Numerik-Vorlesungen

Teil 1

Analyse, Interpolation, Differentiation, Nullstellen

Peter Szyler, Horst Hollatz

Letzte Änderung: February 3, 2016

Contents

Chapter 1

Analyse-Grundlagen numerischer Verfahren

1.1. Einführung

Die numerische Mathematik vermittelt zwischen verschiedenen mathematischen Gebieten; sie dient dem Lösen von Aufgaben aus klassischen Bereichen der Mathematik. Sie untersucht Folgen, die sich aus dem Übertragen mathematischer Methoden auf Rechner ergeben, sofern mit diesen Methoden Aufgaben auf in sich dichten Zahlbereichen zu lösen sind. Die Folgen betreffen sowohl die mathematischen Methoden als auch die Ergebnisse numerischen Rechnens:

1. Entwickeln implementierbarer Algorithmen zum Lösen mathematischer Aufgaben; das sind solche Algorithmen, die man als Grundlage für ein Rechenprogramm verwenden kann.

2. Bewerten von Algorithmen hinsichtlich Effizienz, Konvergenz, Störverhalten, Rundungsfehlereinfluss und ähnlichem. Hier sind zwei Aspekte zu beachten: Einerseits geht es um die mathematische Güte der Algorithmen, andererseits um die Güte des entsprechenden Rechenprogramms.

Um diese Fragen zu motivieren, betrachten wir einige Beispiele.

1.1. Beispiel: Es ist die Größe $w = 9x^4 - y^4 + 2y^2$ für $x = 40545$ und $y = 70226$ zu berechnen. Wir nutzen die folgenden vier mathematisch gleichwertigen Formeln:

$$\begin{aligned} w_1 &= (9 \cdot x \cdot x \cdot x \cdot x - y^4) + 2 \cdot y \cdot y, \\ w_2 &= (3 \cdot x \cdot x - y \cdot y) \cdot (3 \cdot x \cdot x + y \cdot y) + 2 \cdot y \cdot y, \\ w_3 &= 9x^4 + (2 \cdot y \cdot y - y \cdot y \cdot y \cdot y), \\ w_4 &= (9x^4 + 2 \cdot y \cdot y) - y^4. \end{aligned}$$

Bei siebenstelliger dezimaler Rechnung erhalten wir folgende Ergebnisse:

$$\begin{aligned} w_1 &= -9.990137 \cdot 10^{12}, \\ w_2 &= 9.863382 \cdot 10^{9}, \\ w_3 &= 0.000000, \\ w_4 &= -1.000000 \cdot 10^{13}. \end{aligned}$$

Bei vollem Ausnutzen der Stellenzahl eines Taschenrechners (intern etwa 12 Dezimalstellen) und anschließendem Runden auf sieben Dezimalstellen erhalten wir:

$$\begin{aligned} w_1 &= -9.366178 \cdot 10^{8}, \\ w_2 &= 1.000000 \cdot 10^{0}, \\ w_3 &= 1.800000 \cdot 10^{9}, \\ w_4 &= 8.000000 \cdot 10^{8}. \end{aligned}$$

Das sind acht unterschiedliche Resultate. Ähnlich unterschiedlich sind die Ergebnisse eines *C*-Programms auf einer Workstation. Welches Ergebnis kommt dem wahren am nächsten? Vertauscht man die Eingabedaten x und y, erhält man die folgenden Resultate:

$$\begin{array}{rcll} & & 7-stellig & 12-stellig \\ w_1 &=& 2.161918 \cdot 10^{20} & 2.161918 \cdot 10^{20}, \\ w_2 &=& 2.161917 \cdot 10^{20} & 2.161918 \cdot 10^{20}, \\ w_3 &=& 2.161918 \cdot 10^{20} & 2.161918 \cdot 10^{20}, \\ w_4 &=& 2.161918 \cdot 10^{20} & 2.161918 \cdot 10^{20}. \end{array}$$

In diesem Falle gibt es offensichtlich keine Probleme bei der Frage nach dem richtigen Ergebnis. ♡

1.2. Beispiel: Es ist das lineare Gleichungssystem

$$\begin{aligned} x_1 + \tfrac{1}{2}x_2 + \tfrac{1}{3}x_3 + \tfrac{1}{4}x_4 &= 1 \\ \tfrac{1}{2}x_1 + \tfrac{1}{3}x_2 + \tfrac{1}{4}x_3 + \tfrac{1}{5}x_4 &= 1 \\ \tfrac{1}{3}x_1 + \tfrac{1}{4}x_2 + \tfrac{1}{5}x_3 + \tfrac{1}{6}x_4 &= 1 \\ \tfrac{1}{4}x_1 + \tfrac{1}{5}x_2 + \tfrac{1}{6}x_3 + \tfrac{1}{7}x_4 &= 1 \end{aligned}$$

auf einem Rechner zu lösen. Die exakte Lösung lautet

$$x_1 = -4, \quad x_2 = 60, \quad x_3 = -180, \quad x_4 = 140.$$

Gibt man die Koeffizienten $\frac{1}{3}$, $\frac{1}{6}$ und $\frac{1}{7}$ nacheinander mit 4, 5, 6 oder 8 genauen Stellen ein, so wird man folgende Ergebnisse erhalten:

	$4-stellig$	$5-stellig$	$6-stellig$	$8-stellig$
$x_1 =$	-5.8999	-4.1814	-4.0262	-4.0003
$x_2 =$	80.5437	61.9951	60.2963	60.0033
$x_3 =$	-228.5033	-184.7562	-180.7181	-180.0080
$x_4 =$	171.1528	143.0748	140.4694	140.0052

Erkennbar bewirken kleine Änderungen in den Eingabedaten große Änderungen im Ergebnis. Wesentlich ist, dass die Lösung der Aufgabe empfindlich auf Störungen in den Eingabedaten reagiert; dies ist eine vom Algorithmus unabhängige Eigenschaft. ♡

1.3. Beispiel: Es ist das bestimmte Integral

$$I_n = \int_0^1 x^n e^{1-x}\, dx$$

für verschiedene natürliche Zahlen n zu berechnen. Leicht sieht man, dass

$$\begin{aligned} I_0 &= e-1, \\ I_n &= -1 + nI_{n-1}, \quad i = 1,2,\ldots \end{aligned}$$

und

$$0 < I_n = \int_0^1 x^n e^{1-x}\, dx < e\int_0^1 x^n\, dx = \frac{e}{n+1}$$

gilt. Verwendet man nun die obige Rekursionsformel, um I_n für $n = 1,2,\ldots$ zu berechnen, so erhält man folgende Resultate bei 7-stelliger bzw. 16-stelliger

Rechnung:

n	$I_n\,(7-stellig)$	$I_n\,(16-stellig)$
0	1.71828	1.71828
2	0.43656	0.43656
4	0.23877	0.23876
6	0.16304	0.16292
8	0.13024	0.12332
10	0.72160	0.09911
12	82.2512	0.08281
14	14954.72	0.07110
16		0.06506
18		0.90685
20		323.60412
20		149482.10144

Für größere Werte von n sind die Ergebnisse offensichtlich falsch. Beachtet man aber

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

und setzt etwa $I_{10} = 0$, so lassen sich die Werte $I_9, \ldots, I_0$ nach der Formel

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

berechnen; es ergibt sich bei 7-stelliger bzw. 16-stelliger Rechnung:

n	$I_n\,(7-stellig)$	$I_n\,(16-stellig)$
0	1.71828	1.71828
2	0.43656	0.43656
4	0.23876	0.23876
6	0.16290	0.16290
8	0.12222	0.12222
10	0.0	0.0

Obwohl der wahre Wert für I_{10} etwa 0.097 beträgt, der Eingabefehler also relativ groß war, ergeben sich für $I_9, \ldots, I_0$ gute Resultate. ♡

Beim ersten Beispiel sehen wir, wie verschiedene, aber mathematisch äquivalente Algorithmen unterschiedliche Ergebnisse liefern. Das zweite Beispiel zeigt, dass eine kleine Änderung der Eingabedaten eine große Änderung der Ausgabedaten bewirken kann. Einen ähnlichen Effekt bemerken wir beim dritten

Beispiel: Kleine Rundungsfehler, die während der Rechnung auftreten, werden extrem verstärkt und führen zu unsinnigen Resultaten. Ursachen für diese Effekte sollen nun untersucht werden.
Die verschiedenen Fehlerarten begrenzen die Genauigkeit der Lösung einer Aufgabe. Zum Klassifizieren dieser Fehlerarten betrachten wir die allgemeine Vorgehensweise beim Lösen eines Problems. Am Anfang steht die mathematische Modellierung. Das mathematische Modell lässt sich als eine Vektorfunktion φ auffassen, die aus einer Menge $D \subseteq \mathbb{R}^n$ in eine Menge $W \subseteq \mathbb{R}^m$ abbildet. Aus einem Satz von Eingabedaten $x \in D$ ist ein Satz von Ausgabedaten $y \in W$ zu berechnen. Ein numerisches Problem ist nun gerade dadurch gekennzeichnet, dass auf dieser Stufe die Eingabedaten als fehlerbehaftet anzusehen sind. Die Ursachen dieser **Datenfehler** sind unterschiedlich. In den meisten Fällen werden die Eingabedaten aus physikalischen Messungen, statistischen Erhebungen oder ähnlichem gewonnen. Damit unterliegen sie gewissen Unsicherheiten. Es ist auch möglich, dass die Daten des Problems selbst Ergebnis des Lösens eines anderen Problems sind. Auch in diesem Falle müssen wir annehmen, dass sie fehlerbehaftet sind. Wesentliche Eigenschaft dieser Datenfehler ist die, dass sie unabhängig vom Algorithmus auftreten. Folglich muss man anstelle eines Eingabedatensatzes eine Menge von möglichen Eingabedaten

$$X = \{ \bar{x} + \delta x \mid \delta x \in \Delta \}$$

betrachten. Die Menge Δ charakterisiert hier die Datenunsicherheit. Damit gibt es auch eine Menge von möglichen Ergebnissen

$$Y = \{ \varphi(x) \mid x \in X \}.$$

Im nächsten Schritt des Problemlösens ist aus der Menge X der möglichen Eingabedaten ein Datensatz x auszuwählen. Dieser Eingabedatensatz wird sich von den exakten Daten unterscheiden. Damit unterscheidet sich das Ergebnis $\varphi(x)$ auch vom exakten. Dieser Fehler in der Lösung wird nicht durch den Algorithmus beeinflusst; er ist ein **unvermeidbarer Fehler**. Sodann wählen wir für das mathematisches Problem einen numerischen Algorithmus $\bar{\varphi}$ zum Lösen der Aufgabe aus. Wir benötigen eine konkrete Vorschrift zum Berechnen der Ausgabedaten aus den Eingabedaten. Natürlich soll die Aufgabe nach endlicher Zeit gelöst sein. Folglich können nur endlich viele Rechenoperationen ausgeführt werden. Die meisten mathematischen Probleme erfordern aber zum Lösen unendliche Prozesse (Reihenentwicklungen, Grenzübergänge usw.). Diese sind daher durch endliche Prozesse zu ersetzen. Der dabei auftretende Fehler wird als **Verfahrensfehler** bezeichnet. In einem letzten Schritt wird der numerischen Algorithmus $\bar{\varphi}$ auf einem Rechner ablaufen. Wie wir später sehen

werden, arbeitet der Rechner nur mit einer endlichen Menge von Maschinenzahlen. Beim Übergang von den reellen Zahlen x zu Rechnerzahlen $\hat{x}$ und dem effektiven Ausführen eines Algorithmus auf einem Rechner treten **Rundungsfehler** auf. Das berechnete Maschinenergebnis wird sich von dem erwarteten unterscheiden.
Zusammen sind drei Fehlerarten zu konstatieren: Datenfehler, Verfahrensfehler und Rechen- oder Rundungsfehler. Verfahrensfehler untersucht man im Zusammenhang mit dem entsprechenden Algorithmus. Sie hängen eng mit der Konvergenz und der Konvergenzgeschwindigkeit der Verfahren zusammen. Die Auswirkungen von Datenfehlern beim Anwenden numerischer Verfahren sind weitgehend unabhängig vom gewählten Algorithmus; sie sind eine der numerischen Aufgabe innewohnende Eigenschaft: Kleine Änderungen in den Eingabedaten können zu großen Änderungen im Ergebnis führen. Jedoch lassen sich schon jetzt über das Wirken von Daten- und Rundungsfehlern prinzipielle Aussagen machen.

