $\mathbb{R}^n$, d.h. jeder Vektor $\vec{b} \in \mathbb{R}^n$ lässt sich eindeutig schreiben als Linearkombination $\vec{b} = x_1\vec{a}_1 + \cdots + x_n\vec{a}_n$. Die eindeutig durch $\vec{b}$ festgelegten Gewichte sind somit offenbar die eindeutige Lösung von $A\vec{x} = \vec{b}$.

2.3 Konditionsabschätzungen

Modellfehler, Datenfehler und Rechenfehler beim Einlesevorgang der Koeffizienten verfälschen ein lineares Gleichungssystem. Statt des "wahren" Systems

$$A\vec{x} = \vec{b} \tag{2.20}$$

hat man i.a. das verfälschte System

$$\tilde{A}\tilde{\vec{x}} = \tilde{\vec{b}} \tag{2.21}$$

im Rechner. Es müssen daher die Auswirkungen der Datenstörungen auf das Ergebnis, also der *absolute Fehler*

$$\|\tilde{\vec{x}} - \vec{x}\|$$

oder der *relative Fehler*

$$\frac{\|\tilde{\vec{x}} - \vec{x}\|}{\|\vec{x}\|} \quad \text{bzw.} \quad \frac{\|\tilde{\vec{x}} - \vec{x}\|}{\|\tilde{\vec{x}}\|}$$

abgeschätzt werden.

2.3.1 Konditionsabschätzungen bezüglich Störungen von $\vec{b}$

$$\begin{array}{rll} \text{Ungestörtes Problem:} & A\vec{x} = \vec{b}, & (A \dots \text{regulär}) \\ \text{Gestörtes Problem:} & A\tilde{\vec{x}} = \tilde{\vec{b}} = \vec{b} + \Delta\vec{b} & \end{array}$$

$$\Rightarrow\ A(\vec{\tilde{x}} - \vec{x}) = \Delta\vec{b} \quad \Rightarrow\ \Delta\vec{x} := (\vec{\tilde{x}} - \vec{x}) = A^{-1}\Delta\vec{b} \quad \Rightarrow$$

$$\boxed{\|\Delta\vec{x}\| \ = \ \|\vec{\tilde{x}} - \vec{x}\| \ \leq \ \|A^{-1}\|\ \|\Delta\vec{b}\|} \tag{2.22}$$

absolute Konditionsabschätzung

Um eine relative Konditionsabschätzung zu erhalten, werden alle absoluten Fehlergrößen durch relative Fehlergrößen ersetzt, z.B. $\|\Delta\vec{b}\|$ durch $\frac{\|\Delta\vec{b}\|}{\|\vec{b}\|}$ oder $\|\Delta\vec{x}\|$ durch $\frac{\|\Delta\vec{x}\|}{\|\vec{x}\|}$:

$$\|\Delta\vec{x}\| \ \leq \ \|A^{-1}\|\ \|\Delta\vec{b}\| \quad \text{Vgl. (2.22)} \quad \Rightarrow$$

$$\boxed{\frac{\|\Delta\vec{x}\|}{\|\vec{x}\|} \ \leq \ \frac{\|A\|\ \|A^{-1}\|\ \|\Delta\vec{b}\|}{\|A\|\ \|\vec{x}\|} \ \leq \ \|A\|\ \|A^{-1}\|\frac{\|\Delta\vec{b}\|}{\|\vec{b}\|}} \tag{2.23}$$

relative Konditionsabschätzung

Dabei wurde benützt: $\|A\|\ \|\vec{x}\| \geq \|A\vec{x}\| = \|\vec{b}\|$.
Diese Abschätzung ist also vom Typ:

$$\|\ \text{relative Störung des Ergebnisses } \vec{x}\ \| \ \leq\ \text{Faktor} \cdot \|\ \text{relative Störung von } \vec{b}\ \|$$

mit dem Faktor

$$\kappa(A) = \|A\|\ \|A^{-1}\| \tag{2.24}$$

der sogenannten **Konditionszahl**. Man kann zeigen, dass die Abschätzungen (2.22) und (2.23) scharf sind in folgendem Sinn: Zu jedem A gibt es ein $\vec{b}$ und ein $\Delta\vec{b}$, sodass das Gleichheitszeichen angenommen wird.

2.3.2 Konditionsabschätzungen bezüglich Störungen von A

Ungestörtes Problem: $A\vec{x} = \vec{b}$, $(A \ldots \text{regulär})$
Gestörtes Problem: $\tilde{A}\vec{\tilde{x}} = \vec{b} = (A + \Delta A)\vec{\tilde{x}} = \vec{b}$ $(\tilde{A} = A + \Delta A \ldots \text{regulär})$

$$\begin{aligned}
\Rightarrow\ \tilde{A}\vec{\tilde{x}} = (A + \Delta A)(\vec{x} + \Delta\vec{x}) &= \vec{b} \quad \Rightarrow \\
Ax + \Delta Ax + A\Delta x + \Delta A\Delta x &= b \quad \Rightarrow \\
A\Delta\vec{x} = -\Delta A(\vec{x} + \Delta\vec{x}) &= -\Delta A\vec{\tilde{x}} \quad \Rightarrow \\
\Delta\vec{x} &= -A^{-1}\Delta A\vec{\tilde{x}} \quad \Rightarrow
\end{aligned}$$

$$\boxed{\|\Delta\vec{x}\| \ \leq \ \|A^{-1}\|\ \|\vec{\tilde{x}}\|\ \|\Delta A\|} \tag{2.25}$$

absolute Konditionsabschätzung bezüglich Störungen von A

$$\boxed{\frac{\|\Delta\vec{x}\|}{\|\vec{\tilde{x}}\|} \ \leq \ \|A\|\ \|A^{-1}\|\frac{\|\Delta A\|}{\|A\|}} \tag{2.26}$$

relative Konditionsabschätzung bezüglich Störungen von A

Wieder tritt als Faktor die Konditionszahl $\kappa(A)$ auf. Beim relativen Fehler $\frac{\|\Delta\vec{x}\|}{\|\vec{\tilde{x}}\|}$ ist allerdings nicht $\|\vec{x}\|$ sondern $\|\tilde{x}\|$ der Vergleichswert. Will man eine relative Abschätzung bez. $\frac{\|\Delta\vec{x}\|}{\|\vec{x}\|}$, so kann man so vorgehen:

$$\begin{aligned}(A+\Delta A)(x+\Delta x) &= b \quad \Rightarrow\\ Ax+\Delta Ax+\underbrace{A\Delta x+\Delta A\Delta x}_{\tilde{A}\Delta x} &= b \quad \Rightarrow\\ \tilde{A}\Delta x &= -\Delta Ax \quad \Rightarrow\\ \Delta x &= -\tilde{A}^{-1}\Delta Ax \Rightarrow\end{aligned}$$

$$\boxed{\frac{\|\Delta x\|}{\|x\|} \leq \|A\|\,\|\tilde{A}^{-1}\|\frac{\|\Delta A\|}{\|A\|}} \tag{2.27}$$

Die Konditionszahl ist jetzt $\|A\|\,\|\tilde{A}^{-1}\|$, hier ist also die Regularität von $\tilde{A}$, wesentlich. Man kann eine relative Konditionsabschätzung ganz ohne gestörte Größen folgender Form herleiten:

$$\frac{\|\Delta x\|}{\|x\|} \leq \frac{\kappa(A)}{1-\kappa(A)\frac{\|\Delta A\|}{\|A\|}}\cdot\frac{\|\Delta A\|}{\|A\|}$$

(unter der Voraussetzung, dass $\|\Delta A\|$ so klein ist, dass $\kappa(A)\frac{\|\Delta A\|}{\|A\|} < 1$ gilt).

2.4 Gaußelimination

Im Folgenden sei $A \in \mathbb{R}^{n\times n}$, $\vec{b} \in \mathbb{R}^n$ und $\mathrm{Rang}(A) = n$. Die Lösung erfolgt mit einem *Eliminationsverfahren*, meist mit der sogenannten **Gaußelimination**. Dh.

$$A\vec{x} = \vec{b}, \tag{2.28}$$

ist (bis auf Rundungsfehler) exakt lösbar, es tritt **kein Verfahrensfehler** auf. Es können aber Datenfehler auftreten, wenn man aufgrund der Modellierung oder eventuell auf Grund von Messfehlern verfälschte Koeffizienten in A oder ein verfälschtes $\vec{b}$ hat, und Rundungsfehlerm wenn man den Lösungsalgorithmus auf einem Computer mit einer bestimmten Arithmetik ablaufen lässt.

Beschreibung der Gaußelimination: 1. Besonders leicht zu lösendes Systeme sind die sogenannten *gestaffelte Systeme*

$$\begin{array}{ccccccccccc} a_{11}x_1 & + & a_{12}x_2 & + & \cdots & + & a_{1,n-1}x_{n-1} & + & a_{1,n}x_n & = & b_1\\ & & a_{22}x_2 & + & \cdots & + & a_{2,n-1}x_{n-1} & + & a_{2,n}x_n & = & b_2\\ & & & & \ddots & & & & \vdots & & \\ & & & & & & a_{n-1,n-1}x_{n-1} & + & a_{n-1,n}x_n & = & b_{n-1}\\ & & & & & & & & a_{nn}x_n & = & b_n \end{array} \tag{2.29}$$

$$\Rightarrow \quad x_n = \frac{b_n}{a_{nn}}, \quad x_{n-1} = \frac{b_{n-1}-a_{n-1,n}\frac{b_n}{a_{nn}}}{a_{n-1,n-1}}, \quad \ldots \tag{2.30}$$

2. Die Grundidee der Gaußelimination ist ein allgemeines System vom Typ (2.28) in die Form (2.29) bringen, um anschließend die Lösung gemäß (2.30) zu berechnen.

   Ausgangspunkt ist das *volle* System

$$\begin{array}{ccccccccc} a_{11}x_1 & + & a_{12}x_2 & + & \cdots & + & a_{1n}x_n & = & b_1 \\ a_{21}x_1 & + & a_{22}x_2 & + & \cdots & + & a_{2n}x_n & = & b_2 \\ \vdots & & & & & & & & \vdots \\ a_{n1}x_1 & + & a_{n2}x_2 & + & \cdots & + & a_{nn}x_n & = & b_n \end{array} \tag{2.31}$$

   Multiplikation der ersten Gleichung mit $\frac{a_{21}}{a_{11}}$ und anschließende Subtraktion von der zweiten Gleichung lässt in der zweiten Zeile folgendes entstehen:

$$\begin{aligned} \underbrace{(a_{21} - \frac{a_{21}}{a_{11}}a_{11})}_{0}\, x_1 + \underbrace{(a_{22} - \frac{a_{21}}{a_{11}}a_{12})}_{a'_{22}}\, x_2 + \cdots + \\ + \underbrace{(a_{2n} - \frac{a_{21}}{a_{11}}a_{1n})}_{a'_{2n}}\, x_n \;=\; \underbrace{b_2 - \frac{a_{21}}{a_{11}}b_1}_{b'_2} \end{aligned} \tag{2.32}$$

   Multiplikation der ersten Zeile mit $\frac{a_{31}}{a_{11}}$ und anschließende Subtraktion von der dritten Gleichung lässt in der dritten Zeile folgendes entstehen:

$$\begin{aligned} \underbrace{(a_{31} - \frac{a_{31}}{a_{11}}a_{11})}_{0}\, x_1 + \underbrace{(a_{32} - \frac{a_{31}}{a_{11}}a_{12})}_{a'_{32}}\, x_2 + \cdots + \\ + \underbrace{(a_{3n} - \frac{a_{31}}{a_{11}}a_{1n})}_{a'_{3n}} \cdot x_n \;=\; \underbrace{b_3 - \frac{a_{31}}{a_{11}}b_1}_{b'_3} \end{aligned} \tag{2.33}$$

   u.s.w. Durch diese Manipulationen wird (2.31) in ein äquivalentes System (d.h. ein System mit derselben Lösung) von folgender Gestalt umgeformt:

$$\begin{array}{ccccccccc} a_{11}x_1 & + & a_{12}x_2 & + & \cdots & + & a_{1n}x_n & = & b_1 \\ & & a'_{22}x_2 & + & \cdots & + & a'_{2n}x_n & = & b'_2 \\ & & \vdots & & & & & & \vdots \\ & & a'_{n2}x_2 & + & \cdots & + & a'_{nn}x_n & = & b'_n \end{array} \tag{2.34}$$

   Wenn man nun die analoge Vorgangsweise auf das $(n-1) \times (n-1)$-Teilsystem

$$\begin{array}{ccccccc} a'_{22}x_2 & + & \cdots & + & a'_{2n}x_n & = & b'_2 \\ \vdots & & & & & & \vdots \\ a'_{n2}x_2 & + & \cdots & + & a'_{nn}x_n & = & b'_n \end{array}$$

   von (2.34) anwendet, erhält man

$$\begin{array}{ccccccccccc} a_{11}x_1 & + & a_{12}x_2 & + & a_{13}x_3 & + & \cdots & + & a_{1n}x_n & = & b_1 \\ & & a'_{22}x_2 & + & a'_{23}x_3 & + & \cdots & + & a'_{2n}x_n & = & b'_2 \\ & & & & a''_{33}x_3 & + & \cdots & + & a''_{3n}x_n & = & b''_3 \\ & & & & \vdots & & & & \vdots & & \\ & & & & a''_{n3}x_3 & & \cdots & & a''_{nn}x_n & = & b''_n \end{array} \tag{2.35}$$

u.s.w. Es entstehen durch diese Vorgangsweise nach und nach Matrizen mit der Form wie in Abb. 2.1

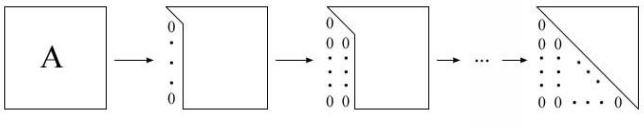


Abbildung 2.1: Reduktion auf Dreiecksgestalt

Am Schluss erhält man ein gestaffeltes System, das äquivalent zum ursprünglichen System ist, und wo man die Lösung gemäß (2.30) berechnen kann.

Die eben beschriebene Vorgangsweise ist ein systematischer Algorithmus und kann leicht programmiert werden.

Frage: Führt dieser Algorithmus immer zum Ziel? Sei

$$A = \begin{pmatrix} 0 & 1 & 1 \\ 1 & 1 & 1 \\ 2 & 1 & 3 \end{pmatrix}$$

dh. $a_{11} = 0$ und die Multiplikatoren $\frac{a_{21}}{a_{11}}$ und $\frac{a_{31}}{a_{11}}$, mit denen die erste Zeile multipliziert wird bevor sie von der zweiten bzw. dritten Zeile abgezogen wird können nicht gebildet werden.

D.h. in diesem Fall muss, bevor der Eliminationsalgorithmus anläuft, die erste Zeile mit der zweiten oder dritten Zeile vertauscht werden. Genau dieselbe Situation tritt ein, wenn später während des Eliminationsvorganges in der Hauptdiagonale eine Null entsteht, sodass der entsprechende Multiplikator wegen Division durch 0 nicht gebildet werden kann. Dann muss die i-te Zeile mit einer weiter unten stehenden Zeile (j-te Zeile mit $j > i$) vertauscht werden, bei der ein nicht verschwindendes Element in der i-ten Spalte steht. (Vgl. Abb. 2.2)

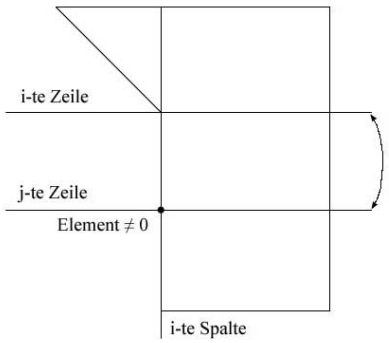


Abbildung 2.2: Vertauschung zweier Zeilen

Nach diesem Vertauschungsprozess kann der Eliminationsvorgang fortgesetzt werden.

Ein echter Zusammenbruch würde nur in folgender Situation entstehen: man kann die i-te Zeile

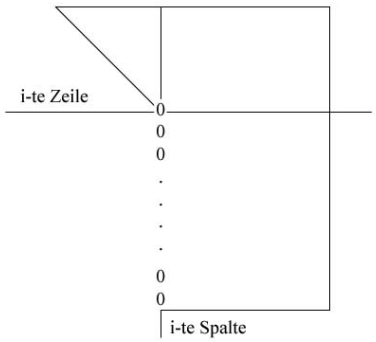


Abbildung 2.3: Zusammenbruch der Gaußelimination

mit der Null in der Hauptdiagonalen nicht wegtauschen, da auch unterhalb der Hauptdiagonalen in der i-ten Spalte lauter Nullen stehen. (Vgl. Abb. 2.3)

Man kann leicht zeigen, dass so ein Fall genau dann eintritt, wenn A singulär ist, also $\text{Rang}(A) < n$ oder $\det A = 0$ vorliegt.

Numerische Problematik: bei rundungsfehlerfreiem exakten Rechnen muss nur dann weggetauscht werden, wenn eine Null in der Hauptdiagonalen enstanden ist, sodass der entsprechende Multiplikator nicht gebildet werdem kann. Beim Rechnen in einer Maschinenarithmetik kann jedoch aufgrund der Rundungsfehler so eine Null verfälscht werden, sodass statt der Null eine sehr kleine Zahl an dieser Stelle in der Hauptdiagonalen steht. Mit dieser nicht verschwindenden Grösse könnte man den entsprechenden Multiplikator berechnen und im Prinzip den Eliminationsvorgang fortsetzen, würde dann jedoch offenbar etwas völlig Unsinniges rechnen. D.h. vom Standpunkt der Numerik aus sollte man nicht nur Nullen sondern auch kleine Elemente wegtauschen. In diesem Sinn hat sich folgende Strategie bewährt:

Vor dem weiteren Eliminationschritt sucht man das betragsgrößte Element aus der i-ten Spalte, aber nur bez. jener Elemente die in und unterhalb der Hauptdiagonalen stehen (vgl. Abb. 2.4).

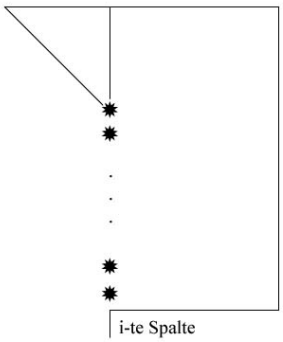


Abbildung 2.4: Suche nach dem betragsgrößten Element

Wenn dieses Element in der j-ten Zeile steht, dann wird die i-te mit der j-ten Zeile vertauscht. Erst anschließend wird der Eliminationsvorgang fortgesetzt (vgl. Abb. 2.5). Diese Vorgangsweise heißt **Spaltenpivotsuche** mit Zeilentausch.

Diese Vorgangsweise lässt sich auch noch anders begründen: Bezügl. der Rundungsfehler ist es

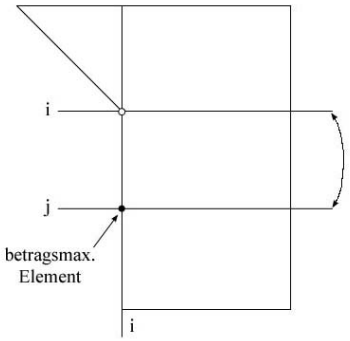


Abbildung 2.5: Spaltenpivotstrategie

i.a. ungünstig, wenn die Koeffizienten von A von stark unterschiedlicher Größenordnung sind; z.B. für

$$\begin{pmatrix} 0.730 & \boxed{0.274} & 0.683 \\ 0.730 & \boxed{21.6} & 0.432 \\ 0.246 & 0.611 & 0.0723 \end{pmatrix}$$

geht bei der Differenzbildung $21.6 - 0.274 = 21.3$ (auf 3 Stellen gerundet) die Information der beiden hinteren Stellen von $0.2\overline{|74}$ verloren. Wenn nun eine Matrix A Koeffizienten unterschiedlicher Größenordnung besitzt, so muss man damit leben, aber man wird vermeiden, dass während dem Eliminationsvorgang die Größenordnung der Koeffizienten von A noch zusätzlich auseinandergezogen wird, beachten Sie: Spaltenpivotsuche mit anschließendem Zeilentausch stellt sicher, dass alle Multiplikationen betragsmässig ≤ 1 sind und vermeidet so dieses zusätzliche Auseinanderziehen der Koeffizienten von A; weiters beachten Sie, dass ohne diese Strategie einige harmlos erscheinende Multiplikatoren der Größenordnung 10 ungünstigen Effekt machen können, wenn sie sich (was im Laufe der Gaußelimination vorkommen kann) ungünstig zusammenmultiplizieren (z.B. können 5 aufeinanderfolgende 10-er einen Effekt der Gößenordnung 10^5 erzielen). Daher ist für die Praxis, insbesonders wenn man Systeme größerer Dimension im Auge hat, wo sich sehr viele harmlose Multiplikatoren zusammenmultiplizieren können, diese Pivotstrategie unerlässlich und man sieht nur Gaußeliminationsalgorithmen mit entsprechenden Pivotstrategien als numerisch stabile Algorithmen an.

Die Spaltenpivotsuche mit anschließendem Zeilentausch wird noch mit der sogenannten **Skalierung** kombiniert. Sie wird anhand von folgendem Beispiel erklärt:

Beispiel

$$\begin{aligned} 0.005x_1 &+ x_2 = 0.5 \\ x_1 &+ x_2 = 1 \end{aligned} \tag{2.36}$$

Die exakte Lösung ist

$$x_1 = \frac{500}{995} = 0.50251\ldots,\ x_2 = \frac{495}{995} = 0.49748\ldots \tag{2.37}$$

Zunächst Rechnung in $\mathbb{M}(10,\, 2, \ldots)$ *ohne Spaltenpivotsuche*, erste Zeile mit $\frac{1}{0.005}$ multiplizieren und von der zweiten Zeile abgeziehen; exakte Rechnung

$$\begin{aligned} 0.005x_1 + x_2 &= 0.5 \\ \underbrace{(1 - 200)}_{-199} x_2 &= \underbrace{(1 - 200 \cdot 0.5)}_{-99} \end{aligned}$$

und Rundung auf 2 Stellen ergibt

$$\begin{aligned} 0.005\tilde{x_1} + \tilde{x}_2 &= 0.5 \\ -200\tilde{x}_2 &= -99 \end{aligned}$$

$$\begin{aligned} \Rightarrow \quad \tilde{x}_2 &= 0.495 \quad \leftarrow \text{ exaktes Divisionsergebnis} \\ &= 0.50 \quad \leftarrow \text{ auf 2 Stellen gerundet} \\ \tilde{x}_1 &= 0. \end{aligned}$$

Rechnung wieder zweistellig, aber *mit Spaltenpivotsuche*, vor der Elimination werden die Zeilen vertauscht

$$\begin{aligned} x_1 + x_2 &= 1 \\ 0.005x_1 + x_2 &= 0.5 \end{aligned}$$

Multiplikation der ersten Zeile mit 0.005 (statt mit $\frac{1}{0.005}$) und Subtraktion von der zweite Zeile liefert

$$\begin{aligned} x_1 + x_2 &= 1 \\ \underbrace{(1 - 0.005)}_{0.995} x_2 &= \underbrace{(0.5 - 0.005)}_{0.495} \end{aligned}$$

und Rundung auf 2 Stellen

$$\begin{aligned} \tilde{x}_1 + \tilde{x}_2 &= 1 \\ \tilde{x}_2 &= 0.5 \end{aligned}$$

$$\Rightarrow \qquad \tilde{x}_2 = 0.5, \qquad \tilde{x}_1 = 0.5$$

Rundet man das exakte Ergebnis (vgl. (2.37)) auf zwei Stellen, so erhält man ebenso $\tilde{x}_1 = 0.5$, $\tilde{x}_2 = 0.5$. Man erkennt also tatsächlich die Überlegenheit der Variante mit Spaltenpivotsuche.

Diese vorteilhafte Wirkung der Spaltenpivotsuche kann aber sehr leicht zerstört werden, wenn man die erste Zeile von (2.36) mit einer hinreichend großen Zahl, etwa 400 multipliziert:

$$\begin{aligned} 2x_1 + 400x_2 &= 200 \\ x_1 + x_2 &= 1 \end{aligned} \tag{2.38}$$

Wenn durch irgendeinen Zufall das System statt in der Form (2.36) in der (dazu äquivalenten) Form (2.38) gegeben wäre, würde trotz Spaltenpivotsuche wegen $2 > 1$ kein Zeilentausch erfolgen und daher die Elimination rundungsfehlermässig in der unverteilhaften Variante ablaufen.

Abhilfe, die solche Effekte vermeidet ist die **Zeilenskalierung**, vor Beginn des Eliminationsprozesses wird jede Zeile mit einem Faktor d_i multipliziert, sodass die maximalen Elemente aller Zeilen dieselbe Größenordnung haben,

$$\text{z.B.} \quad d_i = \frac{1}{\max_{1 \le j \le n} |a_{ij}|} \quad \text{oder} \quad d_i = \frac{1}{\sum_{j=1}^{n} |a_{ij}|}$$

oder etwas Ähnliches. Die so skalierte Version von (2.38) ist dann

$$\begin{aligned} 0.005x_1 + x_2 &= 0.5 \\ 0.5x_1 + 0.5x_2 &= 0.5 \end{aligned} \tag{2.39}$$

Lösung von (2.39) in zweistelliger Arithmetik liefert wieder $\tilde{x}_1 = 0.5$, $\tilde{x}_2 = 0.5$.

2.5 Rundungsfehler bei der Gaußelimination

Satz 2.5.1 (v. Wilkinson)**.** Gegeben sei $A\vec{x} = \vec{b}$ ($A \in \mathbb{R}^{n\times n}$, regulär) mit der exakten (d.h. rundungsfehlerfreien) Lösung $\vec{x}$. Weiters sei $\vec{\tilde{x}}$ die rundungsfehlerbehaftete Lösung, die dadurch zustande kommt, dass man $A\vec{x} = \vec{b}$ mit Gaußelimination mit Spaltenpivotsuche löst, und zwar in einer bestimmten Maschinenarithmetik. Dann lässt sich $\vec{\tilde{x}}$ interpretieren als die exakte, rundungsfehlerfreie Lösung eines gestörten Systems

$$(A + \Delta A)\vec{\tilde{x}} \;=\; \vec{b} \tag{2.40}$$

mit

$$\frac{\|\Delta A\|}{\|A\|} \;\leq\; 1.01\,(n^3 + 3n^2)\,\gamma\, eps. \tag{2.41}$$

Dabei bedeuten

$$\gamma \;:=\; \frac{1}{\|A\|}\max_{i,j,k}|a_{ij}^{(k)}|, \tag{2.42}$$

wo die $a_{ij}^{(k)}$ die während dem Eliminationsvorgang auftretenden Elemente

$$\begin{pmatrix} a_{11} & a_{12} & \cdots & a_{1n} \\ a_{21} & a_{22} & \cdots & a_{2n} \\ \vdots & & & \vdots \\ a_{n1} & a_{n2} & \cdots & a_{nn} \end{pmatrix} \quad\equiv\quad \begin{pmatrix} a_{11}^{(0)} & a_{12}^{(0)} & \cdots & a_{1n}^{(0)} \\ a_{21}^{(0)} & a_{21}^{(0)} & \cdots & a_{2n}^{(0)} \\ \vdots & & \vdots & \\ a_{n1}^{(0)} & a_{n2}^{(0)} & \cdots & a_{nn}^{(0)} \end{pmatrix} \to \begin{matrix}\text{erster El.}\\ \text{Schritt}\end{matrix} \to$$

$$\begin{pmatrix} a_{11}^{(0)} & a_{12}^{(0)} & \cdots & a_{12}^{(0)} \\ 0 & a_{22}^{(1)} & \cdots & a_{2n}^{(1)} \\ \vdots & \vdots & & \vdots \\ 0 & a_{n2}^{(1)} & \cdots & a_{nn}^{(1)} \end{pmatrix} \overset{\text{2.Schritt}}{\to} \begin{pmatrix} a_{11}^{(0)} & a_{12}^{(0)} & \cdots & & a_{1n}^{(0)} \\ 0 & a_{22}^{(1)} & \cdots & & a_{2n}^{(1)} \\ 0 & 0 & a_{33}^{(2)} & \cdots & a_{34}^{(2)} \\ \vdots & \vdots & \vdots & & \\ 0 & 0 & a_{n3}^{(2)} & \cdots & a_{nn}^{(2)} \end{pmatrix} \to \text{u.s.w}$$

vgl. auch (2.31) bis (2.35), wo die $a_{ij}^{(1)}$ und $a_{ij}^{(2)}$ mit a'_{ij} bzw. a''_{ij} bezeichnet werden.

Weiters ist *eps* aus (2.41) definiert als

$$eps \;:=\; \begin{cases} \text{Basis}^{-(\text{Mantissenlänge}-1)} & \ldots \text{ Fall einer Abschneidearithm.} \\ \frac{1}{2}\cdot\text{Basis}^{-(\text{Mantissenlänge}-1)} & \ldots \text{ Fall einer Rundearithmetik.} \end{cases} \tag{2.43}$$

(vgl. 1.19).

Rückwärtsanalyse: Die Idee, die hinter dem Satz v. Wilkinson steht, nämlich die Effekte von Rundungsfehlern als Datenfehlereffekte zu interpretieren ($\vec{\tilde{x}}$ soll das exakte Ergebnis eines gestörten Systems mit der gestörten Matrix $A + \Delta A$ sein), wird als Rückwärtsanalyse bezeichnet. Die Konditionsabschätzung (2.26) ermöglicht nun sofort die *Rundungsfehlerabschätzung*

$$\begin{aligned} \frac{\|\vec{\tilde{x}} - \vec{x}\|}{\|\vec{\tilde{x}}\|} \;&\leq\; \underbrace{\|A\|\,\|A^{-1}\|}_{\kappa(A)}\frac{\|\Delta A\|}{\|A\|} \;\leq \\ &\leq\; \kappa(A)\,1.01\,(n^3 + 3n^2)\,\gamma\, eps \end{aligned} \tag{2.44}$$

Das ist eine Schranke vom Typ

$$\text{Konditionszahl} \cdot \text{moderater Faktor} \cdot \text{Anzahl der Rechenoperationen} \cdot \\ \cdot \text{Maschinengenauigkeit } \text{`}eps\text{'}, \tag{2.45}$$

zumindest in den Fällen, wo während der Elimination keine sehr großen Elemente $a_{ij}^{(k)}$ auftreten. Die Anzahl der Rechenoperationen bei der Gaußelimination ist $\sim \frac{n^3}{3}$.

$\Rightarrow$ *Gaußelimination mit Spaltenpivotsuche ist ein* **numerisch stabiler** *Algorithmus.*

- **Numerische Stabilität** ist gegeben, wenn eine Rundungsfehlerschranke vom Typ (2.45) existiert. Dass in so einer Schranke die Maschinenarithmetik *eps* und die Anzahl der Rechenoperationen von einem Algorithmus auftritt, ist ganz natürlich. Dass weiters nur mehr ein moderater Faktor und die Konditionszahl des Problems aufscheint besagt eben, dass die Rundungsfehlersensitivität nicht schlechter liegt wie die Datenfehlersensitivität, d.h. dass sich die dauernden Störungen während dem Algorithmus, d.h. die Rechenfehler bei den einzelnen Rechenoperationen nicht schlimmer auswirken wie Datenstörungen; das ist offensichtlich das Beste, was man für einem Algorithmus, erhoffen darf.

(2.44) ist eine mathematische Aussage, die numerische Stabilität des Gaußalgorithms mit Spaltenpivotsuche sicherstellt. Sie gilt für alle denkbaren (regulären) linearen Gleichungssysteme. In Einzelfällen wird sie jedoch kaum herangezogen, um das Rundungsfehlerniveau in einem konkreten Fall zahlenmässig abzuschätzen.

Die Schranke (2.44) ist in den meisten konkreten Fällen zu pessimistisch: Wenn man ein System mit Gaußelimination löst, ergibt sich i.a. eine gewisse Kompensation der Rundungsfehler (wenn z.B. bei einem Rechenschritt aufgerundet wird und beim nächsten Rechenschritt wieder abgerundet, heben sich die entsprechenden Rundungsfehler teilweise weg). Die Schranke (2.44) hingegen, die durch eine reine Betragsabschätzung zustande kommt muss sich an den extrem unwahrscheinlichen Fall orientieren, (der aber in sehr sehr seltenen Fällen wirklich einmal eintreten könnte) dass es keine solchen Kompensationen gibt sondern dass alle einzelnen Rundungsfehler immer in dieselbe Richtung liegen und eine völlige Akkumulation dieser Fehler eintritt.

Daher: wenn in einem konkretem Fall das rundungsfehlerbehaftete $\tilde{x}$ vorliegt, kann man mit diesem $\tilde{x}$ *a-posteriori* zu realistischen Abschätzungen des Rundungsfehlerniveaus gelangen. Z.B. aufgrund von folgender Überlegung:

$$\begin{aligned} \tilde{x} - x &= A^{-1}A(\tilde{x} - x) \quad \Rightarrow \\ \frac{\|\tilde{x} - x\|}{\|\tilde{x}\|} &\leq \|A^{-1}\| \frac{\|A\tilde{x} - \overbrace{Ax}^{b}\|}{\|\tilde{x}\|} = \\ &= \underbrace{\|A\|\|A^{-1}\|}_{\kappa(A)} \underbrace{\frac{\|A\tilde{x} - b\|}{\|A\|\|\tilde{x}\|}}_{*)} \end{aligned} \tag{2.46}$$

$*$) ist ein berechenbarer Ausdruck wenn $\tilde{x}$ gegeben. Zähler $A\tilde{x} - b$ ... Residuum, muss in partieller doppelter Genauigkeit berechnet werden!

Ein weiterer wichtiger Begriff zur Theorie der linearen Gleichungssysteme:

•**Numerisch singulär** Wir betrachten jetzt wieder quadratische, $n \times n$-Gleichungssysteme (2.18). Sie sind bekanntlich eindeutig lösbar, wenn die Matrix A vollen Rang besitzt, d.h. $\text{Rang}(A) = n$ gilt. In diesem Fall spricht man von einem **regulären Gleichungssystem**.

Die erste Gleichung

$$\begin{aligned} a_{11}x_1 + a_{12}x_2 + a_{13}x_3 &= b_1 \\ a_{21}x_1 + a_{22}x_2 + a_{23}x_3 &= b_2 \\ a_{31}x_1 + a_{32}x_2 + a_{33}x_3 &= b_3 \end{aligned}$$

stellt eine Ebene im $\mathbb{R}^3$ dar, mit dem Normalvektor (a_{11}, a_{12}, a_{13}). Die zweite Gleichung stellt eine andere Ebene im $\mathbb{R}^3$ dar. Jene Punkte $(x_1, x_2, x_3)^\top \in \mathbb{R}^3$, die beide Gleichungen erfüllen, bilden den Schnitt dieser beiden Ebenen und liegen auf einer Geraden g. Die dritte Gleichung ist eine weitere Ebene. Jene Punkte $(x_1, x_2, x_3)^\top \in \mathbb{R}^3$, die alle drei Gleichungen erfüllen, die schon auf g und auf der durch die dritte Gleichung dargestellten Ebene liegen, sind der im regulären Fall eindeutige Durchstoßpunkt von g mit der dritten Ebene.

Entartungsfälle:

- Rangabfall um 1, d.h.
$$A = \begin{pmatrix} a_{11} & a_{12} & a_{13} \\ a_{21} & a_{22} & a_{23} \\ a_{31} & a_{32} & a_{33} \end{pmatrix}$$
hat nur 2 linear unabhängige Zeilen (Rang von $A = 3 - 1 = 2$): Der Normalvektor (a_{31}, a_{32}, a_{33}) der dritten Ebene ist eine Linearkombination der ersten beiden Normalvektoren: d.h. alle drei Normalvektoren liegen in einer Ebene. Zwei Möglichkeiten:
  a) Alle drei Ebenen schneiden sich in einer Geraden: Statt eindeutiger Lösung liegt eindimensionale Lösungsschar vor: (Dimension der Lösungschar: $n - \text{Rang}(A) = 3 - 1 = 2$;
- Rangabfall um 2, d.h. A hat den Rang $3 - 2 = 1$. Der zweite und dritte Normalvektor sind beides Vielfaches vom ersten Normalvektor. Alle drei Ebenen sind parallel. Es gibt zwei Möglichkeiten:
  a) Alle drei Ebenen fallen zusammen: Lösungsschar ist die Menge aller Punkte dieser Ebene. (Dimension der Lösungsschar: $n - \text{Rang}(A) = 3 - 2 = 1$).
  b) Die parallelen Ebenen fallen nicht zusammen: es gibt keine Lösungen des Systems.

In der reinen Mathematik ist der Begriff des Ranges einer Matrix ein scharfer Begriff. Die Zeilen einer Matrix sind entweder linear unabhängig und die Matrix hat Vollrang, oder sie sind linear abhängig, d.h. es gibt eine nichttriviale Linearkombination, der Zeilen, die verschwindet und die Matrix hat einen Rangabfall. In der Numerik bleibt diese Schärfe jedoch nicht bestehen. Wenn die Koeffizienten einer Matrix in einem Computer eingelesen weden, müssen sie in die Maschinenarithmetik hinein gerundet werden. D.h. die Koeffizienten werden – abhängig von dem verwendeten Computer – verändert und allfällige lineare Abhängigkeiten können zerstört werden bzw. aus linear unabhängigen Zeilen können durch den Rundungsprozess zufällig linear abhängige Zeilen entstehen.

Beispiel 2.5.2.

$$A = \begin{pmatrix} \frac{7}{16} & \frac{3}{5} & \frac{3}{2} \\ \frac{7}{32} & \frac{4}{5} & 2 \\ \frac{7}{8} & \frac{6}{5} & 3 \end{pmatrix} = \begin{pmatrix} 0.4375 & 0.6 & 1.5 \\ 0.21875 & 0.8 & 2 \\ 0.875 & 1.2 & 3 \end{pmatrix}$$

Es gibt die nichttriviale Linearkombination

$$2(\frac{7}{16}, \frac{3}{5}, \frac{3}{2})^\top + 0 \cdot (\frac{7}{72}, \frac{4}{5}, 2)^\top + (-1)(\frac{7}{8}, \frac{6}{5}, 3)^\top = (0, 0, 0)^\top$$

d.h. A hat nicht den vollen Rang. Nach Rundung von A in $\mathbb{M}(10, 3, \ldots)$ ergibt sich die reguläre Matrix

$$\tilde{A} = \begin{pmatrix} 0.438 & 0.6 & 1.5 \\ 0.219 & 0.8 & 2 \\ 0.875 & 1.2 & 3 \end{pmatrix}$$

Bei Rundung von A in $\mathbb{M}(10, 4, \ldots)$ hingegen erhält man

$$\tilde{A} = \begin{pmatrix} 0.4375 & 0.6 & 1.5 \\ 0.2188 & 0.8 & 2 \\ 0.875 & 1.2 & 3 \end{pmatrix}$$

und es gilt dieselbe nichttriviale Linearkombination wie bei A selbst, d.h. $\tilde{A}$ hat in diesem Fall auch keinen vollen Rang.

Man definiert daher:

- *Eine in eine Arithmetik gerundete Matrix $\tilde{A}$ heißt* **numerisch regulär***, wenn in der Menge aller Matrizen A, die nach Rundung in die entsprechende Arithmetik $\tilde{A}$ ergeben, keine singuläre Matrix ist.*

Nur im Fall einer numerisch regulären Matrix $\tilde{A}$ kann man sicher sein, dass die wahre Matrix A auch regulär ist und dass das wahre Gleichungssystem eine eindeutige Lösung besitzt. Nur in diesem Fall ist es sinnvoll, das Gleichungssystem am Computer zu lösen. Ist $\tilde{A}$ hingegen nicht numerisch regulär, könnte das ursprüngliche Gleichungssystem eine singuläre Gleichungsmatrix besitzen und daher nicht lösbar sein. Ein numerischer Lösungsversuch wäre daher in diesem Fall völlig sinnlos.

2.6 Lineares Ausgleichsproblem

Zunächst einige wichtige Begriffe, die für Ausgleichsprobleme bezüglich der $\|\cdot\|_2$-Norm wesentlich sind:

- **Skalarprodukt** von zwei Vektoren $\vec{x} \in \mathbb{R}^n$ und $\vec{y} \in \mathbb{R}^n$:

$$\vec{x} \cdot \vec{y} := \sum_{i=1}^{n} x_i y_i \tag{2.47}$$

oder im Geiste der Matrixmultiplikation:

$$\vec{x} \cdot \vec{y} = \vec{x}^\top \vec{y} = (x_1, \ldots, x_n) \begin{pmatrix} y_1 \\ \vdots \\ y_n \end{pmatrix} = \sum_{i=1}^{n} x_i y_1 \tag{2.48}$$

- **Euklidische Norm** (Länge) eines Vektors:

$$\|\vec{x}\|_2 = \sqrt{\vec{x} \cdot \vec{x}} = \sqrt{\sum_{i=1}^{n} x_i^2} \tag{2.49}$$

- **Orthogonalität:** zwei Vektoren $\vec{x} \in \mathbb{R}^n$, $\vec{y} \in \mathbb{R}^n$ sind *orthogonal* (Schreibweise $\vec{x} \perp \vec{y}$), falls gilt:

$$\vec{x} \cdot \vec{y} = 0 \tag{2.50}$$

Beispiel 2.6.1. (vgl. Abb. 2.6)

$$\vec{x} = \begin{pmatrix} r_x \cos\varphi \\ r_x \sin\varphi \end{pmatrix} \in \mathbb{R}^2, \quad \vec{y} = \begin{pmatrix} r_y(-\sin\varphi) \\ r_y \cos\varphi \end{pmatrix} \in \mathbb{R}^2$$

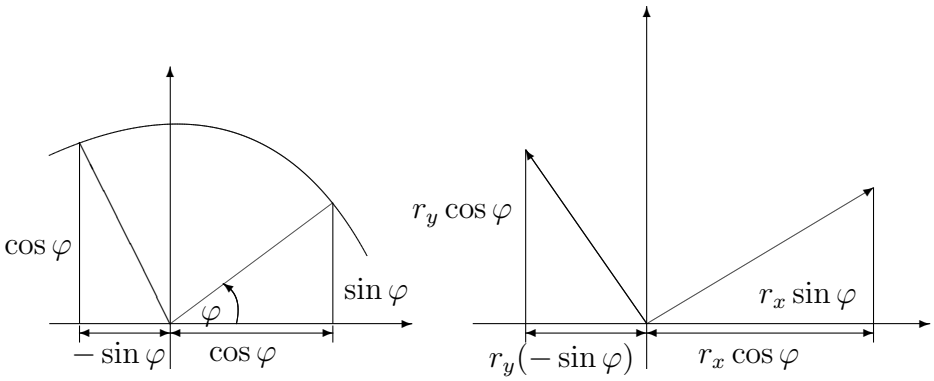


Abbildung 2.6: Beispiel zur Orthogonalität

$$\vec{x} \cdot \vec{y} = r_x r_y \cos\varphi(-\sin\varphi) + r_x r_y \sin\varphi\cos\varphi = 0$$

- **Orthogonales Komplement:** Sei S ein linearer Unterraum des $\mathbb{R}^n$, die Menge T aller Vektoren $\vec{t} \in \mathbb{R}^n$, die orthogonal zu sämtlichen Vektoren $\vec{s} \in S \subset \mathbb{R}^n$ sind, heißt das *orthogonale Komplement* zu S. Wenn S die Dimension k_S besitzt, so hat T die Dimension $k_T = n - k_S$, d.h. es gilt

$$n = k_S + k_T; \tag{2.51}$$

jedes $\vec{x} \in \mathbb{R}^n$ lässt sich in eindeutige Weise schreiben als

$$\vec{x} = \vec{s} + \vec{t}, \quad \vec{s} \in S,\ \vec{t} \in T \tag{2.52}$$

wobei $\vec{s}$ die **Projektion** von $\vec{x}$ auf S und $\vec{t}$ die Projektion von $\vec{x}$ auf T heißt! Man schreibt auch:

$$\mathbb{R}^n = S \oplus T \tag{2.53}$$

d.h. $\mathbb{R}^n$ ist die **direkte Summe** der orthogonalen Teilräume S und T.