1.2. Maschinenzahlen und Computerarithmetik

Üblicherweise werden REAL-Zahlen auf einem Rechner in der sogenannten Gleitpunktdarstellung

$$x = \pm 0.m_1 m_2 \dots m_t \cdot b^e = \pm 0.m_1 m_2 \dots m_t \; {}_b e$$

abgespeichert. Die zweite Form nennt man auch halblogarithmische Darstellung. Diese werden wir vorwiegend verwenden. Wir bezeichnen die Größe

$$m = 0.m_1 m_2 \dots m_t$$

als Mantisse, b als Basis und e als Exponenten. Die Mantissenziffern $m_1, m_2, \dots, m_t$ nehmen die Werte $0, 1, \dots, b-1$ an. Als Basis werden meist Zweierpotenzen verwendet. Um die arithmetischen Grundoperationen über schnelle Mikroprogramme zu realisieren, wird für jede REAL-Zahl gleich viel Speicherplatz genutzt. Damit sind Mantissenlänge und Größe des Exponenten beschränkt. Ein Maschinenzahlbereich

$$\mathbb{M} = \mathbb{M}(b, t, \underline{e}, \bar{e}).$$

ist durch die Angabe der vier Parameter Basis b, Mantissenlänge t, kleinster Exponent $\underline{e}$ und größter Exponent $\bar{e}$ charakterisiert. In der obigen Form ist die Darstellung einer Maschinenzahl noch nicht eindeutig. So ist zum Beispiel durch

$$0.10000 \cdot 10^1, \quad 0.01000 \cdot 10^2$$

jeweils die gleiche Zahl festgelegt. Fordern wir aber von der Gleitpunktdarstellung einer Zahl x, dass ihre erste Mantissenziffer im Falle $x \neq 0$ ungleich Null sein soll, erhalten wir die **normalisierte Gleitpunktdarstellung**. Für sie gilt

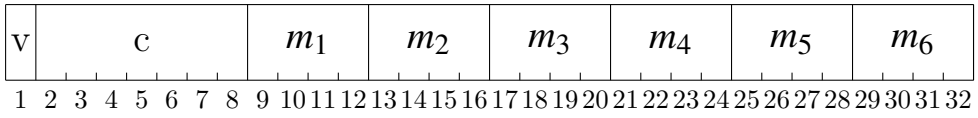
$$\frac{1}{b} \leqq m < 1 \quad \text{für} \quad x \neq 0,$$

$$m = 0 \quad \text{für} \quad x = 0$$

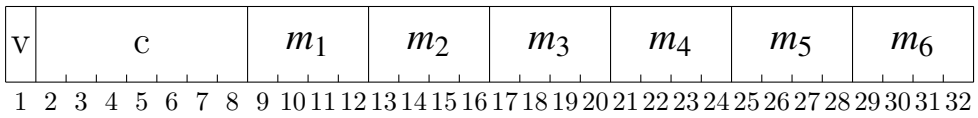
und

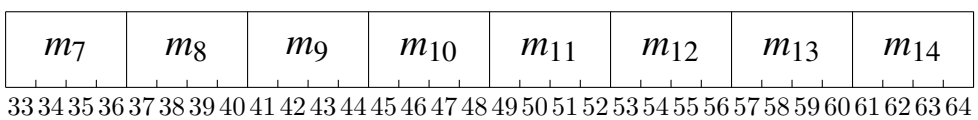
$$\underline{e} \leqq e \leqq \bar{e}.$$

1.4. Beispiel: Bei IBM-Großrechnern wird für eine REAL-Zahl in einfacher Genauigkeit ein Speicherplatz von 4 Bytes bzw. 32 Bits, verwendet. Die Basis der Zahldarstellung ist 16. Die Aufteilung des Speicherplatzes lässt sich an folgendem Schema ablesen.



Das erste Bit enthält das Vorzeichen ($v = 0$ für $x \geqq 0$ und $v = 1$ für $x < 0$). In den folgenden sieben Bits ist der Exponent codiert. Es gilt $c = e + 64$. Wegen $0 \leqq c \leqq 2^7 - 1 = 127$ gilt $-64 \leqq e \leqq 63$, also $\underline{e} = -64$ und $\bar{e} = 63$. Die Bits 9 bis 32 enthalten die sechs Mantissenziffern, wobei für jede Ziffer vier Bits zur Verfügung stehen. Damit darf jede Ziffer Werte zwischen 0 und 15 annehmen. Die Ziffern werden im Hexadezimalsystem üblicherweise mit 0, 1, 2, 3, 4, 5, 6, 7, 8, 9, A, B, C, D, E, F bezeichnet. Es handelt sich hier um den Maschinenzahlbereich $\mathbb{M}(16, 6, -64, 63)$.
Für eine REAL-Zahl doppelter Genauigkeit werden 8 Bytes zur Zahldarstellung genutzt. Die ersten vier Byte sind wie bei einfach-genauen REAL-Zahlen genutzt. Die zusätzlichen vier Bytes nehmen weitere acht Mantissenziffern auf. Es ergibt sich folgende Darstellung:





Wir erhalten den Maschinenzahlbereich $\mathbb{M}(16,14,-64,63)$. ♡

Der auf dem Rechner über Mikroprogramme realisierte Befehlssatz beeinflußt die Zahlendarstellung wesentlich; die arithmetischen Operationen sind aus ökonomischen Gründen über Mikroprogramme realisiert und die Zahldarstellung ist so gewählt, dass die Mikroprogramme möglichst schnell ablaufen.
Offensichtlich ist die Menge der Maschinenzahlen endlich. Die Maschinenzahlen liegen symmetrisch zum Nullpunkt: Mit $x \in \mathbb{M}$ gilt auch $-x \in \mathbb{M}$. Es gibt eine betragsgrößte Maschinenzahl $\max = (1-b^{-t})b^{\bar{e}}$ und außer der Null eine betragskleinste Maschinenzahl $\min = b^{\underline{e}-1}$. Die Anzahl der in $\mathbb{M}(b,t,\underline{e},\bar{e})$ enthaltenen Maschinenzahlen lässt sich berechnen; sie beträgt

$$|\mathbb{M}| = 2(b-1)b^{t-1}(\bar{e}-\underline{e}+1)+1.$$

Für unser erstes Beispiel gilt

$$\begin{aligned} \max &= (1-16^{-6})16^{63} = 16^{63}-16^{57} \approx 7.237\cdot 10^{75}, \\ \min &= 16^{-64-1} = 16^{-65} \approx 5.398\cdot 10^{-79}, \\ |\mathbb{M}| &= 4\,026\,531\,841. \end{aligned}$$

Für das zweite Beispiel erhält man

$$\begin{aligned} \max &= (1-16^{-14})16^{63} = 16^{63}-16^{49} \approx 7.237\cdot 10^{75}, \\ \min &= 16^{-64-1} = 16^{-65} \approx 5.398\cdot 10^{-79}, \\ |\mathbb{M}| &= 17\,293\,822\,566\,102\,704\,641. \end{aligned}$$

Die reellen Zahlen sind in geeigneter Weise auf die Maschinenzahlen abzubilden. Das gilt nicht nur bei der Eingabe, sondern auch während der Rechnung, da das Ergebnis von arithmetischen Operationen mit Maschinenzahlen i. a. keine Maschinenzahl ist. Diese Abbildung werden wir mit rd (Rundung) bezeichnen. Eine sinnvolle Forderung an die Rundung wäre:

$$|x-\mathrm{rd}\,(x)| \leqq |x-y| \quad \forall x \in \mathbb{R} \forall y \in \mathbb{M}.$$

Die reelle Zahl x wird auf jene Maschinenzahl $y = \mathrm{rd}\,(x)$ abgebildet, die ihr am nächsten liegt. (Existieren zwei nächstgelegene Maschinenzahlen, wird z.B. die betragskleinere von ihnen genommen.) Dabei können Probleme auftreten. Es sind drei Fälle zu unterscheiden:

1. $|x| > \max$
   Eine sinnvolle Abbildung von x in $\mathbb{M}$ ist nicht möglich. Diese Situation wird als Exponentenüberlauf (engl. overflow) bezeichnet. Sie führt im allgemeinen zum Programmabbruch.

2. $0 < |x| < \min$
   Es wird $\mathrm{rd}\,(x) = 0$ gesetzt. Die gesamte Information, die in x steckt, geht verloren. Diese Situation wird als Exponentenunterlauf (engl. underflow) bezeichnet und wird i. a. nicht angezeigt.

3. $\min \leqq |x| \leqq \max$
   Nur in diesem Falle ist eine sinnvolle Rundung möglich. Es sei

   $$y = \mathrm{rd}\,(x) = m \cdot b^e.$$

   Dann liegt die Zahl x mit Sicherheit in einem der Intervalle

   $$[y, y + \frac{1}{2} b^{e-t}] \text{ oder } [y - \frac{1}{2} b^{e-t}, y].$$

   Damit gilt

   $$|x - \mathrm{rd}\,(x)| \leqq \frac{1}{2} b^{e-t}.$$

   Beachtet man noch, dass $|x| \geqq b^{e-1}$ gilt, so folgt

   $$\left| \frac{x - \mathrm{rd}\,(x)}{x} \right| \leqq \frac{1}{2} b^{1-t} = \mathrm{eps}.$$

   Durch die Zahl eps ist das relative Rundungsfehlerniveau des Maschinenzahlbereichs $\mathbb{M}$ festgelegt; sie wird als **relative Maschinengenauigkeit** bezeichnet. Für die Rundung gilt somit in diesem Falle

   $$\mathrm{rd}\,(x) = x(1 + \varepsilon_x), \quad |\varepsilon_x| \leqq \mathrm{eps}.$$

Bemerkung: Für einen konkreten Rechner lässt sich die relative Maschinengenauigkeit folgendermaßen ermitteln:

$$\mathrm{eps} = \min \left\{ x \in \mathbb{M} \mid x \geqq 0,\ \mathrm{gl}\,(1 + x) > 1 \right\}.$$

Dabei bezeichnet $\mathrm{gl}\,(\circ)$ das **Gleitpunktergebnis** einer Operation, also das Ergebnis, das ein Rechner nach eventueller Rundung liefert. Die relative Maschinengenauigkeit eps ist die kleinste positive Maschinenzahl x, für die sich das Ergebnis der Operation $1 + x$ in diesem Maschinenzahlbereich von 1 unterscheidet.

1.5. Beispiel: Im Maschinenzahlbereich $\mathbb{M}(16,6,-64,63)$ beträgt die relative Maschinengenauigkeit

$$\text{eps} = \frac{1}{2}16^{1-6} = \frac{1}{2}16^{-5} \approx 0.476837 \cdot 10^{-6}.$$

Im Maschinenzahlbereich $\mathbb{M}(16,14,-64,63)$ erhält man

$$\text{eps} = \frac{1}{2}16^{1-14} = \frac{1}{2}16^{-13} \approx 0.111022 \cdot 10^{-15}.$$

♡

Wir stellen kurz dar, wie die Addition von zwei Zahlen auf einem Rechner abläuft. Dazu betrachten wir den Maschinenzahlbereich $\mathbb{M}(16,6,-64,63)$. Es seien die zwei Zahlen

$$x = 2912.57, \quad y = 27.0165$$

zu addieren. Bei der Eingabe werden diese Zahlen auf entsprechende Maschinenzahlen abgebildet(konvertiert). Wir erhalten

$$\hat{x} = \texttt{0.B6091F}_{16}\texttt{3}, \quad \hat{y} = \texttt{0.1B0439}_{16}\texttt{2}.$$

Diese Werte unterscheiden sich von den ursprünglichen Daten. Es gilt

$$\begin{aligned} \hat{x} &= x + \delta x, \quad \delta x \approx 0.68350 \cdot 10^{-4}, \\ \hat{y} &= y + \delta y, \quad \delta y \approx -0.52492 \cdot 10^{-5}. \end{aligned}$$

Der relative Konvertierungsfehler beträgt

$$\frac{\delta x}{x} \approx 0.23467 \cdot 10^{-7} \quad \text{bzw.} \quad \frac{\delta y}{y} \approx -0.19430 \cdot 10^{-6},$$

liegt also jeweils unterhalb der relativen Maschinengenauigkeit. Vor der Addition der beiden Zahlen findet ein Exponentenangleich statt. Dabei wird die Mantisse der betragskleineren Zahl soweit nach rechts verschoben, bis die Exponenten beider Zahlen übereinstimmen. Erst dann sind die Zahlen addierbar. Man erhält:

$$\texttt{0.B6091F}_{0}{}_{16}\texttt{3} + \texttt{0.01B043}_{9}{}_{16}\texttt{3} = \texttt{0.B7B962}_{9}{}_{16}\texttt{3}.$$

Das Zwischenergebnis wird noch auf eine Maschinenzahl abgebildet und eventuell normalisiert. Insgesamt gilt damit

$$\text{gl}\,(x+y) = \text{rd}\,(\hat{x}+\hat{y}) = \text{rd}\,(\texttt{0.B7B962}_{9}{}_{16}\texttt{3}) = \texttt{0.B7B963}_{16}\texttt{3}.$$

Der absolute Gesamtfehler beträgt

$$\mathrm{gl}\,(x+y)-(x+y)\approx 0.16988\cdot 10^{-3},$$

der eigentliche Rundungsfehler beträgt dagegen

$$\mathrm{rd}\,(\hat{x}+\hat{y})-(\hat{x}+\hat{y})\approx 0.10681\cdot 10^{-3}.$$

Der Unterschied zwischen diesen beiden Fehlern ist der unvermeidbare Fehler; hier also die Auswirkung des Konvertierungsfehlers:

$$(\hat{x}+\hat{y})-(x+y)\approx 0.6307\cdot 10^{-4}.$$

In diesem Falle sind Rundungsfehler und unvermeidbarer Fehler von derselben Größenordnung. Das gilt aber nicht immer. Von einem guten Algorithmus sollte man erwarten, dass der erzeugte Rundungsfehler nicht wesentlich größer ist als der unvermeidbaren Fehler. Ein weiterer, wesentlicher Unterschied beim Rechnen mit Maschinenzahlen gegenüber dem Rechnen mit reellen Zahlen ist der, dass Assoziativ- und Distributivgesetze nicht mehr gelten.

1.6. Beispiel: Es sei die Summe s der drei Maschinenzahlen

$$\hat{x}=\texttt{0.B6091F}_{16}\texttt{3},\quad \hat{y}=\texttt{-0.B6091D}_{16}\texttt{3},\quad \hat{z}=\texttt{0.1B0439}_{16}\texttt{-2}$$

aus dem Maschinenzahlbereich $\mathbb{M}(16,6,-64,63)$ zu berechnen. Dazu addiert man die Zahlen gemäß $s=(\hat{x}+\hat{y})+\hat{z}$ oder $s=\hat{x}+(\hat{y}+\hat{z})$. Man erhält

$$(x+y)+z=\texttt{0.200000}_{16}\texttt{-2}+\texttt{0.1B0439}_{16}\texttt{-2}=\texttt{0.3B0439}_{16}\texttt{-2}$$

beziehungsweise

$$x+(y+z)=\texttt{0.B6091F}_{16}\texttt{3}-\texttt{0.B6091B}_{16}\texttt{3}=\texttt{0.400000}_{16}\texttt{-2}.$$

♡

Diese kleinen Beispiele belegen, dass sich die arithmetischen Grundoperationen auf einem Rechner wesentlich von den entsprechenden Operationen im Bereich der reellen Zahlen unterscheiden können. Um den Unterschied zwischen den Gleitpunktoperationen und den üblichen arithmetischen Operationen deutlich zu machen, verwenden wir die Symbole "‘$\oplus$"’ , "‘$\ominus$"’, "‘$\odot$"’ und "‘$\oslash$"’ statt "‘$+$"’, "‘$-$"’ , "‘$\cdot$"’ und "‘$/$"’. Von diesen "‘Ersatzoperationen"’ erwarten wir im besten Falle, dass das Ergebnis $\hat{x}\circ\hat{y}$ für zwei Maschinenzahlen $\hat{x}$ und $\hat{y}$ im Rahmen der Maschinengenauigkeit mit dem exakten Ergebnis übereinstimmt. Dabei steht "‘$\circ$"’ für eine der vier Operationen "‘$\oplus$"’ , "‘$\ominus$"’, "‘$\odot$"’ und "‘$\oslash$"’. Sind x und

y zwei reelle Zahlen, sowie $\hat{x} = \mathrm{rd}\,(x)$ und $\hat{y} = \mathrm{rd}\,(y)$ die ihnen zugeordneten Maschinenzahlen, so gilt bei den heute üblichen Rechnern:

$$\begin{aligned}
\mathrm{gl}\,(x+y) &= \hat{x} \oplus \hat{y} = \mathrm{rd}\,(\hat{x}+\hat{y}) = (\hat{x}+\hat{y})(1+\varepsilon_1),\\
\mathrm{gl}\,(x-y) &= \hat{x} \ominus \hat{y} = \mathrm{rd}\,(\hat{x}-\hat{y}) = (\hat{x}-\hat{y})(1+\varepsilon_2),\\
\mathrm{gl}\,(x\cdot y) &= \hat{x} \odot \hat{y} = \mathrm{rd}\,(\hat{x}\cdot\hat{y}) = (\hat{x}\cdot\hat{y})(1+\varepsilon_3),\\
\mathrm{gl}\,(x/y) &= \hat{x} \oslash \hat{y} = \mathrm{rd}\,(\hat{x}/\hat{y}) = (\hat{x}/\hat{y})(1+\varepsilon_4) \quad \text{für} \quad \hat{y} \neq 0,
\end{aligned}$$

wobei für die erzeugten relativen Rundungsfehler

$$|\varepsilon_i| \leqq \mathrm{eps}, \quad i = 1,2,3,4$$

gilt.
Diese Rundungsfehler sind typisch fuer arithmetische Operationen. Es ist nun wichtig, zu verfolgen, welchen Einfluss die einzelnen Rundungsfehler auf den Gesamtfehler im Ergebnis haben.

1.3. Fehlerfortpflanzung

Wir betrachten einen Algorithmus als Vektorfunktion

$$\varphi: \quad D \subseteqq \mathbb{R}^n \longrightarrow \mathbb{R}^m.$$

Zunächst wollen wir uns für eine Abschätzung des unvermeidbaren Fehlers interessieren. Es seien dazu durch $x \in \mathbb{R}^n$ die exakten Eingabedaten gegeben. Wir verfügen aber nur über gestörte Daten $x + \delta x$. Der unvermeidbare Fehler, als Auswirkung des Datenfehlers δx, ist durch

$$\delta y = \varphi(x+\delta x) - \varphi(x)$$

gegeben. Ist die Vektorfunktion φ zweimal stetig differenzierbar, so lässt sich der unvermeidbare Fehler abschätzen. Durch Taylor-Entwicklung erhält man

$$\varphi(x+\delta x) \;=\; \varphi(x) + \varphi'(x)\delta x + O(\|\delta x\|^2)$$

Für

$$\varphi(x) = \begin{pmatrix} \varphi_1(x_1, x_2, \ldots, x_n) \\ \varphi_2(x_1, x_2, \ldots, x_n) \\ \vdots \\ \varphi_m(x_1, x_2, \ldots, x_n) \end{pmatrix}$$

gilt

$$\varphi'(x) = \begin{pmatrix} \frac{\partial \varphi_1(x)}{\partial x_1} & \frac{\partial \varphi_1(x)}{\partial x_2} & \dots & \frac{\partial \varphi_1(x)}{\partial x_n} \\ \frac{\partial \varphi_2(x)}{\partial x_1} & \frac{\partial \varphi_2(x)}{\partial x_2} & \dots & \frac{\partial \varphi_2(x)}{\partial x_n} \\ \vdots & \vdots & \ddots & \vdots \\ \frac{\partial \varphi_m(x)}{\partial x_1} & \frac{\partial \varphi_m(x)}{\partial x_2} & \dots & \frac{\partial \varphi_m(x)}{\partial x_n} \end{pmatrix}.$$

Wir werden die Betrachtungen in der sogenannten ersten Näherung fortsetzen. Wir vernachlässigen alle Terme höherer Ordnung, in diesem Falle $O(\|\delta x\|^2)$. Um die erste Näherung zu kennzeichnen, verwenden wir statt des Gleichheitszeichens das Symbol "$\doteq$". Wir erhalten für die einzelnen Komponenten von φ

$$\varphi_i(x+\delta x) \doteq \varphi_i(x) + \sum_{j=1}^{n} \frac{\partial \varphi_i(x)}{\partial x_j} \delta x_j, \quad i = 1, \dots, m,$$

beziehungsweise

$$\delta y_i = \varphi_i(x+\delta x) - \varphi_i(x) \doteq \sum_{j=1}^{n} \frac{\partial \varphi_i(x)}{\partial x_j} \delta x_j, \quad i = 1, \dots, m.$$

Die Größen

$$C_{ij}(\varphi) = \frac{\partial \varphi_i(x)}{\partial x_j}$$

geben an, wie sich der absolute Fehler δx_j der j-ten Komponente von x im absoluten Fehler δy_i der i-ten Komponente von y auswirkt. Sie heißen **absolute partielle Konditionszahlen**. Gelten für alle Eingabefehler δx_j, $j = 1, \dots, n$, Ungleichungen der Form

$$|\delta x_j| \leqq K \text{eps} |x_j|, \quad j = 1, \dots, n,$$

so ergibt sich die folgende Abschätzung für den absoluten unvermeidbaren Fehler der Komponente y_i:

$$\begin{aligned} |\delta y_i| &\doteq \left| \sum_{j=1}^{n} \frac{\partial \varphi_i(x)}{\partial x_j} \delta x_j \right| \\ &\dot{\leq} \sum_{j=1}^{n} \left| \frac{\partial \varphi_j(x)}{\partial x_j} \right| |\delta x_j| \\ &\dot{\leq} K\text{eps} \sum_{j=1}^{n} \left| \frac{\partial \varphi_i(x)}{\partial x_j} \right| |x_j|, \quad i = 1, \ldots, m. \end{aligned}$$

Das Zeichen "$\dot{\leq}$" bedeutet dabei "'kleiner oder gleich in erster Näherung"'. Gilt $x_j \neq 0$ für $j = 1, \ldots, n$ und $y_i \neq 0$ für $i = 1, \ldots, m$, so lassen sich auch die relativen Fehler abschätzen. Man erhält

$$\begin{aligned} \frac{\delta y_i}{y_i} &\doteq \sum_{j=1}^{n} \frac{1}{\varphi_i(x)} \frac{\partial \varphi_i(x)}{\partial x_j} \delta x_j \\ &\doteq \sum_{j=1}^{n} \left(\frac{x_j}{\varphi_i(x)} \frac{\partial \varphi_i(x)}{\partial x_j} \right) \left(\frac{\delta x_j}{x_j} \right). \end{aligned}$$

Durch die Größen

$$c_{ij}(\varphi) = \frac{x_j}{\varphi_i(x)} \frac{\partial \varphi_j(x)}{\partial x_j}$$

sind Faktoren gegeben, mit denen sich relative Eingabefehler $\delta x_j / x_j$ in den Komponenten von x im relativen Fehler $\delta y_i / y_i$ der Ergebniskomponenten verstärken. Sie heißen **relative partielle Konditionszahlen**. Nimmt man nun wieder an, dass die Eingabefehler Ungleichungen der Form

$$|\varepsilon_j| = \left| \frac{\delta x_j}{x_j} \right| \leq K\text{eps}, \quad j = 1, \ldots, n$$

erfüllen, so erhält man die folgenden Abschätzungen für die relativen unvermeidbaren Fehler der Lösungskomponenten δy_i:

$$|\mu_i| = \left| \frac{\delta y_i}{y_i} \right| \dot{\leq} K\text{eps} \sum_{j=1}^{n} \left| \frac{x_j}{\varphi_i(x)} \frac{\partial \varphi_i(x)}{\partial x_j} \frac{\delta x_j}{x_j} \right|, \quad i = 1, \ldots, m.$$

Wie wir an den Abschätzungen erkennen, hängt der unvermeidbare Fehler nur von den Eingabefehlern und vom durch φ repräsentierten mathematischen

Problem ab. Der Algorithmus, den man zum Lösen dieses Problems anwendet, spielt an dieser Stelle keine Rolle.
Beim Abarbeiten eines konkreten Algorithmus auf einem Rechner treten Rundungsfehler auf. Die Auswirkung dieser Rundungsfehler auf das Endresultat studiert man prinzipiell auf die gleiche Weise wie den Eingabe-Fehlereinfluss. Dazu wird die Funktion φ in der Form

$$\varphi = \varphi^{(r)} \circ \varphi^{(r-1)} \circ \ldots \circ \varphi^{(1)}$$

dargestellt. Die Funktionen $\varphi^{(l)}$, $l = 1, \ldots, r$ stellt man sich im einfachsten Falle als arithmetische Verknüpfungen von Zwischenergebnissen vor. Damit gilt

$$\begin{aligned} y^{(1)} &= \varphi^{(1)}(x), \\ y^{(2)} &= \varphi^{(2)}(y^{(1)}), \\ &\vdots \\ y^{(r-1)} &= \varphi^{(r-1)}(y^{(r-2)}), \\ y &= \varphi^{(r)}(y^{(r-1)}). \end{aligned}$$

Auf jeder Stufe l entsteht ein zusätzlicher Rundungsfehler $\varepsilon^{(l)}$. Die Auswirkung dieses Fehlers auf das Endergebnis y wird durch Untersuchung der Restabbildung

$$\psi^{(l)} = \varphi^{(r)} \circ \varphi^{(r-1)} \circ \ldots \circ \varphi^{(l+1)}$$

abgeschätzt. Diese Vorgehensweise bezeichnet man als differentielle Fehleranalyse. Sie ist nur für einfache Algorithmen durchführbar. Man erhält schnell unübersichtliche Resultate. Wir nutzen die differentielle Fehleranalyse zur Untersuchung der Fehlerfortpflanzung bei elementaren Funktionen. Absolute und relative Konditionszahlen gebräuchlicher Funktionen sind in der folgenden Tabelle zusammengestellt.
Wir wollen nun die Fehlerfortpflanzung der elementaren arithmetischen Operationen untersuchen. Dabei werden wir etwas einfacher vorgehen als bei der differentiellen Fehleranalyse. Wir nutzen die Erkenntnisse, die wir am Ende von Abschnitt 1.2. gewonnen haben. Mit x und y bezeichnen wir wie üblich die exakten Daten, und mit $\hat{x} = x(1+\varepsilon_x)$ und $\hat{y} = y(1+\varepsilon_y)$ die gestörten Daten. Die Fehlerterme ε_x und ε_y seien dabei von der Größenordnung der relativen Maschinengenauigkeit eps.