Beweisskizze von (2.51), (2.52): Man dreht die Standardbasis (2.4) so, dass möglichst viele Basisvektoren in S liegen. Diese bilden dann eine Orthonormalbasis von S, die anderen gedrehten Einheitsvektoren bilden eine Orthonormalbasis von T. Man erhält dann sofort die Projektionen $\vec{s}$ und $\vec{t}$ eines beliebigen Vektors $\vec{x} \in \mathbb{R}^n$ (vgl. (2.52)). Zunächst stellen wir den Vektor $\vec{x}$ in der gedrehten Basis dar:

$$\vec{x} = \sum_{i=1}^{n} \bar{x}_i \vec{\bar{e}}_i$$

wobei $\vec{\bar{e}}_i$ die verdrehten Einheitsvektoren sein sollen; sei die Basis von S durch $\{\vec{\bar{e}}_1, \ldots, \vec{\bar{e}}_{k_S}\}$ gegeben und die Basis von T durch $\{\vec{\bar{e}}_{k_S+1}, \ldots, \vec{\bar{e}}_n\}$, so haben wir

$$\vec{s} = \sum_{i=1}^{k_S} \bar{x}_i \vec{\bar{e}}_i, \quad \vec{t} = \sum_{i=k_S+1}^{n} \bar{x}_i \vec{\bar{e}}_i.$$

□

Beispiel 2.6.2. $n = 2$: (vgl. Abb. 2.7)

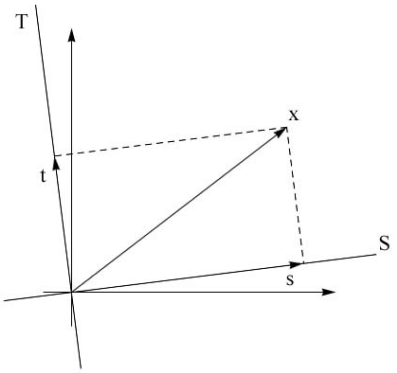


Abbildung 2.7: Bsp. $n = 2$

Beispiel 2.6.3. $n = 3$: (vgl. Abb. 2.8)

- **Lineares Ausgleichsproblem:** Das lineare Gleichungssystem $A\vec{x} = \vec{b}$, $A \in \mathbb{R}^{m \times n}$, $\vec{b} \in \mathbb{R}^m$, $\vec{x} \in \mathbb{R}^n$ sei widersprüchlich (d.h. $m > Rang(A)$, $\vec{b} \notin Bild(A)$). Dann wenigstens: jenes $\vec{x}$ (jene $\vec{x}$) suchen, das (die) die Gleichung so gut wie möglich löst (lösen), d.h. die Forderung lautet

$$\|A\vec{x} - \vec{b}\|_2 = \min \tag{2.54}$$

Frage nach der Existenz von Lösungen, Eindeutigkeit von Lösungen, Dimension von Lösungsscharen beim Ausgleichsproblem.

Zunächst beweisen wir folgendes

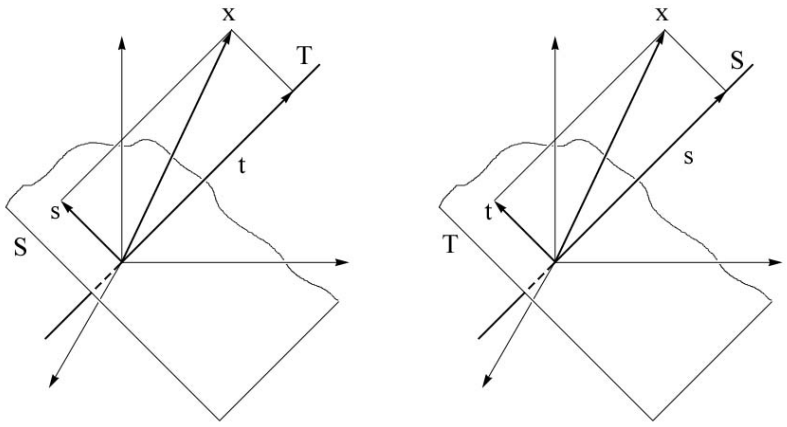


Abbildung 2.8: S ... Ebene durch den Ursprung S ... Gerade durch den Ursprung

Satz 2.6.4.

$$\mathbb{R}^m = Bild(A) \oplus Kern(A^\top) \tag{2.55}$$
$$\mathbb{R}^n = Bild(A^\top) \oplus Kern(A) \tag{2.56}$$

Beweis von (2.55): O sei das orthogonale Komplement von $Kern(A^\top)$. Wir zeigen zunächst: $Bild(A) \subset O$. Es gilt für

$$\vec{y} \in Bild(A) \subset \mathbb{R}^m, \quad \vec{z} \in Kern(A^\top) \subset \mathbb{R}^m \ ^{1)}$$
$$\vec{y} \cdot \vec{z} = \vec{y}^\top \vec{z} = (A\vec{x})^\top \vec{z} = \vec{x}^\top A^\top \vec{z} = \vec{0}$$

$\Rightarrow \quad \vec{y} \perp \vec{z}$ d.h. $Bild(A) \subset O$. Es gilt: $m = \dim(Kern(A^\top)) + Rang(A) = \dim(Kern(A^\top)) + \dim(Bild(A))$ d.h. durch die beiden linearen Teilräume $Kern(A^\top)$ und $Bild(A)$ wird der $\mathbb{R}^m$ aufgespannt und es gilt $Bild(A) = O \Rightarrow$ (2.55).

Beweis von (2.56): Man betrachtet statt der durch A repräsentierten Abbildung, die durch $A^\top$ repräsentierte Abbildung von $\mathbb{R}^m$ in den $\mathbb{R}^n$, dann geht (2.55) in (2.56) über. □

Satz 2.6.5. Das lineare Ausgleichsproblem (2.54) ist immer lösbar.

Beweis: (vgl. Abb. 2.9)

$$A\vec{x} - \vec{b} = \underbrace{A\vec{x} - \vec{b}_1}_{\in Bild(A)} - \underbrace{\vec{b}_2}_{\in Kern(A^\top)}$$

Wegen $A\vec{x} - \vec{b}_1 \perp \vec{b}_2$ lässt sich der Satz von Pythagoras anwenden:

$$\|A\vec{x} - \vec{b}\|_2^2 = \|A\vec{x} - \vec{b}_1\|_2^2 + \|\vec{b}_2\|_2^2$$
$$\uparrow$$
unabhängig von $\vec{x} \in \mathbb{R}^n$

[1)] $\vec{z} \in \mathbb{R}^m$ gilt, da $A^\top$ eine lineare Abbildung vom $\mathbb{R}^m$ in den $\mathbb{R}^n$ darstellt und die Elemente des Kerns stets im Urbildraum liegen. (Vgl. Def. vom Kern S. 33)

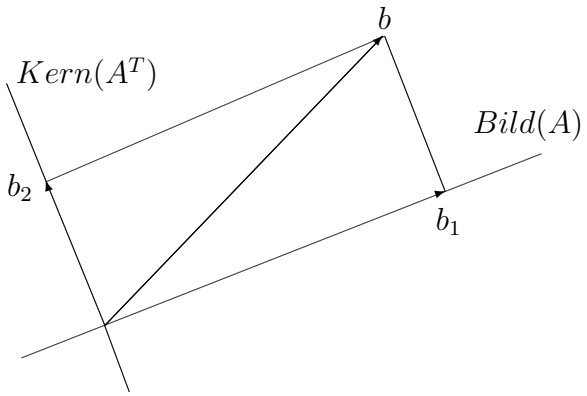


Abbildung 2.9: Zerlegung von $\vec{b}$ gemäß (2.55) und (2.52)

$\Rightarrow$ $\|A\vec{x} - \vec{b}\|_2$ ist genau dann minimal, wenn $\|A\vec{x} - \vec{b}_1\|_2$ minimal ist. $b_1 \in Bild(A) \Rightarrow A\vec{x} = \vec{b}_1$ ist lösbar; d.h. $\|A\vec{x} - \vec{b}_1\|_2$ kann zu Null gemacht werden $\Rightarrow$ alle $\vec{x}$ die das (stets lösbare) Gleichungssystem $A\vec{x} = \vec{b}_1$ lösen, sind auch Lösungen des Ausgleichsproblems. □

Aus dem Beweis folgt auch: Das lineare Ausgleichsproblem (2.54) ist genau dann *eindeutig* lösbar, wenn das lösbare Gleichungssystem $Ax = b_1$ eindeutig lösbar ist, d.h. wenn $Rang(A) = n$ ist. Im Fall $Rang(A) < n$ bestimmt wieder $n - Rang(A)$ die Dimension der Lösungsschar von $A\vec{x} = \vec{b}_1$, d.h. die Dimension der Lösungsschar des linearen Ausgleichsproblems.

- **Gaußsche Normalgleichungen:**

$$Residuum(\vec{x}) := A\vec{x} - \vec{b} = \vec{b}_1 - \vec{b} = -\vec{b}_2 \in Kern(A^\top)$$

$$\begin{aligned} \Rightarrow \quad & A^\top(A\vec{x} - \vec{b}) = \vec{0} \in \mathbb{R}^n \\ \Rightarrow \quad & A^\top A\vec{x} = A^\top\vec{b} \end{aligned} \tag{2.57}$$

  Man beachte: $A^\top A \in \mathbb{R}^{n\times n}$, $A^\top\vec{b} \in \mathbb{R}^n$;

  Im Falle $Rang(A) = n$ gilt $Rang(A^\top A) = n$, d.h. (2.57) ist ein reguläres lineares Gleichungssystem mit quadratischer Matrix; die eindeutig bestimmte Lösung von (2.57) liefert dann die eindeutig bestimmte Lösung des linearen Ausgleichsproblems.

- *Ausgleichsprobleme mit $Rang(A) < n$*: Hier kann man zu eindeutig lösbaren Ausgleichsproblem durch eine Zusatzbedingung gelangen:

$$\|A\vec{x} - \vec{b}\|_2 = \min \qquad \|\vec{x}\|_2 = \min \tag{2.58}$$

  Beschreibung der Abb. 2.10: $\vec{x}_0 \ldots$ Spezielle Lösung des Ausgleichsproblems, also spezielle Lösung von $A\vec{x} = \vec{b}_1$. Die allgemeine Lösung von $A\vec{x} = \vec{b}_1$ erhält man indem man zu $\vec{x}_0$ beliebige Elemente des Kerns von A addiert. $\vec{x}_{0,1}$ sei nun jene Lösung des Ausgleichsproblems, also von $A\vec{x} = \vec{b}_1$, die minimale euklidische Norm hat ($\|\vec{x}\|_2 = \min$ als Zusatzforderung). Ist unabhängig davon, welches Element $\vec{x}_0$ aus der Lösungsschar von $A\vec{x} = \vec{b}_1$ man ausgewählt hat.

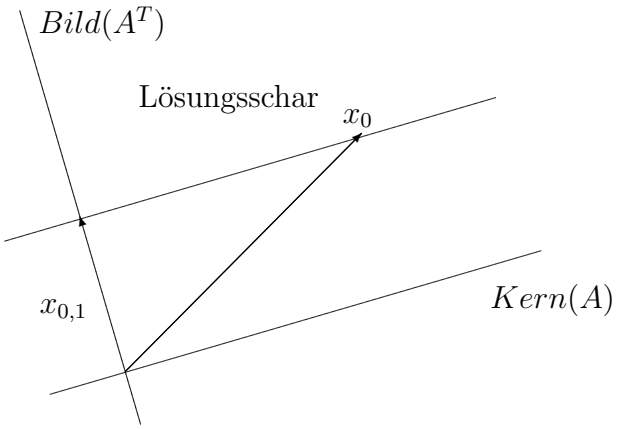


Abbildung 2.10: Schematische Darstellung von $\mathbb{R}^n = Bild(A^\top) \oplus Kern(A)$

Satz 2.6.6. $\|A\vec{x} - \vec{b}\|_2 = \min \quad \|\vec{x}\|_2 = \min$ ist eindeutig lösbar.

Beweis: $\vec{x}_0$ sei spezielle Lösung von $\|A\vec{x} - \vec{b}\|_2 = \min$, also von $A\vec{x} = \vec{b}_1$ (vgl. nun Abb. 2.11)

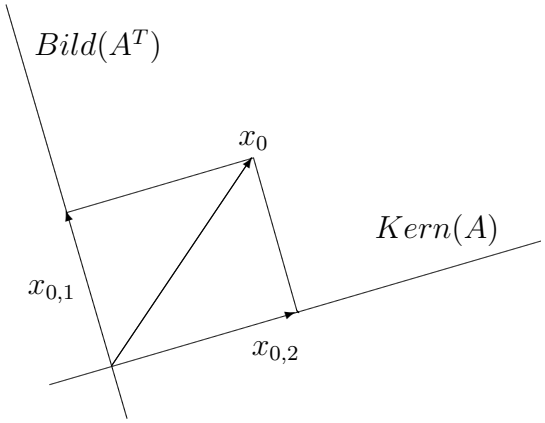


Abbildung 2.11: Zerlegung von $\vec{x}_0$

$\vec{x}_0 = \vec{x}_{0,1} + \vec{x}_{0,2}$ mit $\vec{x}_{0,1} \in Kern(A^\top)$ und $\vec{x}_{0,2} \in Kern(A)$ d.h. $\vec{x}_{0,1} \perp \vec{x}_{0,2}$. Es gilt

$$\vec{x}_0 + \vec{x}_k = \underbrace{\vec{x}_{0,1}}_{\in Bild(A^\top)} + \underbrace{\vec{x}_{0,2} + \vec{x}_k}_{\in Kern(A)}$$

wobei $\vec{x}_k$ ein bel. Element aus $Kern(A)$ bezeichnet. Weiters gilt offenbar $\vec{x}_{0,1} \perp (\vec{x}_{0,2} + \vec{x}_k)$ und mit dem Pythagoras

$$\|\vec{x}_0 + \vec{x}_k\|_2^2 = \|\vec{x}_{0,1}\|_2^2 + \|\vec{x}_{0,2} + \vec{x}_k\|_2^2;$$

$\Rightarrow \|\vec{x}_0 + \vec{x}_k\|_2$ genau minimal für $\vec{x}_k = -\vec{x}_{0,2}$ also für $\vec{x}_{0,1} = \vec{x}_0 - \vec{x}_{0,2}$ □

Es ist klar, dass in der Praxis beim Ausgleichsproblem meist gilt: $\|\vec{b}_1\|_2 \gg \|\vec{b}_2\|_2$ (vgl. Abb. 2.12)

Denn: Hätte man keine fehlerbehafteten Messgeräte, so würde man bei redundanten Messungen den Spezielfall $\vec{b} = \vec{b}_1 \in Bild(A), \vec{b}_2 = \vec{0}$ erhalten, d.h. ein überbestimmtes ($m > n$) aber nicht widersprüchliches ($\vec{b} \in Bild(A)$) Gleichungssystem (vgl. etwa Seite 53). Durch kleine Messfehler fällt man mit b nur wenig aus $Bild(A)$ heraus.

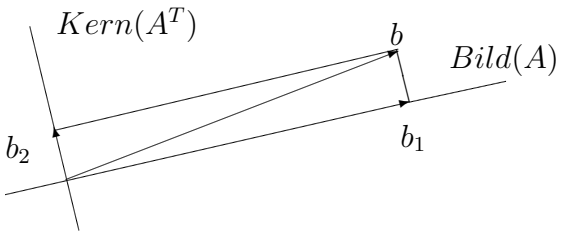


Abbildung 2.12: Verhältnis $\vec{b}_1$ zu $\vec{b}_2$

In vielen Fällen sind die Normalgleichungen starke Rundungsfehlersensitivität numerisch bedenklich. Das Ausgleichsproblem kann dann besser mit dem sogenannten *QR-Algorithmus* behandelt werden, einem Eliminationsalgorithmus, der auf orthogonalen Eliminationsmatrizen aufgebaut ist.

- *Pseudoinverse* (Verallgemeinerte Inverse): Das lineare Ausgleichsproblem

$$\begin{array}{lll} \|A\vec{x}-\vec{b}\|_2 = \min & & Rang(A)=n \\ \|A\vec{x}-\vec{b}\|_2 = \min & \|\vec{x}\|_2 = \min & Rang(A)<n \end{array}$$

ist stets eindeutig lösbar; d.h. man kann zu vorgegebener Matrix A dann die Abbildung betrachten die jeder rechten Seite $\vec{b}$ die eindeutige Lösung $\vec{x}$ des Ausgleichsproblems zuordnet: $\vec{b} \to \vec{x}$. Man kann zeigen: diese Abbildung ist linear wird also durch eine Matrix repräsentiert. Bezeichnung: A^+ d.h.

$$\vec{x} = A^+\vec{b}; \tag{2.59}$$

Sonderfälle:

- $A \in \mathbb{R}^{n\times n},\ Rang(A)=n:\ \|A\vec{x}-\vec{b}\|_2$ wird zu Null und liefert die Lösung dieses regulären Falls: $A^+ = A^{-1}$;
- $A \in \mathbb{R}^{m\times n},\ Rang(A)=n$: eindeutige Lösung des Ausgleichsproblems durch die Gaußschen Normalgleichungen. (2.57) $\Rightarrow A^+ = (A^\top A)^{-1}A^\top$

Die weiteren Fälle sind:

- $A\vec{x}=\vec{b}$ ist lösbar (d.h. $\|A\vec{x}-\vec{b}\|_2=0$) aber nicht eindeutig (wegen $Rang(A)<n$); dann stellt $\vec{x}=A^+\vec{b}$ jenes Element der Lösungsschar dar, für das gilt $\|\vec{x}\|_2=\min$
- $A\vec{x}=\vec{b}$ ist widersprüchlich mit $Rang(A)<n$ und $\vec{x}=A^+\vec{b}$ ist die eindeutig bestimmte Lösung von $\|A\vec{x}-\vec{b}\|_2=\min$ und $\|\vec{x}\|_2=\min$

In den letzten beiden Fällen kann man A^+ mit Hilfe der sogenanten **Singulärwertzerlegung** darstellen.

Beispiel Interpolation. Angenommen, man weiß, dass eine Funktion $f(x)$ die Gestalt

$$f(x) = c_0 + c_1x + c_2x^2$$

hat (Polynom zweiten Grades). Die Parameter c_0, c_1, c_2 (Koeffizienten) seien jedoch unbekannt. Aus drei Messungen von $f(x)$ kann man sie aus einem linearen Gleichungssystem berechnen: $f(x_0)$, $f(x_1)$ und $f(x_2)$ seien die Messwerte bezügl. x_0, x_1, und x_2. Es ergibt sich:

$$\begin{array}{lll} x_0 & \ldots & c_0 + c_1x_0 + c_2x_0^2 = f(x_0) \\ x_1 & \ldots & c_0 + c_1x_1 + c_2x_1^2 = f(x_1) \\ x_2 & \ldots & c_0 + c_1x_2 + c_2x_2^2 = f(x_2) \end{array}$$

d.h. ein lineares Gleichungssystem der Form:

$$\underbrace{\begin{pmatrix} 1 & x_0 & x_0^2 \\ 1 & x_1 & x_1^2 \\ 1 & x_2 & x_2^2 \end{pmatrix}}_{\text{Gleichungsmatrix (Vandermondematrix)}} \cdot \underbrace{\begin{pmatrix} c_0 \\ c_1 \\ c_2 \end{pmatrix}}_{\text{Unbekannten-Vektor}} = \underbrace{\begin{pmatrix} f(x_0) \\ f(x_1) \\ f(x_2) \end{pmatrix}}_{\text{Vektor d. rechten Seite}}$$

Konkretes Zahlenbeispiel:

$$\begin{array}{lllllllll} x_0 & = & 0 & \ldots & f(x_0) & = & f(0) & = & 2 \\ x_1 & = & \frac{1}{2} & \ldots & f(x_1) & = & f(\frac{1}{2}) & = & \frac{5}{2} \\ x_2 & = & 1 & \ldots & f(x_2) & = & f(1) & = & 0 \end{array}$$

liefert das Gleichungssystem:

$$\begin{array}{llllllll} c_0 & & & & = & 2 & & \\ c_0 & + & \frac{1}{2}c_1 & + \; \frac{1}{4}c_2 & = & \frac{5}{2} & \Rightarrow & c_0 = 2 \\ c_0 & + & c_1 & + \; c_2 & = & 0 & & \end{array}$$

$c_0 = 2$ in 2. und 3. Gleichung einsetzen:

$$\begin{array}{llllll} \frac{1}{2}c_1 & + & \frac{1}{4}c_2 & = & \frac{5}{2} - 2 & = \; \frac{1}{2} \\ c_1 & + & c_2 & = & 0 - 2 & = \; -2 \end{array}$$

erste Gleichung mit 2 multipliziert und von der 2. Gleichung abgezogen:

$$\begin{array}{llll} \frac{1}{2}c_1 & + \; \frac{1}{4}c_2 & = & \frac{1}{2} \\ & \frac{1}{2}c_2 & = & -3 \end{array}$$

$\Rightarrow c_2 = -6,\ c_1 = 4$ und somit die gesuchte Funktion

$$f(x) \;=\; 2 + 4x - 6x^2 \tag{2.60}$$

Man kann nun $f(x)$ für beliebige x-Werte auswerten; Interpolation! (vgl. Abb. 2.13)

Bemerkung: Für diese Art der Interpolation – ein Interpolationspolynom an die Interpolationsdaten $(x_0, f(x_0)),\ (x_1, f(x_1)),\ \ldots,\ (x_n, f(x_n))$ (bei unserem Beispiel $n = 2$) anzupassen und das Polynom an einer Stelle $x \neq x_0, x_1, \ldots, x_n$ auszuwerten – gibt es effizientere numerische Algorithmen (vgl. Kap. 4); der Weg, zuerst die Koeffizienten über ein lineares Gleichungssystem zu bestimmen, und dann das so erhaltene Interpolationspolynom an der Stelle x auszuwerten, ist zwar prinzipiell möglich, aber für wirkliche Berechnungen ein Umweg. Für theoretische Fragestellungen sind jedoch die hinter einem linearen Interpolationsprozess stehenden linearen Gleichungssysteme oft wichtig.

Redundante Daten bei der Interpolation: Bei obigem Zahlenbeispiel: zusätzliche Messung für $x_3 = \frac{1}{4}$:

$$f(x_3) \;=\; f(\frac{1}{4}) \;=\; \frac{21}{8}\ . \tag{2.61}$$

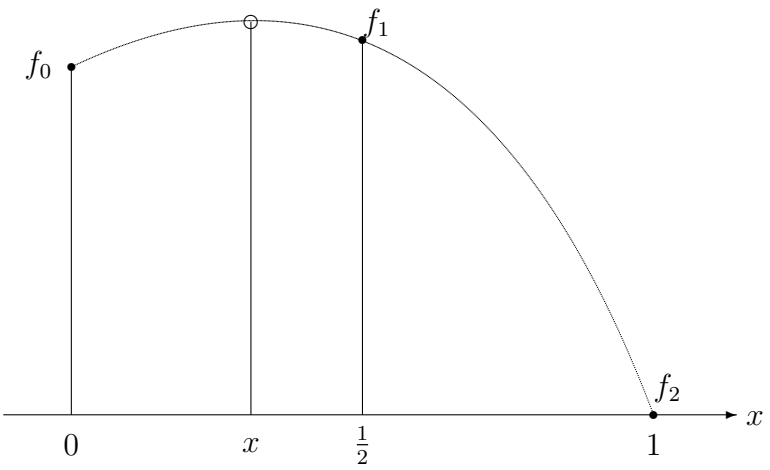


Abbildung 2.13: x ... "Zwischenstelle"; $f(x)$ kann berechnet werden.

Ergäbe das überbestimmte lineare Gleichungssystem

$$\begin{array}{lrcrcrcl} x_0 = 0: & c_0 & & & & & = & 2 \\ x_1 = \frac{1}{2}: & c_0 & + & \frac{1}{2}c_1 & + & \frac{1}{4}c_2 & = & \frac{5}{2} \\ x_2 = 1: & c_0 & + & c_1 & + & c_2 & = & 0 \\ x_3 = \frac{1}{4}: & c_0 & + & \frac{1}{4}c_1 & + & \frac{1}{16}c_2 & = & \frac{21}{8} \end{array}$$

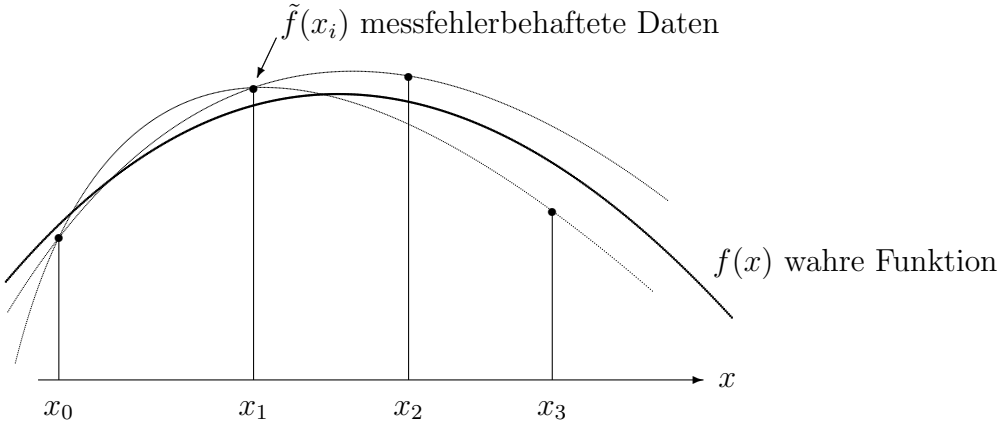
(4 Gleichungen für 3 Unbekannte) Matrixschreibweise:

$$\begin{pmatrix} 1 & 0 & 0 \\ 1 & \frac{1}{2} & \frac{1}{4} \\ 1 & 1 & 1 \\ 1 & \frac{1}{4} & \frac{1}{16} \end{pmatrix} \begin{pmatrix} c_0 \\ c_1 \\ c_2 \end{pmatrix} = \begin{pmatrix} 2 \\ \frac{5}{2} \\ 0 \\ \frac{21}{8} \end{pmatrix}$$

mit einer Rechtecksmatrix von 4 Zeilen und 3 Spalten, d.h. $A \in \mathbb{R}^{4\times 3}$; der Lösungsvektor $\vec{x} = (c_0, c_1, c_2)^\top$ ist aus dem $\mathbb{R}^3$, die rechte Seite $\vec{b} = (2, \frac{5}{2}, 0, \frac{21}{8})^\top$ aus dem $\mathbb{R}^4$. Man kann eine beliebige von diesen 4 Gleichungen weglassen und kann aus den verbleibenden 3 Gleichungen c_0, c_1, c_2 bestimmen. Egal, welche der Gleichungen man weglässt, stets ergeben die restlichen 3 Gleichungen dasselbe Ergebnis $c_0 = 2, c_1 = 4, c_2 = -6$. Das überbestimmte Gleichungssystem ist eben redundant. Trotzdem wird in der Praxis oft dieser Fall (Anzahl der Gleichungen > Anzahl Unbekannten) betrachtet. Und zwar muss man in der Praxis mit Messfehlern (beim Aufstellen des Datensatzes $(x_0, f(x_0)), (x_1, f(x_1)), \ldots$) rechnen. (Vgl. Abb. 2.14)

Also liegt aufgrund von Messfehlern, wenn man ein quadratisches Polynom aus mehr als 3 Datenpunkten $(x_0, f(x_0))$, $(x_1, f(x_1))$, $(x_2, f(x_2))$ festlegen möchte, kein redundantes, sondern ein widersprüchliches, überbestimmtes Gleichungssystem vor. Z.B. durch einen Datensatz von 4 messfehlerbehafteten Datenpunkten kann man i.a. kein quadratisches Polynom (nur drei Koeffizienten!) festlegen.

Aber man kann versuchen, die Information, die in allen 4 (messfehlerbehafteten) Messungen



$- - -$ das quadratische Polynom, das durch die messfehlerbehafteten Daten $(x_0, \tilde{f}(x_0))$, $(x_1, \tilde{f}(x_1))$, $(x_2, \tilde{f}(x_2))$ festgelegt wird

$- \cdot - \cdot -$ das quadratische Polynom, das durch die Daten $(x_0, \tilde{f}(x_0))$, $(x_1, \tilde{f}(x_1))$ und $(x_3, \tilde{f}(x_3))$ festgelegt ist.

Abbildung 2.14: Messfehlerbehaftete Daten

steckt, aus den Daten zu holen und hoffen, dass sich die Messfehler herausmitteln. "Bestimmung des *Ausgleichspolynoms*. (Vgl. Abb. 2.15)

$f(x) = c_0 + c_1 x + c_2 x^2$ soll optimal an Daten $(x_0, \tilde{f}(x_0)), \ \ldots, \ (x_3, \tilde{f}(x_3))$ angepasst werden, z.B. durch die Forderung die c_0, c_1, c_2 so festzulegen, dass

$$\sum_{i=0}^{3} (c_0 + c_1 x_i + c_2 x_i^2 - \tilde{f}(x_i))^2$$

minimal wird.

Etwas allgemeiner: Datensatz: $(x_0, \tilde{f}(x_0))$, $(x_1, \tilde{f}(x_1))$, $\ldots$, $(x_m, \tilde{f}(x_m))$ (vgl. Abb. 2.16)

quadratisches Polynom so festlegen (d.h. die Koeffizienten so bestimmen), dass

$$E(c_0, c_1, c_2) \ := \ \sum_{i=0}^{m} (c_0 + c_1 x_i + c_2 x_i^2 - \tilde{f}(x_i))^2 \tag{2.62}$$

minimal wird. Eine notwendige Bedingung für das Minimum ist:

$$\frac{\partial E(c_0, c_1, c_2)}{\partial c_0} = 0, \quad \frac{\partial E(c_0, c_1, c_2)}{\partial c_1} = 0, \quad \frac{\partial E(c_0, c_1, c_2)}{\partial c_2} = 0 \ . \tag{2.63}$$

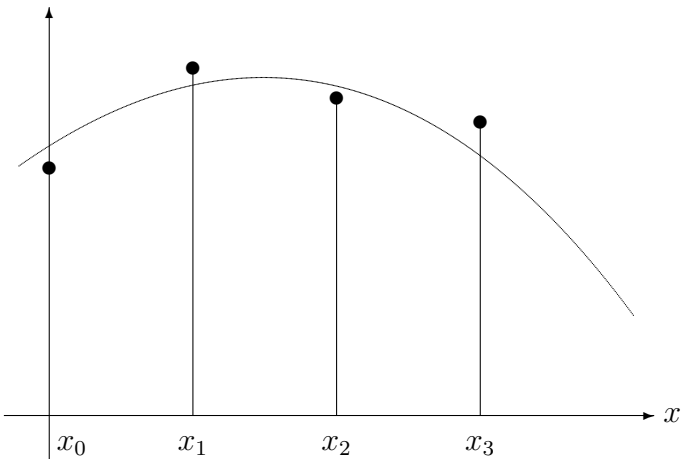


Abbildung 2.15: Ausgleichspolynom

$$\begin{aligned}
\frac{\partial E(c_0,c_1,c_2)}{\partial c_0} &= \sum_{i=0}^{m} 2(c_0 + c_1 x_i + c_2 x_i^2 - \tilde{f}(x_i)) \\
\frac{\partial E(c_0,c_1,c_2)}{\partial c_1} &= \sum_{i=0}^{m} 2(c_0 + c_1 x_i + c_2 x_i^2 - \tilde{f}(x_i)) \cdot x_i \\
\frac{\partial E(c_0,c_1,c_2)}{\partial c_2} &= \sum_{i=0}^{m} 2(c_0 + c_1 x_i + c_2 x_i^2 - \tilde{f}(x_i)) \cdot x_i^2 \ ,
\end{aligned} \tag{2.64}$$

d.h. das Minimum berechnet sich aus dem linearen Gleichungssystem

$$\begin{aligned}
\not{2} \sum_{i=0}^{m} (c_0 + c_1 x_i + c_2 x_i^2 - \tilde{f}(x_i)) &= 0 \\
\not{2} \sum_{i=0}^{m} (c_0 + c_1 x_i + c_2 x_i^2 - \tilde{f}(x_i)) \cdot x_i &= 0 \\
\not{2} \sum_{i=0}^{m} (c_0 + c_1 x_i + c_2 x_i^2 - \tilde{f}(x_i)) \cdot x_i^2 &= 0 \ ,
\end{aligned} \tag{2.65}$$

das man offensichtlich auch so schreiben kann:

$$\begin{aligned}
(m+1) \cdot c_0 + \Big(\sum_{i=0}^{m} x_i\Big) \cdot c_1 + \Big(\sum_{i=0}^{m} x_i^2\Big) \cdot c_2 &= \sum_{i=0}^{m} \tilde{f}(x_i) \\
\Big(\sum_{i=0}^{m} x_i\Big) \cdot c_0 + \Big(\sum_{i=0}^{m} x_i^2\Big) \cdot c_1 + \Big(\sum_{i=0}^{m} x_i^3\Big) \cdot c_2 &= \sum_{i=0}^{m} x_i \tilde{f}(x_i) \\
\Big(\sum_{i=0}^{m} x_i^2\Big) \cdot c_0 + \Big(\sum_{i=0}^{m} x_i^3\Big) \cdot c_1 + \Big(\sum_{i=0}^{m} x_i^4\Big) \cdot c_2 &= \sum_{i=0}^{m} x_i^2 \tilde{f}(x_i)
\end{aligned} \tag{2.66}$$

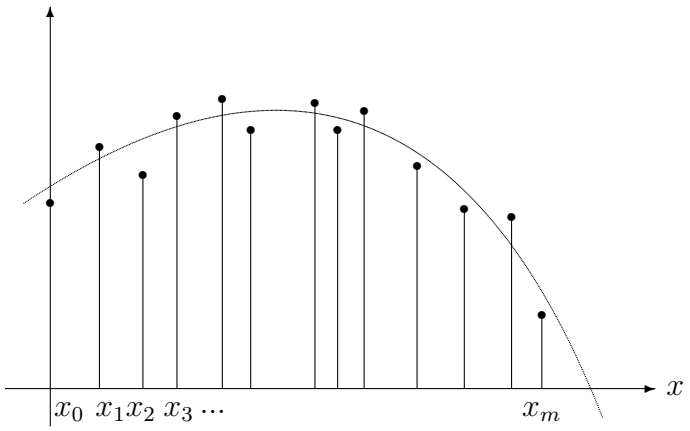


Abbildung 2.16: Ausgleichspolynom zu allgemeinem Datensatz

Also: 3 lineare Gleichungen für die 3 Unbekannten c_0, c_1, c_2. Die Gleichungsmatrix ist

$$\begin{pmatrix} m+1 & \sum x_i & \sum x_i^2 \\ \sum x_i & \sum x_i^2 & \sum x_i^3 \\ \sum x_i^2 & \sum x_i^3 & \sum x_i^4 \end{pmatrix} \tag{2.67}$$

und der Vektor der rechten Seite ist

$$\left(\sum \tilde{f}(x_i), \sum x_i \tilde{f}(x_i), \sum x_i^2 \tilde{f}(x_i)\right)^\top .$$

Ein konkretes Zahlenbeispiel dazu: $f(x) = 2 + 4x - 6x^2 \ldots$ und f soll jetzt nicht wie bei obigen Beispiel durch den exakten, unverfälschten Datensatz

$$(x_0, f(x_0)) = (0, 2), \quad (x_1, f(x_1)) = (\frac{1}{2}, \frac{5}{2}), \quad (x_2, f(x_2)) = (1, 0)$$

festgelegt sein, sondern durch den messfehlerbehafteten Datensatz mit 5 Messpunkten:

$$\begin{array}{lcllcl} (x_0, \tilde{f}(x_0)) & = & (0, 2.002)\ , & (x_1, \tilde{f}(x_1)) & = & (0.25, 2.624) \\ (x_2, \tilde{f}(x_2)) & = & (0.5, 2.498)\ , & (x_3, \tilde{f}(x_3)) & = & (0.75, 1.626) \\ (x_4, \tilde{f}(x_4)) & = & (1, 0.001)\ . & & & \end{array} \tag{2.68}$$

Würde man aus diesen 5 Messdaten die 3 Messdaten zu den Stellen $x_0 = 0$, $x_2 = 0.5$, $x_4 = 1$ auswählen und aus den verfälschten Größen $\tilde{f}(x_0)$, $\tilde{f}(x_2)$ und $\tilde{f}(x_4)$ ganz analog wie auf Seite 51 die Größen $\tilde{c}_0$, $\tilde{c}_1$, $\tilde{c}_2$ berechnen, ergäbe sich:

$$\begin{array}{lrcrcrcl} x_0 = 0: & \tilde{c}_0 & & & & & = & 2.002 \\ x_2 = 0.5: & \tilde{c}_0 & + & 0.5\tilde{c}_1 & + & 0.25\tilde{c}_2 & = & 2.498 \\ x_4 = 1: & \tilde{c}_0 & + & \tilde{c}_1 & + & \tilde{c}_2 & = & 0.001 \end{array} \tag{2.69}$$

mit der Lösung $\tilde{c}_0 = 2.002, \tilde{c}_1 = 3.985, \tilde{c}_2 = -5.986$; wertet man das verfälschte Polynom $\tilde{c}_0 + \tilde{c}_1 x + \tilde{c}_2 x^2$ für $x = 2$ aus, so ergibt sich

$$\tilde{f}(2) = -13.972 \quad \text{statt des wahren Wertes} \quad f(2) = -14,$$

also ein Fehler

$$\tilde{f}(2) - f(2) \;=\; 0.028 \tag{2.70}$$

Nun Einbeziehung aller Daten ergibt für unseren Datensatz folgendes Gleichungssystem:

$$\begin{array}{rcrcrcl} 5c_0 & + & 2.5c_1 & + & 1.875c_2 & = & 8.751 \\ 2.5c_0 & + & 1.875c_1 & + & 1.5625c_2 & = & 3.1255 \\ 1.875c_0 & + & 1.5625c_1 & + & 1.3828125c_2 & = & 1.704125 \end{array}$$

mit der Lösung

$$\begin{array}{rcl} c_0 &=& 2.001628568 \\ c_1 &=& 3.988571429 \\ c_2 &=& -5.98857142. \end{array} \tag{2.71}$$

Ein Vergleich mit der exakten Lösung von (2.69) zeigt eine (allerdings nur geringe) Verbesserung (etwas näher zu der wahren Lösung $c_0 = 2$, $c_1 = 4$, $c_2 = -6$). Berechnet man $\tilde{f}(2)$ mit den Koeffizienten aus (2.71) ergibt sich $\tilde{f}(2) = -13.97551425$ mit dem Fehler

$$\tilde{f}(2) - f(2) \;=\; 0.02448575 \tag{2.72}$$

Um hier eine größere Genauigkeitssteigerung zu erreichen, hätte man noch deutlich mehr Messpunkte einbeziehen müssen.

Bemerkung: Die eben beschriebene Vorgangsweise wurde zum ersten Mal von Carl Friedrich Gauß im Jahre 1801 angewendet. Von einem italienischen Astronomen ist im Jahr 1801 ein Planetoid entdeckt worden und seine Position an vielen Tagen gemessen worden (natürlich immer mit den üblichen Messfehlern). Als er wegen zu großer Sonnennähe nicht mehr beobachtet werden konnte, gelang es nachher nicht mehr, ihn wieder zu finden. Da die Planetoidenbahn durch die Messungen festgelegt ist, hat man aus der großen Zahl von Einzelbeobachtungen auf verschiedene Weise drei Beobachtungsdaten herausgegriffen, anschließend jeweils die entsprechende Position berechnet und dann mit dem Fernrohr die Umgebung der so berechneten Position abgesucht; aber aufgrund der Messfehler der einzelnen Beobachtungen war diese Vorgangsweise erfolglos. Erst als Gauß entsprechend der oben beschriebenen Vorgangsweise die in den einzelnen Messungen enthaltene Gesamtinformation ausnützte, gelang es, den Planetoiden wieder zu entdecken.

Das Gleichungssystem (2.66) kann man sich auch folgendermaßen entstanden denken:

Zuerst das überbestimmte, widersprüchliche Gleichungssystem betrachten:

$$\begin{array}{rcrcl} x_0 & \ldots & c_0 + c_1 x_0 + c_2 x_0^2 & = & \tilde{f}(x_0) \\ x_1 & \ldots & c_0 + c_1 x_1 + c_2 x_1^2 & = & \tilde{f}(x_1) \\ x_2 & \ldots & c_0 + c_1 x_2 + c_2 x_2^2 & = & \tilde{f}(x_2) \\ \vdots & & & \vdots & \\ x_m & \ldots & c_0 + c_1 x_m + c_2 x_m^2 & = & \tilde{f}(x_m); \end{array} \tag{2.73}$$

die Matrix A des Systems ist gegeben gemäß

$$A \;=\; \begin{pmatrix} 1 & x_0 & x_0^2 \\ 1 & x_1 & x_1^2 \\ 1 & x_2 & x_2^2 \\ \vdots & \vdots & \vdots \\ 1 & x_m & x_m^2 \end{pmatrix}. \tag{2.74}$$

Betrachtet auch die transponierte Matrix

$$A^\top = \begin{pmatrix} 1 & 1 & 1 & \cdots & 1 \\ x_0 & x_1 & x_2 & \cdots & x_m \\ x_0^2 & x_1^2 & x_2^2 & \cdots & x_m^2 \end{pmatrix}, \tag{2.75}$$

so entsteht (2.66) offenbar dadurch, dass (2.73) mit $A^\top$ von links multipliziert wird, d.h. aus (2.23) erhält man sofort die Gaußschen Normalgleichungen (2.66) in der Form

$$A^\top A \cdot \begin{pmatrix} c_0 \\ c_1 \\ c_2 \end{pmatrix} = A^\top \begin{pmatrix} \tilde{f}(x_0) \\ \tilde{f}(x_1) \\ \tilde{f}(x_2) \\ \vdots \\ \tilde{f}(x_m) \end{pmatrix}.$$

Kapitel 3

Nichtlineare Gleichungssysteme

3.1 Einleitung und Problemstellung

In diesem Kapitel wird das Lösen von nichtlinearen Gleichungen oder das sogenannte **Nullstellenproblem** behandelt. Sei $\vec{F} : \mathbb{R}^n \to \mathbb{R}^n$ eine i.a. nichtlineare Funktion. Unter dem Nullstellenproblem versteht man die Suche nach allen Lösungen $\vec{x} \in \mathbb{R}^n$ für die $\vec{F}(\vec{x}) = \vec{0}$ gilt. Diese Lösungen $\vec{x} \in \mathbb{R}^n$ werden als *Nullstellen* bezeichnet.

Existenz und *Eindeutigkeit* dieser Nullstellen $\vec{x}$: Nichtlineare Gleichungsysteme können oft nur in einem bestimmten Gebiet, also *lokal* eindeutig gelöst werden.

Einen *Spezialfall* stellt die sogenannte *affine Funktion* $F : \mathbb{R} \to \mathbb{R}$, $F(x) = ax + b,\ a, b \in \mathbb{R}$ dar. Eine eindeutige Lösung existiert für $a \neq 0$. Für $a = 0$ und $b \neq 0$ existiert keine Nullstelle und für $a = b = 0$, also $F(x) \equiv 0$, ist die Lösungsschar ganz $\mathbb{R}$.
Für $\vec{F} : \mathbb{R}^n \to \mathbb{R}^n$ sind verschiedene Glattheitsforderungen ($\vec{F}$ stetig, $\vec{F}$ stetig differenzierbar, etc.) denkbar, die die Existenz und die Anzahl von Lösungen beeinflussen, siehe Abb. 3.1 bis 3.6.

Definition 3.1.1. Eine Nullstelle $\vec{x}$ heißt **isoliert**, wenn gilt

$$\vec{F}(\vec{x}) = \vec{0} \quad \text{und} \quad \det(\vec{F}'(\vec{x})) \neq 0$$

Bemerkungen.

1 Siehe Abbildung 3.7.

2. Im Spezialfall der skalaren Funktion $F(x) = ax+b$ entspricht wegen $F'(x) = a$ die obige Definition dem regulären Fall $a \neq 0$.

3. In Abbildung 3.8 liegt eine Nullstelle vor, die nicht isoliert ist. Eine geringfügige Störung von F, verändert die Situation, siehe Abbildung 3.9. Dieses Problem ist schlecht konditioniert.

4. Unter der Voraussetzung einer isolierten Nullstelle lässt sich lokale Eindeutigkeit zeigen.

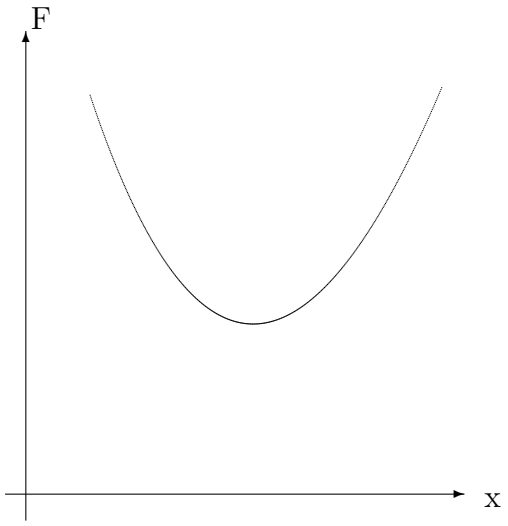


Abbildung 3.1: Beispiel für keine Nullstelle

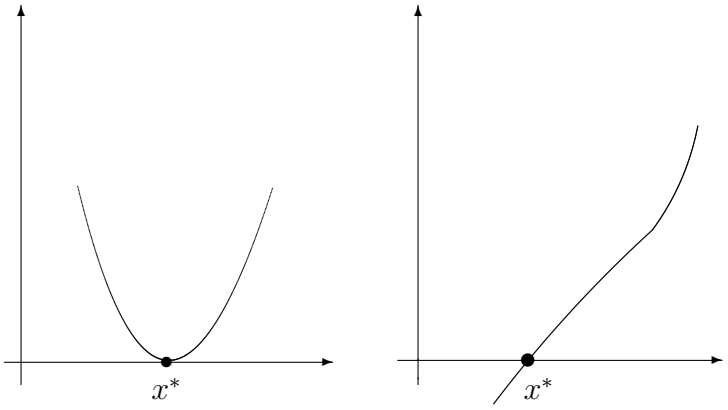


Abbildung 3.2: Beispiele für genau eine Nullstelle

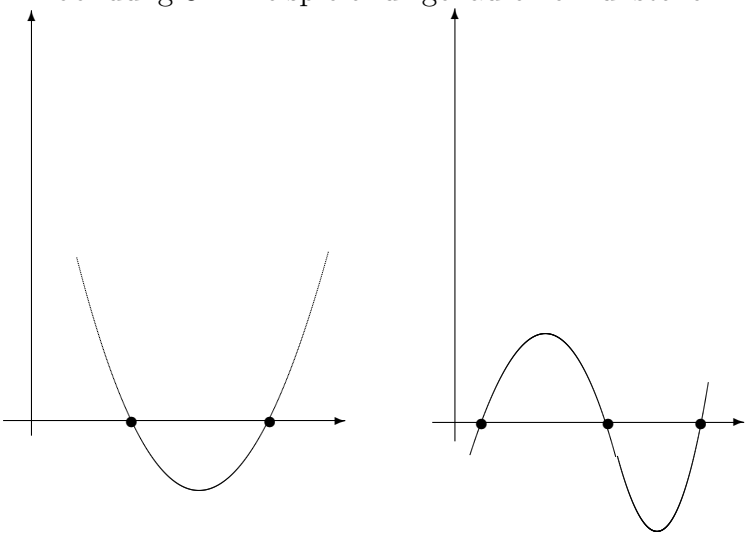


Abbildung 3.3: Beispiele für endlich viele Nullstellen

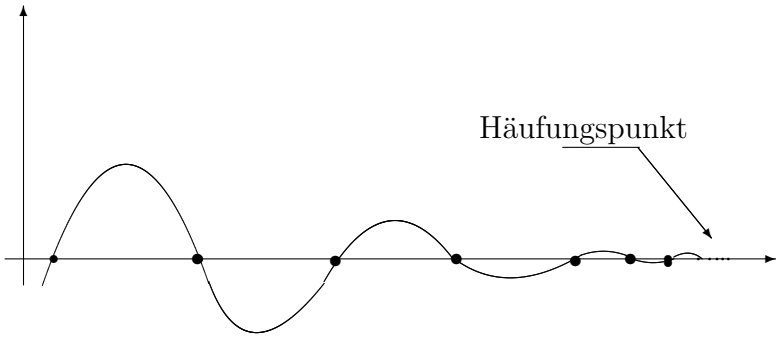


Abbildung 3.4: Beispiel für unendlich viele Nullstellen mit Häufungspunkt

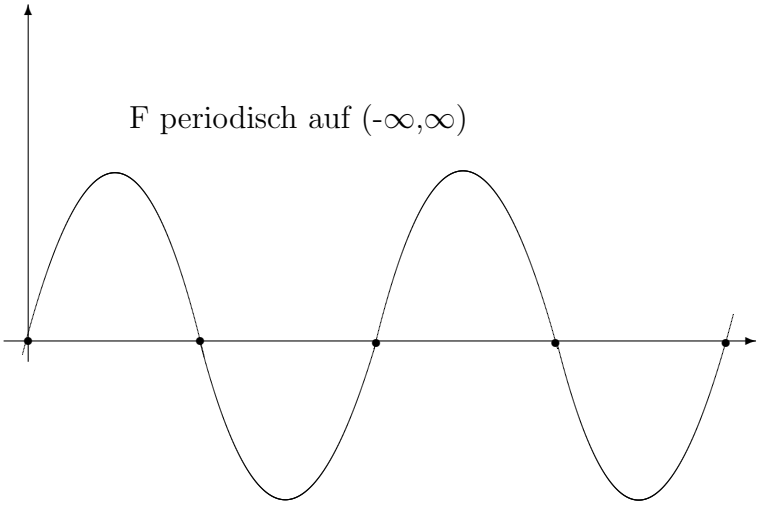


Abbildung 3.5: Beispiel für unendlich viele Nullstellen ohne Häufungspunkt

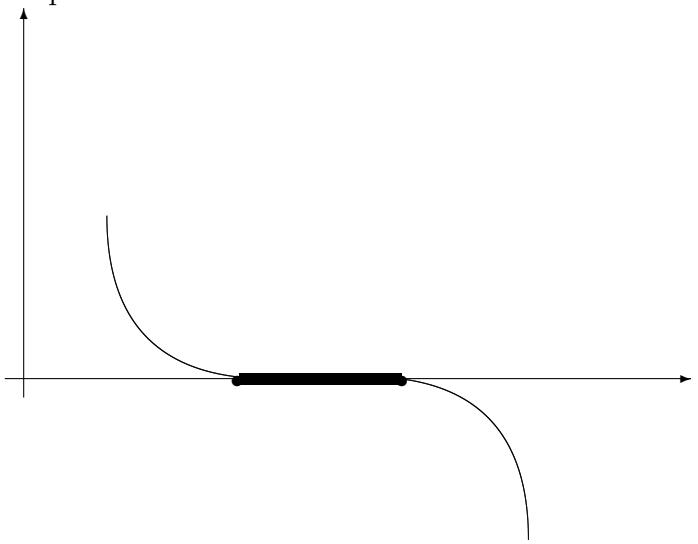


Abbildung 3.6: Beispiel für ein Kontinuum von Nullstellen

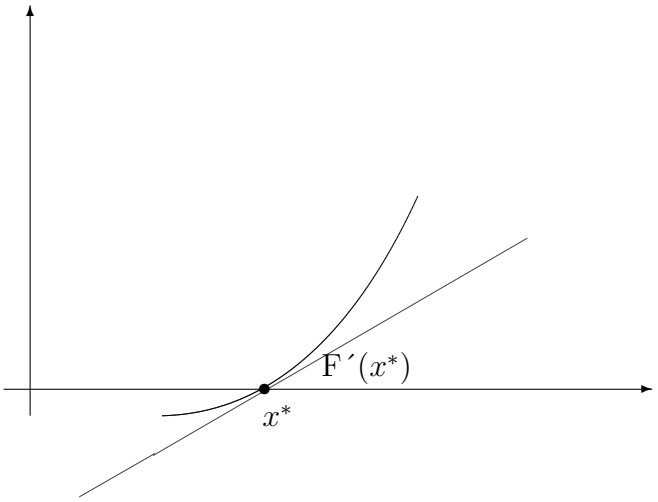


Abbildung 3.7: Beispiel für eine isolierte Nullstelle

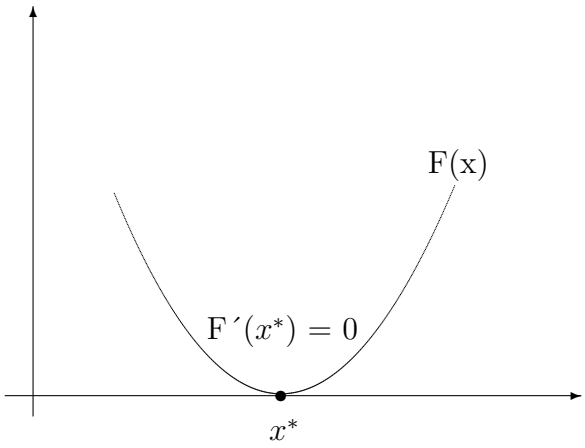


Abbildung 3.8: Beispiel für eine mehrfache Nullstelle

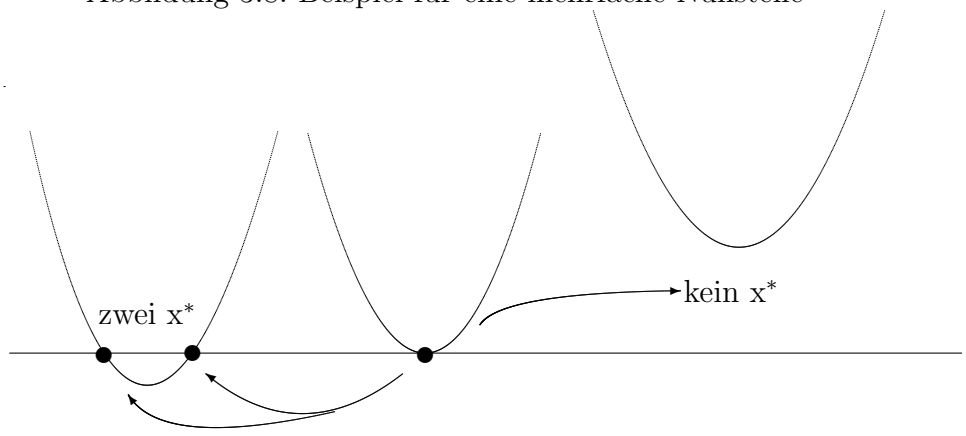


Abbildung 3.9: Einfluss von Störungen einer Funktion auf die Anzahl von Nullstellen

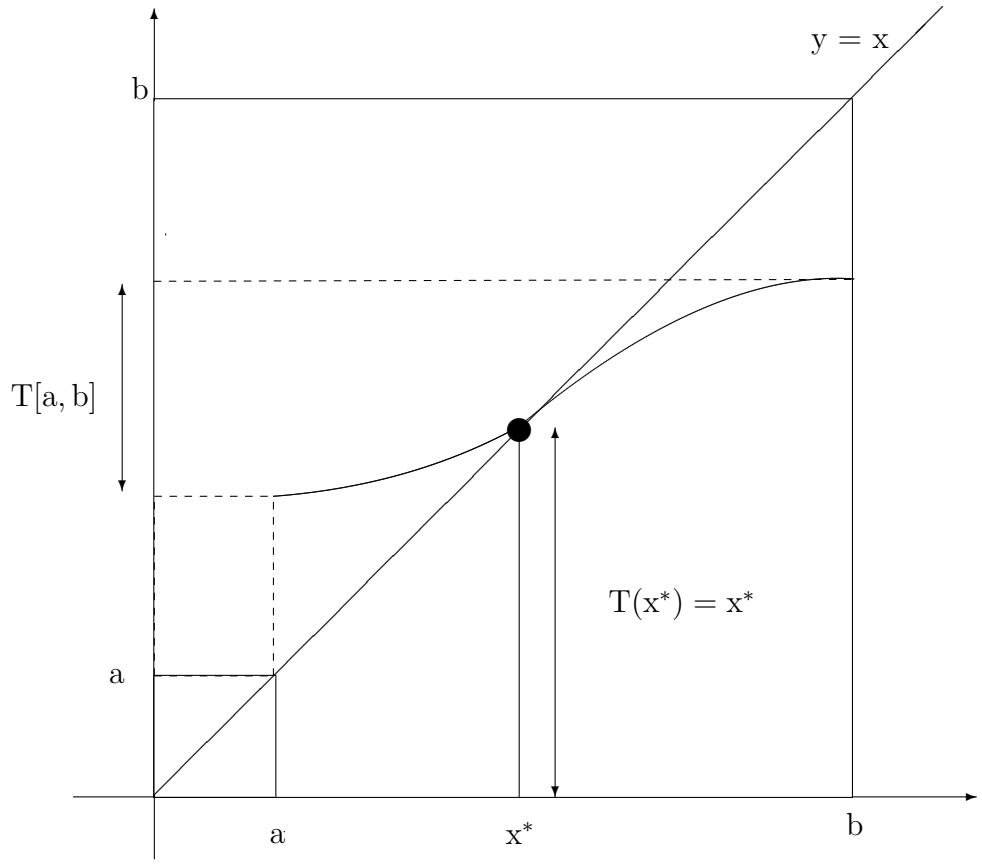


Abbildung 3.10: Beweisidee von Satz 3.1.2

Im Folgenden werden **äquivalente Formulierungen des Nullstellenproblems** angegeben.

1. Sei $R(\vec{x})$ eine von $\vec{x} \in \mathbb{R}^n$ abhängige, reguläre $n \times n$-Matrix, dann gilt:

$$\vec{F}(\vec{x}) = \vec{0} \qquad \Longleftrightarrow \qquad R(\vec{x})\vec{F}(\vec{x}) = \vec{0} \tag{3.1}$$

2. Es gilt weiters

$$\vec{F}(\vec{x}) = \vec{0} \qquad \Longleftrightarrow \qquad \overrightarrow{T}(\vec{x}) := \vec{x} - R(\vec{x})\vec{F}(\vec{x}) = \vec{x} \tag{3.2}$$

Diese äquivalente Formulierung wird als **Fixpunktproblem** bezeichnet. Die Menge der Nullstellen von $\vec{F} : \mathbb{R}^n \to \mathbb{R}^n$ ist gleich der Menge der **Fixpunkte** von $\overrightarrow{T} : \mathbb{R}^n \to \mathbb{R}^n$, also alle $\vec{x} \in \mathbb{R}^n$ mit $\overrightarrow{T}(\vec{x}) = \vec{x}$.

Für viele Fragestellungen ist es einfacher statt dem Nullstellenproblem $\vec{F}(\vec{x}) = \vec{0}$ das dazu äquivalente Fixpunktproblem $\overrightarrow{T}(\vec{x}) = \vec{x}$ zu betrachten.

Für die Diskussion von Existenz und Eindeutigkeit von Lösungen ist der folgende Satz wichtig.

Satz 3.1.2. Sei $I = [a, b] \subset \mathbb{R}$ und $T : \mathbb{R} \to \mathbb{R}$ stetig mit $T(I) \subset I$. Dann hat $T(x) = x$ mindestens eine Lösung in I, d.h. es existiert mindestens ein Fixpunkt von $T(x)$.

Beweis. Das Bild $T(I) = T([a, b])$ des Intervalls I unter der Abbildung T liegt ganz im Intervall I, d.h. es gilt $T(a) \geq a$, also der Funktionswert $T(a)$ liegt oberhalb (genauer: nicht unterhalb) der

Geraden $y = x$. Analog folgt aus $T(I) \subset I$ die Beziehung $T(b) \leq b$.
Da T stetig ist, muss $T(x)$ mindestens einmal die Gerade $y = x$ schneiden, siehe Abb.3.10. □

Die Verallgemeinerung dieses Satzes im $\mathbb{R}^n$.

Satz 3.1.3 (Satz von Brouwer)**.** Sei $D \subset \mathbb{R}^n$ beschränkt, abgeschlossen und konvex und sei $\overrightarrow{T} : D \subset \mathbb{R}^n \to \mathbb{R}^n$ stetig mit $\overrightarrow{T}(D) \subset D$. Dann hat $\overrightarrow{T}(\vec{x}) = \vec{x}$ mindestens eine Lösung in D.

Beweis. Siehe Literatur. □

Bemerkung.

Eine Verallgemeinerung des Fixpunktsatzes von Brower ist der **Fixpunktsatz von Schauder**. Er ist für unendlichdimensionale Banachräume formuliert. Im wesentlichen muss im Satz von Brouwer nur die Eigenschaft beschränkt durch *kompakt* ersetzt werden. Er wird in der Analysis oft angewendet, um die Existenz von Lösungen von Funktionalgleichungen (z.B. Differentialgleichungsprobleme, Intergralgleichungen, etc.) nachzuweisen.

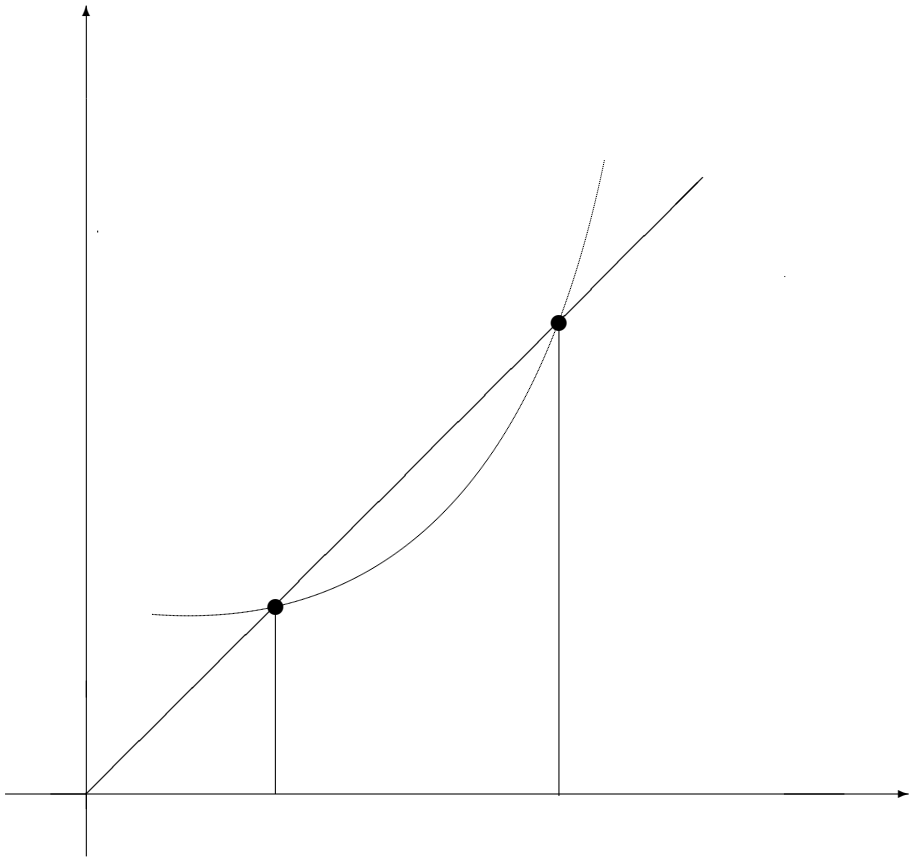


Abbildung 3.11: Zur Mehrdeutigkeit eines Fixpunktes

Will man nicht nur die Existenz von Fixpunkten nachweisen, sondern auch deren lokale Eindeutigkeit zeigen, ist der Begriff der **Kontraktion** von Bedeutung.
In Abbildung 3.11 erkennt man, dass für $T : \mathbb{R} \to \mathbb{R}$ Eindeutigkeit bedeutet, das der Graph von $T : \mathbb{R} \to \mathbb{R}$ flacher verläuft als die Gerade $y = x$, sonst könnte es mehrere Schnittpunkte mit der ersten Mediane, d.h. mehrere Fixpunkte geben, siehe Abbildung 3.11.

Definition 3.1.4. Die Abbildung $\overrightarrow{T} : D \subset \mathbb{R}^n \to \mathbb{R}^n$ heißt **kontrahierend** auf D, falls $\overrightarrow{T}$ Lipschitzstetig ist

$$\left\| \overrightarrow{T}(\vec{x}_1) - \overrightarrow{T}(\vec{x}_2) \right\| < L \left\| \vec{x}_1 - \vec{x}_2 \right\| \qquad \forall x_1, x_2 \in D \tag{3.3}$$

mit Lipschitzkonstante $L < 1$.

Satz 3.1.5. Sei $\overrightarrow{T} : D \subset \mathbb{R}^n \to \mathbb{R}^n$ kontrahierend und $\overrightarrow{T}(D) \subset D$ so besitzt $\overrightarrow{T}(\vec{x}) = \vec{x}$ genau einen Fixpunkt in D.

Beweis. Siehe Literatur. □

3.2 Berechnung von Nullstellen und Fixpunkten

Im Spezialfall von linearen Gleichungssystemen erfolgt die Berechnung von Lösungen bis auf Rundungsfehler exakt durch entsprechende Formelmanipulation, etwa durch Gaußelemination. Im Gegensatz dazu können die Nullstellen nichtlinearer Gleichungen meist nur näherungsweise durch sogenannte **Iterationsverfahren** berechnet werden.

Idee eines Iterationsverfahrens für das Auffinden von Lösungen von $\overrightarrow{T}(\vec{x}) = \vec{x}$. Wählen Sie einen Startwert $\vec{x}_0 \in D$ und berechnen Sie die folgenden Ausdrücke:

$$\begin{aligned} \overrightarrow{T}(\vec{x}_0) &=: \vec{x}_1 \\ \overrightarrow{T}(\vec{x}_1) &=: \vec{x}_2 \\ &\vdots \\ \overrightarrow{T}(\vec{x}_{k-1}) &=: \vec{x}_k \end{aligned} \tag{3.4}$$

Falls $\overrightarrow{T}$ eine kontrahierende Abbildung ist erfolgt Konvergenz, d.h.

$$\lim_{k\to\infty} \overrightarrow{T}(\vec{x}_k) = \vec{x}^* \qquad \vec{x}^* \text{ Fixpunkt von } \overrightarrow{T}. \tag{3.5}$$

Beispiel 3.2.1. Sei $F(x) = e^{x-2} - x$. Es sollen die Nullstellen von F bestimmt werden, also die Gleichung $F(x) = 0$ oder äquivalent das Fixpunktproblem $x = e^{x-2} =: T(x)$ gelöst werden. Als Startwert wird hier $x_0 = 0.25$ gewählt.

$$\begin{array}{lcll} x_0 & = & & 0.25 \\ x_1 & = & T(x_0) & = & 0.1737739435\ldots \\ x_2 & = & T(x_1) & = & 0.1610201033\ldots \\ x_3 & = & T(x_2) & = & 0.158979519\ldots \\ x_4 & = & T(x_3) & = & 0.158554386\ldots \\ x_5 & = & T(x_4) & = & 0.1586040298\ldots \\ x_6 & = & T(x_5) & = & 0.1585958764\ldots \\ x_7 & = & T(x_6) & = & 0.1585945833\ldots \\ x_8 & = & T(x_7) & = & 0.1585943782\ldots \\ x_9 & = & T(x_8) & = & 0.1585943457\ldots \end{array}$$

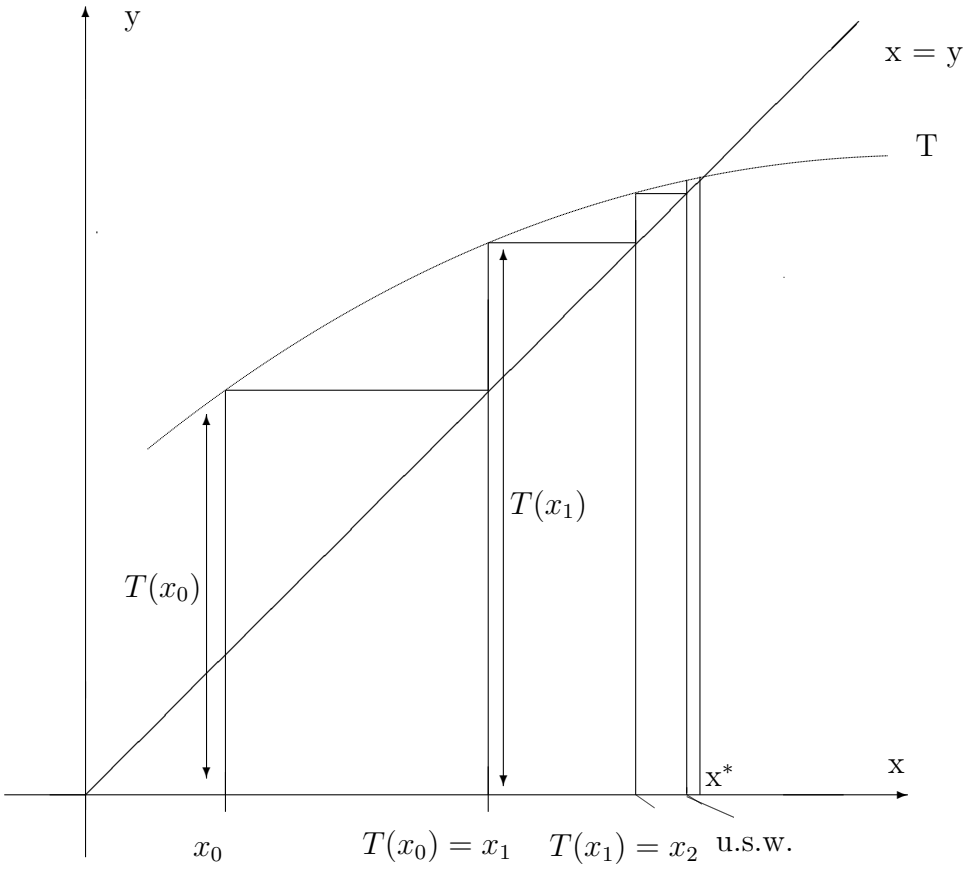


Abbildung 3.12: Visualisierung der Fixpunktiteration für $T : \mathbb{R} \to \mathbb{R}$

$$\begin{array}{rcccl} x_{10} & = & T(x_9) & = & 0.1585943405\ldots \\ x_{11} & = & T(x_{10}) & = & 0.1585943397\ldots \\ x_{12} & = & T(x_{11}) & = & 0.1585943396\ldots \\ x_{13} & = & T(x_{12}) & = & 0.1585943396\ldots \end{array}$$

Ab x_{12} ändert sich die Iteration bis zur zehnten Nachkommastelle nicht mehr. Die Iteration berechnet Werte die im Intervall $[x^*, x_0]$ liegen. Wegen der Monotonie der Exponentialfunktion ist die kleinsmögliche Lipschitzkonstante in diesem Intervall durch

$$L_{opt} = T'(x_0) = e^{0,25-2} = 0,1737739435\ldots$$

gegeben.

Beispiel 3.2.2. Ein weiteres Beispiel mit einer betragsmäßig kleineren Lipschitzkonstante ist durch

$$F(x) = \frac{1}{2} + e^{x-10} - x = 0 \qquad \Longleftrightarrow \qquad T(x) := \frac{1}{2} + e^{x-10} = x$$

gegeben.

$$\begin{array}{rcccl} x_0 & = & & & 0.5 \\ x_1 & = & T(x_0) & = & 0.5000748518\ldots \\ x_2 & = & T(x_1) & = & 0.5000748574\ldots \\ x_3 & = & T(x_2) & = & 0.5000748574\ldots \end{array}$$

Im Intervall $[x_0, x^*]$ ist L_{opt} durch $L_{opt} = T'(x^*) = 0,00007485743331\ldots$ gegeben. Die Konvergenz ist hier sehr viel schneller als in den anderen Beispielen.

Beispiel 3.2.3. Das Fixpunktproblem aus Beispiel 3.2.1 kann in ein dazu äquivalentes Fixpunktproblem der Form

$$T(x) := \ln x + 2 = x$$

umgeformt werden. Mit einem Startwert $x_0 = 0.1586$, welcher in der Nähe von x^* liegt, erhält man:

$$\begin{array}{rcrcl} x_0 & = & & & 0.1586 \\ x_1 & = & T(x_0) & = & 0.158600302\ldots \\ x_2 & = & T(x_1) & = & 0.1588199578\ldots \\ x_3 & = & T(x_2) & = & 0.1600121629\ldots \\ x_4 & = & T(x_3) & = & 0.1674945515\ldots \\ x_5 & = & T(x_4) & = & 0.2131955434\ldots \end{array}$$

Die Lipschitzkonstante für $T(x) = \ln x + 2$ ist auf dem Intervall $[x_0, 1]$ größer 1.

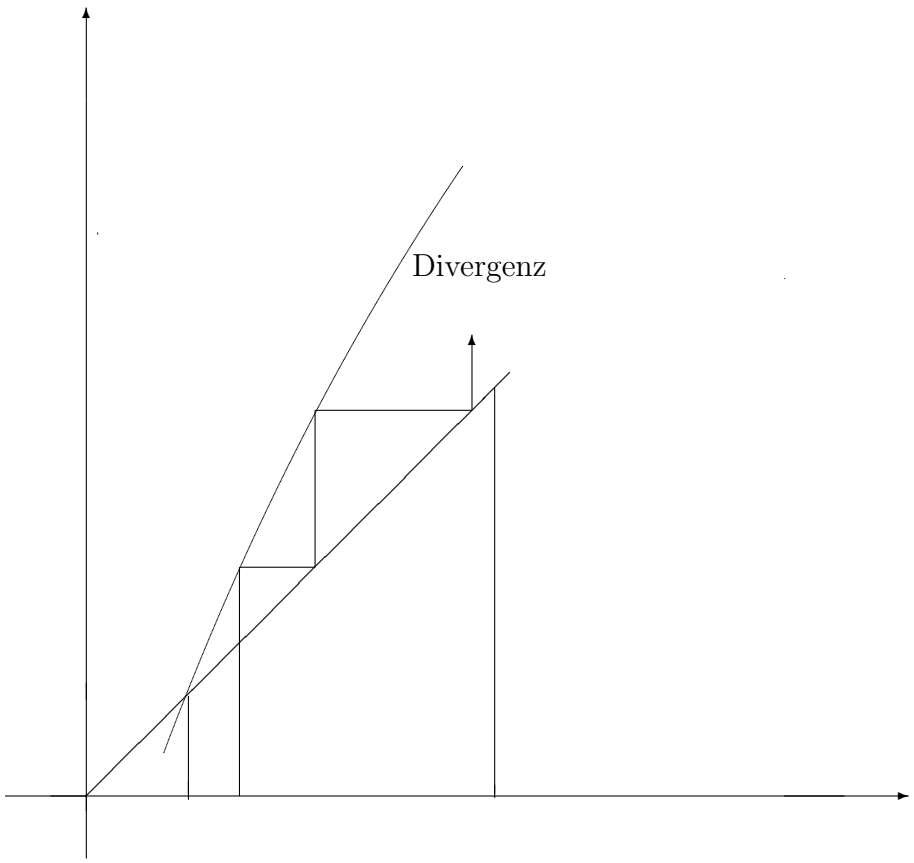


Abbildung 3.13: Visualisierung der Fixpunktiteration bei Divergenz

Satz 3.2.4 (Kontraktionssatz)**.** Sei $\overrightarrow{T} : D \subset \mathbb{R}^n \to \mathbb{R}^n$ und D abgeschlossen, beschränkt und konvex, es gelte $\overrightarrow{T}(D) \subset D$ und $\overrightarrow{T}$ sei kontrahierend auf D. Sei $\vec{x}_0 \in D$ ein beliebig gewählter Startwert für die Iteration $\vec{x}_k = \overrightarrow{T}(\vec{x}_{k-1})$ mit $k = 1, 2, 3, \ldots$.

Unter diesen Voraussetzungen liegt Konvergenz gegen $\vec{x}^*$ vor, wobei

$$\lim_{k\to\infty} \vec{x}_k = \vec{x}^* \tag{3.6}$$

gilt. Weiters gilt

$$\|\vec{x}_k - \vec{x}^*\| \leq L \|\vec{x}_{k-1} - \vec{x}^*\| \tag{3.7}$$

und

$$\|\vec{x}_k - \vec{x}^*\| \leq \frac{L}{1-L} \|\vec{x}_k - \vec{x}_{k-1}\| \leq \frac{L^k}{1-L} \|\vec{x}_1 - \vec{x}_0\|, \tag{3.8}$$

mit $L < 1$ einer Lipschitzkonstante von $\overrightarrow{T}$ auf D.

Beweis. 1. *Eindeutigkeit.* Angenommen es gäbe zwei Fixpunkte $\vec{x}_1^*, \vec{x}_2^*$ mit $\vec{x}_1^* \neq \vec{x}_2^*$. Die Rechnung

$$\|\vec{x}_1^* - \vec{x}_2^*\| = \left\| \overrightarrow{T}(\vec{x}_1^*) - \overrightarrow{T}(\vec{x}_2^*) \right\| \leq L \|\vec{x}_1^* - \vec{x}_2^*\| < \|\vec{x}_1^* - \vec{x}_2^*\|$$

zeigt einen Widerspruch. Die letzte Abschätzung wird durch $L < 1$ gerechtfertigt.

2. *Existenz.* Aus $\overrightarrow{T}(D) \subset D$ folgt, dass zu jedem $\vec{x}_0 \in D$ die Folge $\vec{x}_k$ existiert mit $\vec{x}_k = \overrightarrow{T}(\vec{x}_{k-1})$. Dies gilt, da aus $\vec{x}_0 \in D$ folgt $\vec{x}_1 = \overrightarrow{T}(\vec{x}_0) \in D$, daraus wieder $\vec{x}_2 = \overrightarrow{T}(\vec{x}_1) \in D$, usw. Es sind also sämtliche $\vec{x}_k \in D$. Aus der Rechnung

$$\|\vec{x}_{k+1} - \vec{x}_k\| = \left\| \overrightarrow{T}(\vec{x}_k) - \overrightarrow{T}(\vec{x}_{k-1}) \right\| \leq L \|\vec{x}_k - \vec{x}_{k-1}\| \leq L^2 \|\vec{x}_{k-1} - \vec{x}_{k-2}\| \leq \ldots$$

$$\ldots \leq L^k \|\vec{x}_1 - \vec{x}_0\|$$

folgt,

$$\begin{aligned} \|\vec{x}_{k+r} - \vec{x}_k\| &= \|(\vec{x}_{k+r} - \vec{x}_{k+r-1}) + (\vec{x}_{k+r-1} - \vec{x}_{k+r-2}) + \ldots + (\vec{x}_{k+1} - \vec{x}_k)\| \leq \\ &\leq (1 + L + L^2 + \ldots + L^{r-1}) \|\vec{x}_{k+1} - \vec{x}_k\| < \frac{1}{1-L} \|\vec{x}_{k+1} - \vec{x}_k\| \\ &\leq \frac{L^k}{1-L} \|\vec{x}_1 - \vec{x}_0\| \qquad \forall r \in \mathbb{N} \end{aligned} \tag{3.9}$$

Gleichung (3.9) mit $k = 1$ ergibt, dass sämtliche Elemente der Folge $\vec{x}_1 = \overrightarrow{T}(\vec{x}_0)$, $\vec{x}_2 = \overrightarrow{T}(\vec{x}_1)$, ... in dem beschränkten, abgeschlossenen Bereich

$$D \cap \overline{S}\left(\vec{x}_1, \frac{L}{1-L} \|\vec{x}_1 - \vec{x}_0\|\right), \tag{3.10}$$

liegen, wobei $\overline{S}(\vec{x}_m, r)$ die abgeschlossene Kugel $\{\vec{x} \in \mathbb{R}^n : \|\vec{x} - \vec{x}_m\| \leq r\}$ mit Mittelpunkt $\vec{x}_m \in \mathbb{R}^n$ und dem Radius $r \in \mathbb{R}^+$ bezeichnet.
Wendet man Gleichung (3.9) für beliebiges $k \in \mathbb{N}$ an, so folgt wegen $L^k \to 0$ und $L < 1$, dass die Folge $\vec{x}_0, \vec{x}_1, \vec{x}_2, \ldots$ eine Chauchyfolge ist. Da eine Cauchyfolge in einem beschränkten, abgeschlossenen Bereich in $\mathbb{R}^n$ stets gegen ihren Grenzwert konvergiert, ist die Existenz des Grenzwertes $\vec{x}^* = \lim_{k\to\infty} \vec{x}_k$ sichergestellt. Es gilt daher wegen der Stetigkeit von T

$$\vec{x}^* = \lim_{k\to\infty} \vec{x}_k = \lim_{k\to\infty} \overrightarrow{T}(\vec{x}_{k-1}) = \overrightarrow{T}\big(\lim_{k\to\infty} \vec{x}_{k-1}\big) = \overrightarrow{T}(\vec{x}^*).$$

Der Grenzwert $\vec{x}^*$ ist der eindeutige Fixpunkt der Gleichung $\vec{x} = \overrightarrow{T}(\vec{x})$.

3. *Abschätzung.* Es gilt

$$\|\vec{x}_k - \vec{x}^*\| = \left\| \overrightarrow{T}(\vec{x}_{k-1}) - \overrightarrow{T}(\vec{x}^*) \right\| \leq L \|\vec{x}_{k-1} - \vec{x}^*\|,$$

daraus folgt die Gültigkeit von Gleichung (3.7). Weiters folgt aus Gleichung (3.9)

$$\|\vec{x}_{k+r} - \vec{x}_k\| < \frac{1}{1-L} \|\vec{x}_{k+1} - \vec{x}_k\| = \frac{1}{1-L} \left\| \overrightarrow{T}(\vec{x}_k) - \overrightarrow{T}(\vec{x}_{k-1}) \right\| \leq$$

$$\leq \frac{L}{1-L} \|\vec{x}_k - \vec{x}_{k-1}\| \leq \frac{L^k}{1-L} \|\vec{x}_1 - \vec{x}_0\|.$$

Für $r \to \infty$ erhält man

$$\|\vec{x}^* - \vec{x}_k\| = \|\vec{x}_k - \vec{x}^*\| < \frac{L}{1-L} \|\vec{x}_k - \vec{x}_{k-1}\| \leq \frac{L^k}{1-L} \|\vec{x}_1 - \vec{x}_0\|.$$

□

Bemerkung.

1. Gleichung (3.6) zeigt, dass die Konvergenz tatsächlich umso rascher eintritt, je kleiner L ist.

2. Gleichung (3.7) bietet die Möglichkeit, bei Kenntnis von $\vec{x}_0, \vec{x}_1$ und L die Qualität der k-ten Näherung $\vec{x}_k$ a-priori abzuschätzen.

3. Der Kontraktionssatz gilt nicht nur im Endlichdimensionalen, sondern er kann auch auf allgemeine normierte Räume übertragen werden und ist somit auf unendlichdimensionale Funktionenräume anwendbar. Er ist dann einer der zentralen Sätze der konstruktiven Mathematik und dient in dieser Form zum Nachweis der Existenz und der Eindeutigkeit von Lösungen von Funktionalgleichungen und Operatorgleichungen (Differentialgleichungsprobleme, etc.) und auch zur Gewinnung von Iterationsverfahren zur näherungsweisen Lösung dieser Probleme.

Im Folgenden wird eine *Modifikation des Kontraktionssatzes* vorgestellt, wo auf die Implementierung und Computerarithmetik Bezug genommen wird. Unter Berücksichtigung, dass bei tatsächlichen Implementierungen der Iteration $\vec{x}_k = \overrightarrow{T}(\vec{x}_{k-1})$ auf einem Rechner der Ablauf durch die Computerarithmetik beeinflusst wird, muss im Rechner statt $\vec{x}_k = \overrightarrow{T}(\vec{x}_{k-1})$ die gestörte Version

$$\widetilde{\vec{x}}_k := \widetilde{\overrightarrow{T}}(\widetilde{\vec{x}}_{k-1}) \tag{3.11}$$

betrachtet werden mit $\widetilde{\overrightarrow{T}} : \mathbb{M}^n \to \mathbb{M}^n$, dabei $\mathbb{M}$ bezeichnet die Menge der Maschinenzahlen und $\mathbb{M}^n$ ein n-Tuppel von Maschinenzahlen.