$f(x)$	$\lvert C(f)\rvert = \lvert f'(x)\rvert$	$\lvert c(f)\rvert = \left\lvert \frac{x\cdot f'(x)}{f(x)} \right\rvert$
x^α $(\alpha \in \mathbb{R}_+)$	$\alpha \lvert x\rvert^{\alpha-1}$	α
$\sqrt{x}$	$\frac{1}{2\sqrt{x}}$	$\frac{1}{2}$
x^{-1}	$\frac{1}{x^2}$	1
$\ln x$	$\frac{1}{\lvert x\rvert}$	$\frac{1}{\lvert \ln x\rvert}$
e^x	e^x	$\lvert x\rvert$
$\sin x$	$\lvert \cos x\rvert$	$\lvert x\cot x\rvert$
$\cos x$	$\lvert \sin x\rvert$	$\lvert x\tan x\rvert$
$\tan x$	$\frac{1}{\cos^2 x}$	$\left\lvert \frac{2x}{\sin 2x} \right\rvert$

Table 1.1: Absolute und relative Konditionszahlen elementarer Funktionen

Addition und Subtraktion: Es gilt

$$\begin{aligned}
\mathrm{gl}\,(x+y) &= \hat{x}\oplus\hat{y} = \mathrm{rd}\,(\hat{x}+\hat{y}) = (\hat{x}+\hat{y})(1+\varepsilon), \quad (|\varepsilon| \leqq \mathrm{eps}) \\
&= (x(1+\varepsilon_x)+y(1+\varepsilon_y))(1+\varepsilon) \\
&= x(1+\varepsilon_x)(1+\varepsilon)+y(1+\varepsilon_y)(1+\varepsilon) \\
&= x(1+\varepsilon_x+\varepsilon+\varepsilon_x\varepsilon)+y(1+\varepsilon_y+\varepsilon+\varepsilon_y\varepsilon) \\
&= (x+y)\left(1+\frac{x}{x+y}(\varepsilon_x+\varepsilon+\varepsilon_x\varepsilon)+\frac{y}{x+y}(\varepsilon_y+\varepsilon+\varepsilon_y\varepsilon)\right).
\end{aligned}$$

Arbeiten wir nun in erster Näherung, so vernachlässigen wir alle höheren Potenzen und Produkte von Fehlertermen der Größenordnung eps. Damit erhält man

$$\begin{aligned}
\mathrm{gl}\,(x+y) &\doteq (x+y)\left(1+\frac{x}{x+y}(\varepsilon_x+\varepsilon)+\frac{y}{x+y}(\varepsilon_y+\varepsilon)\right) \\
&\doteq (x+y)\left(1+\frac{x}{x+y}\varepsilon_x+\frac{y}{x+y}\varepsilon_y+\varepsilon\right).
\end{aligned}$$

Für die Subtraktion ergibt sich analog

$$\begin{aligned}
\mathrm{gl}\,(x-y) &\doteq (x-y)\left(1+\frac{x}{x-y}(\varepsilon_x+\varepsilon)-\frac{y}{x-y}(\varepsilon_y+\varepsilon)\right) \\
&\doteq (x-y)\left(1+\frac{x}{x-y}\varepsilon_x-\frac{y}{x-y}\varepsilon_y+\varepsilon\right).
\end{aligned}$$

Relative Fehler in den Eingabedaten x und y werden hier daher mit den Faktoren

$$x/(x\pm y) \text{ bzw. } \pm y/(x\pm y)$$

verstärkt. Diese Faktoren können beliebig groß werden. Das ist bei der Subtraktion insbesondere dann der Fall, wenn $x \approx y$ gilt, falls also zwei Zahlen voneinander subtrahiert werden, die ungefähr gleich groß sind. Fehler in x und y werden dann extrem verstärkt, wohingegen bei der Operation selbst oft kein zusätzlicher Rundungsfehler erzeugt wird. Diese Erscheinung wird **Auslöschung** genannt.

1.7. Beispiel: Wir addieren zwei Zahlen im Zahlbereich $\mathbb{M}(16,6,-64,63)$.

$$\begin{aligned} \hat{x} &= \texttt{0.AB3}\underline{\texttt{32F}}_{16}\texttt{2E} \\ \hat{y} &= \texttt{-0.AB1}\underline{\texttt{DCB}}_{16}\texttt{2E} \\ \hat{x}\oplus\hat{y} &= \texttt{0.001}\underline{\texttt{554}}_{16}\texttt{2E} \\ &= \texttt{0.1}\underline{\texttt{55400}}_{16}\texttt{30} \end{aligned}$$

Die fehlerbehafteten Stellen sind jeweils unterstrichen. Die Addition wird exakt ausgeführt. Aber während in den Eingabedaten noch drei sichere Stellen vorhanden sind, ist es im Ergebnis nur noch eine. Der Verstärkungsfaktor für die Eingabefehler beträgt also etwa $16^2 = 256$. ♡

Addition und Subtraktion sind gefährliche Operationen. Beim Programmieren sollte man darauf achten, dass Auslöschung vermieden wird.
Multiplikation: Es gilt

$$\begin{aligned} \mathrm{gl}\,(x\cdot y) &= \hat{x}\odot\hat{y} = \mathrm{rd}\,(\hat{x}\cdot\hat{y}) = (\hat{x}\cdot\hat{y})(1+\varepsilon), \quad (|\varepsilon| \leqq \mathrm{eps}) \\ &= (x(1+\varepsilon_x)\cdot y(1+\varepsilon_y))(1+\varepsilon) \\ &= (x\cdot y)(1+\varepsilon_x+\varepsilon_y+\varepsilon+\varepsilon_x\varepsilon_y+\varepsilon_x\varepsilon+\varepsilon_y\varepsilon+\varepsilon_x\varepsilon_y\varepsilon) \\ &\doteq (x\cdot y)(1+\varepsilon_x+\varepsilon_y+\varepsilon). \end{aligned}$$

Die relativen partiellen Konditionszahlen sind beide gleich 1. Es findet keine Fehlerverstärkung statt. Die Multiplikation ist eine "'gutartige"' Operation.
Division: Es gilt

$$\begin{aligned} \mathrm{gl}\,(x/y) &= \hat{x}\oslash\hat{y} = \mathrm{rd}\,(\hat{x}/\hat{y}) \\ &= (\hat{x}/\hat{y})(1+\varepsilon), \quad |\varepsilon| \leqq \mathrm{eps}) \\ &= \frac{x(1+\varepsilon_x)}{y(1+\varepsilon_y)}(1+\varepsilon) = \frac{x}{y}\frac{(1+\varepsilon_x)(1+\varepsilon)}{1+\varepsilon_y} \\ &= \frac{x}{y}\frac{(1+\varepsilon_x)(1+\varepsilon)(1-\varepsilon_y)}{1-\varepsilon_y^2} \doteq \frac{x}{y}(1+\varepsilon_x-\varepsilon_y+\varepsilon). \end{aligned}$$

Die Verstärkunsfaktoren der relativen Eingabefehler ε_x und ε_y sind betragsmäßig gleich 1. Die Division ist ebenfalls eine "'gutartige"' Operation.
Nun sind wir in der Lage, kleinere Algorithmen zu analysieren.

1.8. Beispiel: Es ist $z = x^2 - y^2$ zu berechnen. Der unvermeidbaren Fehler ist mittels differentieller Fehleranalyse angebbar. Durch

$$\hat{x} = x + \delta x = x(1+\varepsilon_x), \quad \hat{y} = y + \delta y = y(1+\varepsilon_y)$$

seien die entsprechenden Rechnerdaten gegeben. Dann gilt für den unvermeidbaren Fehler

$$\delta z \doteq 2x\delta x - 2y\delta y$$

beziehungsweise

$$\mu = \frac{\delta z}{z} \doteq 2\frac{x^2}{x^2-y^2}\varepsilon_x - \frac{y^2}{x^2-y^2}\varepsilon_y.$$

Gilt $|\varepsilon_x|, |\varepsilon_y| \leqq K\text{eps}$, so folgt

$$|\mu| \stackrel{\cdot}{\leqq} 2K\text{eps}\frac{x^2+y^2}{|x^2-y^2|}.$$

Wir betrachten zwei Algorithmen zum Berechnen von z.
Algorithmus 1:

$$z_1 = x^2, \quad z_2 = y^2, \quad z = z_1 - z_2.$$

Auf einem Rechner wird aber der folgende Algorithmus realisiert:
Algorithmus 1':

$$\hat{z}_1 = \hat{x} \odot \hat{x}, \quad \hat{z}_2 = \hat{y} \odot \hat{y}, \quad \hat{z} = \hat{z}_1 \ominus \hat{z}_2,$$

Dann gilt:

$$\begin{aligned}
\hat{z}_1 &= (\hat{x}\cdot\hat{x})(1+\varepsilon_1), \\
\hat{z}_2 &= (\hat{y}\cdot\hat{y})(1+\varepsilon_2), \\
\hat{z} &= (\hat{z}_1 - \hat{z}_2)(1+\varepsilon_3) \\
&= [(\hat{x}\cdot\hat{x})(1+\varepsilon_1) - (\hat{y}\cdot\hat{y})(1+\varepsilon_2)](1+\varepsilon_3) \\
&\doteq (\hat{x}^2 - \hat{y}^2)\left(1 + \frac{\hat{x}^2}{\hat{x}^2-\hat{y}^2}\varepsilon_1 - \frac{\hat{y}^2}{\hat{x}^2-\hat{y}^2}\varepsilon_2 + \varepsilon_3\right).
\end{aligned}$$

Der relative Gesamtrundungsfehler dieses Algorithmus ist in erster Näherung durch

$$\mu_1 \doteq \frac{\hat{x}^2}{\hat{x}^2-\hat{y}^2}\varepsilon_1 - \frac{\hat{y}^2}{\hat{x}^2-\hat{y}^2}\varepsilon_2 + \varepsilon_3 \doteq \frac{x^2}{x^2-y^2}\varepsilon_1 - \frac{y^2}{x^2-y^2}\varepsilon_2 + \varepsilon_3$$

gegeben. Er wird gemäß

$$|\mu_1| \stackrel{\cdot}{\leqq} \left(\frac{\hat{x}^2+\hat{y}^2}{|\hat{x}^2-\hat{y}^2|}+1\right)\text{eps} \doteq \left(\frac{x^2+y^2}{|x^2-y^2|}+1\right)\text{eps}$$

abgeschätzt, falls wieder $|\varepsilon_i| \leqq \text{eps}$ für $i = 1,2,3$ gilt.
Algorithmus 2:

$$z_1 = x+y, \quad z_2 = x-y, \quad z = z_1 \cdot z_2.$$

Auf einem Rechner ist dann folgender Algorithmus zu betrachten:
Algorithmus 2':

$$\hat{z}_1 = \hat{x} \oplus \hat{y}, \quad \hat{z}_2 = \hat{x} \ominus \hat{y}, \quad \hat{z} = \hat{z}_1 \odot \hat{z}_2.$$

Es gilt:

$$\begin{aligned}
\hat{z}_1 &= (\hat{x}+\hat{y})(1+\varepsilon_1), \\
\hat{z}_2 &= (\hat{x}-\hat{y})(1+\varepsilon_2), \\
\hat{z} &= (\hat{z}_1 \cdot \hat{z}_2)(1+\varepsilon_3) \\
&= (\hat{x}+\hat{y})(1+\varepsilon_1)\cdot(\hat{x}-\hat{y})(1+\varepsilon_2)\cdot(1+\varepsilon_3) \\
&\doteq (\hat{x}^2-\hat{y}^2)(1+\varepsilon_1+\varepsilon_2+\varepsilon_3) \doteq (x^2-y^2)(1+\varepsilon_1+\varepsilon_2+\varepsilon_3).
\end{aligned}$$

Der relative Gesamtrundungsfehler dieses Algorithmus ist in erster Näherung durch

$$\mu_2 \doteq \varepsilon_1+\varepsilon_2+\varepsilon_3$$

gegeben und wird durch

$$|\mu_2| \stackrel{\cdot}{\leqq} 3\text{eps}$$

abgeschätzt.
Der erste Algorithmus ist daher vorzuziehen, falls $|\mu_1| \leqq |\mu_2|$ gilt; der zweite Algorithmus sollte angewendet werden, falls $|\mu_2| \leqq |\mu_1|$ gilt. Da

$$3 \leqq \frac{x^2+y^2}{|x^2-y^2|}+1$$

genau dann gilt, wenn

$$\frac{1}{3} \leqq \frac{x^2}{y^2} \leqq 3,$$

sollte man den zweiten Algorithmus - also im Falle $1/\sqrt{3} \leqq |x/y| \leqq \sqrt{3}$ - anwenden, andernfalls den ersten. ♡

Bei diesem Beispiel bestimmt die Größe der Eingabedaten den anzuwendenden Algorithmus. Wie das folgende Beispiel zeigt, gibt es auch Fälle, in denen der eine Algorithmus stets besser ist als der andere.