Satz 3.2.5. Die Abbildung $\overrightarrow{T} : D \subset \mathbb{R}^n \to \mathbb{R}^n$ genüge auf D den Voraussetzungen des Kontraktionssatzes und $\widetilde{\overrightarrow{T}}$ erfülle

$$\left\| \widetilde{\overrightarrow{T}}(\widetilde{\vec{x}}) - \overrightarrow{T}(\widetilde{\vec{x}}) \right\| \leq \varepsilon, \quad \widetilde{\vec{x}} \in D \cap \mathbb{M}^n. \tag{3.12}$$

Für den Startwert $\vec{x}_0 \equiv \widetilde{\vec{x}}_0 \in D \cap \mathbb{M}^n$ gilt

$$\widetilde{S}_1 := S\left(\widetilde{x}_1, \frac{1}{1-L} \big(L \left\| \widetilde{\vec{x}}_1 - \widetilde{\vec{x}}_0 \right\| + 2\varepsilon \big) \right) \subset D.$$

Dann gilt:

1. Die Folge $\left\{\widetilde{\vec{x}}_k\right\}$ liegt ganz in der Kugel $\widetilde{S}_1$.
2. Es gilt:
$$\exists k^* : \left\|\widetilde{\vec{x}}_k - \vec{x}^*\right\| \leq \delta := \frac{\varepsilon}{1-L}, \quad k = k^*, k^*+1, \ldots \tag{3.13}$$
3. Wenn $\left\|\widetilde{\vec{x}}_{k-1} - \vec{x}^*\right\| > \delta$ ist, gilt
$$\left\|\widetilde{\vec{x}}_k - \vec{x}^*\right\| < \left\|\widetilde{\vec{x}}_{k-1} - \vec{x}^*\right\|. \tag{3.14}$$
4. Weiters gilt die **a-posteriori Fehlerabschätzung**
$$\left\|\widetilde{\vec{x}}_k - \vec{x}^*\right\| \leq \frac{\varepsilon L \left\|\widetilde{\vec{x}}_k - \widetilde{\vec{x}}_{k-1}\right\|}{1-L}. \tag{3.15}$$

Beweis. Siehe Literatur. □

Die a-posteriori Abschätzung (3.15) liefert eine **Abbruchbedingung** für die Iteration. Wenn die Genauigkeitsforderung
$$\left\|\widetilde{\vec{x}}_k - \vec{x}^*\right\| \leq \mathrm{TOL},$$
wobei TOL die *Toleranz* bezeichnet, erfüllt werden soll, beobachtet man die Differenzen $\left\|\widetilde{\vec{x}}_k - \widetilde{\vec{x}}_{k-1}\right\|$ und bricht ab, sobald die Ungleichung
$$\left\|\widetilde{\vec{x}}_k - \widetilde{\vec{x}}_{k-1}\right\| \leq \frac{1}{L}\big(TOL(1-L) - \varepsilon\big)$$
erfüllt ist. Aus (3.15) folgt:
$$\left\|\widetilde{\vec{x}}_k - \vec{x}^*\right\| \leq \frac{\varepsilon + L\frac{1}{L}\big(TOL(1-L) - \varepsilon\big)}{1-L} = TOL$$

3.3 Newtonverfahren

Aus der Kontraktioneigenschaft der Funktion $\overrightarrow{T}(\vec{x}) = \vec{x} - R(\vec{x})\vec{F}(\vec{x})$ folgt die Konvergenz des Iterationsverfahrens. Die Geschwindigkeit der Konvergenz hängt jedoch von der Lipschitzkonstanten L ab. Je näher L bei 1 liegt, umso langsamer ist die Konvergenz, für $0 < L \ll 1$ hat man rasche Konvergenz. Eine naheliegende Idee ist nun, $R(\vec{x})$ so zu wählen, dass $\overrightarrow{T}'(\vec{x})$ in einer Umgebung von $\vec{x}^*$ möglichst klein wird, was dann in der Umgebung des Fixpunktes $\vec{x}^*$ eine kleine Lipschitzkonstante und damit rasche Konvergenz zu $\vec{x}^*$ zur Folge hat. Es wird $R(\vec{x})$ in $\overrightarrow{T}(\vec{x}) = \vec{x} - R(\vec{x})\vec{F}(\vec{x})$ konkret so gewählt, dass
$$\frac{\mathrm{d}\overrightarrow{T}}{\mathrm{d}\vec{x}}(\vec{x}^*) = \overrightarrow{T}'(\vec{x}^*) = 0_{n\times n} \tag{3.16}$$
gilt. Dies wird mit der Wahl von
$$R(\vec{x}) = \big(\vec{F}'(\vec{x})\big)^{-1} \tag{3.17}$$

erreicht. Es folgt daher

$$\overrightarrow{T}(\vec{x}) = \vec{x} - \big(\vec{F}'(\vec{x})\big)^{-1}\vec{F}(\vec{x}). \tag{3.18}$$

Dass (3.16) tatsächlich gilt ist im Fall $T : \mathbb{R} \longrightarrow \mathbb{R}$ leicht nachzuweisen:

$$T'(x) = \Big(x - \big(F'(x)\big)^{-1}F(x)\Big)' = \underbrace{1 - \underbrace{\big(F'(x)\big)^{-1}F'(x)}_{=1} + \big(F'(x)\big)^{-2}F''(x)F(x)}_{=0} \tag{3.19}$$

Schließlich ist $T'(x^*) = 0$ wegen $F(x^*) = 0$. Im Falle des $\mathbb{R}^n$, also für $\overrightarrow{T} : \mathbb{R}^n \longrightarrow \mathbb{R}^n$ gilt eine analoge Überlegung, es muss lediglich mit der vektorwertigen Funktion $\vec{F}$ und der Funktionalmatrix argumentriert werden:

$$\overrightarrow{T}'(\vec{x}) = \Big(\vec{x} - \big(\vec{F}'(\vec{x})\big)^{-1}\vec{F}(\vec{x})\Big)' = \underbrace{I - \underbrace{\big(\vec{F}'(\vec{x})\big)^{-1}\vec{F}'(\vec{x})}_{=I} + \big(\vec{F}'(\vec{x})\big)^{-1}\vec{F}''(\vec{x})\big(\vec{F}'(\vec{x})\big)^{-1}\vec{F}(\vec{x})}_{=\mathbf{0}} \tag{3.20}$$

Der Term $\big(\vec{F}'(\vec{x})\big)^{-1}\vec{F}''(\vec{x})\big(\vec{F}'(\vec{x})\big)^{-1}\vec{F}(\vec{x})$ ist eine Matrix. Das ist auch der Grund, warum der Ausdruck sich von der Formulierung (3.19) unterscheidet. $\vec{F}(\vec{x})$ ist ein Vektor, $\big(\vec{F}'(\vec{x})\big)^{-1}$ ist die inverse Funktionalmatrix, daher ist $\big(\vec{F}'(\vec{x})\big)^{-1}\vec{F}(\vec{x})$ ebenfalls ein Vektor. $\vec{F}''(\vec{x})$ ist ein bilinearer Operator, der auf zwei Vektoren des $\mathbb{R}^n$ wirkt, also $\vec{F}'' : \mathbb{R}^n \times \mathbb{R}^n \longrightarrow \mathbb{R}^n$. $\vec{F}''(\vec{x})\big(\vec{F}'(\vec{x})\big)^{-1}\vec{F}(\vec{x})$ ist eine Matrix.

Das Iterationsverfahren (3.4) wird mit (3.18) zum **Newtonverfahren**. Dies ist durch den Startvektor $\vec{x}_0 \in \mathbb{R}^n$ und der Iteration

$$\vec{x}_k = \overrightarrow{T}(\vec{x}_{k-1}) = \vec{x}_{k-1} - \big(\vec{F}'(\vec{x}_{k-1})\big)^{-1}\vec{F}(\vec{x}_{k-1}) \tag{3.21}$$

rekursiv definiert. Die geometrische Veranschaulichung für $n = 1$ ist in Abbildung 3.14 dargestellt.

Beispiel 3.3.1. Betrachte $F(x) = e^{x-2} - x$. Für die Newtoniteration ergibt sich:

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

$$\begin{aligned} x_0 &= 0.25 \\ x_1 &= 0.1577418874\ldots \\ x_2 &= 0.1585942711\ldots \\ x_3 &= 0.1585943396\ldots \end{aligned}$$

Die raschere Konvergenz im Vergleich zu Beispiel 3.2.1 ist offensichtlich.

Die rasche Konvergenz des Newtonverfahrens ergibt sich für Startwerte, hinreichend nahe an der gesuchten Nullstelle. Global gesehen, d.h. für beliebige Startwerte, muss das Verfahren aber nicht konvergieren.

Im Folgenden betrachten wir die reelle Funktion $F(x) = \arctan x$. Es gilt $F(0) = 0$. Der Graph dieser Funktion und der **Konvergenzbereich**, also die Menge aller Startwerte, bei denen das Iterationsverfahren konvergiert fnden sich in Abbildung 3.15.

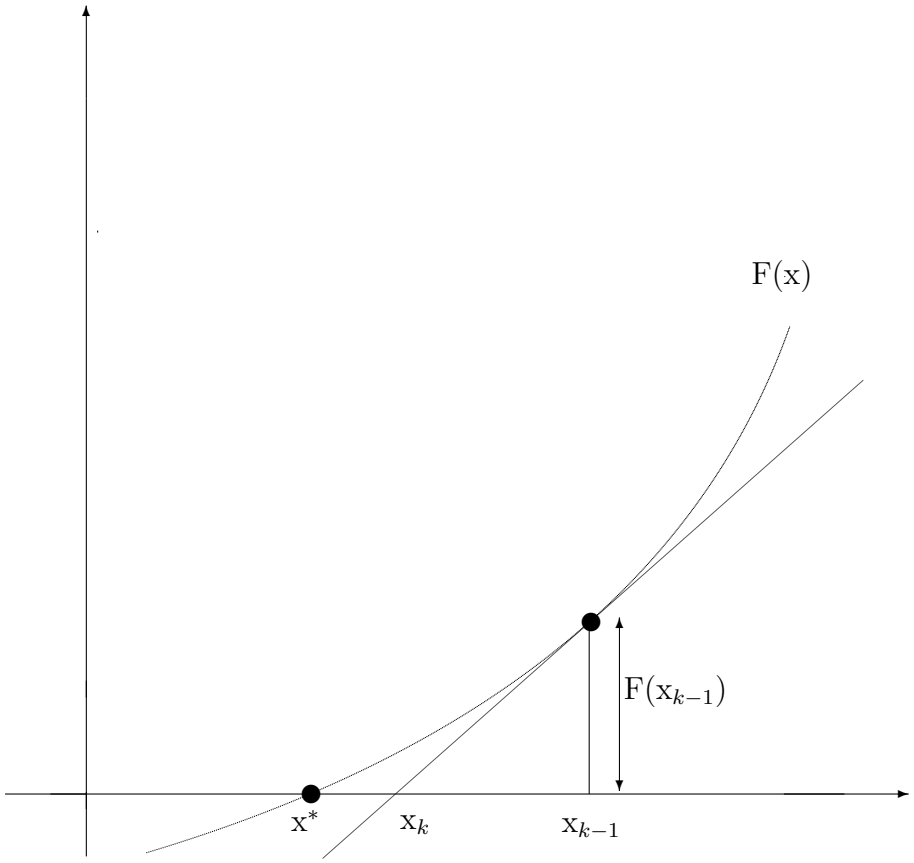


Abbildung 3.14: Geometrische Veranschaulichung des Newtonverfahrens

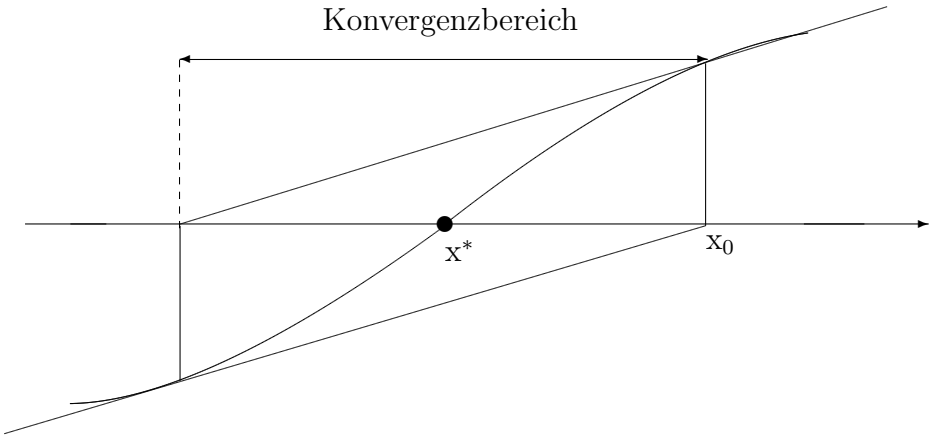


Abbildung 3.15: Konvergenzbereich von $F(x) = \arctan x$

Die Abbildungen 3.16 und 3.17 zeigen Newtoniterationen zu verschiedenen Startwerten x_0. Der Startwerte im Inneren des Einzugsbereiches (Abb. 3.17) ergibt Konvergenz, außerhalb des Einzugsbereiches (Abb. 3.16) *alternierende Divergenz*, $|x_0| < |x_1| < |x_2| < \ldots$ und $x_0 > 0, x_1 < 0, x_2 > 0, \ldots$ Für Startwerte genau am Rand des Einzugsbereiches (Abb.3.15) ergibt sich die Folge

$$x_0, x_1 = -x_0, x_2 = x_0, x_3 = -x_0, \ldots$$

Im Vergleich dazu untersuchen wir die Konvergenzbereiche einer quadratischen Funktion mit Minmalstelle $\bar{x}$ und Nullstellen x_1^*, x_2^*, siehe Abbildung 3.18.

1. Startwert $x_0 > \overline{x}$: Konvergenz gegen x_1^*, der Einzugsbereich von x_1^* ist also $(\overline{x}, \infty)$.
2. $x_0 < \overline{x}$: Konvergenz gegen x_2^*, der Einzugsbereich von x_2^* ist also $(-\infty, \overline{x})$.
3. $x_0 = \overline{x}$: Das Newtonverfahren lässt sich hier nicht durchführen, da $F'(\bar{x}) = 0$.

Im Falle der Divergenz des Newtonverfahrens ergibt sich

$$|F(x_0)| < |F(x_1)| < |F(x_2)| < \ldots$$

Dies liefert die Idee für das **gedämpfte Newtonverfahren**.
Man vergleicht $|F(x_k)|$ mit $|F(x_{k-1})|$. Wenn $|F(x_k)| \geq |F(x_{k-1})|$ gilt, so verkürzt man den Newtonschritt, d.h. statt

$$x_k = x_{k-1} - \big(F'(x_{k-1})\big)^{-1} F(x_{k-1})$$

betrachtet man

$$x_k^{(1)} = x_{k-1} - \lambda\big(F'(x_{k-1})\big)^{-1} F(x_{k-1}) \quad \lambda \in (0,1)$$

und untersucht, ob $|F(x_k^{(1)})| < |F(x_{k-1})|$ gilt. Falls das nicht zutrifft, betrachtet man

$$x_k^{(2)} = x_{k-1} - \frac{\lambda}{2}\big(F'(x_{k-1})\big)^{-1} F(x_{k-1})$$

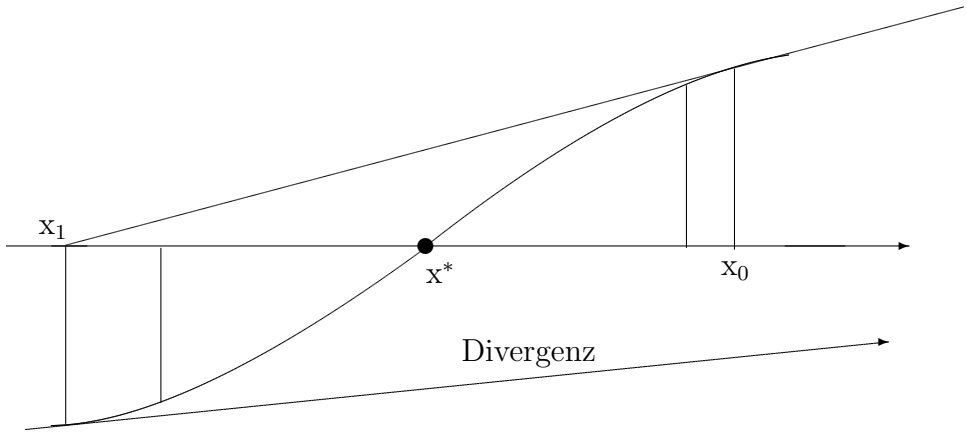


Abbildung 3.16: Divergenz von $F(x) = \arctan x$

usw., solange bis für ein $j \in \mathbb{N}$

$$|F(x_k^{(j)})| < |F(x_{k-1})|$$

gilt. Dann setzt man $x_k := x_k^{(j)}$ und setzt das Newtonverfahren fort. Durch diese Strategie kann man die Konvergenzbereiche des Newtonverfahrens wesentlich vergrößern.

Das gedämpfte Newtonverfahren im Falle $\vec{F} : \mathbb{R}^n \longrightarrow \mathbb{R}^n$ wird ident formuliert, an Stelle der Beträge treten Normen, es werden die Normen $\left\|\vec{F}(\vec{x}_k)\right\|$ kontrolliert.
Im Falle des $\mathbb{R}^n$ ist außerdem zu beachten, dass die Gültigkeit der Ungleichung $\left\|\vec{F}(\vec{x}_k)\right\| < \left\|\vec{F}(\vec{x}_{k-1})\right\|$ von der Norm und der Skalierung von $\vec{F}$ abhängt. Algorithmisch kann das eine Reihe von speziellen Maßnahmen bedeuten.

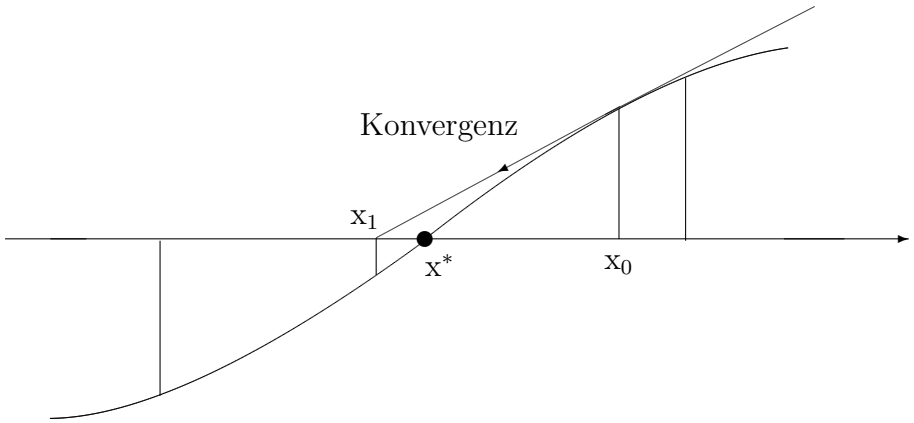


Abbildung 3.17: Konvergenz von $F(x) = \arctan x$

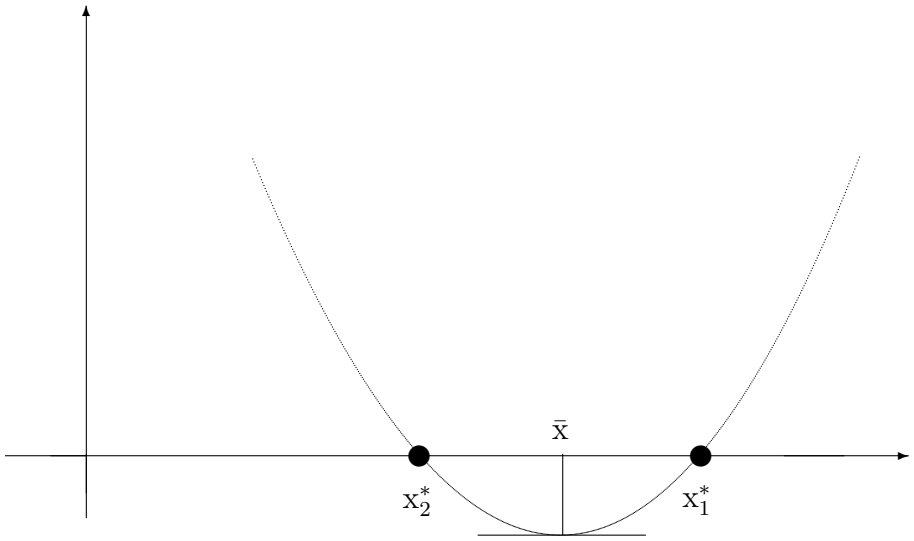


Abbildung 3.18: Konvergenzbereiche einer quadratischen Funktion

Kapitel 4

Interpolation

4.1 Einleitende Betrachtungen

Bei einem **Interpolationsproblem** sind im einfachsten Fall Paare (x_k, y_k) $k = 0, \ldots n$ gegeben und *einfache Funktionen* $g(x)$ mit $g(x_k) = y_k$ gesucht. Klassen einfacher Funktionen können Polynome, stückweise Polynome (Splines) oder auch rationale Funktionen sein.

Verwandt ist das Thema **Approximation**. Gegeben sind dabei eine geeignete Norm $\|\cdot\|$ und eine Funktion f. Gesucht ist wiederum eine *einfache Funktion* g, die jetzt im Sinne der Norm eine gute Approximation sein soll, z.B. mit $\|g - f\|$ minimal im Vergleich zu anderen einfachen Funktionen aus einer gegebenen Klasse von Funktionen, siehe weiterführende Literatur.

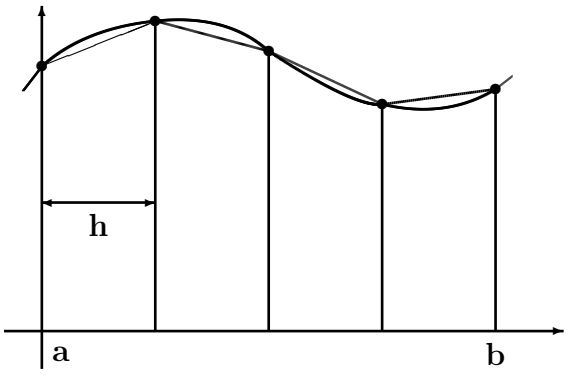


Abbildung 4.1: Trapezregel

Numerische Integration *Trapezregel:* Der Integrand $f(x)$, dessen Stammfunktion $F(x)$ nicht geschlossen darstellbar ist oder nicht bekannt ist wird durch einen interpolierenden Polygonzug $g(x)$ ersetzt. $\int_a^b g(x)\,\mathrm{d}x$ kann dann berechnet werden und ist eine Näherung für $\int_a^b f(x)\,\mathrm{d}x$, siehe Kapitel 1, Seite 16 und Abb. 4.1.

Simpson Regel: Der Integrand wird stückweise durch interpolierende quadratische Polynome

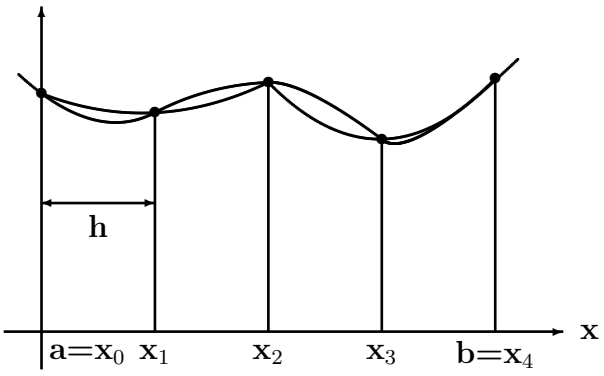


Abbildung 4.2: Simpsonregel

ersetzt, in der Abbildung 4.2 etwa durch zwei Polynome, das erste $c_0^{(1)} + c_1^{(1)}x + c_2^{(1)}x^2$ ist durch die Punkte $(x_0, y_0 = f(x_0))$, $(x_1, y_1 = f(x_1))$, $(x_2, y_2 = f(x_2))$ festgelegt, das zweite Polynom $c_0^{(2)} + c_1^{(2)}x + c_2^{(2)}x^2$ durch die Punkte $(x_2, f(x_2))$, $(x_3, f(x_3))$, $(x_4, f(x_4))$.

Darstellung von Standardfunktionen: Am Computer können nur die 4 Grundrechnungsarten $*$, $/$, $+$, $-$ ausgeführt werden, d.h. es können etwa Polynome bzw. rationale Funktionen, die in endlicher Weise mit den Grundrechnungsarten aufgebaut sind, ausgewertet werden. Elementare Funktionen oder Standardfunktionen wie z.B.

$$\sin x, \quad \arcsin x, \quad e^x, \quad \ln x, \ \ldots$$

können nicht mit Hilfe der Grundrechenoperationen in endlicher Weise dargestellt werden. Um diese Funktionen am Computer auswerten zu können, muss man sie durch in endlich vielen Rechenschritten berechenbare Funktion (d.h. durch Polynome oder rationale Funktionen) ersetzen. Etwa die Taylorreihenentwicklung $e^x = \sum_{j=0}^{\infty} \frac{x^j}{j!}$ kann durch ein (endliches) Taylorpolynom $e^x \approx \sum_{j=0}^{n} \frac{x^j}{j!}$ ersetzt werden, dieses Taylorpolynom stellt aber e^x nicht exakt sondern nur näherungsweise dar.

Anforderungen an die Ersatzfunktionen: Je nach Anwendungsfall werden an diese *einfachen Funktionen* oder *Ersatzfunktionen* verschiedene Anforderungen gestellt. Etwa bei der numerischen Integration sollen die Programme immer wieder auf verschiedenste Integranden f angewendet werden. Das Ersetzen von $f(x)$ durch $g(x)$ muss einfach und unproblematisch möglich sein.

Hingegen bei der Implementierung von Standardfunktionen kann man bei der Aufstellung der Ersatzfunktion einen sehr großen Aufwand treiben. Da so ein Programm für eine Standardfunktion dann immer wieder aufgerufen wird, ist hier die Effizienz besonders wichtig: einerseits hat man strenge Genauigkeitsforderungen (meist wird verlangt, dass man bei der Auswertung nicht mehr wie einen elementaren Rundungsfehler macht), anderseits möchte man dieses Genauigkeitsniveau mit möglichst wenig Rechenoperationen erreichen (z.B. bei polynomialer Ersatzfunktion $g(x)$ mit möglichst niedrigem Polynomgrad). Um dieses schwierige Ziel zu erreichen, lohnt es sich i.a. einen größeren Entwicklungsaufwand vor der Programmierung und Implementierung in Kauf zu nehmen.

Bei der Festlegung von $g(x)$ müssen i.a. zwei Entscheidungen getroffen werden:

(i) Welcher Funktionenklasse soll g angehören?

- Polynom vom Grad 3:

$$c_0 + c_1 x + c_2 x^2 + c_3 x^3 \qquad c_i \in \mathbb{R}$$

- Gerades Polynom vom Grad 4:

$$c_0 + c_2 x^2 + c_4 x^4$$

- Ungerades Polynom von Grad 5:

$$c_1 x + c_3 x^3 + c_5 x^5$$

- Rationale Funktion: Zählerpolynom Grad 3, Nennerpolynom Grad 5

$$\frac{c_0 + c_1 x + c_2 x^2 + c_3 x^3}{d_0 + d_1 x + d_2 x^2 + d_3 x^3 + d_4 x^4 + d_5 x^5}$$

- Stückweises Polynom vom Grad 1 (Polygonzug)
- Stückweises Polynom vom Grad 2 (wie bei der Simpsonregel)

$$\vdots$$

(ii) Wodurch soll g festgelegt werden?

Interpolation: Dabei wird meist verlangt, dass an gewissen Stellen x_i die einfache Funktion $g(x)$ die Werte von f animmt. Also dass gilt:

$$\begin{aligned} g(x_0) &= f(x_0) \\ g(x_1) &= f(x_1) \\ &\vdots \\ g(x_n) &= f(x_n) \end{aligned}$$

für eine bestimmte Menge $x_0, x_1, \ldots, x_n$ von *Interpolationsknoten*. Manchmal wird eine Interpolationsfunktion auch durch andere Interpolationsdaten festgelegt etwa kann ein Polynom $g(x)$ vom Grad 3 durch folgende Forderungen festgelegt werden:

$$\begin{aligned} g(x_0) &= f(x_0) \\ g'(x_0) &= f'(x_0) \\ g''(x_0) &= f''(x_0) \\ g(x_1) &= f(x_1). \end{aligned}$$

Ausgleichende Interpolation: Es gibt mehr Interpolationsdaten als Unbekannte in der zu bestimmenden Ersatzfunktion.

Taylorpolynom: g ist eine endliche Partialsumme

$$g(x) = \sum_{i=0}^{n} \frac{1}{i!} f^{(i)}(x_0)(x - x_0)^i$$

der Taylorreihe

$$\sum_{i=0}^{\infty} \frac{1}{i!} f^{(i)}(x_0)(x - x_0)^i$$

von f, dabei wird hinreichende Glattheit von f vorausgesetzt. Ist etwa f in dem betrachteten Intervall $n + 1$ mal stetig differenzierbar, so existiert nicht nur das Taylorpolynom n-ten Grades, sondern man kann auch noch den Fehler $g(x) - f(x)$ durch eine geeignete Restglieddarstellung des Taylorpolynoms abschätzen.

Beispiel 4.1.1. $f(x) = \sin x,\ x \in [0, \frac{\pi}{2}]$

a) $g(x) \ldots$ **Polynom vom Grad 1**, interpoliert die Daten

$$\begin{aligned} (x_0, f(x_0)) &= (0, \sin(0)) = (0, 0) \\ (x_1, f(x_1)) &= (\tfrac{\pi}{2}, \sin(\tfrac{\pi}{2})) = (\tfrac{\pi}{2}, 1) \end{aligned}$$

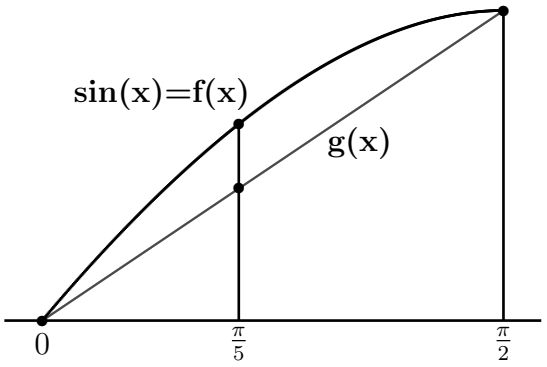


Abbildung 4.3: Vergleich von $g(x) = \frac{2}{\pi}x$ und $\sin x$ an $x = \frac{\pi}{5}$

$$g(x) = \frac{2}{\pi}x = 0.6366197724 \ldots x$$

Vergleich von $g(x)$ und $f(x)$ etwa an der Stelle $x = \frac{\pi}{5}$, siehe Abb. 4.3:

$$\boxed{g\left(\frac{\pi}{5}\right) - \sin\left(\frac{\pi}{5}\right) = -0.1877852523 \ldots}$$

b) $g(x) \ldots$ **Polynom vom Grad 2**, interpoliert die Daten

$$\begin{aligned} (x_0, f(x_0)) &= (0, \sin(0)) = (0, 0) \\ (x_1, f(x_1)) &= (\tfrac{\pi}{2}, \sin(\tfrac{\pi}{2})) = (\tfrac{\pi}{2}, 1) \\ (x_2, f(x_2)) &= (\tfrac{\pi}{4}, \sin(\tfrac{\pi}{4})) = (\tfrac{\pi}{4}, \tfrac{\sqrt{2}}{2}) \end{aligned}$$

also zusätzlich zu den Interpolationsdaten von a) kommt noch $(x_2, f(x_2)) = (\frac{\pi}{4}, \sin(\frac{\pi}{4}))$ dazu.

$$g(x) \;=\; 0.6366197724\ldots x - 0.3357488674\ldots x(x - \frac{\pi}{2})$$

Das ist die *Newtonsche Darstellung* des Interpolationspolynoms, siehe Abschnitt 4.2. Der Koeffizient von x ist wie bei a). Die Newtonsche Darstellung eignet sich sehr gut, um die Interpolationsdatenmenge zu erweitern.

$$\boxed{g\left(\frac{\pi}{5}\right) - \sin\left(\frac{\pi}{5}\right) \;=\; 0.01103725769\ldots,}$$

Der Fehler $g(x) - \sin x$ an der Stelle $x = \frac{\pi}{5}$ ist ungefähr um einen Faktor 10 kleiner als in a).

c) $g(x)\ldots$ **Polynom vom Grad 3**, interpoliert die Daten

$$\begin{aligned}
(x_0, f(x_0)) &= (0, \sin(0)) &&= (0, 0)\\
(x_1, f(x_1)) &= (\tfrac{\pi}{2}, \sin(\tfrac{\pi}{2})) &&= (\tfrac{\pi}{2}, 1)\\
(x_2, f(x_2)) &= (\tfrac{\pi}{4}, \sin(\tfrac{\pi}{4})) &&= (\tfrac{\pi}{4}, \tfrac{\sqrt{2}}{2})\\
(x_3, f(x_3)) &= (\tfrac{\pi}{6}, \sin(\tfrac{\pi}{6})) &&= (\tfrac{\pi}{6}, \tfrac{1}{2})
\end{aligned}$$

Newtonsche Darstellung:

$$\begin{aligned}
g(x) \;=\;& 0.6366197724\ldots x - 0.3357488674\ldots x(x - \frac{\pi}{2}) -\\
&- 0.11214109653\ldots x(x - \frac{\pi}{2})(x - \frac{\pi}{4})
\end{aligned}$$

die Koeffizienten von x und $x(x - \frac{\pi}{2})$ sind wie bei b).

$$\boxed{g\left(\frac{\pi}{5}\right) - \sin\left(\frac{\pi}{5}\right) \;=\; -0.00025624\ldots} \tag{4.1}$$

d) Da $\sin x$ eine ungerade Funktion ist liegt es nahe mit einem **ungeraden Polynom** zu arbeiten:

$$g(x) \;=\; c_1 x + c_3 x^3$$

c_1, c_3 wird so ermittelt, dass bezüglich der Daten

$$\begin{aligned}
(x_0, f(x_0)) &= (\tfrac{\pi}{6}, \sin(\tfrac{\pi}{6})) &&= (\tfrac{\pi}{6}, \tfrac{1}{2})\\
(x_1, f(x_1)) &= (\tfrac{\pi}{3}, \sin(\tfrac{\pi}{3})) &&= (\tfrac{\pi}{3}, \tfrac{\sqrt{3}}{2})
\end{aligned}$$

interpoliert wird. Das führt auf ein lineares Gleichungssystem mit 2 Gleichungen für die 2 Unbekanten c_1, c_3. Berechnet man die beiden Koeffizienten c_1, c_3 durch Lösen des Systems

$$\begin{aligned}
x = \tfrac{\pi}{6}: \quad c_1 \tfrac{\pi}{6} + c_3 (\tfrac{\pi}{6})^3 &= \tfrac{1}{2}\\
x = \tfrac{\pi}{3}: \quad c_1 \tfrac{\pi}{3} + c_3 (\tfrac{\pi}{3})^3 &= \tfrac{\sqrt{3}}{2}
\end{aligned}$$

erhält man:

$$g(x) \;=\; 0.997575097\ldots x - 0.1555519069\ldots x^3$$

$$\boxed{g\left(\frac{\pi}{5}\right) - \sin\left(\frac{\pi}{5}\right) \;=\; 0.00042298\ldots}$$

Hier holt man aus zwei Interpolationsdaten etwa dieselbe Genauigkeit heraus wie bei c) aus vier Interpolationsdaten!

e) **Taylorpolynom**: $\sin x \approx x - \frac{x^3}{3!} = g(x)$

$$\boxed{g\left(\frac{\pi}{5}\right) - \sin\left(\frac{\pi}{5}\right) \;=\; 0.000808442\ldots}$$

Taylorpolynom approximiert besonders gut in der Nähe der Entwicklungsstelle $x = 0$. Für größere x-Werte wird die Approximationsqualität schlechter.

f) $g(x)$ wie bei d) und e) von der Form

$$g(x) \;=\; c_1 x + c_3 x^3$$

Das Interpolationspolynom soll aber im Gegensatz zu d) durch 3 Interpolationsdaten

$$\begin{aligned}
(x_0, f(x_0)) &= (\tfrac{\pi}{6}, \sin(\tfrac{\pi}{6})) = (\tfrac{\pi}{6}, \tfrac{1}{2}) \\
(x_1, f(x_1)) &= (\tfrac{\pi}{4}, \sin(\tfrac{\pi}{4})) = (\tfrac{\pi}{4}, \tfrac{\sqrt{2}}{2}) \\
(x_2, f(x_2)) &= (\tfrac{\pi}{3}, \sin(\tfrac{\pi}{3})) = (\tfrac{\pi}{3}, \tfrac{\sqrt{3}}{2}).
\end{aligned}$$

festgelegt werden d.h. die beiden Koeffizienten c_1, c_3 sind durch drei Bedingungen festgelegt, also **ausgleichende Interpolation**. Die Quadratsumme

$$\begin{aligned}
E(c_1, c_3) \;:=\; &\left(c_1\frac{\pi}{6} + c_3(\frac{\pi}{6})^3 - \frac{1}{2}\right)^2 + \left(c_1\frac{\pi}{4} + c_3(\frac{\pi}{4})^3 - \frac{\sqrt{2}}{2}\right)^2 + \\
&+ \left(c_1\frac{\pi}{3} + c_3(\frac{\pi}{3})^2 - \frac{\sqrt{3}}{2}\right)^2
\end{aligned}$$

soll minimal werden. Notwendige Bedingungen für ein Minimum

$$\frac{\partial E}{\partial c_1} \;=\; 0, \qquad \frac{\partial E}{\partial c_3} \;=\; 0$$

also

$$\begin{aligned}
&2\left(c_1\frac{\pi}{6} + c_3(\frac{\pi}{6})^3 - \frac{1}{2}\right)\frac{\pi}{6} + \\
&\quad +2\left(c_1\frac{\pi}{4} + c_3(\frac{\pi}{4})^3 - \frac{\sqrt{2}}{2}\right)\frac{\pi}{4} + 2\left(c_1\frac{\pi}{3} + c_3(\frac{\pi}{3})^3 - \frac{\sqrt{3}}{2}\right)\frac{\pi}{3} \;=\; 0 \\
&2\left(c_1\frac{\pi}{6} + c_3(\frac{\pi}{6})^3 - \frac{1}{2}\right)\left(\frac{\pi}{6}\right)^3 + \\
&\quad +2\left(c_1\frac{\pi}{4} + c_3(\frac{\pi}{4})^3 - \frac{\sqrt{2}}{2}\right)\left(\frac{\pi}{4}\right)^3 + 2\left(c_1\frac{\pi}{3} + c_3(\frac{\pi}{3})^3 - \frac{\sqrt{3}}{2}\right)\left(\frac{\pi}{3}\right)^3 \;=\; 0
\end{aligned}$$

Es ergibt sich das folgende lineare Gleichungssystem zur Berechnung von c_1, c_3:

$$\begin{aligned}
c_1[(\tfrac{\pi}{6})^2 + (\tfrac{\pi}{4})^2 + (\tfrac{\pi}{3})^2] + c_3[(\tfrac{\pi}{6})^4 + (\tfrac{\pi}{4})^4 + (\tfrac{\pi}{3})^4] &= \pi(\frac{1}{12} + \frac{\sqrt{2}}{8} + \frac{\sqrt{3}}{6}) \\
c_1[(\tfrac{\pi}{6})^4 + (\tfrac{\pi}{4})^4 + (\tfrac{\pi}{3})^4] + c_3[(\tfrac{\pi}{6})^6 + (\tfrac{\pi}{4})^6 + (\tfrac{\pi}{3})^6] &= \frac{1}{2}(\frac{\pi}{6})^3 + \frac{\sqrt{2}}{2}(\frac{\pi}{4})^3 + \frac{\sqrt{3}}{2}(\frac{\pi}{3})^3
\end{aligned}$$

Daraus folgt

$$g(x) \;=\; 0.9964027665\ldots x - 0.1546327342\ldots x^3$$

$$\boxed{g\left(\frac{\pi}{5}\right) - \sin\left(\frac{\pi}{5}\right) \;=\; -0.0000836\ldots}$$

4.2 Lagrange- und Hermiteinterpolation

Unter **Lagrangeinterpolation** versteht man die folgende Interpolationsaufgabe. An die

$$\text{Daten:} \qquad (x_0, f(x_0) =: f_0),\ (x_1, f(x_1) := f_1),\ \ldots,\ (x_n, f(x_n) := f_n) \tag{4.2}$$

soll ein Interpolationspolynom vom Grad n

$$p(x) \;=\; c_0 + c_1 x + \cdots + c_n x^n \tag{4.3}$$

angepasst werden. Die $n+1$ Interpolationsbedingungen $p(x_i) = f(x_i)$, $i = 0, 1, 2, \ldots n$ legen die $n+1$ Koeffizienten $c_0, \ldots, c_n$ eindeutig fest, wenn sämtliche x_i verschieden sind:

$$\begin{array}{rll}
x_0: & c_0 + c_1 x_0 + c_2 x_0^2 + \cdots + c_n x_0^n &= f(x_0) \\
x_1: & c_0 + c_1 x_1 + c_2 x_1^2 + \cdots + c_n x_1^n &= f(x_1) \\
x_2: & c_0 + c_1 x_2 + c_2 x_2^2 + \cdots + c_n x_2^n &= f(x_2) \\
\vdots & & \vdots \\
x_n: & c_0 + c_1 x_n + c_2 x_n^2 + \cdots + c_n x_n^n &= f(x_n)
\end{array} \tag{4.4}$$

Das ist ein lineares Gleichungssystem mit dem Unbekanntenvektor $(c_0, c_1, \ldots, c_n)^\top$. Die Koeffizientenmatrix ist eine sogenannte *Vandermondematrix*

$$\begin{pmatrix}
1 & x_0 & x_0^2 & \ldots & x_0^n \\
1 & x_1 & x_1^2 & \ldots & x_1^n \\
\vdots & \vdots & \vdots & & \vdots \\
1 & x_n & x_n^2 & \ldots & x_n^n
\end{pmatrix}$$

Aus der Regularität folgt die eindeutige Lösbarkeit der Lagrangeschen Interpolationsaufgabe. (4.3) entspricht der Darstellung des Interpolationspolynoms $p(x)$ bezüglich der sogannten *Monombasis*

$$1, x, x^2, \ldots, x^n \tag{4.5}$$

dabei wird der Basisbegriff wie in der linearen Algebra verwendet. Z.B. lässt sich jeder Vektor $\vec{x} \in \mathbb{R}^3$ als Linearkombination dreier Basisvektoren darstellen:

$$\vec{x} \;=\; \begin{pmatrix} x_1 \\ x_2 \\ x_3 \end{pmatrix} \;=\; x_1 \begin{pmatrix} 1 \\ 0 \\ 0 \end{pmatrix} + x_2 \begin{pmatrix} 0 \\ 1 \\ 0 \end{pmatrix} + x_3 \begin{pmatrix} 0 \\ 0 \\ 1 \end{pmatrix}$$

Analog ist jedes Polynom $p(x)$ auch eindeutig festgelegt als Linearkombination

$$p(x) \;=\; c_0 e_0 + c_1 e_1 + \cdots + c_n e_n$$

etwa mit den Basisfunktionen

$$e_0 = e_0(x) \equiv 1, \quad e_1 = e_1(x) \equiv x, \quad e_2 = e_2(x) \equiv x^2, \dots \quad e_n = e_n(x) \equiv x^n \tag{4.6}$$

Im Prinzip kann man das Lagrange-Interpolationspolynom tatsächlich durch Lösen von (4.4) erhalten das ist aber i.a. nicht effizient und schlecht konditioniert. Für andere Möglichkeiten der Berechnung werden weitere Basisdarstellungen von Polynomen vom Grad n studiert. Andere Basen für den Raum der Polynome vom Grad n sind z.B. die Lagrange-Polynome

$$\varphi_i(x) := \frac{(x-x_0)(x-x_1)\cdots(x-x_{i-1})(x-x_{i+1})\cdots(x-x_n)}{(x_i-x_0)(x_i-x_1)\cdots(x_i-x_{i-1})(x_i-x_{i+1})\cdots(x_i-x_n)} \qquad i = 0, 1, \dots, n$$

oder die Newton-Polynome

$$1, (x-x_0), (x-x_0)(x-x_1), \dots \prod_{j=0}^{n-1}(x-x_j).$$

Bei der **Hermiteinterpolation** sind im Gegensatz zur Lagrangeinterpolation an den Interpolationsknoten auch mehr oder weniger hohe Ableitungswerte vorgegeben, an die das Interpolationspolynom von entsprechend höherem Grad angepasst werden soll. Z.B. an den Datensatz (x_0, f_0, f_0'), $(x_1; f_1)$, (x_2, f_2, f_2', f_2'') kann man ein Polynom vom Grad 5 anpassen, siehe Kapitel 4.7.

4.3 Die Lagrange - Polynome

Die $n+1$ Funktionen $\varphi_i(x),\ i = 0, \dots, n$

$$\varphi_i(x) := \frac{(x-x_0)(x-x_1)\cdots(x-x_{i-1})(x-x_{i+1})\cdots(x-x_n)}{(x_i-x_0)(x_i-x_1)\cdots(x_i-x_{i-1})(x_i-x_{i+1})\cdots(x_i-x_n)} = \frac{\prod_{j=0,j\neq i}^{n}(x-x_j)}{\prod_{j=0,j\neq i}^{n}(x_i-x_j)} \tag{4.7}$$
$$i = 0, 1, \dots, n.$$

sind offenbar Polynome vom Grad n und es gilt

$$\varphi_i(x_k) = \delta_{ik}, \quad i, k = 0, 1, \dots, n, \tag{4.8}$$

wobei δ_{ik} das Kroneckersymbol ist, d.h. $\delta_{ik} = 1$ ist für $i = k$ und 0 für $i \neq k$. (4.8) folgt sofort aus (4.7):

$$i = k \quad \varphi_i(x_i) = \frac{(x_i-x_0)(x_i-x_1)\cdots(x_i-x_{i-1})(x_i-x_{i+1})\cdots(x_i-x_n)}{(x_i-x_0)(x_i-x_1)\cdots(x_i-x_{i-1})(x_i-x_{i+1})\cdots(x_i-x_n)} = 1$$

da für $x = x_i$ der Zähler gleich dem Nenner wird. Für $i \neq k$ verschwindet ein Faktor des Zählers

$$(x_k-x_0)(x_k-x_1)\cdots(x_k-x_{i-1})(x_k-x_{i+1})\cdots(x_k-x_n)$$

Aufgrund von (4.8) folgt sofort, dass

$$p(x) = \sum_{i=0}^{n} f(x_i)\varphi_i(x) \tag{4.9}$$

das gesuchte Interpolationspolynom $p(x)$ zu den Interpolationsdaten $(x_i, f(x_i))$, $i = 0, 1, 2, \ldots, n$ ist. Denn für einen beliebigen Knotenpunkt x_k folgt

$$\begin{aligned} p(x_k) &= f(x_0)\underbrace{\varphi_0(x_k)}_{\delta_{0k}=0} + f(x_1)\underbrace{\varphi_1(x_k)}_{\delta_{1k}=0} + \cdots + \\ &\quad + f(x_k)\underbrace{\varphi_k(x_k)}_{\delta_{kk}=1} + \cdots + f(x_n)\underbrace{\varphi_n(x_k)}_{\delta_{nk}=0} = \\ &= f(x_k) \cdot 1 = f(x_k) \end{aligned} \tag{4.10}$$

Die $n+1$ Polynome (4.7) bilden eine Basis im Raum der Polynome vom Maximalgrad n, sie spannen diesen Raum auf: Jedes Polynom

$$p(x) = c_0 + c_1 x + \cdots + c_n x^n$$

kann man an den Knotenstellen x_0, $x_1, \ldots, x_n$ betrachten, wo es die Werte $p(x_0)$, $p(x_1), \ldots$, $p(x_n)$ annimmt, d.h. man kann es als Lagrangesches Interpolationspolynom zu dem Datensatz

$$(x_0, p(x_0)), \quad (x_1, p(x_1)), \quad \ldots, \quad (x_n, p(x_n))$$

auffassen und es daher aufgrund von (4.9) als Linearkombination

$$p(x) = p(x_0)\varphi_0(x) + p(x_1)\varphi_1(x) + \cdots + p(x_n)\varphi_n(x) \tag{4.11}$$

der $n+1$ Lagrange-Polynome schreiben. Jedes Polynom von Grad n ist also tatsächlich Linearkombination der $\varphi_i(x)$. Auch die Eindeutigkeit der Darstellung folgt unmittelbar: Wenn zwei Polynome an den Knotenstellen $x_0, \ldots, x_n$ dieselben Werte haben, wenn also die Gewichte in der Linearkombination (4.11) übereinstimmen, so müssen sie identisch sein andernfalls hätte das nichtverschwindende Differenzpolynom vom Grad n die $n+1$ Nullstellen $x_0, x_1, \ldots, x_n$ im Widerspruch zum Fundamentalsatz der Algebra.

Beispiel 4.3.1. Interpolationsddaten $(0, 0)$, $(\frac{\pi}{4}, \frac{\sqrt{2}}{2})$, $(\frac{\pi}{2}, 1)$

$$\begin{aligned} \varphi_0(x) &= \frac{(x-\frac{\pi}{4})(x-\frac{\pi}{2})}{(0-\frac{\pi}{4})(0-\frac{\pi}{2})} = \\ &= 1 - 1.909859317\ldots x + 0.8105694692\ldots x^2 \\ \varphi_1(x) &= \frac{(x-0)(x-\frac{\pi}{2})}{(\frac{\pi}{4}-0)(\frac{\pi}{4}-\frac{\pi}{2})} = \\ &= 2.546479089\ldots x - 1.621138938\ldots x^2 \\ \varphi_2(x) &= \frac{(x-0)(x-\frac{\pi}{4})}{(\frac{\pi}{2}-0)(\frac{\pi}{2}-\frac{\pi}{4})} = \\ &= -0.6366197724\ldots x + 0.8105694692\ldots x^2 \Rightarrow \\ p(x) &= 0 \cdot \varphi_0(x) + \frac{\sqrt{2}}{2}\varphi_1(x) + 1 \cdot \varphi_2(x) = \\ &= 1.16408357\ldots x - 0.3357488674\ldots x^2 \end{aligned}$$

Wenn man zu einer Knotenmenge $x_0, x_1, \ldots, x_n$ die Lagrange - Polynome $\varphi_i(x)$ schon aufgestellt hat, dann kann nach (4.9) sofort das Interpolationspolynom hingeschrieben werden. Wenn man zu einer bestimmten Knotenmenge $x_0, \ldots, x_n$ verschiedene Datensätze

$$\begin{array}{llll} (x_0, f_0^0), & (x_1, f_1^0), & \ldots, & (x_n f_n^0) \\ (x_0, f_0^1), & (x_1, f_1^1), & \ldots, & (x_n, f_n^1) \\ \vdots & & & \vdots \\ (x_0, f_0^r), & (x_1, f_1^r), & \ldots, & (x_n, f_n^r) \end{array}$$

interpoliert, dann ist offenbar das Arbeiten mit Lagrange - Polynomen optimal.

Ungünstig ist hingegen die Situation, wo man zunächst einen Datensatz (x_0, f_0), ..., (x_n, f_n) interpoliert hat, und dann bemerkt, dass die Genauigkeit des Interpolationspolynoms vom Grad n für den vorliegenden Zweck nicht ausreicht. Wenn man nun um die Genauigkeit zu steigern den Datensatz erweitert und mit Polynomen höheren Grades arbeitet, dann muss man bezüglich der neuen Knotenmenge die Lagrange - Polynome komplett neu berechnen. Für so eine Situation ist die Darstellung des Interpolationpolynom bezüglich *Newton - Polynome* optimal.

4.4 Newton - Polynome

Newton - Polynome bezüglich der Knotenmenge $x_0, x_1, \ldots, x_n$:

$$\begin{aligned} &1,\ (x - x_0),\ (x - x_0)(x - x_1),\ (x - x_0)(x - x_1)(x - x_2), \\ &\quad \ldots,\ (x - x_0)(x - x_1) \cdots (x - x_{n-1}) \end{aligned} \tag{4.12}$$

Jedes Polynom $p(x)$ vom Grad n lässt sich als Linearkombination dieser Basispolynome schreiben

$$\begin{aligned} p(x) \;=\; & p_0 \cdot 1 + p_1(x - x_0) + p_2(x - x_0)(x - x_1) + p_3(x - x_0)(x - x_1)(x - x_2) + \\ & + \cdots + p_n(x - x_0)(x - x_1) \cdots (x - x_{n-1}) \end{aligned} \tag{4.13}$$

Um das Interpolationspolynom zum Datensatz

$$(x_0, f_0), \quad (x_1, f_1), \quad \ldots, \quad (x_n, f_n)$$

aufzustellen, können die Gewichte p_i, $i = 0, 1, 2, \ldots n$ in der Linearkombination (4.13) nach und nach aus den Interpolationsbedingungen berechnet werden:

$x = x_0$ in (4.13)

$$f_0 \;=\; p(x_0) \;=\; p_0 \tag{4.14}$$

$x = x_1$ in (4.13)

$$\begin{aligned} f_1 &= p(x_1) \;=\; p_0 + p_1(x_1 - x_0) \\ f_1 &= f_0 + p_1(x_1 - x_0) \quad \Rightarrow \\ p_1 &= \frac{f_1 - f_0}{x_1 - x_0} \end{aligned} \tag{4.15}$$

$x = x_2$ in (4.13)

$$\begin{aligned} f_2 &= p(x_2) = p_0 + p_1(x_2 - x_0) + p_2(x_2 - x_0)(x_2 - x_1) \\ f_2 &= f_0 + \frac{f_1 - f_0}{x_1 - x_0}(x_2 - x_0) + p_2(x_2 - x_0)(x_2 - x_1) \end{aligned}$$

$$\begin{aligned} f_2 - f_1 + f_1 - f_0 - \frac{f_1 - f_0}{x_1 - x_0}(x_2 - x_0) &= p_2(x_2 - x_0)(x_2 - x_1) \\ \frac{f_2 - f_1}{x_2 - x_1} + \frac{(f_1 - f_0)(x_1 - x_0)}{(x_1 - x_0)(x_2 - x_1)} - \frac{(f_1 - f_0)(x_2 - x_0)}{(x_1 - x_0)(x_2 - x_1)} &= p_2(x_2 - x_0) \\ \frac{f_2 - f_1}{x_2 - x_1} - \frac{(f_1 - f_0)(x_0 - x_1) + (f_1 - f_0)(x_2 - x_0)}{(x_1 - x_0)(x_2 - x_1)} &= p_2(x_2 - x_0) \\ \frac{f_2 - f_1}{x_2 - x_1} - \frac{f_1 - f_0}{x_1 - x_0} &= p_2(x_2 - x_0) \\ \frac{\frac{f_2 - f_1}{x_2 - x_1} - \frac{f_1 - f_0}{x_1 - x_0}}{x_2 - x_0} &= p_2 \qquad (4.16) \\ &\vdots \end{aligned}$$

Es ist offensichtlich, dass bei dieser Vorgangsweise die Menge der Interpolationsdaten problemlos erweitert werden kann.

(4.14), (4.15) und (4.17) legt die Definition der sogenannten **dividierten Differenzen** nahe:

Nullte dividierte Differenz

$$f[x_0] := f(x_0) = f_0$$

Erste dividierte Differenz

$$f[x_0, x_1] := \frac{f(x_1) - f(x_0)}{x_1 - x_0} = \frac{f_1 - f_0}{x_1 - x_0} = \frac{f[x_1] - f[x_0]}{x_1 - x_0}$$

Zweite dividierte Differenz

$$f[x_0, x_1, x_2] := \frac{\frac{f_2 - f_1}{x_2 - x_1} - \frac{f_1 - f_0}{x_1 - x_0}}{x_2 - x_0} = \frac{f[x_1, x_2] - f[x_0, x_1]}{x_2 - x_0}$$

In analoger Weise ergibt sich offensichtlich

Dritte dividierte Differenz

$$f[x_0, x_1, x_2, x_3] := \frac{\frac{\frac{f_3 - f_2}{x_3 - x_2} - \frac{f_2 - f_1}{x_2 - x_1}}{x_3 - x_1} - \frac{\frac{f_2 - f_1}{x_2 - x_1} - \frac{f_1 - f_0}{x_1 - x_0}}{x_2 - x_0}}{x_3 - x_0} = \frac{f[x_1, x_2, x_3] - f[x_0, x_1, x_2]}{x_3 - x_0}.$$

Dies lässt folgendes Bildungsgesetz für die k-te dividierte Differenz $f[x_0, x_1, \ldots, x_k]$ erkennen: Im Nenner steht die Differenz $x_k - x_0$, im Zähler steht eine Differenz von (k–1)-ten dividierten Differenzen, und zwar wird von der $(k-1)$-ten Differenz, bei der sämtliche Knoten um *1 nach rechts geshiftet* sind (d.h. die Knoten $x_1, \ldots, x_k$ betrachtet werden) die $(k-1)$-te dividierte Differenz bezüglich der *ungeshifteten* Knoten $x_0, x_1, \ldots, x_{k-1}$ abgezogen, es ergibt sich folgende rekursive Definition:

$$f[x_0, x_1, \ldots, x_k] := \frac{f[x_1, \ldots, x_k] - f[x_0, \ldots, x_{k-1}]}{x_k - x_0} \qquad (4.17)$$

Bemerkung: Der Zusammenhang zwischen den höheren dividierten Differenzen und den höheren Ableitungen ist offenkundig. Wir nehmen die spezielle Knotenlage $x_1 = x_0 + h$, $x_2 = x_0 + 2h$, $x_3 = x_0 + 3h, \ldots$ an und betrachten kleine h-Werte. Dann haben wir offenbar (streng gelten die folgenden Beziehungen nur für $h \to 0$):

$$\begin{aligned} f[x_0, x_1] &= \frac{f_1 - f_0}{h} \approx f'(x_0) \\ f[x_0, x_1, x_2] &= \frac{f[x_1, x_2] - f[x_0, x_1]}{2h} \approx \frac{1}{2}\frac{f'(x_1) - f'(x_0)}{h} \approx \frac{1}{2}f''(x_0) \\ f[x_0, x_1, x_2, x_3] &= \frac{f[x_1, x_2, x_3] - f[x_0, x_1, x_2]}{3h} \approx \frac{1}{3}\frac{\frac{1}{2}f''(x_1) - \frac{1}{2}f''(x_0)}{h} \approx \frac{1}{2 \cdot 3}f'''(x_0) \end{aligned}$$

u.s.w. also allgemein gilt

$$f[x_0, x_1, \ldots, x_k] \approx \frac{1}{k!}f^{(k)}(x_0) \tag{4.18}$$

Aufgrund von (4.14), (4.15) und (4.17) folgt

$$\begin{aligned} p(x) &= f[x_0] + f[x_0, x_1](x - x_0) + \\ &\quad + f[x_0, x_1, x_2](x - x_0)(x - x_1) + \cdots + f[x_0, x_1, \ldots, x_n](x - x_0) \cdots (x - x_{n-1}) \end{aligned} \tag{4.19}$$

d.h. die Gewichte $p_0, \ldots, p_n$ des Interpolationspolynom $p(x)$ in der Darstellung (4.13) lassen sich als dividierte Differenzen rekursiv berechnen.

4.5 Berechnung von Werten des Interpolationspolynoms

Wenn nur einzelne Werte des Interpolationspolynoms benötigt werden, ist eine Möglichkeit natürlich zunächst das Polynom aufzustellen und dann für die Stelle $x = \bar{x}$, für die man den Wert des Interpolationspolynoms wissen möchte, das Polynom auszuwerten. Für die Auswertung von (4.13) gibt es einen Hornerartigen[1)] Algorithmus:

$$p(\bar{x}) = \Big(\cdots\Big(p_n(\bar{x} - x_{n-1}) + p_{n-1}\Big)\Big(\bar{x} - x_{n-2}\Big) + \cdots + p_1\Big)\Big(\bar{x} - x_0\Big) + p_0 \tag{4.20}$$

Es ist i.a. jedoch ökonomischer, nicht zunächst das Interpolationspolynom aufzustellen und anschließend für $x = \bar{x}$ auszuwerten, sondern mit dem sogenannten **Neville-Schema** direkt $p(\bar{x})$ zu berechnen. Nur wenn man ein Interpolationspolynom an *vielen* Stellen $\bar{x}_i$ auswerten möchte, ist der Weg über (4.19) und (4.20) effizienter.

Nevilleschema: Zunächst Annahme, dass die Interpolationsknoten aufsteigend indiziert sind: $x_0 < x_1 < \cdots < x_n$. Grundlage für das Nevilleschema ist die Identität

$$p_{i,i+k}(x) = \frac{(x - x_i)p_{i+1,i+k}(x) - (x - x_{i+k})p_{i,i+k-1}(x)}{x_{i+k} - x_i}, \tag{4.21}$$

[1)] Hornerschema zur Auswertung von Polynomen: Statt ein Polynom z.B. von Grad 3 direkt $p(\bar{x}) = c_0 + c_1\bar{x} + c_2\bar{x}^2 + c_3\bar{x}^3$ auszuwerten, wertet man $p(\bar{x}) = ((c_3\bar{x} + c_2)\bar{x} + c_1)\bar{x} + c_0$ aus, was Rechenoperationen erspart. Konkret bei Polynomgrad 3: statt 5 Multiplikationen nur 3 Multiplikationen.

wobei $p_{i,i+k}(x)$ das Interpolationspolynom zu dem Datensatz $(x_i, f_i), \ldots, (x_{i+k}, f_{i+k})$ bezeichnet und $p_{i+1,i+k}(x)$ bzw. $p_{i,i+k-1}(x)$ die Interpolationspolynome zu den Datensätzen $(x_{i+1}, f_{i+1}), \ldots, (x_{i+k}, f_{i+k})$ bzw. $(x_i, f_i), \ldots, (x_{i+k-1}, f_{i+k-1})$.

$p_{i,i+k}(x)$ hat offenbar den Grad k, während $p_{i+1,i+k}(x)$ und $p_{i,i+k-1}(x)$ den Grad $k-1$ haben. (4.21) ist eine Beziehung zwischen den Interpolationspolynomen $p_{i,i+k}(x)$, $p_{i,i+k-1}(x)$ und $p_{i+1,i+k}(x)$. Man kann aber, wenn man sich diese Polynome für $x = \bar{x}$ ausgewertet denkt, (4.21) auch als eine Beziehung zwischen den Werten dieser Polynome ansehen:

$$p_{i,i+k}(\bar{x}) \quad = \quad \frac{(\bar{x} - x_i)p_{i+1,i+k}(\bar{x}) - (\bar{x} - x_{i+k})p_{i,i+k-1}(\bar{x})}{x_{i+k} - x_i} \tag{4.22}$$

Wir erläutern dieses Schema zunächst für $n = 2$, d.h. wir schreiben sämtliche Neville-Identitäten, die für den Aufbau des Interpolationspolynoms $p_{0,2}$ vom Grad 2 nötig sind, in Dreiecksgestalt an:

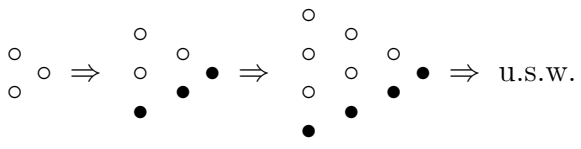
$$\begin{array}{l|l|l|l} x_0 & f_0 \equiv p_{0,0}(x)\searrow & & \\ & & \frac{(x-x_0)p_{1,1}(x)-(x-x_1)p_{0,0}(x)}{x_1-x_0} = p_{0,1}(x)\searrow & \\ x_1 & f_1 \equiv p_{1,1}(x)\nearrow\searrow & & \frac{(x-x_0)p_{1,2}(x)-(x-x_2)p_{0,1}(x)}{x_2-x_0} = p_{0,2}(x) \\ & & \frac{(x-x_1)p_{2,2}(x)-(x-x_2)p_{1,1}(x)}{x_2-x_1} = p_{1,2}(x)\nearrow & \\ x_2 & f_2 \equiv p_{2,2}(x)\nearrow & & \end{array}$$

Denkt man nun diese Identitäten nicht für die Polynome $p_{0,0}$, $p_{1,1}$, $p_{2,2}$, $p_{0,1}$, $p_{1,2}$, $p_{0,2}$ hingeschrieben, sondern für die jeweiligen Werte $p_{0,0}(\bar{x})\,(= f_0)$, $p_{1,1}(\bar{x})\,(= f_1)$, $p_{2,2}(\bar{x})\,(= f_2)$, $p_{0,1}(\bar{x})$, $p_{1,2}(\bar{x})$, $p_{0,2}(\bar{x})$, so kann man dieses Dreiecksschema rein zahlenmässig auf einem Computer ablaufen lassen und so den gewünschten Wert $p_{0,2}(\bar{x})$ berechnen, ohne das Polynom $p_{0,2}$ tatsächlich aufzustellen.

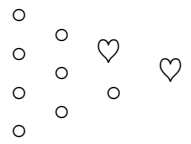
Das Nevilleschema ist ein Dreiecksschema.

$$\begin{array}{llll} x_0 & f_0 & =: & p_{0,0} \\ & & & \searrow \\ & & & p_{0,1} \quad \searrow \\ x_1 & f_1 & =: & p_{1,1} \quad \nearrow \quad p_{0,2} \quad \searrow \\ & & & \searrow \quad \nearrow \\ & & & p_{1,2} \\ x_2 & f_2 & =: & p_{2,2} \qquad\qquad \ddots \\ & & & \qquad\qquad\qquad p_{0,n} \\ \vdots & \vdots & & \qquad\qquad\qquad \iddots \\ & & & \searrow \\ x_{n-1} & f_{n-1} & =: & p_{n-1,n-1} \quad \nearrow \\ & & & \searrow \\ & & & p_{n-1,n} \quad \nearrow \\ x_n & f_n & =: & p_{n,n} \end{array}$$

Die Schleifenorganisation ist so, dass das Dreieck in folgender Weise aufgebaut wird:



(• steht für neu dazugekommene $p_{i,i+k}$) Das ist für viele Anwendungen sehr wichtig: Durch Vergleich von $p_{0,n}$ und $p_{0,n+1}$



(die ♡'s werden verglichen) schafft man sich einen Eindruck von der Approximationsqualität (Genauigkeit) von $p_{0,n}$. Falls diese für den vorliegenden Zweck nicht ausreicht, kann man versuchen durch Erweiterung des Datensatzes um (x_{n+1}, f_{n+1}) und Berechnung von $p_{0,n+1}$ die Genauigkeit zur steigern. D.h. dieser Aufbau des Nevilleschemas ermöglicht in sehr ökonomischer Weise die Erreichung eines bestimmten, vorgegebenen Genauigkeitsniveaus. Sobald dieses ereicht ist, bricht man den Rechenvorgang ab.

4.6 Interpolationsfehler

Es ist klar, dass der Interpolationsfehler $p(x) - f(x)$ für $x \neq x_i$ beliebig groß werden kann, wenn man keine einschränkenden Voraussetzungen an f stellt.

Der folgende Satz liefert für hinreichend glatte Funktionen f eine Darstellung des bei der Polynominterpolation auftretenden Fehlers.

Satz 4.6.1. Ist f $(n+1)$-mal stetig differenzierbar, so gibt es zu jedem x eine Zahl ϑ aus dem kleinsten Intervall $I[x_0, \ldots, x_n; x]$, das alle x_i und das x enthält, sodass für das Interpolationspolynom zum Datensatz $(x_0, f_0), \ldots, (x_n, f_n)$ die Beziehung

$$p(x) - f(x) \;=\; -\frac{f^{(n+1)}(\vartheta)}{(n+1)!}\omega(x) \tag{4.23}$$

gilt mit

$$\omega(x) \;:=\; (x - x_0)(x - x_1)\cdots(x - x_n) \tag{4.24}$$

Beweis: Angenommen $x \neq x_i$, da für $x = x_i$ (4.23) trivial erfüllt. Dann ist $\omega(x) \neq 0$ ist und es existiert eine Größe K, sodass

$$f(x) - p(x) - K\omega(x) \;=\; 0$$

ist. $K = \frac{f(x)-P(x)}{\omega(x)}$ ist wohldefiniert, da der Nenner $\omega(x)$ nicht verschwindet. Mit dieser der Stelle x entsprechenden Größe K bilden wir jetzt die Funktion

$$F(\tau) \;:=\; f(\tau) - p(\tau) - K\omega(\tau), \quad \tau \in I[x_0, \ldots, x_n; x].$$

$F(\tau)$ besitzt in $I[x_0, \ldots, x_n; x]$ mindestens die $n+2$ Nullstellen $x_0, \ldots, x_n; x$. Nach dem Satz von Rolle besitzt dann $\frac{dF(\tau)}{d\tau}$ in $I[x_0, \ldots, x_n; x]$ mindestens $n+1$ Nullstellen, vergleiche Abb. 4.4.

Daraus folgt wieder nach dem Satz von Rolle, dass $\frac{d^2F(\tau)}{d\tau^2}$ mindestens n Nullstellen besitzt, ... und schließlich dass $\frac{d^{n+1}F(\tau)}{d\tau^{n+1}}$ mindestens eine Nullstelle $\vartheta \in I[x_0, \ldots, x_n; x]$ besitzt. Da aber die $(n+1)$-te Ableitung des Interpolationspolynoms $p(x)$ vom Grad n verschwindet, haben wir:

$$\frac{d^{n+1}F(\vartheta)}{d\tau^{n+1}} \;=\; f^{(n+1)}(\vartheta) - K(n+1)! \;=\; 0 \tag{4.25}$$

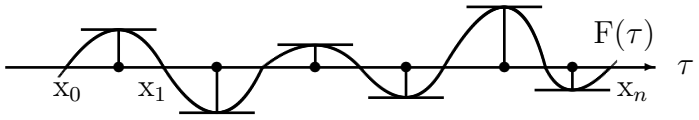


Abbildung 4.4: Nullstellen von $\frac{dF(\tau)}{d\tau}$

Die dabei benützte Aussage $\omega^{(n+1)}(x) = (n+1)!$ folgt leicht durch Induktion:

$$\begin{aligned} n=1: \quad \omega(x) &= (x-x_0)(x-x_1) \\ \omega'(x) &= (x-x_0)+(x-x_1) \\ \omega''(x) &= 1+1=2=2! \end{aligned}$$

Induktionsannahme: für $\omega(x) = (x-x_0)\cdots(x-x_n)$ gilt $\omega^{(n+1)}(x) = (n+1)!$ Betrachte nun

$$\begin{aligned} \omega(x) &= (x-x_0)(x-x_1)\cdots(x-x_{n+1}) \quad \Rightarrow \\ \omega'(x) &= (x-x_1)(x-x_2)\cdots(x-x_{n+1}) + \\ &\quad +(x-x_0)(x-x_2)\cdots(x-x_{n+1})+\cdots+ \\ &\quad +(x-x_0)(x-x_1)\cdots(x-x_n) \quad \leftarrow \quad (n+2) \text{ Summanden} \end{aligned} \tag{4.26}$$

Die $n+2$ Summanden rechts in (4.26) entstehen dadurch, dass man auf alle möglichen Arten aus $(x-x_0)\cdots(x-x_{n+1})$ jeweils einen Faktor streicht. Differenziert man (4.26) nun $n+1$ mal, so erhält man aufgrund der Induktionsannahme

$$\begin{aligned} \omega^{(n+2)}(x) &= \underbrace{(n+1)!+(n+1)!+\cdots+(n+1)!}_{n+2 \text{ Summanden}} = \\ &= (n+2)(n+1)! = (n+2)!, \end{aligned}$$

sodass der Induktionsbeweis abgeschlossen ist. Aus (4.25) folgt

$$K = \frac{f^{n+1}(\vartheta)}{(n+1)!}$$

woraus die zu (4.23) äquivalente Behauptung

$$f(x)-p(x) = K\omega(x) = \frac{f^{(n+1)}(\vartheta)}{(n+1)!}\omega(x)$$

folgt. □

Aus der Fehlerdarstellung (4.23), die man i.a. nicht direkt verwenden kann, da ϑ unbekannt ist, folgt sofort die Abschätzung

$$|p(x)-f(x)| \leq |\omega(x)|\frac{M_{n+1}}{(n+1)!} \tag{4.27}$$

wenn man über eine Schranke M_{n+1} für $|f^{(n+1)}(x)|$, $x \in I[x_0,\ldots,x_n;x]$ verfügt.

Beispiel 4.6.2. $f(x) = \sin x$, $x_0 = 0$, $x_1 = \frac{\pi}{6}$, $x_2 = \frac{\pi}{4}$, $x_3 = \frac{\pi}{2}$ und $x = \frac{\pi}{5}$, das Polynom $p(x)$ hat den Grad $n = 3$ und $M_{n+1} = 1$:

$$\boxed{\left|p\left(\frac{\pi}{5}\right) - \sin\left(\frac{\pi}{5}\right)\right| \leq \left|\frac{\pi}{5}\left(\frac{\pi}{5} - \frac{\pi}{6}\right)\left(\frac{\pi}{5} - \frac{\pi}{4}\right)\left(\frac{\pi}{5} - \frac{\pi}{2}\right)\right| \frac{1}{4!} = 0.002435227276\ldots}$$

Der tatsächliche Fehler

$$\boxed{p\left(\frac{\pi}{5}\right) - \sin\left(\frac{\pi}{5}\right) = -0.00025624\ldots}$$

ist etwa um einen Faktor 10 kleiner, siehe Beispiel 4.1.1.

Analog zu (4.23) *gibt es auch Aussagen über die Approximation der Ableitungen* $p^{(k)}(x) - f^{(k)}(x)$*:*

$$p^{(k)}(x) - f^{(k)}(x) = -\sum_{i=0}^{k} \frac{k!}{(n+k-i+1)!i!}\omega^{(i)}(x) f^{n+k-i+1}(\vartheta_i) \tag{4.28}$$

$$\vartheta_i \in I[x_0, \ldots, x_n; x]$$

Beweis. Siehe Literatur. □

f soll auf $[a, b]$ durch Polynome vom Grad n interpoliert werden und zwar, siehe Abbildung 4.5

1) von einem Polynom zu den Daten $(x_0, f_0), \ldots, (x_n, f_n)$ mit den äquidistanten Knoten

$$x_0 = a,\ x_1 = a + \frac{b-a}{n},\ x_2 = a + 2\frac{b-a}{n},\ \ldots,\ x_n = b$$

(Knotenabstand $h = \frac{b-a}{n}$)

2) von zwei Polynomen zu den Datensätzen

$$(x_0, f_0),\ \ldots,\ (x_n, f_n) \quad \text{und} \quad (x_n, f_n),\ \ldots,\ (x_{2n}, f_{2n})$$

mit den äquidistanten Knoten

$$x_0 = a,\ x_1 = a + \frac{b-a}{2n},\ \ldots\ x_{2n} = b$$

(Knotenabstand $h = \frac{b-a}{2n}$)

3) von drei Polynomen zu den Datensätzen $(x_0, f_0), \ldots, (x_n, f_n)$ und $(x_n, f_n), \ldots, (x_{2n}, f_{2n})$ und $(x_{2n}, f_{2n}), \ldots, (x_{3n}, f_{3n})$ mit den äquidistanten Knoten $x_0 = a$, $x_1 = a + \frac{b-a}{3n}$, …, $x_{3n} = b$. (Knotenabstand $h = \frac{b-a}{3n}$)

4) Analoge Konstruktion fortsetzen, wobei man nach und nach immer mehr und mehr Polynome stückweise aneinanderreiht.

Wegen $|x - x_i| \leq const \cdot h,\ i = kn, kn+1, \ldots, (k+1)n$ und $k \in \mathbb{N}_0$, wobei x beliebig aus $[a, b]$ ist und die Knoten $x_{kn}, \ldots, x_{(k+1)n}$ Interpolationsknoten bezüglich der stückweisen polynomialen Interpolation mit Polynomen vom Grad n sind, hat man $|\omega(x)| \leq \text{const} \cdot h^{n+1}$. Aus (4.23) folgt somit sofort

$$|p(x) - f(x)| \leq \text{const} \cdot h^{n+1} \quad \text{oder}$$

$$p(x) - f(x) = O(h^{n+1}) \tag{4.29}$$

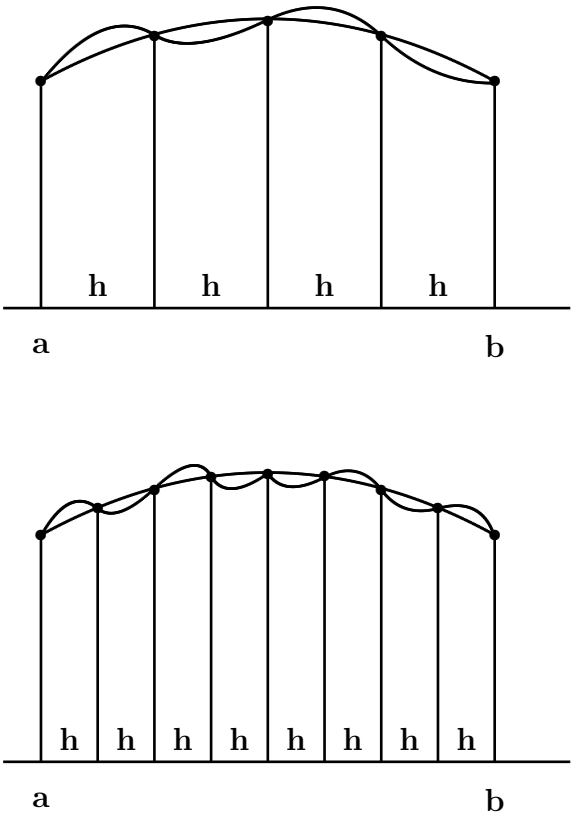


Abbildung 4.5: ein Polynom, zwei Polynome, ...

Für $h \to 0$ (immer feinere Gitter, gleichzeitig immer mehr Polynome vom festgehaltenen Grad n bezüglich der stückweisen polynomialen Interpolationsfunktion) ergibt sich somit für $(n+1)$ mal differenzierbares f Konvergenz mit der Ordnung $n+1$.

Bezüglich der Approximationsqualität von Ableitungen $p^{(k)}(x)$ des Interpolationspolynoms verglichen mit $f^{(k)}(x)$ ergibt sich aus (4.28) sofort:

$$p^{(k)}(x) - f^{(k)}(x) \;=\; O(h^{n+1-k}) \quad k = 0, 1, \ldots, n, \tag{4.30}$$

denn für die höchste rechts in (4.28) auftrefende Ableitung von ω hat man[2)]

$$\omega^{(k)}(x) \;=\; O(h^{n+1-k}).$$

Skalierungsdiskussion: Von (4.23), (4.27) und (4.29) folgt, dass h-Potenzen den Interpolationsfehler klein machen, aber erst für $h < 1$. Dies widerspricht auf den ersten Blick der Anschauung, siehe Abbildung 4.6. f sei ein zeitabhängiger Vorgang, der zwischen den Zeitpunkten x_0 und x_1

[2)] $\omega(x) = (x-x_0)(x-x_1)\cdots(x-x_n) = O(h^{n+1})$, da ω $n+1$ Faktoren $x - x_i$ der Größenordnung h hat.

$$\begin{aligned}\omega'(x) \;&=\; (x-x_1)\cdots(x-x_n) + (x-x_0)(x-x_2)\cdots(x-x_n) + \cdots \\ &\quad +(x-x_0)\cdots(x-x_{n-1}) \;=\; O(h^n),\end{aligned}$$

da jeder Summand in ω' n Faktoren $x - x_i$ der Größenordnung h hat.

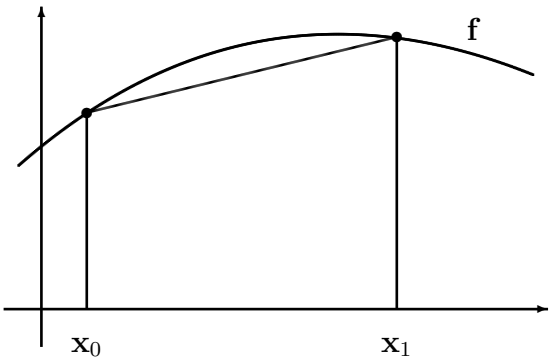


Abbildung 4.6: zeitabhängiger Vorgang interpoliert

durch ein interpolierendes Geradenstück approximiert werden soll. Ob der Stützstellenabstand $h = x_1 - x_0$ zahlenmässig klein ($\ll 1$) oder groß ($\gg 1$) ist, hängt von der gewählten Zeiteinheit (Jahre oder Sekunden) ab. Wie aus der Skizze ersichtlich ist, ist die Approximationsqualität $p(x) - f(x)$ von der Zeitskalierung aber unabhängig. Dieser scheinbare Widerspruch löst sich sofort, wenn man die Fehlerformel (4.23) bzw. (4.27) betrachtet. Bei einer Umskalierung $x = c\xi$ ändert sich nicht nur h und damit entsprechend auch ω, sondern diese Änderung wird durch die inneren Ableitungen (Kettenregel beim Differenzieren) bei der Berechnung von $\frac{d^{n+1}f(c\cdot\xi)}{d\xi^{n+1}}$ kompensiert.

Zusammenhang zum Taylorpolynom: (vgl. Abb. 4.7)

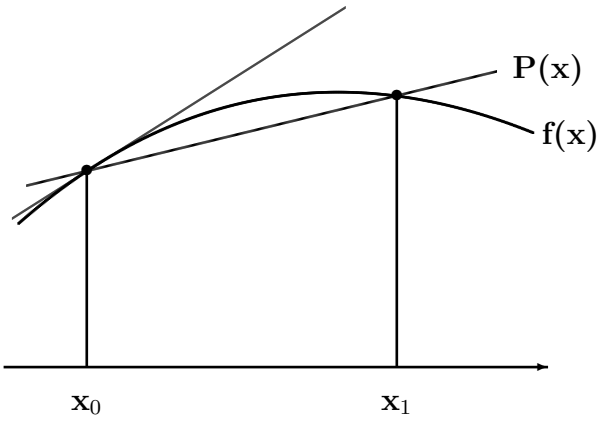


Abbildung 4.7: Sehne wird zur Tangente

$p(x)$ ... lineares Interpolationspolynom, das (x_0, f_0) und (x_1, f_1) interpoliert. Lässt man den zweiten Knoten x_1 gegen x_0 wandern, so konvergiert $p(x)$ offenbar gegen die Tangente $f(x_0) + (x - x_0)f'(x_0)$, d.h. gegen das Taylorpolynom vom Grad 1 bezüglich f, entwickelt um x_0. Man kann zeigen, dass ganz allgemein ein Interpolationspolynom $p(x)$ vom Grad n zum Datensatz $(x_0, f_0), \ldots, (x_n, f_n)$ gegen das

Taylorpolynom

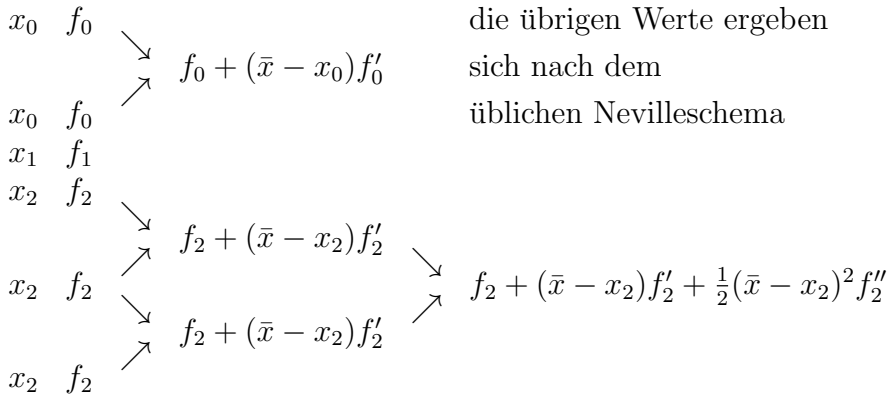
$$f(x_0) + f'(x_0)(x - x_0) + \frac{1}{2!}f''(x_0)(x - x_0)^2 + \cdots + \frac{1}{n!}f^{(n)}(x - x_0)^n \tag{4.31}$$

strebt, wenn sämtliche Interpolationsknoten x_i gegen x_0 konvergieren, siehe die Darstellung (4.19) der Interpolationspolynome und (4.18)). Parallel dazu strebt das Interpolationsrestglied (4.23) gegen das entsprechende Taylorrestglied:

$$\frac{f^{(n+1)}(\vartheta)}{(n+1)!}(x - x_0)(x - x_1)\cdots(x - x_n) \quad \to \quad \frac{f^{(n+1)}(\vartheta)}{(n+1)!}(x - x_0)^{n+1}$$

4.7 Hermiteinterpolation

Im Gegensatz zur Lagrangeinterpolation sind an den Interpolationsknoten auch mehr oder weniger hohe Ableitungswerte vorgeschrieben, an die das Interpolationspolynom von entsprechend höherem Grad angepasst werden soll. Z.B. an den Datensatz (x_0, f_0, f_0'), $(x_1; f_1)$, (x_2, f_2, f_2', f_2'') kann man ein Polynom vom Grad 5 anpassen. Um Werte des Interpolationspolynoms für $x = \bar{x}$ zu ermitteln, muss das Nevilleschema modifiziert werden



Begründung für dieses modifizierte Nevilleschema:

- ein Element $p_{i,i+k}$ des Nevilleschemas ist das Interpolationspolynom zum Teildatensatz (x_i, f_i), $\ldots$, (x_{i+k}, f_{i+k})
- Wenn die Knoten eines Interpolationspolynoms gegen einen Punkt konvergieren, dann konvergieren die Interpolationspolynome gegen die entsprechenden Taylorpolynome. Wenn also z.B. $x_1 = x_0 + h$ gegen x_0 konvergiert, so konvergiert $p_{0,1}$ (Wert des linearen Interpolationspolynoms zum Datensatz (x_0, f_0), $(\tilde{x}_1, \tilde{f}_1)$ für $x = \bar{x}$) gegen den entsprechenden Wert $f_0 + (\bar{x} - x_0)f_0'$ des Taylorpolynoms vom Grad 1.

Auch bezüglich der Fehlerformel (4.23) ist sofort klar, wie sie aufgrund des oben beschriebenen Grenzprozesses zu modifizieren ist: Etwa für den Datensatz

$$(x_0, f_0, f_0'), \quad (x_1, f_1), \quad (x_2, f_2, f_2', f_2'')$$

hat man

$$p(x) - f(x) \;=\; -\frac{f^{(6)}(\vartheta)}{6!}\omega(x)$$

mit $\omega(x) = (x - x_0)^2(x - x_1)(x - x_2)^3$ oder für einen Datensatz

$$(x_0, f_0, f_0'), \quad (x_1, f_1, f_1'), \quad \ldots, \quad (x_n, f_n, f_n')$$

hat man

$$\begin{aligned} p(x) - f(x) &= -\frac{f^{(2n+2)}(\vartheta)}{(2n+2)!}\omega^2(x) \qquad p(x) \text{ hat Grad } 2n+1 \\ \omega^2(x) &= (x - x_0)^2(x - x_1)^2 \cdots (x - x_n)^2 \end{aligned} \tag{4.32}$$

und

$$p(x) - f(x) = O(h^{2n+2}) \tag{4.33}$$

für äquidistante Knoten $x_i - x_{i-1} = h$.