1.9. Beispiel: Es ist die betragskleinere Lösung der quadratischen Gleichung

$$x^2 - 2px + q = 0, \quad p \geqq 0, \quad q \geqq 0, \quad p^2 - q \geqq 0$$

zu berechnen. Die Eingabedaten sind in diesem Falle p und q, und die Lösung lautet

$$x = p - \sqrt{p^2 - q} = \varphi(p,q).$$

Der erste Algorithmus ergibt sich aus der naiven Auswertung dieser Lösungsformel.
Algorithmus 1:

$$x_1 = p^2, \quad x_2 = x_1 - q, \quad x_3 = \sqrt{x_2}, \quad x = p - x_3.$$

Auf einem Rechner ist der folgende Algorithmus realisiert:
Algorithmus 1':

$$\hat{x}_1 = \mathrm{gl}\,(p^2), \quad \hat{x}_2 = \mathrm{gl}\,(\hat{x}_1 - q), \quad \hat{x}_3 = \mathrm{gl}\,(\sqrt{\hat{x}_2}), \quad \hat{x} = \mathrm{gl}\,(p - \hat{x}_3).$$

Es gilt:

$$\begin{aligned}
\hat{x}_1 &= [p(1+\varepsilon_p)]^2(1+\varepsilon_1) \\
&\doteq p^2(1+2\varepsilon_p+\varepsilon_1) = x_1(1+2\varepsilon_p+\varepsilon_1),
\end{aligned}$$

$$\begin{aligned}
\hat{x}_2 &= [\hat{x}_1 - q(1+\varepsilon_q)](1+\varepsilon_2) \\
&\doteq [p^2(1+2\varepsilon_p+\varepsilon_1) - q(1+\varepsilon_q)](1+\varepsilon_2) \\
&\doteq (p^2-q)\left(1 + \frac{2p^2}{p^2-q}\varepsilon_p - \frac{q}{p^2-q}\varepsilon_q + \frac{p^2}{p^2-q}\varepsilon_1 + \varepsilon_2\right) \\
&= x_2\left(1 + \frac{2p^2}{p^2-q}\varepsilon_p - \frac{q}{p^2-q}\varepsilon_q + \frac{p^2}{p^2-q}\varepsilon_1 + \varepsilon_2\right),
\end{aligned}$$

$$\begin{aligned}
\hat{x}_3 &= \sqrt{\hat{x}_2}(1+\varepsilon_3) \\
&\doteq \sqrt{x_2\left(1+\frac{2p^2}{p^2-q}\varepsilon_p-\frac{q}{p^2-q}\varepsilon_q+\frac{p^2}{p^2-q}\varepsilon_1+\varepsilon_2\right)}(1+\varepsilon_3) \\
&\doteq x_3\left(1+\frac{p^2}{p^2-q}\varepsilon_p-\frac{q}{2(p^2-q)}\varepsilon_q+\frac{p^2}{2(p^2-q)}\varepsilon_1+\frac{1}{2}\varepsilon_2+\varepsilon_3\right),
\end{aligned}$$

$$\begin{aligned}
\hat{x} &= [p(1+\varepsilon_p)-\hat{x}_3](1+\varepsilon_4) \\
&\doteq \left[p(1+\varepsilon_p)\right. \\
&\qquad \left.-x_3\left(1+\frac{p^2}{p^2-q}\varepsilon_p-\frac{q}{2(p^2-q)}\varepsilon_q+\frac{p^2}{2(p^2-q)}\varepsilon_1+\frac{1}{2}\varepsilon_2+\varepsilon_3\right)\right](1+\varepsilon_4) \\
&\doteq (p-x_3)\left[1+\frac{p}{p-x_3}\varepsilon_p-\frac{p^2}{(p-x_3)\sqrt{p^2-q}}\varepsilon_p+\frac{q}{2(p-x_3)\sqrt{p^2-q}}\varepsilon_q\right. \\
&\qquad \left.-\frac{p^2}{2(p-x_3)\sqrt{p^2-q}}\varepsilon_1-\frac{\sqrt{p^2-q}}{2(p-x_3)}\varepsilon_2-\frac{\sqrt{p^2-q}}{(p-x_3)}\varepsilon_3+\varepsilon_4\right].
\end{aligned}$$

Mit

$$p-x_3 = p-\sqrt{p^2-q} = \frac{q}{p+\sqrt{p^2-q}},$$

also

$$\frac{1}{p-x_3} = \frac{p+\sqrt{p^2-q}}{q}$$

erhält man daraus

$$\begin{aligned}
\hat{x} &\doteq x\left[1+\frac{p\sqrt{p^2-q}-p^2}{(p-\sqrt{p^2-q})\sqrt{p^2-q}}\varepsilon_p+\frac{p+\sqrt{p^2-q}}{2\sqrt{p^2-q}}\varepsilon_q-\frac{p^2(p+\sqrt{p^2-q})}{2q\sqrt{p^2-q}}\varepsilon_1\right. \\
&\qquad \left.-\frac{\sqrt{p^2-q}(p+\sqrt{p^2-q})}{2q}\varepsilon_2-\frac{\sqrt{p^2-q}(p+\sqrt{p^2-q})}{q}\varepsilon_3+\varepsilon_4\right] \\
&\doteq x\left[1-\frac{p}{\sqrt{p^2-q}}\varepsilon_p+\frac{p+\sqrt{p^2-q}}{2\sqrt{p^2-q}}\varepsilon_q\right. \\
&\qquad \left.-\frac{p^2(p+\sqrt{p^2-q})}{2q\sqrt{p^2-q}}\varepsilon_1-\frac{\sqrt{p^2-q}(p+\sqrt{p^2-q})}{q}\left(\frac{1}{2}\varepsilon_2+\varepsilon_3\right)+\varepsilon_4\right].
\end{aligned}$$

Der relative Fehler ist in erster Näherung durch

$$\varepsilon_x^{(1)} = \bar{\varepsilon}_x + \tilde{\varepsilon}_x^{(1)}$$

mit dem unvermeidbaren Fehler

$$\bar{\varepsilon}_x \doteq -\frac{p}{\sqrt{p^2-q}}\varepsilon_p + \frac{p+\sqrt{p^2-q}}{2\sqrt{p^2-q}}\varepsilon_q$$

und dem in diesem Algorithmus erzeugten Rundungsfehler

$$\tilde{\varepsilon}_x^{(1)} \doteq -\frac{p^2(p+\sqrt{p^2-q})}{2q\sqrt{p^2-q}}\varepsilon_1 - \frac{\sqrt{p^2-q}(p+\sqrt{p^2-q})}{q}\left(\frac{1}{2}\varepsilon_2+\varepsilon_3\right)+\varepsilon_4$$

gegeben. Für $q \approx p^2$ liegt der Rundungsfehlereinfluss in der Größenordnung des unvermeidbaren Fehlers. Für $q \ll 1$ wird der Rundungsfehler i. a. beliebig groß, auch wenn der unvermeidbare Fehler klein ist!
Algorithmus 2:

$$x_1 = p^2, \quad x_2 = x_1 - q, \quad x_3 = \sqrt{x_2}, \quad x_4 = p + x_3, \quad x = q/x_4.$$

Auf einem Rechner sei der folgende Algorithmus realisiert:
Algorithmus 2':

$$\hat{x}_1 = \mathrm{gl}\,(p^2), \quad \hat{x}_2 = \mathrm{gl}\,(\hat{x}_1 - q),$$
$$\hat{x}_3 = \mathrm{gl}\,(\sqrt{\hat{x}_2}), \quad \hat{x}_4 = \mathrm{gl}\,(p+\hat{x}_3), \quad \hat{x} = \mathrm{gl}\,(q/\hat{x}_4).$$

Die Berechnungen bis x_3 entsprechen denen aus dem ersten Algorithmus. Wir übernehmen daher die Fehleranalyse bis einschließlich x_3. Es gilt

$$\hat{x}_3 \doteq x_3\left(1 + \frac{p^2}{p^2-q}\varepsilon_p - \frac{q}{2(p^2-q)}\varepsilon_q + \frac{p^2}{2(p^2-q)}\varepsilon_1 + \frac{1}{2}\varepsilon_2 + \varepsilon_3\right).$$

Weiter erhält man

$$\begin{aligned}
\hat{x}_4 &= (p(1+\varepsilon_p)+\hat{x}_3)(1+\varepsilon_4) \\
&\doteq p(1+\varepsilon_p)(1+\varepsilon_4) \\
&\quad +x_3\left(1+\frac{p^2}{p^2-q}\varepsilon_p-\frac{q}{2(p^2-q)}\varepsilon_q+\frac{p^2}{2(p^2-q)}\varepsilon_1+\frac{1}{2}\varepsilon_2+\varepsilon_3\right)(1+\varepsilon_4) \\
&\doteq x_4\left[1+\left(\frac{p}{p+\sqrt{p^2-q}}+\frac{p^2}{(p+\sqrt{p^2-q})\sqrt{p^2-q}}\right)\varepsilon_p\right. \\
&\quad -\frac{q}{2(p+\sqrt{p^2-q})\sqrt{p^2-q}}\varepsilon_q+\frac{p^2}{2(p+\sqrt{p^2-q})\sqrt{p^2-q}}\varepsilon_1 \\
&\quad \left.+\frac{\sqrt{p^2-q}}{p+\sqrt{p^2-q}}\left(\frac{1}{2}\varepsilon_2+\varepsilon_3\right)+\varepsilon_4\right]
\end{aligned}$$

und

$$\begin{aligned}
\hat{x} &= \frac{q(1+\varepsilon_q)}{\hat{x}_4}(1+\varepsilon_5) \\
&\doteq \frac{q}{x_4}\left[1+\varepsilon_q-\frac{p}{\sqrt{p^2-q}}\varepsilon_p+\frac{q}{2(p+\sqrt{p^2-q})\sqrt{p^2-q}}\varepsilon_q\right. \\
&\quad \left.-\frac{p^2}{2(p+\sqrt{p^2-q})\sqrt{p^2-q}}\varepsilon_1-\frac{\sqrt{p^2-q}}{p+\sqrt{p^2-q}}\left(\frac{1}{2}\varepsilon_2+\varepsilon_3\right)-\varepsilon_4+\varepsilon_5\right] \\
&\doteq x\left[1-\frac{p}{\sqrt{p^2-q}}\varepsilon_p+\frac{p+\sqrt{p^2-q}}{2\sqrt{p^2-q}}\varepsilon_q\right. \\
&\quad \left.-\frac{p^2}{2(p+\sqrt{p^2-q})\sqrt{p^2-q}}\varepsilon_1-\frac{\sqrt{p^2-q}}{p+\sqrt{p^2-q}}\left(\frac{1}{2}\varepsilon_2+\varepsilon_3\right)-\varepsilon_4+\varepsilon_5\right].
\end{aligned}$$

Der relative Fehler ist in diesem Falle durch

$$\varepsilon_x^{(2)} = \bar{\varepsilon}_x + \tilde{\varepsilon}_x^{(2)}$$

gegeben. Die Größe $\bar{\varepsilon}_x$ ist dabei wieder der unvermeidbare Fehler und $\tilde{\varepsilon}_x^{(2)}$ ist der erzeugte Rundungsfehler. Für ihn gilt in erster Näherung:

$$\tilde{\varepsilon}_x^{(2)} \doteq -\frac{p^2}{2(p+\sqrt{p^2-q})\sqrt{p^2-q}}\varepsilon_1-\frac{\sqrt{p^2-q}}{p+\sqrt{p^2-q}}\left(\frac{1}{2}\varepsilon_2+\varepsilon_3\right)-\varepsilon_4+\varepsilon_5.$$

Für $p^2 \approx q$ liegt der Rundungsfehler in der Größenordnung des unvermeidbaren Fehlers. Der Fall $q \ll 1$ ist bei diesem Algorithmus nicht kritisch. Die Verstärkungsfaktoren der Fehler ε_1 (Berechnen von p^2), ε_2 (Subtraktion) und ε_3 (Quadratwurzel) sind beim ersten Algorithmus um den Faktor

$$\frac{(p+\sqrt{p^2-q})^2}{q} = \frac{p+\sqrt{p^2-q}}{p-\sqrt{p^2-q}} > 1$$

größer als beim zweiten. Der zweite Algorithmus ist daher in jedem Falle vorzuziehen. Der Grund für das schlechte Verhalten des ersten Algorithmus liegt darin, dass im Falle $q \ll 1$ bei der Subtraktion $p - \sqrt{p^2-q}$ Auslöschung auftritt. Diese wird im zweiten Algorithmus vermieden. ♡

Wir wollen ein Beispiel betrachten, um zwei grundlegende Prinzipien der Fehleranalyse zu erläutern.

1.10. Beispiel: Für die Eingabedaten $x_i \in \mathbb{R}$, $i = 1,\ldots,n$, ist die Summe

$$z^* = \sum_{i=1}^{n} x_i$$

zu berechnen. Praktisch stehen aber statt der exakten Daten x_i nur die fehlerbehafteten Daten $\tilde{x}_i = x_i(1+\vartheta_i)$, $i = 1,\ldots,n$, zur Verfügung. Es wird daher bestenfalls die Summe

$$\tilde{z} = \sum_{i=1}^{n} \tilde{x}_i$$

berechnet. Die Differenz zwischen z^* und $\tilde{z}$ ist gerade der absolute unvermeidbare Fehler:

$$\overline{\delta z} = \tilde{z} - z^* = \sum_{i=1}^{n} (\tilde{x}_i - x_i) = \sum_{i=1}^{n} \vartheta_i x_i.$$

Falls die relativen Eingabefehler der Abschätzung

$$|\vartheta_i| \leq K\text{eps}, \quad i = 1,\ldots,n$$

genügen, gilt für den unvermeidbaren Fehler

$$|\overline{\delta z}| = \left| \sum_{i=1}^{n} \vartheta_i x_i \right| \leq \sum_{i=1}^{n} |\vartheta_i|\,|x_i| \leq K\text{eps} \sum_{i=1}^{n} |x_i|.$$

Naheliegend ist der folgende Algorithmus zum Berechnen von $\tilde{z}$.