Beispiel 4.7.1. $f(x) = \sin x$,

$$\begin{aligned} (x_0, f_0, f_0') &= (0, \sin(0), \cos 0) = (0, 0, 1) \\ (x_1, f_1, f_1') &= (\tfrac{\pi}{4}, \sin(\tfrac{\pi}{4}), \cos\tfrac{\pi}{4}) = (\tfrac{\pi}{4}, \tfrac{\sqrt{2}}{2}, \tfrac{\sqrt{2}}{2}) \end{aligned}$$

Nevilleschema bezüglich $x = \bar{x} = \frac{\pi}{5}$

$$\begin{array}{cc} 0 & 0 \\ 0 & 0 \\ \frac{\pi}{4} & \frac{\sqrt{2}}{2} \\ \frac{\pi}{4} & \frac{\sqrt{2}}{2} \end{array} \quad \begin{array}{c} 0.6283185307\ldots \\ 0.5656854249\ldots \\ 0.5960347077\ldots \end{array} \quad \begin{array}{c} 0.5782120461\ldots \\ 0.5877852523\ldots \end{array} \quad 0.5876142901\ldots$$

$$\boxed{p(\bar{x}) - \sin(\bar{x}) = -0.000170962194\ldots}$$

Die Fehlerschranke (4.32)

$$\boxed{\left|p\left(\frac{\pi}{5}\right) - \sin\left(\frac{\pi}{5}\right)\right| \le \frac{1}{4!}\left(\frac{\pi}{5}\right)^2\left(\frac{\pi}{5} - \frac{\pi}{4}\right)^2 = 0.0004058712127\ldots}$$

verglichen mit dem tatsächlichen Fehler ist also etwa um einen Faktor 4 zu pessimistisch.

4.8 Bestapproximation

Aufgabenstellung: Gegeben ist eine Funktion f aus einem bestimmten Funktionenraum Φ (etwa $\Phi = C^0[a, b]$ = Menge der auf $[a, b]$ stetigen Funktionen, oder $\Phi = C^2[0, \infty)$ = Menge der auf $[0, \infty)$ zweimal stetig differenzierbaren Funktionen oder ...).

Weiters betrachtet man einen endlichdimensionalen Teilraum $\Gamma \subset \Phi$. *Endlichdimensional* heißt hier, dass die Funktionen $g \in \Gamma$ durch endlich viele Parameter charakterisiert werden können (z.B. rationale Funktionen, also Quotienten von zwei Polynomen)

$$g(x) = \frac{c_0 + c_1 x + c_2 x^2}{d_0 + d_1 x + d_2 x^2 + d_3 x^3};$$

Γ wäre hier 7-dimensional, da jedes $g \in \Gamma$ durch die sieben Parameter $c_0, c_1, c_2,\ d_0, d_1, d_2, d_3$, festgelegt werden kann. Sehr oft werden hier *lineare* endlichdimensionale Teilräume Γ betrachtet, bei denen jedes $g \in \Gamma$ als endliche Linearkombination von Basisfunktionen geschrieben werden kann, Die Parameter, die $g \in \Gamma$ charakterisieren, sind dann die Gewichte dieser Linearkombination (z.B. Γ ist der Raum der Polynome vom Maximalgrad 3: Γ ist dann 4-dimensional, jedes $g \in \Gamma$ lässt sich schreiben als $g(x) = c_0 + c_1 x + c_2 x^2 + c_3 x^3$, also als Linearkombination der Basisfunktionen $1, x, x^2, x^3$, die Γ aufspannen, mit den Gewichten c_0, c_1, c_2, c_3)

Weiters betrachtet man in Φ und Γ eine bestimmte Norm. Z.B wenn Φ ein Funktionenraum ist mit Funktionen f die auf $[a, b]$ definiert sind, die bf Maximumnorm

$$\|f\|_\infty \;=\; \max_{x\in[a,b]} |f(x)| \tag{4.34}$$

oder die *L_p-Norm*

$$\|f\|_p \;=\; \left(\int_a^b |f(x)|^p dx\right)^{\frac{1}{p}} \quad p \geq 1 \tag{4.35}$$

(Spezialfall $p = 2 \ldots$ ist die uuklidischen Norm) oder außer der euklidischen Norm noch **Skalarproduktnormen**

$$\|f\|_G \;=\; \left(\int_a^b f^2(x) G(x) dx\right)^{\frac{1}{2}} \tag{4.36}$$

mit der *Gewichtsfunktion* $G(x)$.

Die Bestapproximierende $g^* \in \Gamma$ zu $f \in \Phi$ ist nun jenes Element g^* aus Γ, für das gilt

$$\begin{array}{lll} \|g^* - f\| & & \text{ist minimal, d.h.} \\ \|g^* - f\| & \leq & \|g - f\| \quad \text{für alle } g \in \Gamma. \end{array} \tag{4.37}$$

Je nachdem, welche Normdefinition (vgl. (4.34)-(4.36)) man in (4.37) zugrunde legt, führt das auf sehr verschiedene Aufgabenstellungen aus der Approximationstheorie.

Z.B.: Wenn man Skalarproduktnormen (4.36) oder (4.35) mit $p = 2$ (Euklidische Norm ist eine Skalarproduktnorm mit $G(x) \equiv 1$) zugrunde legt, führt das in das weite Feld der **Fourierreihen**. Dabei kommt die Grundidee der Funktionalanalysis – einfache geometrische Konzepte aus dem Endlichdimensionalen auf unendlichdimensionale Funktionenräume zu übertragen – besonders deutlich zum Ausdruck:

Endlichdimensional, z.B. $\mathbb{R}^3$:

1) Skalarprodukt:

$$\begin{aligned} \langle a, b\rangle \;&=\; a_1 b_1 + a_2 b_2 + a_3 b_3 \;= \\ &=\; \sum_{i=1}^{3} a_i b_i \end{aligned}$$

2) Euklidische Norm:

$$\|a\|_2 \;=\; \langle a, a\rangle^{\frac{1}{2}} \;=\; \sqrt{\sum_{i=1}^{3} a_i^2}$$

3) Orthonormale Basis:

$$e_1 = \begin{pmatrix} 1 \\ 0 \\ 0 \end{pmatrix}, \quad e_2 = \begin{pmatrix} 0 \\ 1 \\ 0 \end{pmatrix}, \quad e_3 = \begin{pmatrix} 0 \\ 0 \\ 1 \end{pmatrix}$$

4) Orthonormalitätsrelationen bezügl. der Basiselemente:

$$\langle e_i, e_k \rangle \; = \; \delta_{ik} \quad i,k = 1,2,3$$

5) Darstellung von a:

$$\begin{aligned} a &= a_1 e_1 + a_2 e_2 + a_3 e_3 \; = \\ &= \langle a, e_1 \rangle e_1 + \langle a, e_2 \rangle e_2 + \langle a, e_3 \rangle e_3 \end{aligned}$$

Funktionenraum, z.B. $f \in C^0[-\pi, +\pi]$:

1) Skalarprodukt:

$$\langle f_1, f_2 \rangle \;\; = \;\; \int_{-\pi}^{\pi} f_1(x) f_2(x) dx$$

2) Euklidische Norm:

$$\|f\|_2 \; = \; \langle f, f \rangle^{\frac{1}{2}} \; = \; \sqrt{\int_{-\pi}^{\pi} \Big(f(x) \Big)^2 dx}$$

3) Orthornormale Basis:

$$\frac{1}{\sqrt{2\pi}}, \; \frac{1}{\sqrt{\pi}} \cos x, \; \frac{1}{\sqrt{\pi}} \sin x, \; \ldots \; \frac{1}{\sqrt{\pi}} \cos lx, \; \frac{1}{\sqrt{\pi}} \sin lx, \; \ldots$$

4) Orthogonalitätsrelationen bez. der Basiselemente:

$$\begin{aligned} \int_{-\pi}^{\pi} \cos lx \cos mx \, dx \;\; &= \;\; \int_{-\pi}^{\pi} \sin lx \sin mx \, dx \; = \\ &= \;\; \int_{-\pi}^{\pi} \cos lx \sin mx \, dx \; = \; 0 \quad \text{für } l \neq m; \end{aligned}$$

weiters:

$$\int_{-\pi}^{\pi} \cos lx \sin lx \, dx \; = \; 0;$$

ferner gilt:

$$\int_{-\pi}^{\pi} (\cos lx)^2 dx \; = \; \int_{-\pi}^{\pi} (\sin lx)^2 dx \; = \; \pi$$

(Vgl. Lehrbuchliteratur).

5) Darstellung von $f \ldots$ *Fourierreihe:*

$$\begin{aligned} f &= \langle f, \frac{1}{\sqrt{2\pi}}\rangle \frac{1}{\sqrt{2\pi}} + \langle f, \frac{1}{\sqrt{\pi}}\cos x\rangle \frac{1}{\sqrt{\pi}}\cos x + \langle f, \frac{1}{\sqrt{\pi}}\sin x\rangle \frac{1}{\sqrt{\pi}}\sin x + \cdots \\ &\quad \cdots + \langle f, \frac{1}{\sqrt{\pi}}\cos lx\rangle \frac{1}{\sqrt{\pi}}\cos lx + \langle f, \frac{1}{\sqrt{\pi}}\sin lx\rangle \frac{1}{\sqrt{\pi}}\sin lx + \cdots \end{aligned}$$

- Für jedes $f \in C^0[-\pi, \pi]$ konvergiert die Fourierreihe gegen f. (Vollständigkeitsrelation ist erfüllt)
- Die Vollständigkeitsrelation gilt sogar unter viel schwächeren Voraussetzungen an f: und zwar für (im Lebesgueschen Sinne) auf $[-\pi, \pi]$ quadratintegrierbaren Funktionen.
- Konvergenz der Fourierreihe gegen f nicht nur wie bei der Vollständigkeitsrelation bez. der Euklidischen Norm, sondern auch bez. der Maximumnorm (wenn f den Dirichletschen Bedingungen genügt – vgl. Lehrbuchliteratur).
- Abgebrochene Fourierreihen (d.h. endliche Teilsummen der Fourierreihen) sind die Bestappoximierenden bez. der Euklid'schen Norm von f; der Teilraum $\Gamma \subset \Phi$ ist dabei der durch die endlich vielen Basisfunktionen

  $$\frac{1}{\sqrt{2\pi}},\ \frac{1}{\sqrt{\pi}}\cos x,\ \frac{1}{\sqrt{\pi}}\sin x,\ \ldots,\ \frac{1}{\sqrt{\pi}}\cos lx,\ \frac{1}{\sqrt{\pi}}\sin lx$$

  aufgespannte Teilraum, die der abgebrochenen Fourierreihe entsprechen.

Außer dem historisch ältesten Orthonormalsystem

$$\frac{1}{\sqrt{2\pi}},\ \frac{1}{\sqrt{\pi}}\cos x,\ \frac{1}{\sqrt{\pi}}\sin x,\ \ldots \quad \text{bezüglich } [a, b] = [-\pi, \pi]$$

gibt es noch zahlreiche weitere Orthonormalsysteme bez. anderer Intervalle (z.B. $[a, b] = [-1, +1]$ und $[a, b] = (-\infty, +\infty)$) und bezüglich verschiedener Skalarproduktdefinitionen $\int_a^b f_1(x) f_2(x) G(x) dx$ mit verschiedenen Gewichtsfunktionen $G(x)$. Z.B. Legendre-Polynome, Tschebyscheff-Polynome, Laguerre-Polynome, Hermite-Polynome .... Die Theorie der Fourierreihen ist ein sehr umfassendes Teilgebiet der Approximationstheorie.

4.8.1 Tschebyscheff Approximation

Hier soll nur kurz die **Tschebyscheff–Approximation** gestreift werden, bei der in (4.37) die Maximumnorm (4.34) zugrunde gelegt wird, d.h. zu einem vorgegebenen f soll jenes g^* aus Γ gefunden werden, für das gilt

$$\|g^* - f\|_\infty \ \leq \ \|g - f\|_\infty, \quad \forall g \in \Gamma \tag{4.38}$$

oder anders gesagt: es soll jenes g^* aus Γ gefunden werden, für das die Maximalabweichung von f

$$\max_{x \in [a,b]} |g^*(x) - f(x)|$$

so klein wie möglich wird,

Ein besonders einfacher Fall:

f auf $[a, b]$ konvex (oder konkav) und Γ sei Menge der Polynome vom Maximalgrad 1

D.h. folgendes Konstruktionsprinzip:

- Sehne durch $(a, f(a)), (b, f(b)) \ldots$ Gerade g_1
- Tangente, die zu g_1 parallel ist $\ldots$ Gerade g_2
- Mittelparallele von g_1 und $g_2 \ldots$ **Bestappoximierende**

Beweis: Die Maximalabweichung

$$\max_{x\in[a,b]} |g^*(x) - f(x)|,$$

die für g^* so klein wie möglich werden soll, wird bezüglich der Mittelparallelen an drei Stellen angenommen.

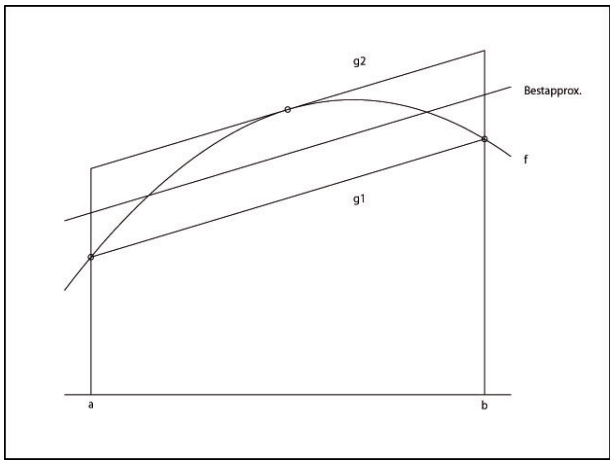


Abbildung 4.8: Alternantenpunkte

1. für $x = a$ (linkes Intervallende)
2. für $x = b$ (rechtes Intervallende)
3. für $x = x_1$ (jene Stelle, wo die zu g_1 parallele Tangente g_2) die Funktion f berührt)

Nun Beweis indirekt: Angenommen die Bestapproximierende g^* stimmt nicht mit der Mittelparallelen überein. Dann kann die Bestapproximierende für $x = a$ nicht oberhalb von der Mittelparallelen liegen (da sonst $\max_{x\in[a,b]} |g^*(x) - f(x)| \geq |g^*(a) - f(a)| >$ Maximalabweichung der Mittelparallelen wäre, d.h. $\|g^* - f\|_\infty >$ ‖Mittelparallele $- f\|_\infty$ gelten würde). Aus dem selben Grund kann die Bestapproximierende nicht oberhalb der Mittelparallelen bezüglich $x = b$ liegen. Wenn also die Bestapproximierende von der Mittelparallelen verschieden sein soll, muss sie im Inneren von $[a, b]$ ganz unterhalb von der Mittelparallelen liegen, woraus bezüglich $x = x_1$ folgt:

$$\begin{aligned}\max_{x\in[a,b]} |g^*(x) - f(x)| &\geq |g^*(x_1) - f(x_1)| > \\ &> |\text{Mittelparallele}(x_1) - f(x_1)| = \\ &= \|\text{Mittelparallele} - f\|_\infty\end{aligned}$$

Also gilt: $\|g^* - f\|_\infty >$ ‖Mittelparallele $- f\|_\infty$ und g^* kann nicht die Bestapproximierende sein $\Rightarrow$ Mittelparallele ist tatsächlich die Bestapproximierende. □

In dem eben besprochenen Fall wird also der Maximalabstand $\max_{x\in[a,b]} |g^*(x) - f(x)|$ in den drei Punkten $x_0 = a$, x_1 und $x_2 = b$ angenommen, wobei die Abweichung $g^*(x) - f(x)$ in diesen drei Punkten *alternierendes* Vorzeichen annimmt. Man bezeichnet diese Punkte x_0, x_1, x_2 deshalb als **Alternantenpunkte**. Man kann das bestappoximierende Polynom vom Grad 1 offenbar dadurch charakterisieren, dass die Maximalabweichung an drei Alternantenpunkten angenommen wird, an denen das Vorzeichen von $g(x) - f(x)$ alterniert. Wenn – wie in dem oben besprochenen Fall f konkav oder konvex ist, so gibt es offenbar *genau* drei Alternantenpunkte, wo die Maximalabweichung angenommen wird und zwei davon müssen die Randpunkte $x_0 = a, x_2 = b$ sein. Wenn f im Inneren von $[a, b]$ auch Wendepunkte aufweist, muss man diese Aussage etwas abschwächen: dann lässt sich nur mehr sagen, dass es *mindestens* drei Alternantenpunkte gibt, wo die Maximalabweichung angenommen wird und $g(x) - f(x)$ alternierendes Vorzeichen annimmt, und die Randpunkte a, b müssen auch nicht mehr Alternantenpunkte sein.

Diese Alternanteneigenschaft gilt auch für bestapproximierende Polynome höheren Grades (und auch noch für allgemeinere endlichdimensionale Teilräume $\Gamma \subset \Phi$). Für Polynome lautet die ensprechende Aussage:

Satz 4.8.1. Zu beliebigem $f \in C[a, b]$ existiert ein eindeutig bestimmtes bestapproximierendes Polynom P^* vom Maximalgrad n; P^* ist folgendermaßen charakterisiert: ein beliebiges Polynom vom Grad n ist dann und nur dann bestapproximierendes Polynom im Tschebyscheff'schen Sinn, wenn (mindestens) $n + 2$ Punkte

$$a \le x_0 < x_1 < \cdots < x_{n+1} \le b$$

existieren, für die die Fehlerfunktion $g(x) - f(x)$ maximal wird und für zwei aufeinanderfolgende Punkte x_{i-1}, x_i entgegengesetztes Vorzeichen hat.

$$x_0, x_1, \ldots, x_{n+1} \quad \text{heißen } \textbf{Alternantenpunkte}.$$

(Ohne Beweis.)

Bemerkung: Dieser sogenannte Alternantensatz ist nicht konstruktiv, d.h. um in konkreten Fällen die Bestapproximierende aufgrund der Charakterisierung des Alternantensatzes wirklich aufzubauen, müsste man einen extrem hohen algorithmischen Aufwand treiben (*Remez-Algorithmus*). Der Alternantensatz bietet aber die Grundlage dafür, mit viel einfacheren Mitteln eine sehr gute *Näherung* für die Bestapproximierende zu gewinnen.

Heuristische Begründung dieser Vorgangsweise Da in den $n + 2$ Alternantenpunkten der Fehler verschiedenes Vorzeichen aufweist, muss das bestapproximierende Polynom die Funktion f jeweils zwischen zwei Alternantenpunkten schneiden – also an $n + 1$ Punkten; d.h. das bestapproximierende Polynom vom Grad n ist daher auch ein Lagrange'sches Interpolationspolynom bezüglich der durch diese Schnittpunkte definierten Interpolationsdaten. Um Verwechslungen bezüglich der Notation zu vermeiden bezeichnen wir ab sofort die $n + 2$ Alternantenpunkte mit $x_0^{(a)}, x_1^{(a)}, \ldots, x_{n+1}^{(a)}$ und die dazwischenliegenden Knoten der Schnittpunkte von f und P^* mit $x_0, x_1, \ldots, x_n$; wir haben also

$$x_0^{(a)} < x_0 < x_1^{(a)} < x_1 < \cdots < x_n^{(a)} < x_n < x_{n+1}^{(a)}$$

$P^*(x)$ ist Interpolationspolynom vom Grad n zum Datensatz $(x_0, f_0), \ldots, (x_n, f_n)$. Das Interpolationspolynom zu so einem Datensatz aufzustellen ist – im Gegensatz zur Aufgabenstellung *das bestapproximierende Polynom $P^*(x)$ im Tschebyscheffschen Sinn zu ermitteln* – eine sehr einfache Aufgabenstellung. Nur scheitert diese Vorgangsweise (die Bestimmung der Bestapproximierenden auf die

Langrangeinterpolation zurückzuführen) leider daran, dass man die Lage der Interpolationsknoten $x_0, \ldots, x_n$ nicht kennt.

$\Rightarrow$ näherungsweise Bestimmung dieser Interpolationsknoten: es wird sich herausstellen, dass eine zweckmässige Wahl von $x_0, \ldots, x_n$ die (geeignet transformierten) Nullstellen der sogenannten **Tschebyscheffpolynome** ist.

Zunächst angenommen wir würden genau die Knoten $x_0, \ldots, x_n$ kennen, sodass das Lagrangesche Interpolationspolynom zum Datensatz $(x_0, f_0), \ldots, (x_n, f_n)$ gleich dem bestapproximierenden Polynom $P^*(x)$ bezüglich f ist. Aufgrund von (4.23) schreibt sich der Interpolationsfehler dann

$$P^*(x) - f(x) \;=\; -\frac{f^{n+1}(\vartheta)}{(n+1)!}\omega(x) \tag{4.39}$$

wobei $\omega(x)$ das Polynom $(x - x_0) \cdots (x - x_n)$ vom Grad $n+1$ ist. Andrerseits schwingt aufgrund des Alternantensatzes die Fehlerfunktion $P^*(x) - f(x)$ gleichmässig zwischen den Abweichungsmaxima.

Wenn man nun annimmt, dass f so glatt ist, dass die $(n+1)$-te Ableitung von f in $[a, b]$ nicht stark schwankt, sodass $f^{(n+1)}(\vartheta)$ beziehungsweise genauer $f^{(n+1)}(\vartheta(x))$ (man beachte: ϑ hängt auch von x ab – siehe den Beweis von (4.23)) annähernd konstant ist, kann man (4.39) auch schreiben:

$$P^*(x) - f(x) \;\approx\; \text{const.}\omega(x)$$

und muss daher – aufgrund des Alternantesatzes – die Interpolationsknoten $x_0, \ldots, x_n$ so legen, dass $\omega(x) = (x - x_0) \cdots (x - x_n)$ den gleichmässig schwingenden Verlauf aufweist. Es wird sich herausstellen, dass das vorliegt, wenn $\omega(x)$ das Tschebyscheffpolynom vom Grad $n+1$ ist, wenn also die $x_0, \ldots, x_n$ die $n+1$ Nullstellen dieses Tschebyscheffpolynoms sind. Bis auf die Ungenauigkeit, dass $f^{(n+1)}(\vartheta(x))$ nicht konstant ist hat dann der Interpolationsfehler genau den im Alternantensatz verlangten Verlauf, sodass das Interpolationspolynom zu den (entsprechend transformierten) Nullstellen der Tschebyscheffpolynome als Interpolationsknoten eine gute Approximation zur Bestapproximierenden sein sollte.

Tschebyscheffpolynome: Einen gleichmässig schwingenden Verlauf weist die Funktion $\cos\varphi$ auf; auf dem Intervall $[0, (n+1)\pi]$ hat $\cos\varphi$ die $(n+1)$ Nullstellen $\frac{\pi}{2}, \frac{3\pi}{2}, \frac{5\pi}{2}, \ldots, \frac{(2n+1)\pi}{2}$. Mit $\psi = \frac{1}{(n+1)}\varphi$ (d.h. $\varphi = (n+1)\psi$) durchläuft ψ das Intervall $[0, \pi]$, wenn φ das Intervall $[0, (n+1)\pi]$ durchläuft, d.h. die Funktion $\cos((n+1)\psi)$ hat für $0 \le \psi \le \pi$ $(n+1)$ Nullstellen.

Die Funktion $\cos(n+1)\psi$ hat also auf $[0, \pi]$ den idealschwingenden Verlauf mit $n+1$ Nullstellen (und $n+2$ Extremata), ist aber leider kein Polynom. Aufgrund der bekannten Formel

$$\begin{aligned}\cos k\psi \;&=\; (\cos\psi)^k - \binom{k}{2}(\cos\psi)^{k-2}(1 - \cos^2\psi) + \\ &\quad + \binom{k}{4}\cos^{k-4}\psi(1 - \cos^2\psi)^2 - + \cdots,\end{aligned}$$

ist jedoch $\cos(n+1)\psi$ ein Polynom vom Grad $n+1$ in $\cos\psi$. Die Transformation $t = \cos\psi$ beziehungsweise $\psi = \arccos t$ führt somit auf das Polynom

$$T_{n+1}(t) \;=\; \cos((n+1)\arccos t)$$

in t, das vom Grad $n+1$ ist; wenn t das Intervall $[-1,+1]$ durchläuft, so durchläuft ψ offenbar das Intervall $[0,\pi]$, d.h. sämtliche Nullstellen von $T_{n+1}(t)$ liegen in dem Intervall $[-1,+1]$. Allgemein definiert man

$$T_k(t) \;=\; \cos(k \arccos t) \qquad \begin{array}{l}\text{Tschebyscheffpolynom}\\ \text{vom Grad } k.\end{array}$$

$k=0:\ \cos(0\cdot\arccos t)=\cos 0=1$ d.h. $T_0(t)\equiv 1$

$k=1:\ \cos(1\cdot\arccos t)=t$ d.h. $T_1(t)\equiv t$

Rekursion:

$$\begin{aligned}\cos((k+1)\psi) &= 2\cos\psi\cos k\psi-\cos((k-1)\psi)^{3)} \quad \Rightarrow\\ T_{k+1}(t) &= 2t\cdot T_k(t)-T_{k-1}(t) \quad \Rightarrow\\ T_2(t) &= 2t^2-1\\ T_3(t) &= 4t^3-3t,\ldots\end{aligned}$$

Die Tschebyscheffpolynome zeigen also den ideal gleichmässigen Schwingungsverlauf, d.h. die Nullstellen von T_{n+1} sind die idealen Nullstellen für das gleichmässig ausschwingendes Polynom ω vom Grad $n+1$. Dabei ist aber zu beachten, dass $T_{n+1}(t)$ bezüglich des Intervalls $[-1,+1]$ betrachtet wurde (d.h. die Nullstellen symmetrisch um $t=0$ in diesem Intervall liegen), während ω sich auf das Intervall $a\le x\le b$ bezieht. Die Nullstellen[4)] $t_0,t_1,\ldots,t_n\in[-1,+1]$ von $T_{n+1}(t)$ müssen daher noch transformiert werden gemäss

$$x_i \;=\; \frac{b-a}{2}\cdot t_i+\frac{a+b}{2}$$

.

Beispiel: $f(x)=\sin x,\ x\in[0,\frac{\pi}{2}]$. Bestapproximierendes Polynom vom Grad 1 gemäss obigem dem Konstruktionsprinzip konstruieren:

$$g_1 \;=\; \frac{2}{\pi}\cdot x \;=\; 0.6366197724\ldots\cdot x$$

Anstieg von g_2 ist ebenfalls $\frac{2}{\pi}$;

$$\frac{d\sin x}{dx}=\cos x \;\Rightarrow x_1^{(a)} \text{ festgelegt durch } \cos x_1^{(a)}=\frac{2}{\pi}$$

d.h. $x_1^{(a)}=\arccos(\frac{2}{\pi})=0.8806892354\ldots\ \Rightarrow$

$$\begin{aligned}P^*(x) &= \frac{2}{\pi}x+\frac{1}{2}(\sin x_1^{(a)}-\frac{2}{\pi}x_1^{(a)}) \;=\\ &= 0.6366197724\ldots\cdot x+0.1052568312\ldots\end{aligned}$$

Maximalabweichung von $P^*(x)$ und $\sin x$ ist gegeben durch $\frac{1}{2}(\sin x_1^{(a)}-\frac{2}{\pi}x_1^{(a)})=0.1052568312\ldots$

---

3) Folgt für $\alpha=k\psi$ und $\beta=\psi$ durch Addition der bekannten Formeln $\cos(\alpha+\beta)=\cos\alpha\cos\beta-\sin\alpha\sin\beta$ und $\cos(\alpha-\beta)=\cos\alpha\cos\beta+\sin\alpha\sin\beta$.

4) Die Nullstellen von $T_k(t)$ sind offenbar $\cos(\frac{2k+1-2i}{2k}\pi),\ i=1(1)k$.

Verglichen mit dem Interpolationspolynom vom Grad 1 zum Datensatz $(0, \sin 0), (\frac{\pi}{2}, \sin \frac{\pi}{2})$ (vgl. Seite 79 a)) für das sich

$$g(\frac{\pi}{5}) - \sin(\frac{\pi}{5}) \;=\; -0.1877852523\ldots$$

ergeben hat, haben wir jetzt:

$$P^*(\frac{\pi}{5}) - \sin(\frac{\pi}{5}) \;=\; -0.08252842109\ldots$$

Beispiel: $f(x) = \sin x,\ x \in [0, \frac{\pi}{2}]$; soll mit einem Lagrangeinterpolationspolynom vom Grad 3 interpoliert werden, aber bezüglich der 4 transformierten Nullstellen des Tschebyscheffpolynoms vom Grad 4 als Interpolationsknoten.
Die 4 Nullstellen von $T_4(t)$ in $(-1, +1)$ sind

$$\begin{aligned} t_0 &= \cos(\tfrac{7\pi}{8}), & t_1 &= \cos(\tfrac{5\pi}{8}) \\ t_2 &= \cos(\tfrac{3\pi}{8}), & t_3 &= \cos(\tfrac{\pi}{8}), \end{aligned}$$

die transformierten Nullstellen aus $[0, \frac{\pi}{2}]$ sind also

$$\begin{aligned} x_0 &= \tfrac{\pi}{4} \cdot \cos(\tfrac{7\pi}{8}) + \tfrac{\pi}{4} &&= 0.05978487536\ldots \\ x_1 &= \tfrac{\pi}{4} \cdot \cos(\tfrac{5\pi}{8}) + \tfrac{\pi}{4} &&= 0.4848392985\ldots \\ x_2 &= \tfrac{\pi}{4} \cdot \cos(\tfrac{3\pi}{8}) + \tfrac{\pi}{4} &&= 1.085957028\ldots \\ x_3 &= \tfrac{\pi}{4} \cdot \cos(\tfrac{\pi}{8}) + \tfrac{\pi}{4} &&= 1.511011451\ldots \end{aligned}$$

Mit dem Nevilleschema berechnen wir nun $P(\frac{\pi}{5})$, wobei P jetzt das Interpolationspolynom vom Grad 3 zu diesen Interpolationsknoten ist:

$$\begin{array}{lllll} x_0 & 0.059749\ldots & & & \\ & & 0.6032204\ldots & & \\ x_1 & 0.466066\ldots & & 0.582599\ldots & \\ & & 0.5660007\ldots & & 0.586865\ldots \\ x_2 & 0.884749\ldots & & 0.593487\ldots & \\ & & 0.7625883\ldots & & \\ x_3 & 0.998213\ldots & & & \end{array}$$

$$P(\frac{\pi}{5}) - \sin(\frac{\pi}{5}) \;=\; -0.000920249794\ldots$$

Kapitel 5

Numerische Integration

5.1 Motivation

Das Ziel der **numerischen Integration** oder **numerischen Quadratur** ist das Auffinden von Näherungswerten I für Integrale von Funktionen $f : [a, b] \subset \mathbb{R} \to \mathbb{R}$ der Form

$$\int_a^b f(t)\,\mathrm{dt}$$

oder im Mehrdimensionalen $\vec{f} : [a_1, b_1] \times \ldots \times [a_n, b_n] \to \mathbb{R}^n$ mit $n \in \mathbb{N}$

$$\int_{a_1}^{b_1} \ldots \int_{a_n}^{b_n} \vec{f}(t_1, \ldots, t_n)\,\mathrm{dt}_n \ldots \mathrm{dt}_1.$$

Im $\mathbb{R}^n$ werden oft auch Integrale über ein Gebiet $\mathbb{G}$ der Form

$$\int_{\mathbb{G}} \vec{f}(t_1, \ldots, t_n)\mathrm{dt}_1 \ldots \mathrm{dt}_n$$

berechnet. Der Näherungswert I soll einer bestimmten Genauigkeitsbedingung genügen, wie beispielsweise für Integrale über $\mathbb{R}$

$$\left| I - \int_a^b f(t)\,\mathrm{dt} \right| < \varepsilon \qquad \varepsilon > 0 \tag{5.1}$$

In diesem Kapitel wird nahezu ausschließlich der eindimensionale Fall besprochen.

Die Grundidee für numerische Quadraturverfahren ist, dass der Integrand $f(t)$ auf dem Intervall $[a, b]$ durch eine einfache Funktion $g(t)$ ersetzt wird und $\int_a^b g(t)\,\mathrm{dt}$ als Näherungswert für $\int_a^b f(t)\,\mathrm{dt}$ genommen wird.
Für die Wahl der Funktion $g(t)$ gibt es im wesentlichen zwei Aspekte:

1. $\int_a^b g(t)\,\mathrm{dt}$ soll leicht zu berechnen sein, g ist daher meistens ein Polynom oder eine stückweise polynomiale Funktion.
2. Um g leicht berechnen zu können, kommt sehr häufig das Interpolationsprinzip zur Anwendung.

Wir werden also g als stückweise polynomiale Funktion konstruieren, welche die Funktionswerte $f(t_i) = f_i,\ i = 0, 1, \ldots, n$, für eine Zerlegung $(t_i)_{i\in\mathbb{N}}$ des Intervalls $[a, b]$, interpoliert. Einzelne Verfahren arbeiten auch mit *einem* Interpolationspolynom auf dem ganzen Intervall. Um für so ein Polynom die Genauigkeitsanforderung zu erfüllen, muss jedoch ein ausreichend hoher Polynomgrad ausgewählt werden.

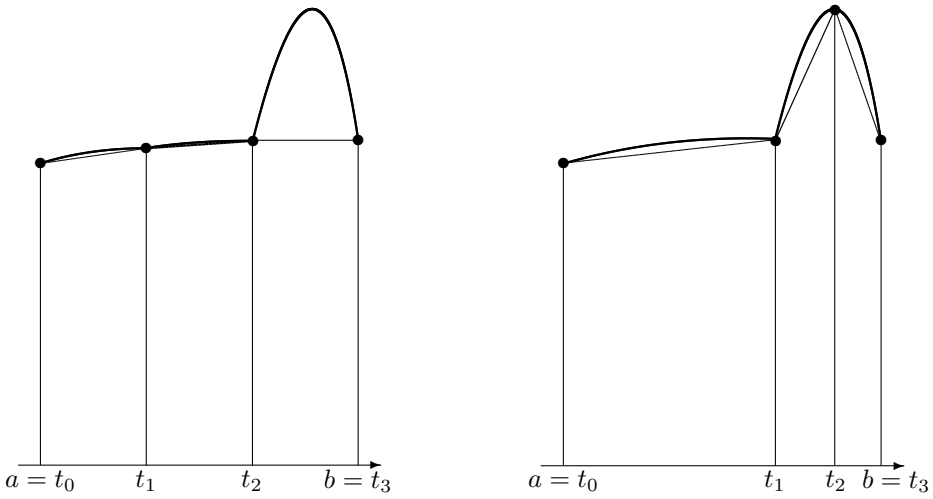


Abbildung 5.1: Äquidistantes und angepasstes Gittter

Um die Genauigkeitsanforderung (5.1) aus Effizienzgründen mit möglichst wenigen Funktionsauswertungen erfüllen zu können wird i.a. nicht an äquidistanten Knoten interpoliert. Gute Algorithmen generieren sich automatisch optimale Gitter, siehe Abbildung 5.1.

5.2 Newton - Cotes - Formeln

Wir betrachten zunächst eine Funktion $g : [a, b] \to \mathbb{R}$ die stückweise aus Polynomen zweiten Grades zusammengesetzt ist. Durch die Daten $(a = t_0, f(t_0))$, $(t_1, f(t_1))$ und $(t_2, f(t_2))$ ist das erste Interpolationspolynom festgelegt, durch $(t_2, f(t_2))$, $(t_3, f(t_3))$ und $(t_4, f(t_4))$ das zweite Interpolationspolynom, ... Eine Annahme ist, dass die Interpolationsknoten (t_0, t_1, t_2) des ersten Interpolationsintervalls äquidistant liegen, ebenso die Knoten (t_2, t_3, t_4) des zweiten Interpolationsintervalls, ...
Die beschriebene Vorgangsweise ergibt die sogenannte **Simpsonregel** zur Berechnung einer Näherung I für $\int_a^b f(t)\,\mathrm{dt}$.

Wir betrachten zunächst o.B.d.A. nur ein (Teil)Intervall $[0, 1]$. Die äquidistanten Knoten sind dann $t_k = 0 + k\frac{1-0}{2} = k\frac{1}{2}$ für $k = 0, 1, 2$. Die φ_i stellen die Lagrange - Polynome dar, siehe (4.7):

$$I = \int_0^1 \left(f(0)\varphi_0(t) + f\left(\frac{1}{2}\right)\varphi_1(t) + f(1)\varphi_2(t)\right)\mathrm{dt} =$$

$$= f(0)\int_0^1 \frac{(t-t_1)(t-t_2)}{(t_0-t_1)(t_0-t_2)}\,\mathrm{dt} + f(\tfrac{1}{2})\int_0^1 \frac{(t-t_0)(t-t_2)}{(t_1-t_0)(t_1-t_2)}\,\mathrm{dt} + f(1)\int_0^1 \frac{(t-t_0)(t-t_1)}{(t_2-t_0)(t_2-t_1)}\,\mathrm{dt}$$

Einsetzen vom t_k für $k = 0, 1, 2$

$$\int_0^1 \frac{(t-\frac{1}{2})(t-1)}{(-\frac{1}{2})(-1)}\,\mathrm{dt} = \int_0^1 (2t^2 - 3t + 1)\,\mathrm{dt} = \frac{1}{6}$$

$$\int_0^1 \frac{t(t-1)}{\frac{1}{2}(-\frac{1}{2})}\,\mathrm{dt} = \frac{4}{6} \qquad \int_0^1 2t(t-\frac{1}{2})\,\mathrm{dt} = \frac{1}{6}$$

Dies ergibt die *Simpsonregel*, die sich im Spezialfall, dass in allen Teilintervallen $[t_0, t_2]$,$[t_2, t_4]$, ... $[t_{N-2}, t_N]$ der selbe Knotenabstand $h = \frac{t_{j+2}-t_j}{2}$ für $j = 0, 1, 2, \ldots, N-2$ gewählt wird, folgendermaßen schreiben lässt:

$$\begin{aligned} I &= \int_a^b g(t)\,\mathrm{dt} = \int_a^{t_2} g(t)\,\mathrm{dt} + \int_{t_2}^{t_4} g(t)\,\mathrm{dt} + \ldots \int_{t_{N-2}}^{t_N} g(t)\,\mathrm{dt} = \\ &= h\left[\frac{1}{6}f(a) + \frac{4}{6}f(t_1) + \frac{2}{6}f(t_2) + \frac{4}{6}f(t_3) + \frac{2}{6}f(t_4) + \ldots + \frac{2}{6}f(t_{N-2}) + \frac{4}{6}f(t_{N-1}) + \frac{1}{6}f(b)\right] \end{aligned} \tag{5.2}$$

mit

$$t_i = a + \frac{b-a}{2N}i = a + hi \quad \text{mit} \quad i = 0, 1, \ldots, 2N \quad \text{und} \quad N \in \mathbb{N}\,.$$

Eine *alternative Herleitungsidee* ist die folgende: Wie betrachten auf dem Teilintervall $[0, 1]$ die Quadraturformel

$$\int_0^1 f(t)\,\mathrm{dt} \approx c_0 f(0) + c_1 f\left(\frac{1}{2}\right) + c_2 f(1) \tag{5.3}$$

mit noch nicht festgelegten Koeffizienten c_0, c_1, c_2. Diese sogenannten Gewichte c_i werden dadurch festgelegt, dass die Quadraturformel für Polynome mit Grad kleiner gleich zwei exakt ist. Speziell ergibt sich:

$$\begin{array}{lrcrcl} f(t) \equiv 1: & 1c_0 + 1c_1 + 1c_2 & = & \int_0^1 1\,\mathrm{dt} & = & 1 \\ f(t) \equiv t: & \frac{1}{2}c_1 + c_2 & = & \int_0^1 t\,\mathrm{dt} & = & \frac{1}{2} \\ f(t) \equiv t^2: & \frac{1}{4}c_1 + 4c_2 & = & \int_0^1 t^2\,\mathrm{dt} & = & \frac{1}{3} \end{array} \tag{5.4}$$

Lösung des linearen Gleichungssystem (5.4) ergibt $c_0 = \frac{1}{6}$, $c_1 = \frac{4}{6}$ und $c_2 = \frac{1}{6}$.

Die vorgeschlagene Methode lässt sich verallgemeinern. Statt Polynome zweiten Grades werden Polynome n-ten Grades verwendet, es werden also Daten $(t_0, f(t_0))$, $(t_1, f(t_1))$,..., $(t_n, f(t_n))$, $n \in \mathbb{N}$ mit äquidistanten Knotenabstand $t_i - t_{i-1} = \frac{t_n - a}{n}$, interpoliert. Im zweiten Interpolationsintervall $[t_n, t_{2n}]$werden die Daten $(t_n, f(t_n))$,$(t_{n+1}, f(t_{n+1}))$,...,$(t_{2n}, f(t_{2n}))$ interpoliert, wobei der Knotenabstand ebenfalls äquidistant mit $t_i - t_{i-1} = \frac{t_{2n}-t_n}{n}$ ist. Diese Idee wird auf die folgenden Intervalle weiter angewandt.

Ein wichtiger Spezialfall ist, dass in dem ganzen Integrationsintervall ein äquidistanter Knotenabstand vorliegt, d.h. das die Schrittweite $h = \frac{t_{ln} - t_{(l-1)n}}{n}$ in allen Interpolationsintervallen gleich ist. Diese interpolatorischen Quadraturformeln, die auf stückweise polynomialer Interpolation mit Polynomen vom Grad n basieren und wo in jedem Interpolationsintervall die Interpolationsknoten äquidistant liegen, werden als **Newton - Cotes - Formeln** bezeichnet.

Für den Verfahrensfehler bezüglich eines Interpolationsintervalls $[t_{(l-1)n}, t_{ln}]$ gilt folgender

Satz 5.2.1. Sei $f(t)$ eine $(n+2)$-mal stetig differenzierbare Funktion, falls n gerade, oder $(n+1)$-mal stetig differenzierbar, falls n ungerade. Dann gilt

$$\int_{t_{(l-1)n}}^{t_{ln}} g(t)\,\mathrm{dt} - \int_{t_{(l-1)n}}^{t_{ln}} f(t)\,\mathrm{dt} = \begin{cases} -\frac{M_{n,g}}{(n+2)!}h^{n+3}f^{(n+2)}(\xi) & \text{n gerade} \\ -\frac{M_{n,u}}{(n+1)!}h^{n+2}f^{(n+1)}(\xi) & \text{n ungerade} \end{cases} \tag{5.5}$$

wobei $\xi \in [t_{(l-1)n}, t_{ln}]$ und $h = \frac{t_{ln}-t_{(l-1)n}}{n}$ gilt, sowie

$$M_{n,g} = \int_0^n t^2(t-1)(t-2)\dots(t-n)\,\mathrm{dt}$$

und

$$M_{n,u} = \int_0^n t(t-1)(t-2)\dots(t-n)\,\mathrm{dt}.$$

Beweis. siehe Isaacson-Keller, Analysis of Numerical Methods □

Die Gewichte für spezielle Newton - Cotes - Formeln sind in Tabelle 5.1 für das Standardintervall [0.1] und äquidistanten Knoten $t_i = i\frac{1}{n}$ $i = 0, 1, 2, \dots n$ zusammengestellt. Für die Fehlerschranke wird die Länge des zu integrierenden Intervalls $h = b - a$ verwendet.

n	Gewichte c_i							Fehler	Name
1	$\frac{1}{2}$	$\frac{1}{2}$						$\frac{h^3}{12}f''(\xi)$	Trapezregel
2	$\frac{1}{6}$	$\frac{4}{6}$	$\frac{1}{6}$					$\frac{(\frac{h}{2})^5}{90}f^{(iv)}(\xi)$	Simpsonregel
3	$\frac{1}{8}$	$\frac{3}{8}$	$\frac{3}{8}$	$\frac{1}{8}$				$\frac{3(\frac{h}{3})^5}{80}f^{(iv)}(\xi)$	3/8 - Regel oder Pulcherrima
4	$\frac{7}{90}$	$\frac{32}{90}$	$\frac{12}{90}$	$\frac{32}{90}$	$\frac{7}{90}$			$\frac{8(\frac{h}{4})^7}{945}f^{(vi)}(\xi)$	Milneregel
5	$\frac{19}{288}$	$\frac{75}{288}$	$\frac{50}{288}$	$\frac{50}{288}$	$\frac{75}{288}$	$\frac{19}{288}$		$\frac{275(\frac{h}{5})^7}{12096}f^{(vi)}(\xi)$	6 - Punkt - Regel
6	$\frac{41h}{140}$	$\frac{216h}{140}$	$\frac{27}{140}$	$\frac{272}{140}$	$\frac{27}{140}$	$\frac{216}{840}$	$\frac{41}{840}$	$\frac{9(\frac{h}{6})^9}{1400}f^{(viii)}(\xi)$	Weddleregel

Tabelle 5.1: Gewichte und Fehlerschranken einiger Newton - Cotes - Formeln

Für große n sind diese Formeln aus praktischer Sicht unbrauchbar, da viele Funktionsauswertungen notwendig sind. Dabei kommt es vermehrt zu Rundungsfehlern und Auslöschung. Ab $n = 7$ treten in etlichen Formlen sogar negative Gewichte auf.