1.11. Rekursive Summation:

$z_0 = 0$
for $i = 1$ ***to*** n ***do***
  $z_i = z_{i-1} + x_i$
endfor
$z = z_n$

Wir wollen den Rundungsfehlereinfluss in diesem Algorithmus untersuchen. Dazu betrachten wir wieder die Computerrealisierung des obigen Algorithmus:

1.12. Computersummation:

$\hat{z}_0 = 0$
for $i = 1$ ***to*** n ***do***
  $\hat{z}_i = \hat{z}_{i-1} \oplus \tilde{x}_i = (\hat{z}_{i-1} + \tilde{x}_i)(1+\varepsilon_i)$
endfor
$\hat{z} = \hat{z}_n$

Die relativen Rundungsfehler ε_i, $i = 1, \ldots, n$ genügen den Ungleichungen

$$|\varepsilon_i| \leqq \text{eps}, \quad i = 1, \ldots, n.$$

Im Einzelnen ergibt sich

$$\begin{aligned}
\hat{z}_1 &= \tilde{x}_1(1+\varepsilon_1), \\
\hat{z}_2 &= \tilde{x}_1(1+\varepsilon_1)(1+\varepsilon_2) + \tilde{x}_2(1+\varepsilon_2), \\
\hat{z}_3 &= \tilde{x}_1(1+\varepsilon_1)(1+\varepsilon_2)(1+\varepsilon_3) + \tilde{x}_2(1+\varepsilon_2)(1+\varepsilon_3) + \tilde{x}_3(1+\varepsilon_3, \\
&\vdots \\
\hat{z} = \hat{z}_n &= \tilde{x}_1(1+\varepsilon_1)\cdots(1+\varepsilon_n) + \tilde{x}_2(1+\varepsilon_2)\cdots(1+\varepsilon_n) + \cdots \tilde{x}_n(1+\varepsilon_n) \\
&= \sum_{i=1}^{n} \tilde{x}_i \prod_{j=i}^{n} (1+\varepsilon_j).
\end{aligned}$$

Für die auftretenden Produkte gilt

$$\begin{aligned}
\prod_{j=i}^{n}(1+\varepsilon_j) &= 1 + \sum_{j=i}^{n} \varepsilon_j + O(\text{eps}^2) \\
&\doteq 1 + \sum_{j=i}^{n} \varepsilon_j,
\end{aligned}$$

und weiter

$$\left|\prod_{j=i}^{n}(1+\varepsilon_j)\right| \overset{\cdot}{\leq} 1+\sum_{j=i}^{n}|\varepsilon_j| \overset{\cdot}{\leq} 1+(n-j+1)\text{eps}.$$

Damit gilt

$$\hat{z}=\sum_{i=1}^{n}\tilde{x}_i(1+\varepsilon_i^{(n)})$$

mit

$$\varepsilon_i^{(n)}=\prod_{j=i}^{n}(1+\varepsilon_j)-1$$

und

$$\begin{aligned}|\varepsilon_i^{(n)}| &\overset{\cdot}{\leq} \min\{n-i+1,n-1\}\cdot\text{eps} \quad \text{(Man beachte, dass} \quad \varepsilon_1=0 \quad \text{gilt!)}\\ &\overset{\cdot}{\leq} (n-1)\text{eps}.\end{aligned}$$

Nun schätzen wir den gesamten erzeugten Rundungsfehler ab. Es gilt

$$\begin{aligned}|\hat{z}-\tilde{z}| &= \left|\sum_{i=1}^{n}\left(\tilde{x}_i(1+\varepsilon_i^{(n)})-\tilde{x}_i\right)\right|\\ &= \left|\sum_{i=1}^{n}\varepsilon_i^{(n)}\tilde{x}_i\right|\\ &\overset{\cdot}{\leq} \sum_{i=1}^{n}|\varepsilon_i^{(n)}|\,|\tilde{x}_i|\\ &\overset{\cdot}{\leq} (n-1)\text{eps}\sum_{i=1}^{n}|\tilde{x}_i|\\ &\doteq (n-1)\text{eps}\sum_{i=1}^{n}|x_i|.\end{aligned}$$

♡

Die im letzten Beispiel erhaltenen Abschätzungen lassen zwei Interpretationen zu.

Interpretation 1
Wir vergleichen die Abschätzung für den unvermeidbaren Fehler

$$|\overline{\delta z}| \stackrel{\cdot}{\leqq} K\text{eps}\sum_{i=1}^{n}|x_i|$$

mit der Abschätzung für den erzeugten Rundungsfehler

$$|\tilde{\delta} z| \stackrel{\cdot}{\leqq} (n-1)\text{eps}\sum_{i=1}^{n}|x_i|.$$

Der erzeugte Rundungsfehler ist höchstens das $\frac{n-1}{K}$-fache des unvermeidbaren Fehlers. Das führt zu folgender Definition. Ein numerischer Algorithmus heißt **stabil** mit der Fehlerkonstanten F, falls der erzeugte Rundungsfehler höchstens das F-fache des unvermeidbaren Fehlers beträgt. Um die Stabilität eines Algorithmus zu untersuchen, braucht man einerseits möglichst realistische Abschätzungen für den unvermeidbaren Fehler und andererseits gute Abschätzungen für den erzeugten Rundungsfehler. Diese Vorgehensweise heißt **Vorwärtsanalyse**.

Interpretation 2
Wir vergleichen die Darstellung

$$z^* = \sum_{i=1}^{n} x_i = \sum_{i=1}^{n} \frac{\tilde{x}_i}{1+\vartheta_i} \doteq \sum_{i=1}^{n} \tilde{x}_i(1-\vartheta_i), \quad |\vartheta_i| \leqq K\text{eps}, \quad i = 1,\ldots,n$$

des gesuchten Ergebnisses mit der Darstellung

$$\hat{z} = \sum_{i=1}^{n} \tilde{x}_i(1+\varepsilon_i^{(n)}), \quad |\varepsilon_i^{(n)}| \leqq (n-1)\text{eps}, \quad i = 1,\ldots,n$$

des berechneten Ergebnisses. Man sieht, dass man das berechnete Ergebnis $\hat{z}$ als exakte Summe der Eingabedaten

$$\hat{x} = \tilde{x}_i(1+\varepsilon_i^{(n)}), \quad i = 1,\ldots,n,$$

auffassen darf. Die relativen Fehler $\varepsilon_i^{(n)}$ dieser Daten betragen höchstens das $\frac{n-1}{K}$-fache des Störungsniveaus Keps, das wir für die relativen Eingabefehler ϑ_i annehmen mussten. Wir haben statt der eigentlichen Aufgabe eine "'benachbarte"' Aufgabe gelöst. Der Faktor $\frac{n-1}{K}$ gibt dabei an, wie weit die "'benachbarte"' Aufgabe von der ursprünglichen Aufgabe entfernt ist. Die Vorgehensweise, die dieser Interpretation zugrunde liegt, heißt **Rückwärtsanalyse**. Wir haben hier versucht, Eingabedaten für den Algorithmus zu konstruieren,

die bei exakter Rechnung das Ergebnis liefern, das wir als rundungsfehlerbehaftetes Ergebnis erhalten. Damit ist eine zweite wünschenswerte Eigenschaft eines Algorithmus definierbar. Ein numerischer Algorithmus heißt **gutartig** mit der Fehlerkonstanten F, falls das berechnete Ergebnis als exaktes Ergebnis einer gestörten Aufgabe interpretierbar ist, wobei das Störungsniveau höchstens das F-fache der Datenunsicherheit beträgt. **Bemerkung:** Für lokal lipschitzstetige Aufgaben folgt aus der numerischen Gutartigkeit die numerische Stabilität. Numerische Gutartigkeit ist die bestmögliche Qualität eines Algorithmus. Numerische Stabilität dagegen ist eine Mindestanforderung, die wir an einen Algorithmus stellen. Auch bei stabilen Algorithmen wird der Rundungsfehler i. a. beliebig groß! (Wenn der unvermeidbare Fehler unbeschränkt ist.)

1.13. Beispiel: Es ist die Summe von 2^L Zahlen zu berechnen. Wir wenden folgenden Summationsalgorithmus an:

1.14. Binäre Summation:

for $k = 1$ ***to*** 2^L ***do***
  $z_k^{(0)} = x_k$
endfor
for $l = 1$ ***to*** L ***do***
  for $k = 1$ ***to*** 2^{L-l} ***do***
    $z_k^{(l)} = z_{2k-1}^{(l-1)} + z_{2k}^{(l-1)}$
  endfor
endfor
$z = z_1^{(L)}$

Hier lässt sich zeigen, dass für das berechnete Ergebnis $\hat{z}$ gilt:

$$\hat{z} = \sum_{i=1}^{2^L} \tilde{x}_i(1+\varepsilon_i), \quad |\varepsilon_i| \leq L\text{eps}, \quad i = 1, \ldots, 2^L.$$

Die $\tilde{x}_i$ bezeichnen wie im vorigen Beispiel die fehlerbehafteten Eingabedaten. Für den gesamten erzeugten Rundungsfehler folgt damit

$$|\tilde{\delta}z| = |\hat{z} - \tilde{z}| \stackrel{\cdot}{\leq} L\text{eps} \sum_{i=1}^{2^L} |x_i|.$$

Dieser Algorithmus ist numerisch gutartig und stabil mit der Fehlerkonstanten L. Er ist damit bedeutend günstiger als die rekursive Summation, bei der die

Fehlerkonstante gleich 2^L ist. Weitere Verbesserungen erzielt man, wenn man negative und positive Summanden für sich binär aufaddiert und dies in absolut aufsteigender Reihenfolge, wobei nach jedem Durchlauf neu sortiert wird. Dabei kann einerseits frühestens am Ende einmal Auslöschung eintreten; andererseits wird der absolute Fehler so klein wie möglich. ♡

1.4. Auslöschung und konvergente Folgen

Die Abweichung einer Näherung y vom exakten Wert x haben wir bisher mit Hilfe des absoluten Fehlers $\delta(x,y) = |x-y|$ oder des relativen Fehlers $\varepsilon(x,y) = \frac{|x-y|}{|x|}$ gemessen. Dabei sollte man den absoluten Fehler im Falle $|x| \ll 1$ und den relativen Fehler im Falle $|x| \gg 1$ verwenden. Meist ist in einem Algorithmus nicht von vornherein bekannt, in welcher Größenordnung die Ergebnisse liegen. Aber die relative und die absolute Differenz sind im allgemeine als Abbruchkriterium ungeeignet.

1.15. Beispiel: Wählt man die relative Differenz

$$|x_{k+1} - x_k| < \delta |x_k|,$$

so folgt bei der Folge

$$x_k = 10^{-k}, \qquad \delta = 10^{-4}.$$

$$\frac{|x_{k+1} - x_k|}{x_k} = 0,9 > \delta.$$

Wählt man die absolute Differenz

$$|x_{k+1} - x_k| < \delta$$

und die Folge

$$x_k = 10^{12}\left(1 + 10^{-k}\right), \qquad \delta = 10^{-4},$$

so ergibt sich zunächst für die entsprechenden, gerundeten Rechnerzahlen bei $k > 7$

$$\bar{x}_k = \begin{cases} 10^{12} & \text{für} \quad i \text{ gerade} \\ 10^{12}(1+\text{eps}) & \text{für} \quad i \text{ ungerade} \end{cases}$$

und damit

$$|\bar{x}_{l+1} - \bar{x}_k| = 10^{12}\text{eps} > \delta.$$

♡

Wenn der Auslöschungswert ein Maß für die Anzahl der sich auslöschenden führenden Mantissenstellen zweier Zahlen sein soll, bietet sich die folgende Definition der Auslöschung zwischen zwei Zahlen, die sich in normalisierter Gleitpunktdarstellung befinden, an:

$$\mathrm{canc}(x,y) = /(|x \breve{-} y| \breve{+} r) \breve{-} r/.$$

Hierin bedeuten $\breve{-}, \breve{+}$ die Subtraktion bzw. Addition ohne Normalisierung, $/\cdot/$ die Mantisse und r die charakteristische Maschinenzahl ($r = 1$). Wählen wir z. B. $x = 0.12345 \cdot 10^3, y = 0.12458 \cdot 10^3$, so ergibt sich bei der Basis 10, 5-stelliger Mantisse und $r = 1$:

$$\mathrm{canc}(x,y) = /(0.00113 \cdot 10^3 \breve{+} 0.00100 \cdot 10^3) \breve{-} 0.00100 \cdot 10^3/ = 0.00113.$$

Diese Definition ist in einem Assembler-Programm sehr effizient programmierbar.
Es seien

$$x = m_x b^{\xi}, \quad y = m_y b^{\eta}, \quad r = m_r b^{\rho}$$

normalisierte Gleitpunktzahlen bei gegebener Basis b. Ferner nehmen wir eine unendliche Mantissenlänge an. Dann folgt mit $\mu = \max\{\xi, \eta\}$ und $\nu = \max\{\xi, \eta, \rho\}$

$$\begin{aligned} |x \breve{-} y| &= |m_x \cdot b^{\xi} \breve{-} m_y \cdot b^{\eta}| \\ &= |\left(m_x b^{\xi-\mu}\right) \cdot b^{\mu} \breve{-} \left(m_y b^{\eta-\mu}\right) \cdot b^{\mu}| \\ &= \frac{|x-y|}{b^{\mu}} \cdot b^{\mu} \end{aligned}$$

und daraus

$$\begin{aligned} \mathrm{canc}(x,y) &= /\left(\frac{|x-y|}{b^{\mu}} \cdot b^{\mu} \breve{+} m_r b^{\rho}\right) \breve{-} m_r b^{\rho}/ \\ &= /\left(\frac{|x-y|}{b^{\nu}} \cdot b^{\nu} \breve{+} \frac{r}{b^{\nu}} \cdot b^{\nu}\right) \breve{-} \frac{r}{b^{\nu}} \cdot b^{\nu}/ \\ &= /\frac{|x-y|}{b^{\nu}} \cdot b^{\nu}/ = \frac{|x-y|}{b^{\nu}} \cdot b^{\nu} \end{aligned}$$

Für die folgenden mathematischen Untersuchungen verwenden wir daher die folgende Definition. Wir definieren als Auslöschung zweier Zahlen x, y in normalisierter Gleitpunktdarstellung

$$\mathrm{aus}(x,y) = |x-y|\, b^{-\max\{\xi,\eta,\rho\}}.$$

Bei der Betrachtung konvergenter Folgen interessieren keine Zahlen x, y mit unterschiedlichen Vorzeichen, deren Mantissendifferenz nicht kleiner als die charakteristische Maschinenzahl r ist (etwa $r = 1$), da diese Situation für höchstens endlich viele Folgeglieder eintreten kann, die außerdem noch weit entfernt vom Grenzwert liegen.
Die Auslöschung bildet keine Metrik auf der Menge der reellen Zahlen. Lediglich in jedem der Intervalle

$$\begin{aligned} I_n^- &= (-b^{n+1}, -b^n), \qquad n \geqq \rho \\ I_0 &= (-b^\rho, +b^\rho), \\ I_n^+ &= [b^n, b^{n+1}], \qquad n \geqq \rho \end{aligned}$$

wird eine Metrik definiert, die sich vom Betrag nur um den Faktor $\frac{1}{b^{\max(\rho,n+1)}}$ unterscheidet:

$$\text{aus}(x,y) = \frac{1}{b^{\max(\rho,n+1)}}|x-y| \quad \text{mit } x,y \in I,$$

wobei I eines der obigen Intervalle darstellt. Daraus folgt: Die Auslöschung verhält sich in diesen Intervallen ähnlich wie der betrag. Interessant sind also nur Untersuchungen für Zahlen aus verschiedenen Intervallen und bei Konvergenzuntersuchungen nur solche Zahlenfolgen, die gegen Randpunkte der obigen Intervalle konvergieren.

1.16. Verhalten der Auslöschung bei linearer Konvergenz: *Es sei* $\{x_k\}$ *eine linear konvergierende, nicht konstante Zahlenfolge, d. h.*

$$\frac{|x_{k+1} - x_k|}{|x_k - x_{k-1}|} \leqq q < 1, \quad k = 1, 2, \cdot$$

Dann gibt es ein $i_0 \in \mathbb{N}$ *und paarweise disjunkte Indexmengen* N_1, N_2, N_3 *mit folgenden Eigenschaften:*

(i) Die Indexmengen enthalten ab i_0 *alle natürlichen Zahlen:*

$$N_1 \cup N_2 \cup N_3 = \{\, i \in \mathbb{N} \mid i > i_0 \,\}$$

(ii) Für alle $i, j \in N_1$ *mit* $i < j$ *existiert* $k \in N_3$ *mit* $i < k < j$.

(iii) Für alle $k \in N_1$ *gilt*

$$\frac{aus(x_{k+1}, x_k)}{aus\left(x_{(k)}, x_{k-1}\right)} \in (q, q \cdot b]\,.$$

Für alle $k \in N_2$ gilt

$$\frac{aus(x_{k+1}, x_k)}{aus(x_k, x_{k-1})} \in \left(\frac{q}{b}, q\right].$$

Für alle $k \in N_1$ gilt

$$\frac{aus(x_{k+1}, x_k)}{aus(x_k, x_{k-1})} \in \left[0, \frac{q}{b}\right].$$

Beweis: Es sei

$$x = m_x \cdot b^{\xi}$$

die normalisierte Gleitpunktdarstellung des Grenzwertes und o. B. d. A. $x \geqq 0$. Wir zeigen (i). Es sei $m_x \neq b^{-1}$ oder $\xi \leqq \rho$, d. h. die Folge konvergiert gegen eine Zahl, die innerhalb eines der obigen Intervalle I_n^-, I_o, I_n^+ liegt. Dann existiert eine Zahl i_0 derart, dass alle x_k für $k \geqq i_0$ in diesem Intervall liegen und es folgt für $k > i_0$

$$\frac{\text{aus}\,(x_{k+1}, x_k)}{\text{aus}\,(x_k, x_{k-1})} = \frac{|x_{k+1} - x_k|}{|x_k - x_{k-1}|} \leqq q,$$

woraus die Behauptung folgt mit $N_1 = \emptyset$.
Es sei nun $m_x = b^{-1}$ und $\xi > \rho$, d. h. die Folge konvergiere gegen einen der Randpunkte der Intervalle I_n^-, I_o, I_n^+.
Zunächst existiert ein i_0 derart, dass $x_k \in \left(b^{\xi-2}, b^{\xi}\right)$ für alle $k \geqq i_0$ gilt, d. h. alle diese Folgeglieder haben den Exponenten ξ oder $\xi - 1$:

$$\xi_k = \xi \quad \text{oder} \quad \xi_k = \xi - 1.$$

Wir definieren die folgenden Indexmengen

$$\begin{aligned} N_1^* &= \{\, k > i_0 \mid \xi_{k-1} = \xi, \xi_k = \xi_{k+1} = \xi - 1 \,\}, \\ N_3^* &= \{\, k > i_0 \mid \xi_{k-1} = \xi_k = \xi - 1, \xi_{k+1} = \xi \,\}, \\ N_2^* &= \{\, k > i_0 \mid i \notin N_1^* \cup N_3^* \,\} \end{aligned}$$

und zeigen, dass sie die Eigenschaft (ii) erfüllen. Es sei $i, j \in N_1^*$ und $i < j$. Als Index k wählen wir die kleinste Zahl, die größer als die Zahl i ist und für die $\xi_{k+1} = \xi$ gilt. Wegen $\xi_{j-1} = \xi$ gilt $k < j$. Nach Wahl von k und wegen $\xi_{i+1} = \xi_i = \xi - 1$ gilt $\xi_l = \xi - 1$ für $l = i, i+1, \ldots, k$, insbesondere also für $l = k-1, k$.

Es ist also $\xi_{k-1} = \xi_k = \xi - 1$ und $\xi_{k+1} = \xi$, folglich $k \in N_3^*$.
Nun defineren wir die Indexmengen

$$\begin{aligned} N_1 &= \left\{ k > i_0 \mid \frac{\operatorname{aus}(x_{k+1}, x_k)}{\operatorname{aus}(x_k, x_{k-1})} \in (q, q \cdot b] \right\}, \\ N_2 &= \left\{ k > i_0 \mid \frac{\operatorname{aus}(x_{k+1}, x_k)}{\operatorname{aus}(x_k, x_{k-1})} \in (\frac{q}{b}, q] \right\}, \\ N_3 &= \left\{ k > i_0 \mid \frac{\operatorname{aus}(x_{k+1}, x_k)}{\operatorname{aus}(x_k, x_{k-1})} \in [0, \frac{q}{b}] \right\}. \end{aligned}$$

Die Eigenschaften (i) und (iii) sind für diese Indexmengen offensichtlich; die Eigenschaft (ii) gilt wegen $N_1 \subset N_1^*$ und $N_3 \supset N_3^*$. ❀

1.17. Bemerkung: *Der 2. Teil des Beweises gilt auch für Grenzwerte in anderen Punkten, insbesondere in der Nähe von Randpunkten.*

1.18. Hinreichendes Konvergenzkriterium: *Es sei $\{x_k\}$ eine Zahlenfolge zu der ein $q \in (0,1)$ und ein $i_0 \in \mathbb{N}$ derart existieren, dass die im letzten Satz behaupteten Indexmengen mit den angegebenen Eigenschaften (i),(ii) und (iii) existieren. Dann konvergiert die Zahlenfolge.*

Beweis: Aus den Voraussetzungen folgt zunächst für $i \geqq j \geqq 0$ und o. B. d. A. $i_0 = 1$

$$\operatorname{aus}(x_{k+1}, x_k) \leqq q^j b \cdot \operatorname{aus}\left(x_{k-j+1}, x_{i-j}\right),$$

woraus sich ergibt, dass die zugeordnete Auslöschungsfolge eine Nullfolge ist. Wir zeigen, dass die Zahlenfolge beschränkt ist und nehmen an, sie ist unbeschränkt. Da die Auslöschungsfolge eine Nullfolge ist, existiert ein Index i^*, so dass für alle $i \geqq i^*$

$$\operatorname{aus}(x_{k+1}, x_k) < \frac{(b-1)(1-q)}{b^3}$$

ausfällt. Dann gilt für alle $k \geqq i^*$ mit $\max(\xi_k, \xi_{k+1}) \geqq \rho$ die Ungleichung

$$|\xi_k - \xi_{k+1}| \leqq 1.$$

Wäre nämlich $|\xi_k - \xi_{k+1}| \geqq 2$, also etwa $\xi_{k+1} \geqq \xi_k + 2$, dann folgte

$$\begin{aligned} \operatorname{aus}(x_{k+1}, x_k) &= \frac{|x_{k+1} - x_k|}{b^{\max(\xi_{k+1}, \xi_k, \rho)}} = \frac{|x_{k+1} - x_k|}{b^{\xi_{k+1}}} \\ &\geqq \frac{|x_{k+1}| - |x_k|}{b^{\xi_{k+1}}} \geqq b^{-1} - b^{-2} > \frac{(b-1)(1-q)}{b^3}, \end{aligned}$$

was im Widerspruch zur Voraussetzung steht. Wir wählen nun einen Index $n \geqq i^*$ derart, dass $|x_n| > b^\rho$ ausfällt; außerdem sei der Index N so gewählt, dass $|x_N| \geqq b^{\xi_n+1}$ gilt. Schließlich sei $m \in [n,N]$ der größte Index mit $|x_m| < b^{\xi_n}$ und $M \in [m,N]$ der kleinste Index mit $|x_M| \geqq b^{\xi_n+1}$. Wegen $|\xi_k - \xi_{k+1}| \leqq 1$ haben alle x_k mit $k \in (m,M)$ in der normalisierten Gleitpunktdarstellung den Exponenten $\xi_n + 1$ und x_M hat den Exponenten $\xi_n + 2$. Außerdem gilt für alle $k \in [m,M)$

$$\text{aus}\,(x_{k+1},x_k) = \frac{|x_{k+1}-x_k|}{b^{\max(\xi_{k+1},\xi_k,\rho)}} = \frac{|x_{k+1}-x_k|}{b^{\xi_n+1}} \geqq \frac{|x_{k+1}-x_k|}{b^{\xi_n+2}}$$

und

$$\text{aus}\,(x_{M-1},x_M) = \frac{|x_M - x_{M-1}|}{b^{\xi_n+2}}.$$

Nun folgt

$$\begin{aligned}
|x_M - x_m| &\leqq |x_{m+1}-x_m| + \cdots + |x_M - x_{M-1}| \\
&\leqq b^{\xi_n+2}\left[\text{aus}(x_{m+1},x_m) + \cdots + \text{aus}(x_M,x_{M-1})\right] \\
&\leqq b^{\xi_n+2}\text{aus}(x_{m+1},x_m)\left[1 + qb + q^2b + \cdots\right] \\
&\leqq b^{\xi_n+2}\text{aus}(x_{m+1},x_m)\cdot\frac{b}{1-q}.
\end{aligned}$$

Außerdem folgt wegen der Wahl von m und M

$$|x_M - x_m| \geqq |x_M| - |x_m| \geqq b^{\xi_n+1-b^{\xi_n}} = (b-1)b^{\xi_n}$$

und daraus

$$\begin{aligned}
b^{\xi_n+2}\text{aus}(x_{m+1},x_m)\cdot\frac{b}{1-q} &\geqq (b-1)b^{\xi_n} \\
\text{aus}(x_{m+1},x_m) &\geqq \frac{(b-1)(1-q)}{b^3}
\end{aligned}$$

im Widerspruch zur Wahl von i^*. Folglich ist die Folge $\{x_k\}$ beschränkt.
Zu zeigen ist, dass es sich um eine Fundamentalfolge handelt. Wegen der Beschränktheit der Folge $\{x_k\}$ existiert der Wert $\bar{\xi} = \max_k(\xi_k,\rho)$ und wir erhalten

$$|x_{k+1}-x_k| = \text{aus}(x_{k+1},x_k)b^{\max(\xi_{k+1},\xi_k,\rho)} \leqq \text{aus}(x_1,x_0)q^k b^{\bar{\xi}+1} = C\cdot q^k.$$

Mit dieser Abschätzung schließen wir

$$\begin{aligned} |x_{n+m}-x_n| &\leqq |x_{n+m}-x_{n+m-1}|+|x_{n+m-1}-x_{n+m-2}|+\cdots+|x_{n+1}-x_n| \\ &\leqq C(q^{n+m-1}+q^{n+m-2}+\cdots+q^n) \\ &= C\cdot q^n(1+q+q^2+\cdots+q^{m-1}) \\ &\leqq Cq^n\cdot\frac{1}{1-q}, \end{aligned}$$

was uns zeigt, dass es sich um eine Fundamentalfolge handelt. ❀

1.19. Verhalten der Auslöschung bei überlinear konvergenten Zahlenfolgen.: *Es sei $\{x_k\}$ eine nicht konstante Zahlenfolge, die von m-ter Ordnung konvergiert:*

$$\frac{|x_{k+1}-x_k|}{|x_k-x_{k-1}|^m}\leqq c.$$

Dann gibt es ein $k_0, c^*>0$ und paarweise elementfremde Indexmengen N_1, N_2, N_3 mit folgenden Eigenschaften:

(ii)

$$N_1\cup N_2\cup N_3=\{\, i\in\mathbb{N}\mid i>k_0\,\}$$

(ii) Für alle $i,j\in N_1$ existiert ein $k\in N_3$ mit $i<k<j$.