Für die Ordnungsaussage (5.5) in Satz 5.2.1 bzw. in der obigen Tabelle gibt es ein einfaches heuristisches Argument. Der Integrationsfehler

$$\int_{t_{l-1}n}^{t_{ln}} \big(g(t) - f(t)\big)\mathrm{d}t,$$

kann leicht mittels des Interpolationsfehlers, siehe (4.23)

$$-\int_{t_{(l-1)n}}^{t_{ln}} \frac{f^{(n+1)}(\theta(t))}{(n+1)!}\omega(t)\mathrm{d}t \tag{5.6}$$

dargestellt werden. Nimmt man nun an, dass $\theta(t)$ eine glatte Funktion ist so, dass $f^{(n+1)}(\theta(t)) = O(1)$ gilt, so ist der Integrand in der Größenordnung von $\omega(t)$, d.h. $O(h^{n+1})$. Da die Länge des Integrationsintervalls $t_{ln} - t_{(l-1)n} = O(h)$ ist, ist das Integral (5.6) von der Größenordnung $O(h^{n+2})$ was für ungerades n in Übereinstimmung mit (5.5) bzw. mit der Tabelle oben ist. Das für gerades n noch eine Potenz von h mehr möglich ist, lässt sich heuristisch aus den Symmetrieeigenschaften von ω begründen, da in diesen Fällen ω schiefsymmetrisch bezüglich des Intervallmittelpunkt $\frac{t_{(l-1)n}+t_{ln}}{2}$ verläuft so, dass

$$\int_{t_{(l-1)n}}^{t_{ln}} \omega(t)\,\mathrm{dt} = 0$$

gilt. Wäre $f^{(n+1)}(t)$ konstant, also unabhängig von t, so würde das Integral (5.6) verschwinden und die numerische Integration wäre exakt. Anders ausgedrückt, die entsprechende Newton - Cotes - Formel ist für gerades n nicht für Polynome vom Grad n, sondern sogar für Polynome vom Grad $n+1$ exakt.

Gemäß der Trapezregel mit $n = 1$, N Interpolationsintervalle und Schrittweite $h = \frac{b-a}{N}$ gilt:

$$\begin{aligned}\left|T(h) - \int_a^b f(t)\mathrm{d}t\right| &= \sum_{r=1}^{N} \frac{h^3}{12} f''(\xi_r) \le N\frac{h^3}{12}\max_{t\in[a,b]}|f''(t)| = \\ &= \frac{(b-a)h^2}{12}\max_{t\in[a,b]}|f''(t)|\end{aligned} \tag{5.7}$$

Gemäß der Simpsonregel mit $n = 2$, N Interpolationsintervalle und Schrittweite $h = \frac{b-a}{2N}$ gilt:

$$\begin{aligned}\left|S(h) - \int_a^b f(t)\,\mathrm{dt}\right| &= \sum_{r=1}^{N} \left(\frac{h}{2}\right)^5 \frac{1}{90} f^{(iv)}(\xi_r) \le N\left(\frac{h}{2}\right)^5 \frac{1}{90}\max_{t\in[a,b]}|f^{(iv)}(t)| = \\ &= \left(\frac{h}{2}\right)^4 \frac{(b-a)}{360}\max_{t\in[a,b]}|f^{(iv)}(t)|\end{aligned} \tag{5.8}$$

5.2.1 Daten- und Rundungsfehler

Der Effekt von verfälschten Funktionswerten $\widetilde{f}(t_i)$ lässt sich folgendermaßen abschätzen: Sei $i \in \mathbb{N}$ und $c_i > 0$, dann gilt

$$\begin{aligned}\left|\sum_i c_i\widetilde{f}(t_i) - \sum_i c_i f(t_i)\right| &\le \sum_i |c_i|\left|\widetilde{f}(t_i) - f(t_i)\right| \le \\ &\le (b-a)\max_i\left|\widetilde{f}(t_i) - f(t_i)\right|.\end{aligned} \tag{5.9}$$

Auch in diesem allgemeinen Fall gilt $\sum_i |c_i| = \sum_i c_i = b - a$, da die Quadratur von $f(t) \equiv 1$ exakt ist. Allgemein gilt

$$\int_a^b f(t)\mathrm{d}t \approx c_0 f(t_0) + c_1 f(t_1) + \ldots + c_N f(t_N)$$

und für $f(t) \equiv 1$

$$\int_a^b 1\mathrm{d}t \approx c_0 + c_1 + \ldots + c_N$$

Vergleicht man (5.9) mit

$$\left| \int_a^b \widetilde{f}(t)\mathrm{d}t - \int_a^b f(t)\mathrm{d}t \right| \leq (b-a) \max_{t\in[a,b]} |\widetilde{f}(t) - f(t)| \tag{5.10}$$

so erkennt man, dass die Empfindlichkeit bezüglich Funktionsverfälschungen im Fall der exakten Integration und der numerischen Integration gleich ist.

Bezüglich der Rechenfehler ist i.a. nur der Additionsfehler wesentlich. Da der Gesamtadditionsfehler mit der Anzahl der Summanden wächst, nimmt der Gesamtrechenfehler mit kleinerwedendem h zu. Er kann jedoch durch Verwendung partieller doppelter Genauigkeit i.a. hinreichend klein gehalten werden. bei monotonem Verlauf von f empfiehlt es sich, die Summation von der Seite der absolut kleineren Werten her zu beginnen.

5.2.2 Effizienz der Newton - Cotes - Formeln

Konvergenzordnungen der Newton - Cotes - Formeln:

$$\begin{array}{rrcll} \text{Trapezregel} & T(h) - \int_a^b f(t)\mathrm{d}t & = & O(h^2) & (5.7) \\ \text{Simpsonregel} & S(h) - \int_a^b f(t)\mathrm{d}t & = & O(h^4) & (5.8) \\ \text{3/8 - Regel oder Pulcherrima} & P(h) - \int_a^b f(t)\mathrm{d}t & = & O(h^4) & \\ \text{Milneregel} & M(h) - \int_a^b f(t)\mathrm{d}t & = & O(h^6) & \end{array}$$

Diese Ordnungsaussagen gelten natürlich nur für hinreichend glatte also hinreichend oft stetig differenzierbare Funktionen.

Anhand des folgenden Beispiels soll nun die Effizienz der unterschiedlichen Newton - Cotes - Formeln für verschiedene Genauigkeitsniveaus analysiert werden.

Beispiel 5.2.2. Es soll $\int_{-1}^{1} e^{8t}\,\mathrm{d}t$ auf ein Genauigkeitsniveau von

(a) 10^{-1}

(b) 10^{-10}

berechnet werden. Wie ist die Schrittweite zu wählen, dass dieses Genauigkeitsniveau erreicht wird mit

(i) Trapezregel

(ii) Simpsonregel.

Dabei soll nur der Verfahrensfehler berücksichtigt werden.

Fall **(a)(i)**: Die Fehlerschranke nach (5.7)

$$\frac{b-a}{12} h^2 \max_{t\in[a,b]} |f''(t)| = h^2 \frac{2}{12} 64\, e^8 \approx 3{,}18 \cdot 10^4 h^2$$

hat

$$3{,}18 \cdot 10^4 h^2 \approx 10^{-1}$$

und damit $h \approx 0{,}002$ zur Folge.

Fall **(a)(ii)**: Die Fehlerschranke nach (5.8)

$$\frac{b-a}{360}\left(\frac{h}{2}\right)^4 \max_{t\in[a,b]}|f^{(iv)}| = \left(\frac{h}{2}\right)^4 \frac{2}{360} 4096\ e^8 \approx 1{,}35\cdot 10^5 \left(\frac{h}{2}\right)^4$$

hat

$$1{,}35\dot{1}0^5 \left(\frac{h}{2}\right)^4 \approx 10^{-1}$$

und damit $h \approx 0{,}06$ zur Folge.

Fall **(b)(i)**: Analog folgt

$$3{,}18\cdot 10^4 h^2 \approx 10^{-10}$$

und damit $h \approx 6\cdot 10^{-8}$.

Fall **(b)(ii)**: Analog folgt

$$1{,}35\cdot 10^5 \cdot \left(\frac{h}{2}\right)^4 \approx 10^{-10}$$

und damit $h \approx 4\cdot 10^{-4}$.

Fazit: Bei geringeren Genauigkeitsanforderungen sind Verfahren höherer Ordnung kaum vorteilhaft, bei sehr großer Genauigkeitsanforderung jedoch ist ein Verfahren höherer Ordnung viel effizienter. Natürlich ist diese Aussage nur aus dem eben besprochenen Beispiel gewonnen und auch nicht an den tatsächlichen Quadraturfomeln, sondern nur an den Fehlerschranken, welche sehr pessimistisch sein können. Die Aussage ist aber doch richtig, und gilt nicht nur im Zusammenhang mit Quadraturverfahren, sondern auch bezüglich Verfahren für Differentialgleichungen, welche auf der Quadratur-Idee aufbauen.

Vom Effizienzstandpunkt aus betrachtet, wünscht man sich für starke Genauigkeitsforderungen nicht nur Verfahren hoher Ordnung, sondern auch eine hohe Ordnung mit möglichst wenig Knoten also möglichst wenig Funktionsauswertungen in einem Interpolationsintervall. Quadraturverfahren zu konstruieren, bei denen mit möglichst wenig Funktionsauswertungen im Interpolationsintervall möglichst hohe Ordnungen erzielt werden, stellen beispielsweise *Gauß-Verfahren* dar.

5.3 Gauß - Verfahren

Grundidee: *Bei den Newton - Cotes - Formeln sind die Interpolationsknoten äquidistant im Interpolationsintervall. Die Ordnungen werden nur aufgrund der geeigneten Definition der Gewichte erreicht. Es liegt daher nahe, zusätzlich auch noch die Interpolationsknoten raffinierter zu wählen, um noch höhere Ordnung zu erreichen.*

Bei den Gauß - Formeln liegen die Knoten innerhalb der Interpolationsintervalle im Gegensatz zu den Newton - Cotes - Formeln nicht äquidistant. Es gibt daher keine feste Schrittweite h. Wenn man im Zusammenhang mit Gaußformeln von Ordnungsaussagen *Fehlerniveau* $= O(h^{\text{Potenz}})$ spricht, versteht man unter h eine mittlere Schrittweite oder eventuell die Länge eines einzelnen Interpolationsintervalls. Ein weiterer Gegensatz zu den Newton - Cotes - Formeln besteht darin, dass die Randpunkte

der Interpolationsintervalle keine Gitterpunkte der Quadratur sind.

Um die relative Lage der nicht äquidistanten Knoten der Gaußquadratur zu beschreiben, wird nur ein Interpolationsintervall betrachtet und ohne Beschränkung der Allgemeinheit der Standardfall angenommen, dass dieses Intervall das Intervall $[-1,1]$ ist. Jedes beliebige Intervall kann durch eine lineare Transformation auf $[-1,1]$ transformiert werden. Nur im Zusammenhang mit Ordnungsaussagen (*Fehlerniveau* $= O(h^{\text{ Potenz}})$) wird nicht an das Standardintervall $[-1,1]$ gedacht, da eine asymptotische Betrachtung $h \to 0$ und festes n, also fester Polynomgrad inkompatibel mit einem festgehaltenen Intervall $[-1,1]$ wären. Siehe etwa Formel (5.17), wo nicht von -1 bis $+1$ integriert wird sondern über ein Interpolationsintervall.

Man möchte also die *Gewichte* c_i *und die Knotenstellen* t_i in der Quadraturformel

$$I := \sum_{i=0}^{n} c_i f(t_i) \approx \int_{-1}^{+1} f(t)dt \tag{5.11}$$

so bestimmen, dass die Ordnung optimal wird.

Bei den Newton - Cotes - Formeln wurde mit Lagrangeinterpolation gearbeitet, jetzt werden Quadraturformeln basierend auf der Hermiteinterpolation aufgebaut, siehe Abschnitt 4.7. Konkret sei $g(t)$ ein Polynom vom Grad $2n+1$, das folgenden Datensatz

$$(t_0, f(t_0), f'(t_0)),\ (t_1, f(t_1), f'(t_1)),\ \ldots,\ (t_n, f(t_n), f'(t_n))$$

interpoliert.

Wir bezeichnen die entsprechenden Basispolynome mit $\psi_i(t)$ und $\rho_i(t)$, ψ_i und ρ_i seien Polynome vom Grad $2n+1$, für die gilt

$$\begin{array}{llll} \psi_i(t_k) = \delta_{ik}, & \psi_i'(t_k) = 0, & & \\ \rho_i(t_k) = 0, & \rho_i'(t_k) = \delta_{ik}, & & i,k = 0,1,\ldots,n \end{array} \tag{5.12}$$

wobei δ_{ik} das Kroneckersymbol ist, also $\delta_{ik} = 0$ für $i \neq k$ und $\delta_{ik} = 1$ für $i = k$. Aus

$$g(t) \;=\; \sum_{i=0}^{n} f(t_i)\psi_i(t) + \sum_{i=0}^{n} f'(t_i)\rho_i(t), \tag{5.13}$$

folgt die Quadraturformel

$$I \;=\; \int_{-1}^{+1} g(t)dt \;=\; \sum_{i=0}^{n} f(t_i) \int_{-1}^{+1} \psi_i(t)dt + \sum_{i=0}^{n} f'(t_i) \int_{-1}^{+1} \rho_i(t)dt \tag{5.14}$$

mit noch unbekannten Knoten t_i für $i = 0,1,2,\ldots n$. Wenn es nun gelingt, spezielle Knoten $t_0, t_1, \ldots, t_n$ so zu finden, dass

$$\int_{-1}^{+1} \rho_i(t)dt \;=\; 0, \quad i = 0,1,\ldots,n \tag{5.15}$$

gilt, entsteht wieder eine Quadraturformel von der üblichen Gestalt

$$I \;=\; \sum_{i=0}^{n} f(t_i) \underbrace{\int_{-1}^{+1} \psi_i(t)dt}_{c_i}. \tag{5.16}$$

Verglichen mit den Newton - Cotes - Formeln ist das asymptotische Ordnungsniveau natürlich höher: Für den Quadraturfehler gilt

$$\int\limits_{\substack{\text{Interpolations-}\\ \text{intervall}}} (g(t) - f(t))dt \;=\; \int\limits_{\substack{\text{Interpolations-}\\ \text{intervall}}} \frac{\omega^2(t) f^{(2n+2)}(\vartheta(t))}{(2n+2)!} dt \tag{5.17}$$

mit

$$\omega^2(t) \;=\; (t-t_0)^2(t-t_1)^2 \cdots (t-t_n)^2 \tag{5.18}$$

Bezeichnet man mit h den mittleren Knotenabstand etwa definiert durch $h = \frac{|\text{ Interpolationsintervall }|}{n}$, so ist offenbar $\omega^2(t) = O(h^{2n+2})$. Da die Länge des Interpolationsintervalls $O(h)$ ist, ist das Integral (5.17) von der Größenordnung $O(h^{2n+3})$. Diese Ordnungsaussage gilt natürlich nur für eine ausreichend glatte Funktion f und bezieht sich auf ein Interpolationsintervall. Bei Aufsummation über alle Interpolationsintervalle verliert man eine h-Potenz, sodass die Gauß - Formeln insgesamt von der Ordnung $2n+2$ sind.

Die entscheidende Frage ist nun, ob es wirklich gelingt Knoten $t_0, \ldots, t_n \in [-1, 1]$ zu finden, sodass (5.15) gilt.

Die Basispolynome $\psi_i(t), \rho_i(t)$ der Hermiteinterpolation (5.12), lassen sich mit Hilfe der Basispolynome $\varphi_i(t)$ der Lagrangeinterpolation (4.7) folgendermaßen schreiben

$$\begin{aligned} \psi_i(t) &= (1 - 2\varphi_i'(t_i)(t-t_i))[\varphi_i(t)]^2 \\ \rho_i(t) &= (t-t_i)[\varphi_i(t)]^2 \end{aligned} \tag{5.19}$$

Man verifiziert sofort das Erfülltsein von (5.12). $\rho_i(t)$ schreibt sich ausführlich

$$\begin{aligned} \rho_i(t) &= (t-t_i)\left[\frac{(t-t_0)\cdots(t-t_{i-1})(t-t_{i+1})\cdots(t-t_n)}{(t_i-t_0)\cdots(t_i-t_{i-1})(t_i-t_{i+1})\cdots(t_i-t_n)}\right]^2 = \\ &= \prod_{k=0}^{n}(t-t_k)\frac{(t-t_0)\cdots(t-t_{i-1})(t-t_{i+1})\cdots(t-t_n)}{[(t_i-t_0)\cdots(t_i-t_{i-1})(t_i-t_{i+1})\cdots(t_i-t_n)]^2} = \\ &\quad\uparrow \qquad\qquad\qquad \uparrow \\ &= \omega(t)\cdot \text{ Polynom vom Grad } n \end{aligned}$$

Also (5.15) ist sicher erfüllt, wenn

$$\int_{-1}^{+1} \omega(t)p(t)dt \;=\; 0 \tag{5.20}$$

gilt für *jedes* Polynom $p(t)$ vom Grad $\leq n$.

Ein kurzer Einschube über die sogenannten **Legendre - Polynomen**, die durch $P_0(t) = 1,\ P_1(t) = t$, $P_2(t) = \frac{1}{2}(3t^2-1)$, ..., und durch die Rekursionsformel

$$P_{n+1}(t) \;=\; \frac{2n+1}{n+1}P_n(t) - \frac{n}{n+1}P_{n-1}(t)$$

definiert sind. Die Legendre - Polynome bilden ein sogenanntes Orthogonalsystem:

$$\langle P_i, P_j\rangle \;=\; \int_{-1}^{+1} P_i(t)P_j(t)dt \;=\; \begin{cases} 0 & i \neq j \\ \frac{2}{2n+1} & i = j, \end{cases}$$

Da die Legendre - Polynome auch linear unabhängig sind, bilden die ersten $n+1$ Legendre - Polynome eine orthogonale Basis im Raum der Polynome vom Maximalgrad n, d.h. jedes Polynom vom Grad n läst sich in eindeutiger Weise als Linearkombination von $P_0(t), \ldots, P_n(t)$ schreiben. Die Nullstellen dieser Polynome liegen alle im Intervall $[-1, 1]$.

Satz 5.3.1. Das Legendre-Polynom P_n ($n \in \mathbb{N}$) hat in $(-1, +1)$ genau n paarweise verschiedene Nullstellen.

Beweis: $t_j,\ j = 1, 2, \ldots, l$ seien die verschiedenen Nullstellen von $P_n(t)$ in $(-1, +1)$ mit den Vielfachheiten α_j, also

$$P_n(t) \;=\; (t-t_1)^{\alpha_1}(t-t_2)^{\alpha_2}\cdots(t-t_l)^{\alpha_l}Q_n(t)$$

mit $Q_n(t) \neq 0$ in $(-1, 1)$. Sei

$$\beta_j = \begin{cases} 0 \\ 1 \end{cases} \quad \text{falls} \quad \alpha_j \quad \begin{matrix} \text{gerade} \\ \text{ungerade} \end{matrix}$$

$$P_n^*(t) \;:=\; (t-t_1)^{\beta_1}(t-t_2)^{\beta_2}\cdots(t-t_l)^{\beta_l}$$

Der Grad von P_n^* ist also sicher $\leq l$.

Falls $l < n$, dann folgt aus der Orthogonalität von P_n zu allen Polynomen von einem Grad $< n$, also auch zu P_n^* mit dem Grad $\leq l < n$

$$0 \;=\; \int_{-1}^{+1} P_n(t)P_n^*(t)dt \;=\; \int_{-1}^{+1} (t-t_1)^{\alpha_1+\beta_1}\cdots(t-t_l)^{\alpha_l+\beta_l}Q_n(t)dt$$

und das ist ein Widerspruch, da der Integrand rechts in $(-1, 1)$ das Vorzeichen nicht wechselt. Es muss also gelten $l = n$ und $\alpha_j = 1,\ j = 1, 2, \ldots n$. □

Wählt man nun die $n+1$ Knoten $t_0, \ldots, t_n$ aus $[-1, +1]$ als die Nullstellen des Legendre - Polynoms $P_{n+1}(t)$ vom Grad $n+1$, so stimmt $\omega(t)$ in (5.20) bis auf einen multiplikativen Faktor mit $P_{n+1}(t)$ überein. $p(t)$ in (5.20) ist ein beliebiges Polynom vom Maximalgrad n und lässt sich somit als Linearkombination von $P_0(t), \ldots, P_n(t)$ schreiben

$$\begin{aligned}
\int_{-1}^{+1} \omega(t)p(t)dt \;&=\; \int_{-1}^{+1} \text{Faktor } P_{n+1}(t)\Big(c_0P_0(t) + \cdots + c_nP_n(t)\Big)dt \;= \\
&=\; \text{Faktor}\, c_0 \int_{-1}^{+1} P_{n+1}(t)P_0(t)dt + \text{Faktor}\, c_1 \int_{-1}^{+1} P_{n+1}(t)P_1(t)dt + \\
&\quad + \cdots + \text{Faktor}\, c_n \int_{-1}^{+1} P_{n+1}(t)P_n(t)dt = 0
\end{aligned}$$

wegen der Orthogonalität der $P_i(t)$. (5.20) ist also tatsächlich erfüllt!

Die niedrigsten Gauß - Formeln sind

$$\begin{aligned}
\int_{-1}^{+1} f(t)dt &\approx 2f(0) \\
\int_{-1}^{+1} f(t)dt &\approx f(-\frac{1}{\sqrt{3}}) + f(\frac{1}{\sqrt{3}}) \\
\int_{-1}^{+1} f(t)dt &\approx \frac{5}{9}f(-\sqrt{\frac{3}{5}}) + \frac{8}{9}f(0) + \frac{5}{9}f(\sqrt{\frac{3}{5}}) \\
&\vdots
\end{aligned} \tag{5.21}$$

Bei anderen Integrationsintervallen wie $[-1, 1]$, bzw. bei Unterteilung des Integrationsintervalls in Teilintervalle und Anwendung der Gauß - Formeln auf jedes Teilintervall müssen die Quadraturgewichte c_i und die Knotenstellen t_i natürlich entsprechend transformiert werden.

Bemerkungen: Es gibt noch weitere Quadraturformeln, die auf ähnlichen Ideen basieren wie die Gauß - Formeln.

Radau-Formeln: Es wird -1 oder 1 in die Menge der Knoten t_i aufgenommen und alle weiteren Knoten und Gewichte dann so gewählt, dass eine Quadraturformel möglichst hoher Ordnung entsteht. Die Ordnung ist um 1 niedriger wie die entsprechende Gauß-Formel der gleichen Knotenanzahl.

Lobatto-Formeln: Es wird -1 und $+1$ in die Menge der Knoten t_i aufgenommen und alle weiteren Knoten und Gewichte bezüglich der erreichbaren Ordnung optimal gewählt. Die Ordnung ist um 2 niedriger wie bei der entsprechenden Gauß-Formel.

Gauß-Kronrod-Formeln: Um adaptive Schrittweitenstrategien zu realisieren, ist es notwendig, Fehlerschätzungen bezüglich der Näherungswerte durchzuführen, siehe Abschnitt 5.6. Dies geschieht entweder dadurch, dass man dieselbe Quadraturformel, die man auf einem Teilintervall angewendet hat, nochmals auf das jeweils halbe Teilintervall anwendet oder dadurch, dass man auf dem Teilintervall noch eine genauere Quadraturformel heranzieht und aus der Differenz *genauer Wert − ungenauerer Wert* auf das Fehlerniveau schließt. In beiden Fällen erweisen sich die irrationalen Gaußknoten als Nachteil:

a) Z.B die Gaußknoten $-\frac{1}{\sqrt{3}}$, $\frac{1}{\sqrt{3}}$. Bei Halbierung des Intervalls sind Funktionsauswertungen an völlig neuen Stellen notwendig, siehe Abbildung 5.3. Die alten Funktionswerte

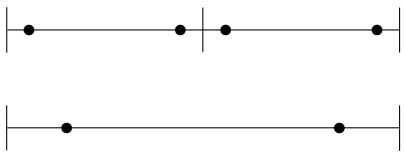


Abbildung 5.2: Intervallhalbierung

können für die Berechnung der genaueren Näherung *nicht* verwendet werden.

b) Auc hbei Verwendung von Formelpaaren, gilt wieder, dass die Gaußknoten der genaueren Gaußformel völlig verschieden sind von den Knoten der ungenaueren Gaußformel.

$$f(-\frac{1}{\sqrt{3}}) + f(\frac{1}{\sqrt{3}})$$

$$\frac{5}{9} \cdot f(-\sqrt{\frac{3}{5}}) - \frac{8}{9} f(0) + \frac{5}{9} f(\sqrt{\frac{3}{5}}).$$

Gauß-Kronrod-Formelpaar: Die $(n+1)$-punktige Gaußformel (ungenaure Formel des Paares) wird durch weitere Knoten und Gewichte zu einer weiteren Quadraturformel (genauere Formel des Paares) ergänzt, d.h. die Funktionsauswertungen in der Gaußformel werden auch in der genaueren Formel verwendet.

5.4 Asymptotische Fehlerentwicklungen

5.4.1 Euler - Maclaurinsche Summenformel

Satz 5.4.1. Für $f \in C^{2m+2}[a,b]$ besitzt die Trapezsumme die Entwicklung

$$T(h) \;=\; \tau_0 + \tau_1 h^2 + \tau_2 h^4 + \cdots + \tau_m h^{2m} + \alpha_{m+1}(h) h^{2m+2} \tag{5.22}$$

mit $\tau_0 := \int_a^b f(t)dt$. Dabei sind die τ_i von h unabhängige Konstanten, die von f abhängen, und $\alpha_{m+1}(h)$ ist eine beschränkte Funktion von h, d.h. $|\alpha_{m+1}(h)| \le M$ für alle $h = \frac{b-a}{n}$, $n \in \mathbb{N}$.

Bemerkungen:

- Die Koeffizienten τ_j können explizit mit Hilfe der sogenannten Bernoulli-Polynome $B_k(t)$ angegeben werden

  $$\tau_j = \frac{B_{2j}(0)}{(2j)!}(f^{(}2j-1)(b) - f^{(}2j-1)(a))$$

  Diese Bernoulli-Polynome $B_k(t)$ $k = 1, 2, \ldots$ sind rekursiv definiert durch

  $$\begin{aligned} &\text{a)} && B_0(t) &&\equiv\; 1 \\ &\text{b)} && B_k'(t) &&=\; kB_{k-1}(t) \quad k \ge 1 \\ &\text{c)} && \textstyle\int_0^1 B_k(t)dt &&=\; 0 \quad k \ge 1 \end{aligned}$$

  Wegen (5.23) hat jedes $B_k(t)$ eine Darstellung der Form

  $$B_k(t) \;=\; A_k + k \int_0^t B_{k-1}(\tau)d\tau, \quad k \ge 1,$$

  wobei die Konstante $A_k = B_k(0)$ so zu bestimmen ist, dass $\int_0^1 B_k(t)dt = 0$ gilt. Es folgt sofort, dass $B_k(t)$ ein Polynom k-ten Grades ist. Insbesondere findet man

  $$\begin{aligned} B_1(t) &= t - \tfrac{1}{2} & \qquad B_2(t) &= t^2 - t + \tfrac{1}{6} \\ B_3(t) &= t^3 - \tfrac{3}{2}t^2 + \tfrac{1}{2}t & B_4(t) &= t^4 - 2t^3 + t^2 - \tfrac{1}{30} \end{aligned}$$

  Es gelten folgende wichtige Eigenschaften:

  $$\begin{aligned} &\text{(i)} && B_k(0) = B_k(1) \quad \text{für } k \ge 2 \\ &\text{(ii)} && \text{Die } B_k(t) \text{ sind für gerades } k \text{ gerade Funktionen} \\ &&& \text{und für ungerades } k \text{ ungerade Funktionen} \\ &&& \text{bezüglich der Stelle } t = \tfrac{1}{2} \\ &\text{(iii)} && B_{2k+1}(0) = B_{2k+1}(1) = 0 \quad \text{für alle } k \ge 1 \end{aligned} \tag{5.24}$$

- Setzt man $g(t) := f(a + th)$ für $0 \le t \le n$, enstspricht einer Intervalltransformation $[a, b] \to [0, n]$, dann ist (5.22) äquivalent zur sogenannten **Euler - MacLaurinschen Summenformel**

$$\frac{g(0)}{2} + g(1) + \cdots + g(n-1) + \frac{g(n)}{2} - \int_0^n g(\tau)d\tau \; = \\ = \sum_{k=1}^{m} \frac{B_{2k}(0)}{(2k)!}[g^{(2k-1)}(n) - g^{(2k-1)}(0)] + R_{m+1} \tag{5.25}$$

$$R_{m+1} \;=\; -\frac{1}{(2m+2)!}\int_0^n [S_{2m+2}(\tau) - S_{2m+2}(0)]g^{(2m+2)}(\tau)d\tau$$

  wobei $S_k(t) := B_k(t - i)$ für $j \le t \le j + 1$ $j = 0, 1, 2, \ldots$

- Im Gegensatz zu (1.15), wo der Verfahrensfehler der Trapezregel nur abgeschätzt wird, beschreibt die Gleichung (5.22) die Struktur des Fehlers. Man spricht von einer **asymptotischen Entwicklung** des Verfahrensfehlers der Trapezregel. Ein viel einfacheres Beispiel einer asymptotischen Entwicklung hatten wir schon in (1.11), siehe S. 14 kennengelernt.

Es sei erwähnt, dass auch bezüglich anderer Quadraturformeln, etwa bezüglich aller Newton - Cotes - Formeln und aller Gauß - Formeln solche asymptotischen Felherentwicklungen existieren.

5.4.2 Hauptanwendung asymptotischer Fehlerentwicklungen

Extrapolationsalgorithmen

Am Beispiel der Trapezregel sieht das folgendermaßen aus, siehe auch Abb. 5.3:

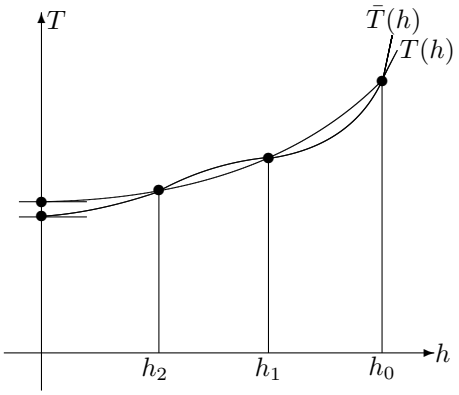


Abbildung 5.3: Trapezsummenextrapolation

Man betrachtet T als Funktion von h. Dann gilt

$$\int_a^b f(t)\,\mathrm{dt} \;=\; \lim_{h\to 0} T(h) \;=\; T(0)$$

Für $h = 0$ ist natürlich $T(0)$ nicht als Trapezsumme definiert sondern nur als $\lim_{h\to 0} T(h)$. Aus $\lim_{h\to 0} T(h) = T(0)$ könnte man schließen, dass es sinnvoll ist, Trapezsummen mit extrem kleinen Schrittweiten als Approximation des gesuchten Integrals zu nehmen, das scheitert aber aus den

bekannten Gründen: Für $h \to 0$ steigt der Rechenaufwand wegen der immer mehr werdenden Funktionsauswertungen gegen Unendlich, und damit auch der Rundungsfehler.

Es liegt daher folgender Gedanke nahe: Man ersetzt die Funktion $T(h)$ durch eine Ersatzfunktion $\bar{T}(h)$, die leicht für $h = 0$ auszuwerten ist und nimmt $\bar{T}(0)$ als Näherungswert für das gesuchte Integral. Konkret denkt man sich bei der **Trapezsummenextrapolation** oder **Rombergintegration** folgende Vorgangsweise. Man berechnet für einige Schrittweiten, in der Abbildung 5.3 für $h_0 > h_1 > h_2 > 0$ die Trapezsummen $T(h_i)$ und interpoliert die Daten $(h_i, T(h_i))$ mit einem Interpolationspolynom $\bar{T}(h)$. Die Auswertung von $\bar{T}(h)$ erfolgt einfach mit dem Neville-Schema. Da der Wert 0 außerhalb des die Knoten umfassenden Intervalls liegt, spricht man hier von *Extrapolation.*

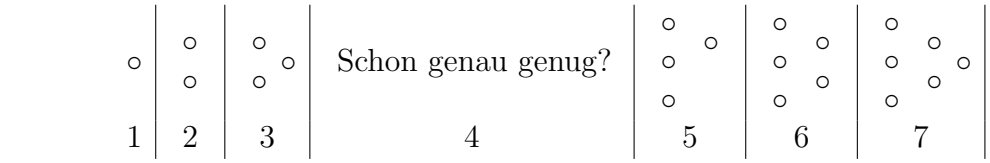
Bemerkungen:

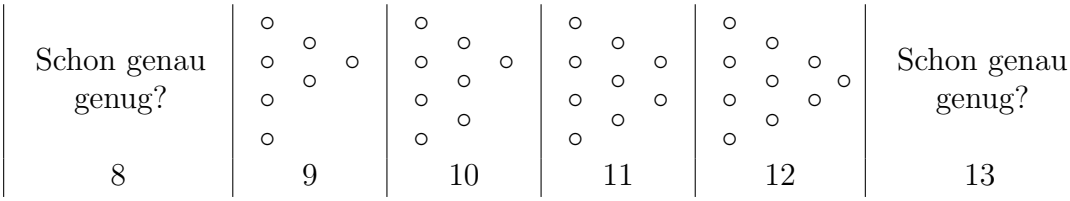
(i) Genaugenommen ist $T(h)$ keine stetige Funktion, wie dies in der Abbildung 5.3 dargestellt ist, sondern $T(h)$ ist nur für die Argumentwerte $h = \frac{b-a}{n}$, $n \in \mathbb{N}$ definiert, die sich in 0 häufen. Dies ist aber offensichtlich ganz irrelevant für die oben beschriebene Idee, die der Trapezsummenextrapolation zu Grunde liegt.

(ii) Wie aus (5.22) ersichtlich, ist $T(h)$ bis auf das Restglied von der Ordnung $O(h^{2m+2})$ ein Polynom in h^2. Dem entspricht auch die waagrechte Tangente von $T(h)$ für $h = 0$ in der Abbildung 5.3. Es liegt daher nahe, $\bar{T}$ nicht als Polynom in h sondern gleich als Polynom in h^2 anzusetzen, sodass das Neville-Schema konkret folgende Gestalt bekommt, siehe (4.21):

$$\begin{aligned} &T_{i,i} := T(h_i) \qquad i = 1, 2, \dots n \\ &T_{i,i+k} := T_{i,i+k-1} + (-h_i^2)\frac{T_{i,i+k-1} - T_{i+1,i+k}}{h_{i+k}^2 - h_i^2} \\ &\bar{T}(0) := T_{0,n} \end{aligned} \tag{5.26}$$

(5.26) ensteht aus (4.21) durch folgende Substitutionen: $p_{i,i+k} \to T_{i,i+k}$, $f_i \to T(h_i)$, $\bar{x} \to 0$, $x_i \to h_i^2$, $x_{i+k} \to h_{i+k}^2$. Natürlich wird man in der praktischen Rechnung n nicht von Haus aus festsetzen weil man ja ein bestimmtes Genaugkeitsniveau ohne unnötigen Rechenaufwand erreichen will. Der zeitliche Ablauf in der praktischen Rechnung wird daher folgender sein:





usw. Man berechnet also immer erst dann eine neue Trapezsumme, die auf dem feineren Gitter zusätzliche Funktionsauswertungen kostet, wenn das Genauigkeitsniveau noch nicht erreicht ist.

Ein weiterer algorithmischer Aspekt ist auch noch die Frage, wie die Schrittweitfolge $h_0, h_1, h_2, \dots$ zweckmäßigerweise zu wählen ist. Dabei sollte man einerseits darauf achten, dass möglichst bereits vorliegende Funktionsauswertungen auch in den späteren Trapezsummen mit den feineren

Gittern verwendet werden, was z.B. durch $h_0 = b - a$, $h_1 = \frac{h_0}{2}$, $h_2 = \frac{h_1}{2}$, $h_3 = \frac{h_2}{2}, \ldots$ gegeben ist, andererseits sollte die Gitterfeinheit nicht zu rasch zunehmen, um Rechenarbeit für die Berechnung neuer $T(h_i)$ zu sparen. Man vewendet daher oft die Folge

$$h_0 = b - a,\ h_1 = \frac{h_0}{2},\ h_2 = \frac{h_0}{3},\ h_3 = \frac{h_0}{4},\ h_4 = \frac{h_0}{6},\ h_5 = \frac{h_0}{8},\ \ldots$$

Es sei hier nur bemerkt, dass mit den Newton - Cotes-Formeln und den Gauß - Radau - Lobatto - und Gauß - Kronrod - Formeln und der Rombergintegration noch nicht alle eindimensionalen Quadraturformeln besprochen worden sind.

Ein Formeltyp von allerdings nur theoretischer Bedeutung sind Qudraturformeln mit gleichen Koeffizienten $c_i = c$. In

$$\int_a^b f(t)\,\mathrm{dt} \;\approx\; c \cdot \sum_{i=0}^{n} f(t_i)$$

werden nur die Knoten t_i so gewählt, dass sich hohe Ordnungen ergeben.

Ein anderes wichtiges Kapitel, das hier ganz unberücksichtigt geblieben ist, sind *uneigentliche Integrale*, also Integrale, die existieren, aber entweder ein unendliches Integrationsintervall haben oder bei denen der Integrand eine oder mehrere Singularitäten aufweist. Singularitäten und unendliches Integrationsintervall können natürlich auch gemeinsam auftreten.

5.5 Mehrdimensionale Integrale

Es werden nur ganz einfache Ideen und nur im zweidimensionalen Fall besprochen. Wir betrachten, siehe Abbidlung 5.4

$$\iint_{\mathbb{G}} f(t_1, t_2)\ \mathrm{dt}_1\,\mathrm{dt}_2 \tag{5.27}$$

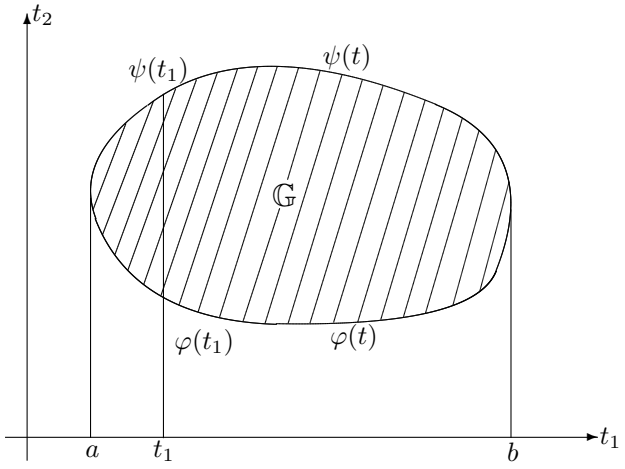


Abbildung 5.4: Integral über das Gebiet $\mathbb{G}$

Es gilt

$$\iint\limits_{\mathbb{G}} f(t_1,t_2)\ \mathrm{dt}_1\,\mathrm{dt}_2 \;=\; \int_a^b \Big[\int_{\varphi(t_1)}^{\psi(t_1)} f(t_1,t_2)dt_2\Big]\,\mathrm{dt}_1. \tag{5.28}$$

D.h. die zweidimensionale Quadratur kann auf die Berechnung des eindimensionalen Integrals

$$\int_a^b g(t_1)dt_1 \quad \text{mit} \quad g(t_1)=\int_{\varphi(t_1)}^{\psi(t_1)} f(t_1,t_2)\,\mathrm{dt}_2$$

zurückgeführt werden. Man kann daher grundsätzlich mit den bereits bekannten Quadraturformeln das Integral $\int_a^b g(t_1)\,\mathrm{dt}_1$ berechnen.

$$\int_a^b g(t_1)\,\mathrm{dt}_1 \;\approx\; c_0 g(t_1^0)+c_1 g(t_1^1)+\cdots+c_n g(t_1^n),$$

wobei bei jedem Funktionsaufruf $g(t_1^i)$ das Integral

$$\int_{\varphi(t_1^i)}^{\psi(t_1^i)} f(t_1^i,t_2)\,\mathrm{dt}_2$$

berechnet werden muss, d.h. wieder eine – meist dieselbe – Quadraturformel verwendet werden muss.

Den Fehler, den man bei der numerischen Berechnung des inneren Integrals $\int_{\varphi(t_1^i)}^{\psi(t_1^i)} f(t_1^i,t_2)\,\mathrm{dt}_2$ macht, kann man als Datenfehler bei der Funktionsauswertung von $g(t_1^i)$ betrachten und kann somit leicht den Gesamtverfahrensfehler abschätzen, wobei man also nur Datenfehlerabschätzungen und Verfahrensfehlersabschätzungen der eindimensionalen Quadratur benötigt.

Beispiel Es soll $\int_{-1}^{+1}\int_{-1}^{+1} e^{\frac{1}{10}(t_1+t_2)}\,\mathrm{dt}_1\,\mathrm{dt}_2$ mit Hilfe der dreipunktige Gauß - Formel berechnet werden:

$$\int_{-1}^{+1} f(t)\,\mathrm{dt} \;\approx\; \frac{5}{9}f(-\sqrt{\frac{3}{5}})+\frac{8}{9}f(0)+\frac{5}{9}f(\sqrt{\frac{3}{5}})$$

$$\begin{aligned}
&\int_{-1}^{+1}\int_{-1}^{+1} e^{\frac{1}{10}(t_1+t_2)}\,\mathrm{dt}_1\,\mathrm{dt}_2 \;\approx\\
&\approx \frac{5}{9}\int_{-1}^{+1} e^{\frac{1}{10}(-\sqrt{\frac{3}{5}}+t_2)}\,\mathrm{dt}_2+\frac{8}{9}\int_{-1}^{+1} e^{\frac{1}{10}t_2}\,\mathrm{dt}_2+\frac{5}{9}\int_{-1}^{+1} e^{\frac{1}{10}(\sqrt{\frac{3}{5}}+t_2)}\,\mathrm{dt}_2 \;\approx\\
&\approx \frac{5}{9}\left[\frac{5}{9}e^{\frac{2}{10}(-\sqrt{\frac{3}{5}})}+\frac{8}{9}e^{\frac{1}{10}(-\sqrt{\frac{3}{5}})}+\frac{5}{9}e^{0}\right]+\\
&\quad+\frac{8}{9}\left[\frac{5}{9}e^{\frac{1}{10}(-\sqrt{\frac{3}{5}})}+\frac{8}{9}e^{0}+\frac{5}{9}e^{\frac{1}{10}(\sqrt{\frac{3}{5}})}\right]+\\
&\quad+\frac{5}{9}\left[\frac{5}{9}e^{0}+\frac{8}{9}e^{\frac{1}{10}\sqrt{\frac{3}{5}}}+\frac{5}{9}e^{\frac{2}{10}\sqrt{\frac{3}{5}}}\right] \;=\\
&= \frac{25}{81}e^{-\frac{1}{5}\sqrt{\frac{3}{5}}}+\frac{80}{81}e^{-\frac{1}{10}\sqrt{\frac{3}{5}}}+\frac{114}{81}e^{0}+\frac{80}{81}e^{\frac{1}{10}\sqrt{\frac{3}{5}}}+\frac{25}{81}e^{\frac{1}{5}\sqrt{\frac{3}{5}}} \;=\\
&= 4.013351122\ldots \qquad \text{Der exakte Integralwert ist } 4.013351122\ldots
\end{aligned}$$

Bei hochdimensionalen Integralen steigt allerdings die Anzahl der Funktionsauswertungen rasch an. Arbeitet man etwa mit einer $(n+1)$-punktigen Gauß - Formel, so hat man bei einem k-dimensionalen Integral $(n+1)^k$ Funktionsauswertungen (z.B. $n = 5$ und $k = 10$ ergibt $(n+1)^k \approx 6 \cdot 10^7$!). Bei hochdimensionalen Integralen scheitern wegen der hohen Rechenzeit und dem schlechten Rundungsfehlerniveau diese einfachen Verfahren. Alternativen sind hier:

(i) Monte - Carlo - Methoden, sogenannte randomisierte Algorithmen

(ii) Methoden, die mit Hilfe von zahlentheoretischen Betrachtungen (Gleichverteilung) hergeleitet werden.

5.6 Aspekte bezüglich praktischer Implementierungen

In den vorhergehenden Abschnitten haben wir zahlreiche Quadraturformeln kennengelernt. Um gute Quadratur-Software zu produzieren, genügt es nicht, einfach diese Quadraturverfahren zu programmieren. Man muss auch umfangreiche Überlegungen anstellen, um u.a. sicherzustellen, dass

(i) das gewünschte Genauigkeitsniveau mit sehr hoher Warscheinlichkeit auch wirklich erreicht wird und dass

(ii) dies ohne unnötigen Rechenaufwand also *effizient* geschieht.

Ein wichtiger Aspekt bezüglich (ii) sind die sogenannten **adaptiven Schrittweitensteuerungen**, die automatisch a-posteriori, also während des Rechnungsverlaufes dafür sorgen, dass in den Bereichen, wo der Integrand welliger ist d.h. größere Ableitungen hat, die Gitterpunkte dichter liegen. Wir deuten all diese Fragen im folgenden nur an:

5.6.1 Fehlerschätzungen

Es soll nicht nur der Näherungswert I für $\int_a^b f(t)\,\mathrm{dt}$ produziert werden, sondern auch eine Schätzung für $e(I) := I - \int_a^b f(t)mboxdt$. Dafür sind i.w. zwei Vorgangsweisen verbreitet:

$(h - \frac{h}{2})$-Kriterium: Wie schon erwähnt, besitzen praktisch alle Quadraturformeln eine asymptotische Fehlerentwicklung, d.h. für ein Verfahren der Ordnung p gilt:

$$e(I) := I_h - \int_a^b f(t)\,\mathrm{dt} \quad = \quad \tau_p h^p + O(h^{p+1}) \tag{5.29}$$

τ_p hängt vom Integranden f und von den Integrationsgrenzen a, b ab und ist daher in konkreten Fällen nicht bekannt. Zur Schätzung des Fehlers liegt nun folgende Vorgangsweise nahe, man berechnet mit derselben Quadraturformel eine Näherung $I_{\frac{h}{2}}$ basierend auf der Schrittweite $\frac{h}{2}$. Für diese Näherung gilt dann

$$I_{\frac{h}{2}} - \int_a^b f(t)\,\mathrm{dt} \quad = \quad \tau_p(\frac{h}{2})^p + O(h^{p+1}). \tag{5.30}$$

Bildet man die Differenz (5.29) − (5.30), so folgt

$$I_h - I_{\frac{h}{2}} \;=\; \tau_p \underbrace{(h^p - \frac{h^p}{2^p})}_{\frac{2^p-1}{2^p}h^p} + O(h^{p+1}) \tag{5.31}$$

$$(I_h - I_{\frac{h}{2}})\frac{1}{2^p-1} \;=\; (\frac{h}{2})^p\tau_p + O(h^{p+1}) \;=\; I_{\frac{h}{2}} - \int_a^b f(t)\,\mathrm{dt} + O(h^{p+1}) \tag{5.32}$$

Man hat also zusätzlich zum genaueren Näherungswert $I := I_{\frac{h}{2}}$ auch die Fehlerschätzung

$$e(I) \;=\; \frac{1}{2^p-1}(I_h - I_{\frac{h}{2}}) \tag{5.33}$$

zur Verfügung.

Formelpaare: Z.B. Gauß-Kronrod. Man veschafft sich zwei Näherungswerte $I^{(1)}$ und $I^{(2)}$ für $\int_a^b f(t)\,\mathrm{dt}$, die auf zwei verschiedenen Quadraturformeln mit unterschiedlicher Ordnung basieren. Zweckmäßigerweise nimmt man immer an, dass in der genaueren Formel auch alle Funktionsauswertungen der ungenaueren Formel verwendet werden und natürlich auch noch einige zusätzliche. Für hinreichend kleine Schrittweiten ist dann

$$I^{(1)} - I^{(2)} \;\approx\; I^{(1)} - \int_a^b f(t)\,\mathrm{dt}$$

In der Praxis nimmt man auch hier die genauere Näherung als endgültigen Näherungswert des Integrals, also $I := I^{(2)}$. Die Größe

$$e(I) \;:=\; I^{(1)} - I^{(2)} \tag{5.34}$$

ist dann i.a. eine starke *Überschätzung* des Fehlers.

5.6.2 Schrittweitensteuerungen

Hier sind zwei Varianten verbreitet: die *globale Strategie* und die *lokale Strategie*.

Globale Strategie: 1. Erster Schritt: Berechnung von $I_{[a,b]}$ und $e(I_{[a,b]})$

$$\text{Abfrage:} \quad |e(I_{[a,b]})| < \varepsilon$$

falls ja: fertig, falls nein:

2. Zweiter Schritt: Berechnung von $I_{[a,\frac{a+b}{2}]},\ I_{[\frac{a+b}{2},b]}$ und $e(I_{[a,\frac{a+b}{2}]}),\ e(I_{[\frac{a+b}{2},b]})$.

$$\text{Abfrage:} \quad |e(I_{[a,\frac{a+b}{2}]})| + |e(I_{[\frac{a+b}{2},b]})| < \varepsilon$$

falls ja: fertig,[1)] falls nein:

[1)] Falls tatsächlich vorzeichenbehaftete Schätzung vorliegt: Abfrage eventuell

$$|e(I_{[a,\frac{a+b}{2}]}) + e(I_{[\frac{a+b}{2},b]})| < \varepsilon$$

3. Es wird jenes Intervall nochmals halbiert, dem die betragsgrößere Fehlerschätzung entspricht usw.

Lokale Strategie: 1. Erster Schritt wie bei globaler Strategie

2. Zweiter Schritt: Berechnung von $I_{[a,\frac{a+b}{2}]},\ I_{[\frac{a+b}{2},b]}$ und $e(I_{[a,\frac{a+b}{2}]}),\ e(I_{[\frac{a+b}{2},b]})$

$$\text{Abfragen:} \quad \begin{aligned} |e(I_{[a,\frac{a+b}{2}]})| &< \frac{1}{2}\varepsilon \\ |e(I_{[\frac{a+b}{2},b]})| &< \frac{1}{2}\varepsilon \end{aligned}$$

Wenn beide Bedingungen erfüllt: fertig.
Wenn nur eine von beiden erfüllt: Das andere Intervall wird weiter halbiert.
Wenn beide nicht erfüllt: Beide Intervalle werden weiter halbiert usw.

Der Vorteil der lokalen Strategie ist ein geringerer Organisationsaufwand, es muss nicht immer wie bei der globalen Strategie ein Sortiervorgang erfolgen. Dies ist aber bei der Geschwindigkeit moderner Rechner kein wesentlicher Aspekt.

Der Nachteil der lokalen Strategie ist, dass unter gewissen Umständen lokal eine ganz sinnlos hohe Genauigkeit erreicht wird, da die Größe ε immer mit der lokalen Intervalllänge gewichtet ist, siehe Abbildung 5.5. In den Unstetigkeitsstellen, wo wegen der geringen Glattheit des Integranden die

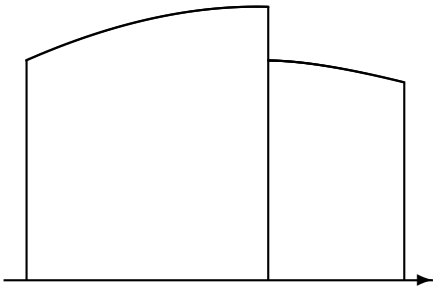


Abbildung 5.5: Unstetigkeitsstelle

Quadraturformeln schlecht arbeiten, werden ganz kurze Intervalle entstehen. Die Abfrage $|e(.)| <$ (extrem kurze Länge) $\cdot\, \varepsilon$ wird sehr schwer zu erfülllen sein und es entstehen dort immer noch kürzere Intervalle, obwohl der Beitrag dieser Intervalle zum Gesamtintegral längst vernachlässigbar ist.

5.7 Ein abschliessendes Zahlenbeispiel

Testbeispiel

$$\int_0^1 e^t \,\mathrm{dt} \;=\; e^t\Big|_0^1 \;=\; e - 1 \;=\; 1.718281828\ldots$$

Vergleich numerischer Verfahren mit annähernd gleichem Rechenaufwand, jeweils 3 Funktionsauswertungen:

Trapezregel mit 2 Interpolationsintervallen $[0, \frac{1}{2}]$, $[\frac{1}{2}, 1]$ $h = \frac{1}{2}$, $t_0 = 0$, $t_1 = h = \frac{1}{2}$, $t_2 = 2h = 1$

$$\begin{aligned}\int_0^1 e^t \,\mathrm{dt} &\approx \frac{1}{2}he^0 + he^{\frac{1}{2}} + \frac{1}{2}he^1 = \\ &= \underline{1.7}53991092\ldots\end{aligned}$$

Simpson: $h = 1$, $t_0 = 0$, $t_1 = \frac{1}{2}$ $t_2 = 1$

$$\begin{aligned}\int_0^1 e^t \,\mathrm{dt} &\approx \frac{1}{6}he^0 + \frac{4}{6}he^{\frac{1}{2}} + \frac{1}{6}he^1 = \\ &= \underline{1.718}861152\ldots\end{aligned}$$

Trapezsummenextrapolation: $h_0 = 1$, $h_1 = \frac{1}{2}$

$$\begin{aligned}T_{0,0} &= 1.859140914\ldots \\ & \qquad\qquad T_{0,1} = \underline{1.718}861151\ldots \\ T_{1,1} &= \underline{1.7}53991092\ldots\end{aligned}$$

Bis auf Rundungsfehler ergibt die Trapezsummenextrapolation und die Simpsonregel denselben Wert.

Gaußformel (dreipunktig) bezüglich $[-1, +1]$:

$$\int_{-1}^{+1} f(x)\,\mathrm{dx} \approx \frac{5}{9}f(-\sqrt{\frac{3}{5}}) + \frac{8}{9}f(0) + \frac{5}{9}f(\sqrt{\frac{2}{5}})$$

Transformation des Intervalls: $-1 \le x \le 1 \;\rightarrow\; a = 0 \le t \le 1 = b$

Transformation der Knoten: $t_i = \frac{a+b}{2} + \frac{b-a}{2}x_i = \frac{1}{2} + \frac{1}{2}x_i$

$$\begin{aligned}x_0 &= -\sqrt{\tfrac{3}{5}} \;\rightarrow\; t_0 = \tfrac{1}{2} - \tfrac{1}{2}\sqrt{\tfrac{3}{5}} \\ x_1 &= 0 \;\rightarrow\; t_1 = \tfrac{1}{2} \\ x_2 &= \sqrt{\tfrac{3}{5}} \;\rightarrow\; t_2 = \tfrac{1}{2} + \tfrac{1}{2}\sqrt{\tfrac{3}{5}}\end{aligned}$$

Transformation der Gewichte:

$$\begin{aligned}c_{i,[-1,+1]} &= \int_{-1}^{+1} \psi_i(x)\,\mathrm{dx} \quad \text{siehe (5.16)} \\ c_{i,[a,b]} &= \int_a^b \psi_{i,[a,b]}(t)\,\mathrm{dt}\end{aligned}$$

Riemannsumme bezüglich $\psi_i(x)$ und analoge Riemannsumme bezüglich $\psi_{i,[a,b]}(t)$

$\Rightarrow \quad c_{i,[a,b]} : c_{i,[-1,+1]} = (b-a) : 2$

$$\Rightarrow \quad c_{i,[a,b]} = \frac{b-a}{2}c_{i,[-1,+1]} = \frac{1}{2}c_{i,[-1,+1]}$$

$$\begin{aligned}\int_0^1 f(t)\;\mathrm{dt} &\approx \frac{5}{9}\frac{1}{2}f(\frac{1}{2} - \frac{1}{2}\sqrt{\frac{3}{5}}) + \frac{8}{9}\frac{1}{2}f(\frac{1}{2}) + \frac{5}{9}\frac{1}{2}f(\frac{1}{2} + \frac{1}{2}\sqrt{\frac{3}{5}}) = \\ &= \frac{5}{18}e^{(\frac{1}{2} - \frac{1}{2}\sqrt{\frac{3}{5}})} + \frac{4}{9}\,e^{\frac{1}{2}} + \frac{5}{18}e^{(\frac{1}{2} + \frac{1}{2}\sqrt{\frac{3}{5}})} = \underline{1.7182809}051\ldots\end{aligned}$$

Kapitel 6

Numerische Lösung von Differentialgleichungen

6.1 Anfangswertprobleme

Wir betrachten zunächst Anfangswertprobleme von gewöhnlichen Differentialgleichungen (ODE):

$$y'(t) = \frac{dy}{dt} = f(t, y(t)), \quad f : \mathcal{D} \subset [t_0, t_{end}] \times \mathbb{R}^n \to \mathbb{R}^n \tag{6.1a}$$

$$y(t_0) = y_0, \quad \text{Lösung } y(t) : [t_0, t_{end}] \to \mathbb{R}^n \tag{6.1b}$$

Der folgende Satz gibt eine Aussage über Existenz und Eindeutigkeit von Lösungen von (6.1):

Satz 6.1.1 (Picard-Lindelöf)**.** Sei $\mathcal{D} := \{(t, y) : t_0 \le t \le a, \|y - y_0\| \le b\}$ und die Funktion $f : \mathcal{D} \to \mathbb{R}^n$ stetig und beschränkt mit $\|f(t, y)\| \le M$ in $\mathcal{D}$. Erfüllt f eine Lipschitzbedingung in y mit Lipschitzkonstante L,

$$\|f(t, y_1) - f(t, y_2)\| \le L\|y_1 - y_2\| \qquad (t, y_i) \in D \quad i = 1, 2.$$

so existiert genau eine Lösung $y(t)$ von (6.1) für $t_0 \le t \le T := \min(a, \frac{b}{M})$.

L ist auch ein wichtiger Parameter für die Sensitivität der Lösung:

Satz 6.1.2 (Konditionsabschätzung für Anfangswertprobleme)**.** Betrachten (6.1) und ein gestörtes Problem

$$\tilde{y}'(t) = \frac{d\tilde{y}}{dt} = \tilde{f}(t, \tilde{y}(t)) \tag{6.2a}$$

$$\tilde{y}(t_0) = \tilde{y}_0. \tag{6.2b}$$

$f, \widetilde{f} : \mathcal{D} \to \mathbb{R}^n$ erfüllen die Anforderungen des Satzes von Picard-Lindelöf. Es gelte

$$\|y_0 - \widetilde{y_0}\| \le \delta_0 \tag{6.3}$$

$$\|f(t, y) - \widetilde{f}(t, y)\| \le \delta \quad \text{in } D, \tag{6.4}$$

dann erfüllen die beiden Lösungen $y(t)$ und $\tilde{y}(t)$ die folgende Ungleichung

$$\|y(t) - \widetilde{y}(t)\| \leq e^{Lt}\delta_0 + \frac{e^{Lt}-1}{L}\delta \text{ für } t \in [0,T]. \tag{6.5}$$

Beweis: Lemma von Gronwall

Lemma 6.1.3. (Gronwall) Angenommen die Funktion $v : [0,T] \to \mathbb{R}$ erfüllt

$$v'(t) \leq \omega v(t) + \delta \qquad t \in [0,T] \tag{6.6a}$$
$$v(0) \leq \delta_0 \tag{6.6b}$$

mit δ, $\delta_0 \geq 0$. Dann gilt

$$v(t) \leq e^{\omega t}\delta_0 + \frac{e^{\omega t}-1}{\omega}\delta \qquad t \in [0,T]$$

Beweis: Multiplikation von (6.6a) mit $e^{\omega t}$ führt auf $(e^{-\omega t}v(t))' \leq e^{-\omega t}\delta$, Integration und Multiplikation mit $e^{\omega t}$ liefert die Behauptung. □

Beispiel 6.1.4.

$$y' = \lambda y \tag{6.7a}$$
$$y(0) = y_0 \qquad \text{Störung:} \quad y(0) = y_0 + \delta_0 \tag{6.7b}$$

Die exakte Lösung ist $y(t) = y_0 e^{\lambda t}$.

$$y_0 e^{\lambda t} - (y_0 + \delta_0)e^{\lambda t} = \underline{\delta_0 e^{\lambda t}}$$

Mit $L = |\lambda|$ ergibt sich aus obiger Abschätzung $\|y(t) - \widetilde{y}(t)\| \leq e^{|\lambda|t}\delta_0$ was für $\lambda < 0$ extrem unrealistisch ist.

Das Anfangswertproblem

$$y' = \lambda y, \qquad y(0) = y_0, \qquad \text{mit } \lambda \ll 0$$

ist ein sehr einfaches sogenanntes **steifes** Problem. Es zeigt das folgende charakteristische Verhalten

(i) Die Lipschitzkonstante $L = |\lambda|$ ist sehr groß.

(ii) Für $y_0 = 0$ ist die Lösung $y(t) \equiv 0$ glatt.

(iii) Für $y_0 \neq 0$ fällt die Lösung $y(t) = e^{\lambda t}y_0$ unmittelbar nach dem Start rasch ab, ist also unglatt (dh. Ableitungen sind sehr groß). Weg von $t = 0$ wird die Lösung wieder schnell glatt und nähert sich der Lösung (ii).

Das ist genau die Situation wo Konditionsabschätzungen basierend auf der üblichen Lipschitzkonstanten L extrem unrealistisch sind, obwohl diese Probleme sehr gut konditioniert sind. Störungen werden sehr schnell weggedämpft.

Für allgemeine Probleme kann **Steifheit** folgendermaßen definiert werden: Ein System von gewöhnlichen Differentialgleichungen heißt *steif*, wenn die Jacobimatrix $f_y = (\frac{\partial f_i}{\partial x_j})$ für $i, j = 1, 2, 3, ..., n$ in

der Nähe der Lösung Eigenwerte λ_k mit Realteil $\lambda_k \ll 0$ hat, neben Eigenwerten von moderater Größenordnung. Zur Lösung solcher Probleme sind **impliziten Verfahren** besser geeignet.

Da die klassische Lipschitzkonstante für steife Probleme im allgemeinen extrem groß ist, wird für Abschätzungen im Zusammenhang mit steifen Probleme die sogenannte *einseitige Lipschitzkonstante* $m \in \mathbb{R}$ verwendet:

Eine Funktion $f : \mathcal{D} \to \mathbb{R}^{\backslash}$ erfüllt eine einseitige Lipschitzkonstante bezüglich y in $\mathcal{D}$ falls

$$\langle y - \tilde{y}, f(t,y) - f(t,\tilde{y})\rangle \leq m\|y - \tilde{y}\|_2^2$$

für $(t,y),(t,\tilde{y}) \in \mathcal{D}$.

6.2 Euler Verfahren; Konsistenz, Stabilität, Konvergenz

Entwickeln wir die Lösung $y(t)$ von (6.1) im Punkt $t+h$ in eine Taylorreihe und setzen $y'(t) = f(t,y(t))$ ein, so erhalten wir

$$y(t+h) = y(t) + hf(t,y(t)) + O(h^2) \approx y(t) + hf(t,y(t)).$$

Das motiviert das folgende einfachste Diskretisierungsverfahren:

$$\begin{aligned} \eta_0 &= y_0 && (6.8a)\\ \eta_1 &= \eta_0 + hf(t_0,\eta_0) && (6.8b)\\ \frac{\eta_\nu - \eta_{\nu-1}}{h} &= f(t_{\nu-1},\eta_{\nu-1}) \qquad t_\nu = t_0 + \nu h \qquad \nu = 1,2,... && (6.8c)\end{aligned}$$

(Die Schrittweite h kann auch variabel sein.) Dabei sind $\eta_1 \approx y(t_1), \eta_2 \approx y(t_2), ...$ Näherungen für die exakte Lösung $y(t)$ zum Zeitpunkt t_1, t_2,... .

Ein **explizites** Verfahren schließt von $\eta_{\nu-1}$ auf η_ν.

$$\eta_\nu = \eta_{\nu-1} + \mathrm{h}f(t_{\nu-1},\eta_{\nu-1}) \qquad \nu = 1,2,\ldots$$

Der Fehler in einem Schritt wird als **lokaler (Diskretisierungs)fehler** l_ν bezeichnet.

$$\begin{aligned} y(t_\nu) - y(t_{\nu-1}) - hf(t_{\nu-1},y(t_{\nu-1})) &= l_\nu h && (6.9a)\\ \frac{y(t_\nu) - y(t_{\nu-1})}{h} - f(t_{\nu-1},y(t_{\nu-1})) = \frac{l_\nu}{h}h &= l_\nu && (6.9b)\end{aligned}$$

Das explizite Eulerverfahren ist **konsistent** für $y(t)$ (falls y zweimal stetig differenzierbar ist), d.h. $\|l_\nu\| \to 0$ falls $h \to 0$,

$$l_\nu = \frac{y(t_\nu) - y(t_{\nu-1})}{h} - y'(t_{\nu-1}) = h\int_0^1 y''(t_{\nu-1} + \theta h)(1-\theta)\mathrm{d}\theta \qquad 0 < \theta < 1,$$

also:

$$\|l_\nu\| = O(h) = O(h^p) \quad p = 1 \qquad (6.10)$$

Die **Konsistenzordnung** für das explizite Eulerverfahren ist 1

$$\| l_\nu \| \leq \frac{M_2 \mathrm{h}}{2}$$

wobei M_2 eine Schranke für $\|y''(t)\|$ ist.

Globaler (Diskretisierung)fehler $e_\nu = \eta_r - y(t_r)$
Ein Diskretisierungsverfahren heißt **konvergent**, falls $\|e_\nu\| \to 0$ für $h \to 0$.

$$\|e_\nu\| = O(\mathrm{h}^p) \quad p = 1, 2, ...$$

Die **Konvergenzordnung** für das explizite Eulerverfahren ist $p = 1$.

Aus Konsistenz eines Dirkretisierungsverfarens folgt nicht notwendigerweise Konvergenz. Man braucht zusätzlich die Eigenschaft der **Stabilität**, dass der globale Effekt der lokalen Fehler gleichmäßig für $h \to 0$ beschränkt bleibt.

Betrachten wir zwei parallele Schritte eines Diskretisierungsverfahrens:

$$(t_{\nu-1}, \eta_{\nu-1}) \to (t_\nu, \eta_\nu)$$
$$(t_{\nu-1}, \tilde{\eta}_{\nu-1}) \to (t_\nu, \tilde{\eta}_\nu).$$

Ein Verfahren heißt **stabil**, falls

$$\|\eta_\nu - \tilde{\eta}_\nu\| \leq (1 + Sh)\|\eta_{\nu-1} - \tilde{\eta}_{\nu-1}\|$$

gleichmäßig für $h < h_0$ gilt, wobei die Konstante S unabhängig von h ist. Für das Eulerverfahren gilt $S = L$.

Theorem (Konvergenz des Eulerverfahrens):
Sei $y(t) \in C^2[0, t_{end}]$ und $M_2 = \sup_{t \in [0, t_{end}]} \|y''(t)\|$. Dann gilt

$$\|e_\nu\| = \|\eta_\nu - y(t_\nu)\| \leq e^{Lt_\nu}\|e_0\| + \frac{e^{Lt_\nu} - 1}{L}\frac{M_2 h}{2}$$

wobei $e_0 = \eta_0 - y(t_0) = O(h)$.

Beweis: Lemma von Gronwall

Lemma 6.2.1. (Gronwall, diskrete Version)
Angenommen die nichtnegative Folge (ζ_ν) $\nu = 0, 1, 2, ...$ erfüllt

$$\begin{aligned} \zeta_0 &\leq \delta_0 \\ \zeta_\nu &\leq (1 + \omega)\zeta_{\nu-1} + \delta, \qquad \nu = 1, 2, 3, ... \end{aligned}$$

mit ω, δ_0, $\delta \geq 0$. Dann gilt

$$\zeta_\nu \leq e^{\nu\omega}\delta_0 + \frac{e^{\nu\omega} - 1}{\omega}\delta \qquad \text{für alle } \nu.$$

Beweis: Die Rekursion

$$\begin{aligned} \zeta_1 &\leq (1+\omega)\delta_0 + \delta \\ \zeta_2 &\leq (1+\omega)\zeta_1 + \delta \leq (1+\omega)^2\delta_0 + (1+\omega)\delta + \delta \\ &\vdots \end{aligned} \tag{6.12}$$

(6.13)

führt auf

$$\begin{aligned} \zeta_\nu &\leq (1+\omega)^\nu\delta_0 + (1 + (1+\omega) + ... + (1+\omega)^\nu)\delta \\ &= (1+\omega)^\nu\delta_0 + \frac{(1+\omega)^\nu - 1}{\omega}\delta(1+\omega)\delta + \delta \\ &\leq e^{\nu\omega}\delta_0 + \frac{e^{\nu\omega} - 1}{\omega}\delta \end{aligned} \tag{6.14}$$

für $1 + \omega \leq e^\omega$ und $\omega > 0$.

Bemerkungen:

- ◇ Konsistenz + Stabilität = Konvergenz
- ◇ auch für variables h entsprechende Abschätzungen möglich
- ◇ Wachstumsfaktoren wie in Konditionsabschätzungen

Beim **impliziten Euler Verfahren**

$$\eta_\nu = \eta_{\nu-1} + hf(t_\nu, \eta_\nu) \qquad \nu = 1, 2, ... \tag{6.15}$$

muss in jedem Schritt ein (nicht)lineares Gleichungssystem gelöst werden. Wendet man dieses Verfahren auf das Modellproblem $y' = \lambda y$ mit Anfangswert δ_0 an, ergibt sich

$$\eta_\nu = (\frac{1}{1 - h\lambda})^\nu\delta_0$$

Für $\lambda \ll 0$ ist $1/(1 - h\lambda)^\nu$ klein, auch für nicht so kleine Schrittweiten h und daher spiegelt (6.15) das Verhalten von $e^{\lambda t}\delta_0$ sehr gut wider.

6.3 Einschrittverfahren allgemein

Ein allgemeines explizites Einschrittverfahren:

$$\eta_0 = y_0 \qquad t_\nu = h\nu \quad \text{oder variables } h \tag{6.16}$$

$$\frac{\eta_\nu - \eta_{\nu-1}}{h} = \underbrace{\varphi(t_{\nu-1}, \eta_{\nu-1}; h)}_{\text{Inkrementfunktion z.B. } \varphi(t,y;h) = hf(t,y)} \qquad \nu = 1, 2, ... \tag{6.17}$$

mit geeigneter Wahl von φ, sodass Konvergenzordnung $p > 1$.

Die Definition des allegemeinen Einschrittverfahren (6.16) erlaubt es eine Rekursion für $\eta_{\nu-1} \to \eta_\nu$ in der Form

$$\eta_\nu = \eta_{\nu-1} + h\varphi(t_{\nu-1}, \eta_{\nu-1}; h)$$

Lokaler (Diskretisierungs)fehler:

$$l_\nu := \frac{y(t_\nu) - y(t_{\nu-1})}{h} - \varphi((t_{\nu-1}), y(t_{\nu-1}); h),$$

hl_ν ist die Differenz zwischen exaktem Lösungswert $y(t_\nu)$ und Näherung mit Startwert $y(t_{\nu-1})$.

Die Wahl der Inkrementfunktion φ erlaubt es höhere Konvergenzordnungen $p > 1$ zu realisieren. Die Konstruktion von Inkrementfunktionen basiert auf der Approximation des Integrals in

$$y(t_\nu) = y(t_{\nu-1}) + \int_{t_{\nu-1}}^{t_\nu} f(t, y(t))\mathrm{dt}.$$

Das Euler Verfahren arbeitet mit der einfachen Approximation

$$\int_{t_{\nu-1}}^{t_\nu} f(t, y(t))dt \approx hf(t_{\nu-1}, y(t_{\nu-1}))$$

Eine bessere Approximation kann durch zusätzliche Funktionsauswertungen an Zwischenwerten im Intervall $[t_{\nu-1}, t_\nu]$ erreicht werden. Diese Funktionswerte sind zwar unbekannt, können aber geschätzt werden. Ein einfaches Beispiel dazu ist das **verbesserte Euler Verfahren**

$$\begin{aligned} Y_1 &= \eta_{\nu-1} + \frac{h}{2} f(t_{\nu-1}, y(t_{\nu-1})) \\ \eta_\nu &= \eta_{\nu-1} + hf(t_{\nu-1} + \frac{h}{2}, Y_1) \end{aligned}$$

mit $\varphi(t, y; h) = f(t + \frac{h}{2}, y + \frac{h}{2} f(t, y))$

Das *Verfahren von Heun* ist ein explizites Einschrittverfahren zweiter Ordnung

$$\begin{aligned} Y_1 &= f(t_{\nu-1}, \eta_{\nu-1}) \\ Y_2 &= f(t_{\nu-1} + h, \eta_{\nu-1} h Y_1) \\ \eta_\nu &= \eta_{\nu-1} + h\varphi(t_{\nu-1}, \eta_{\nu-1}, h) \\ \varphi(t_{\nu-1}, \eta_{\nu-1}, h) &= \frac{Y_1 + Y_2}{2} \end{aligned}$$

Die Funktion φ ist also ein Mittelwert zweier Anstiege.

Eine Verallgemeinerung dieser Idee führt auf die **Runge-Kutta Verfahren**. Hier werden für die Berechnung $\eta_{\nu-1} \to \eta_\nu$, $s \in \mathbb{N}$ definierte Zwischenapproximationen oder Stufen Y_i an den Stellen $\tau_i = t_{\nu-1} + c_i h,\ i = 1, 2, ...s,\ c_1 = 0$ verwendet.

$$\begin{aligned} Y_1 : &= \eta_{\nu-1} \qquad s \text{ Stufen } Y_i,\ \tau_i = t_{\nu-1} + c_i h,\ i = 1, 2, ...s,\ c_1 = 0 \\ Y_2 : &= \eta_{\nu-1} + ha_{21} f(\tau_1, Y_1) \\ Y_3 : &= \eta_{\nu-1} + h(a_{31} f(\tau_1, Y_1) + a_{32} f(\tau_2, Y_2)) \\ &\vdots \\ Y_s : &= \eta_{\nu-1} + h(a_{s1} f(\tau_1, Y_1) + ha_{s2} f(\tau_2, Y_2) + ... + ha_{s,s-1} f(\tau_{s-1}, Y_{s-1})) \\ \eta_\nu &= \underbrace{\eta_{\nu-1} + h(hb_1 f(\tau_1, Y_1) + b_2 f(\tau_2, Y_2) + ... + b_s f(\tau_s, Y_s))}_{\varphi(t_{\nu-1}, \eta_\nu - 1; h)} \end{aligned}$$

Die Runge-Kutta Verfahren werden durch die sogenannten **Butcher Tableaus** dargstellt:

$$\begin{array}{c|c} c & A \\ \hline & b^T \end{array}$$

mit

$$A = \begin{pmatrix} 0 & \dots & \dots & 0 \\ a_{21} & 0 & \dots & 0 \\ a_{31} & a_{32} & 0 & \dots \\ \vdots & \vdots & \ddots & 0 \\ a_{s1} & a_{s2} & \dots & a_{s,s-1} \end{pmatrix},$$

$c = (c_1 = 0, c_2, c_3, ..., c_s)^T$ und $b = (b_1, b_2, ...b_s)^T$.

Im folgenden ist das Butcher Tableau für das Euler Verfahren und das 4-stufige Runge-Kutta Verfahren der Ordnung $p = 4$ angegeben:

$$\begin{array}{c|c} 0 & \\ \hline & 1 \end{array} \qquad \begin{array}{c|cccc} 0 & & & & \\ \frac{1}{2} & \frac{1}{2} & & & \\ \frac{1}{2} & 0 & \frac{1}{2} & & \\ 1 & 0 & 0 & 1 & \\ \hline & \frac{1}{6} & \frac{1}{3} & \frac{1}{3} & \frac{1}{6} \end{array}$$

Ein **implizites s-stufiges Runge-Kutta Verfahren** mit den Stufen Y_i an den Stellen $\tau_i = t_{\nu-1} + c_i h,\ i = 1, 2, ...s,\ c_1 = 0$ ist durch das folgende $n \times s$ System von nichtlinearen Gleichungen gegeben:

$$\begin{aligned} Y_1 : &= \eta_{\nu-1} + h(a_{11} f(\tau_1, Y_1) + h a_{12} f(\tau_2, Y_2) + ... + h a_{1s} f(\tau_s, Y_s)) \\ Y_2 : &= \eta_{\nu-1} + h(a_{21} f(\tau_1, Y_1) + h a_{22} f(\tau_2, Y_2) + ... + h a_{2s} f(\tau_s, Y_s)) \\ &\vdots \\ Y_s : &= \eta_{\nu-1} + h(a_{s1} f(\tau_1, Y_1) + h a_{s2} f(\tau_2, Y_2) + ... + h a_{ss} f(\tau_s, Y_s)) \\ \eta_\nu &= \underbrace{\eta_{\nu-1} + h(h b_1 f(\tau_1, Y_1) + b_2 f(\tau_2, Y_2) + ... + b_s f(\tau_s, Y_s))}_{\varphi(t_{\nu-1}, \eta_\nu - 1; h)} \end{aligned}$$

Das implizite Euler Verfahren wird durch das Butcher Tableau

$$\begin{array}{c|c} 1 & 1 \\ \hline & 1 \end{array}$$

charakterisiert. Der **impliziten Mittelpunktsregel** mit $\eta_\nu = \eta_{\nu-1} + h f(t_{\nu-1} + \frac{h}{2}, \frac{1}{2}(\eta_\nu + \eta_{\nu-1})$ entspricht das Butcher Tableau

$$\begin{array}{c|c} \frac{1}{2} & \frac{1}{2} \\ \hline & 1 \end{array}$$

6.4 Lineare Mehrschrittverfahren allgemein

Einschrittverfahren basieren auf der einfachen Rekursion $\eta_{\nu-1} \to \eta_\nu$ und verwenden keine Informationen aus den vorhergehenden Schritten. Um höhere Ordnungen zu erreichen sind zusätzliche Aus-

wertungen der rechten Seite $f(t,y)$ notwendig. Das ist aber ineffizient falls Funktionsauswertungen sehr aufwändig sind!

Ein **lineares Mehrschrittverfahren** zur Lösung eines Anfangswertproblems ist von der Form

$$\underbrace{\frac{1}{h}(\alpha_0\eta_{\nu-k}+...+\alpha_{k-1}\eta_{\nu-1}+\alpha_k\eta_\nu)}_{\approx y'} = \underbrace{\beta_0 f(t_{\nu-k},\eta_{\nu-k})+...+\beta_{k-1}f(t_{\nu-1},\eta_{\nu-1})+\beta_k f(t_\nu,\eta_\nu),}_{\varphi(t_{\nu-k},\eta_{\nu-k},...,t_\nu,\eta_\nu;h)}$$

Das ist eine k-Schritt Rekursion $\eta_{\nu-k},...,\eta_{\nu-1}\to\eta_\nu$. Dieses Verfahren heißt linear, weil die $f(t_i,\eta_i)$ und die η_i auf der linken Seite nur linear vorkommen. Es gilt $\sum\alpha_i=0, \alpha_k\neq 0$, $\varphi(t_{\nu-k},\eta_{\nu-k},...,t_\nu,\eta_\nu;h)$ ist die **Inkrementfunktion** . Für dieses k-Schrittverfahren ist nur eine weitere f-Auswertung pro Schritt erforderlich. Es werden aber $k-1$ Startwerte benötigt. Falls $\beta_k=0$ heißt das Verfahren **explizit**, für $\beta_k\neq 0$ **implizit**.

Beispiele:

$$\frac{\eta_\nu-\eta_{\nu-1}}{h} = f(t_{\nu-1},\eta_{\nu-1}) \qquad \text{explizites Eulerverfahren} \tag{6.20}$$

$$\frac{\eta_\nu-\eta_{\nu-1}}{h} = f(t_\nu,\eta_\nu) \qquad \text{implizites Eulerverfahren} \tag{6.21}$$

Für $\alpha_k=1,\ \alpha_{k-1}=-1$ und alle übrigen $\alpha_i=0$ erhält man eine wichtige Spezialklasse von linearen Mehrschrittverfahren (Adamsverfahren):

$$\frac{\eta_\nu-\eta_{\nu-1}}{h}=\beta_0 f(\eta_{\nu-k},t_{\nu-k})+\cdots+\beta_{k-1}f(\eta_{\nu-1},t_{\nu-1})+\beta_k f(\eta_\nu,t_\nu).$$

Explizite Adamsverfahren (Adams-Bashford):

k	β_0	β_1	β_2	β_3	β_4	p
1	1	0				1
2	$-\frac{1}{2}$	$\frac{3}{2}$	0			2
3	$\frac{5}{12}$	$-\frac{16}{12}$	$\frac{23}{12}$	0		3
4	$-\frac{9}{24}$	$\frac{37}{24}$	$-\frac{59}{24}$	$\frac{55}{24}$	0	4

Implizite Adamsverfahren (Adams-Moulton):

k	β_0	β_1	β_2	β_3	β_4	p
1	$\frac{1}{2}$	$\frac{1}{2}$				2
2	$-\frac{1}{12}$	$\frac{8}{12}$	$\frac{5}{12}$			3
3	$\frac{1}{24}$	$-\frac{5}{24}$	$\frac{19}{24}$	$\frac{9}{24}$		4
4	$-\frac{19}{720}$	$\frac{106}{720}$	$-\frac{264}{720}$	$\frac{646}{720}$	$\frac{251}{720}$	5

Beispiel: y'=y+t auf [0,1] mit $y(0)=1$
exakte Lösung: $y(t)=2e^t-(t+1)$, daher $y(1)=3.436563$
Schrittweite: $h=0.2$; jeweils 5 Schritte

1. **Implizites Adamsverfahren der Ordnung 2 (Trapezregel)**

$$\eta_{\nu+1} = \eta_\nu + 0.1(t_{\nu+1} + \eta_{\nu+1} + \eta_\nu + t_\nu)$$

t_ν	0	0,2	0,4	0,6	0,8	1
η_ν	1	1.2444	1.5877	2.0516	2.663	3.4548

2. **Explizites Adamsverfahren der Ordnung 2:**

$$\eta_{\nu+1} = \eta_\nu + 0.1(3(t_\nu + \eta_\nu) - (t_{\nu+1} + \eta_{\nu+1}))$$

t_ν	0	0,2	0,4	0,6	0,8	1
η_ν	1	1.24	1.572	2.0196	2.608	3.369

Eine wichtige Klasse von implziten Mehrschrittverfahren sind die **Backward Differentiation Formulas (BDF)**.

Lokaler Diskretisierungsfehler:

$$l_\nu = \frac{1}{h}\sum_{i=0}^{k}(\alpha_i y(t_{\nu-k+i}) - h\beta_i f(t_{\nu-k+i}, y(t_{\nu-k+i}))) \tag{6.22}$$

$$= \frac{1}{h}\sum_{i=0}^{k}(\alpha_i y(t_{\nu-k+i}) - h\beta_i y'(t_{\nu-k+i}))). \tag{6.23}$$

Die Konsistenz, also die Ordnung von l_ν, ist bei Mehrschrittverfahren im Gegensatz zu Einschrittverfahren relativ einfach zu studieren (Theorie von Differenzengleichungen).

Prediktor-Korrektor Verfahren

Lässt sich die Gleichung für $\eta_{\nu+1}$ des impliziten Verfahrens nicht explizit machen, dann kann man sie durch ein Iterationsverfahren lösen. Bei einem relativ genauen Startwert $\eta^{(0)}_{\nu+1}$ wird man schon durch eine Iteration die Lösung $\eta_{\nu+1}$ der impliziten Gleichung ziemlich exakt erhalten. Den Startwert $\eta^{(0)}_{\nu+1}$ bestimmt man mit einem expliziten Verfahren. Das ergibt z.B.:

$$\eta^{(0)}_{\nu+1} = \eta_\nu + \frac{h}{2}(3(f(t_\nu, \eta_\nu) - f(t_{\nu-1}\eta_{\nu-1}))$$

$$\eta_{\nu+1} = \eta_\nu + \frac{h}{12}(5f(t_{\nu+1}, \eta^{(0)}_{\nu+1}) + 8f(t_\nu, \eta_\nu) - f(t_{\nu-1}, \eta_{\nu-1}))$$

Die erste Gleichung heißt Prediktor, die zweite Korrektor, das Verfahren heißt **Prediktor-Korrektor Verfahren**. (Aus den beiden Werten $\eta^{(0)}_{\nu+1}$ und $\eta_{\nu+1}$ kann man den lokalen Fehler abschätzen.)

Literaturverzeichnis

[1] W. Auzinger, D. Praetorius: Numerische Mathematik, Skriptum für TM, TU Wien, 2007.

[2] G. Bärwolff: *Numerik für Ingenieure, Physiker und Informatiker*, Elsevier, Spektrum akademischer Verlag, 2007.

[3] R. Plato: *Numerische Mathematik kompakt*, Vieweg Verlag, 2000.

[4] H. Schneider: *Numerik für Informatiker*, Springer Verlag, 2002.

Index