(iii)

$$\frac{aus(x_{k+1},x_k)}{aus(x_k,x_{k-1})^m}\in\begin{cases}(c^*,c^*\cdot b] & für \quad k\in N_1\\ \left(\frac{c^*}{b^m},c^*\right] & für \quad k\in N_2\\ \left[0,\frac{c^*}{b^m}\right] & für \quad k\in N_3\end{cases}$$

Beweis: Der Beweis verläuft analog zum Beweis des Satzes über das Verhalten der Auslöschung bei linearer Konvergenz. Für die Abschätzungen beachte man lediglich:

$$\frac{\mathrm{aus}(x_{k+1},x_k)}{\mathrm{aus}(x_k,x_{k-1})^m}=\frac{|x_{k+1}-x_k|}{|x_k-x_{k-1}|^m}\cdot b^{(m-1)\bar\xi}\leqq cb^{(m-1)\bar\xi}=c^*\quad(\bar\xi=\max(\xi,\rho)),$$

sowie

$$\begin{aligned}\frac{\mathrm{aus}(x_{k+1},x_k)}{\mathrm{aus}(x_k,x_{k-1})^m} &= \frac{|x_{k+1}-x_k|}{|x_k-x_{k-1}|^m}\cdot\frac{b^{m\cdot\max(\xi_k,\xi_{k-1})}}{b^{\max(\xi_{k+1},\xi_k)}}\\ &= \begin{cases} cb^{m(\bar\xi-1)-\bar\xi}=\frac{c^*}{b^m} & \text{für} \quad in\in N_3\\ cb^{m\bar\xi-\bar\xi+1}=c^*b & \text{für} \quad in\in N_1\\ cb^{m\bar\xi-\bar\xi}=c^* & \text{für} \quad in\in N_2\end{cases}\end{aligned}$$

und schließlich die Beziehungen

$$N_1 = \left\{ k > k_0 \mid \frac{\text{aus}(x_{k+1}, x_k)}{\text{aus}(x_k, x_{k-1})^m} \in (c^*, c^* b] \right\}$$
$$N_2 = \left\{ k > k_0 \mid \frac{\text{aus}(x_{k+1}, x_k)}{\text{aus}(x_k, x_{k-1})^m} \in \left(\frac{c^*}{b^m}, c^* \right] \right\}$$
$$N_3 = \left\{ k > k_0 \mid \frac{\text{aus}(x_{k+1}, x_k)}{\text{aus}(x_k, x_{k-1})^m} \in \left[0, \frac{c^*}{b^m} \right] \right\}.$$

❀

1.20. Hinreichendes Konvergenzkriterium: *Es sei $\{x_k\}$ eine Zahlenfolge, $c^* > 0$, $k_0 \in \mathbb{N}$ und N_1, N_2, N_3 paarweise elementfremde mengenmit den Eigenschaften aus dem letzten Satz. Weiterhin gelte*

$$aus(x_{k_0}, x_{k_0)+1})^{m-1} < \frac{1}{c^* b}.$$

Dann existiert ein k_1 derart, dass die Folge $\{x_k\}$ ab dem Index k_1 von m-ter Ordnung konvergiert.

Beweis: Wir setzen

$$q = c^* b \cdot \text{aus}(x_{k_0}, x_{k_0+1})^{m-1}.$$

Offenbar gilt $q < 1$. Induktiv beweisen wir für $k \geqq k_0$

$$\text{aus}(x_{k+1}, x_k)^{m-1} \leqq \frac{q}{c^* b} :$$

Gilt diese Ungleichung für ein $k \geqq k_0$, so folgt aus den Voraussetzungen

$$\text{aus}(x_{k+2}, x_{k+1}) \leqq c^* b \cdot \text{aus}(x_{k+1}, x_k)^m \leqq q \cdot \text{aus}(x_{k+1}, x_k),$$

also auch

$$\text{aus}(x_{k+2}, x_{k+1})^{m-1} \leq \frac{q}{c^* b}.$$

Damit sind die Voraussetzungen für das hinreichende Konvergenzkriterium 1.18 erfüllt; die Folge konvergiert. Es sei $x = m_x \cdot b^\xi$ der Grenzwert; dann existiert ein k_1, so dass für alle $k \geqq k_1$

$$|x_k| \in [b^{\xi - 1}, b^{\xi + 1}).$$

Daraus folgt

$$\begin{aligned}\frac{|x_{k+1}-x_k|}{|x_k-x_{k-1}|^m} &= \frac{\mathrm{aus}(x_{k+1},x_k)}{\mathrm{aus}(x_k,x_{k-1})^m}\cdot\frac{b^{\max(\xi_{k+1},\xi_k,\rho)}}{b^{\max(\xi_k,\xi_{k-1},\rho)\cdot m}}\\ &\leqq c^*b\cdot\frac{b^{\max(\xi+1,\rho)}}{b^{\max(\xi-1,\rho)}}=c^{**}\end{aligned}$$

und die Konvergenz von m-ter Ordnung ist bewiesen. ❀

Wir wollen nun das Verhalten der Auslöschung bei konvergenten Vektorfolgen untersuchen. Dazu sei im Folgenden $||\cdot||$ die Maximumnorm für $x\in\mathbb{R}^n$

$$||x||=\max_i|x_i|.$$

Unter der Auslöschung der Vektoren $x,y\in\mathbb{R}^n$ verstehen wir die Größe

$$\mathrm{Aus}(x,y)=\max_i\mathrm{aus}(x_i,y_i).$$

1.21. Verhalten der Auslöschung bei linear konvergierenden Vektorfolgen: *Es sei $\{x^k\}$ eine linear konvergierende Vektorfolge:*

$$\frac{||x^{k+1}-x^k||}{||x^k-x^{k-1}||}\leqq q<1\qquad\forall k.$$

Dann gibt es ein k_0 und ein $c>0$, so dass

$$Aus(x^k,x^{k+1})\leqq cq^k\qquad\forall k\geqq k_0.$$

Beweis: Zunächst seien $\{\nu_k\}$ die Folge der Komponentenindices, wo die Vektornorm angenommen wird:

$$||x^{k+1}-x^k||=|x^{k+1}_{\nu_k}-x^k_{\nu_k}|,$$

$\{\mu_k\}$ die Folge der Komponentenindices, wo die Auslöschung angenommen wird:

$$\mathrm{Aus}(x^{k+1},x^k)=\mathrm{aus}(x^{k+1}_{\mu_k},x^k_{\mu_k}),$$

$\{a_k\}$ die Folge

$$a_k=\frac{|x^{k+1}_{\nu_k}-x^k_{\nu_k}|}{|x^{k+1}_{\mu_k}-x^k_{\mu_k}|}\geqq 1,$$

$$x = \lim x^k, \quad x_i = m_{x_i} \cdot b^{\xi_i}, \quad x_i^k = m_{x_i^k} \cdot b^{\xi_i^k}$$

die normalisierten Gleitpunktdarstellungen der Vektorkomponenten, sowie

$$N_k = \max(\xi_{\mu_k}^{k+1}, \xi_{\mu_k}^k, \rho).$$

Es folgt

$$\begin{aligned}
\mathrm{Aus}(x^{k+1}, x^k) &= \mathrm{aus}(x_{\mu_k}^{k+1}, x_{\mu_k}^k) = \frac{|x_{\mu_k}^{k+1}, x_{\mu_k}^k|}{b^{N_k}} \\
&= \frac{1}{a_k b^{N_i}} |x_{\nu_k}^{k+1} - x_{\nu_k}^k| = \frac{1}{||x^{k+1} - x^k||} \\
&\leqq \frac{1}{a_k b^{N_i}} q ||x^k - x^{k-1}|| \\
&= \frac{1}{a_k b^{N_i}} q |x_{\nu_{k-1}}^k - x_{\nu_{k-1}}^{k-1}| = \frac{a_{k-1}}{a_k b^{N_i}} q |x_{\mu_{k-1}}^k - x_{\mu_{k-1}}^{k-1}| \\
&= \frac{a_{k-1}}{a_k b^{N_i}} q b^{N_{k-1}} \mathrm{aus}(x_{\mu_{k-1}}^k, x_{\mu_{k-1}}^{k-1}) \\
&= q \frac{a_{k-1}}{a_k} b^{N_{k-1} - N_k} \mathrm{Aus}(x^k, x^{k-1})
\end{aligned}$$

Durch fortgesetztes Abschätzen folgt daraus

$$\mathrm{Aus}(x^{k+1}, x^k) \leqq q^k \frac{a_0}{a_k} b^{N_0 - N_k} \mathrm{Aus}(x^1, x^0).$$

❀

1.5. Aufgaben

1. Man berechne die Summe von 2^L Zahlen x_k, $k = 1, \ldots, 2^L$ mittels binärer Summation, untersuche den Rundungsfehler durch Vorwärts- und Rückwärtsanalyse und vergleiche mit der rekursiven Summation.

2. Die folgenden Ausdrücke sind so umzuformen, dass bei ihrem Berechnen Auslöschung vermieden wird:

   (a)

   $$\frac{1}{1+2x} - \frac{1-x}{1+x}, \quad |x| \ll 1,$$

(b)

$$\sqrt{x+\frac{1}{x}}-\sqrt{x-\frac{1}{x}}, \quad |x| \gg 1,$$

(c)

$$\frac{1-\cos x}{x}, \quad x \neq 0, \quad |x| \ll 1.$$

3. Man schreibe ein Programm zur binären Summation von 2^L reellen Zahlen, ohne diese zu überspeichern; au"serdem dürfen neben einigen Hilfsvariablen nur ein INTEGER- und ein REAL-Feld der Länge L verwendet werden.

Chapter 2

Interpolation

2.1. Einführung

Es sei eine Funktion

$$\Phi: \quad D \subseteq \mathbb{R} \longrightarrow \mathbb{R} \quad (\text{oder } D \subseteq \mathbb{C} \longrightarrow \mathbb{C}),$$

gegeben, die zusätzlich von unbekannten Parametern abhängen möge:

$$\Phi = \Phi(x; a_0, a_1, \ldots, a_n).$$

Das Interpolationsproblem besteht darin, die $n+1$ Parameter $a_0, a_1, \ldots, a_n$ so zu bestimmen, dass gewisse Nebenbedingungen erfüllt sind. Im einfachsten Falle sind Paare (x_i, y_i), $i = 0, \ldots, n$, reeller (oder komplexer) Zahlen mit paarweise verschiedenen Daten x_i, $i = 0, \ldots, n$, gegeben. Die Parameter a_0 bis a_n sind dann so zu wählen, dass

$$\Phi(x_i; a_0, a_1, \ldots, a_n) = y_i \quad i = 0, \ldots, n$$

gilt. In manchen Anwendungen sind neben den Werten y_i an den Stellen x_i noch gewisse Ableitungswerte vorgegeben. Man hat dann Interpolationsbedingungen der Form

$$\Phi^{(k)}(x_i; a_0, a_1, \ldots, a_n) = y_i^{(k)} \quad k = 0, \ldots, n_i - 1,\ i = 0, \ldots, m$$

mit $n_0 + \cdots + n_m = n$.
Die Werte x_i heißen **Stützstellen** und die zugehörigen y_i **Stützwerte**. Die daraus gebildeten Paare $(x_0, y_0), \ldots, (x_n, y_n)$ heißen **Stützpunkte**. Interpolationsprobleme lassen sich in lineare und nichtlineare Probleme unterteilen. Ein Interpolationsproblem heißt **linear**, falls die Interpolationsfunktion linear von den Parametern abhängt:

$$\Phi(x; a_0, a_1, \ldots, a_n) \equiv a_0 \varphi_0(x) + a_1 \varphi_1(x) + \cdots + a_n \varphi_n(x).$$

Alle anderen Interpolationsprobleme heißen **nichtlinear**. **Beispiele für lineare Interpolationsaufgaben**

- Polynominterpolation: $\Phi(x) = a_0 + a_1 x + a_2 x^2 + \cdots + a_n x^n$.
- Trigonometrische Interpolation: $\Phi(x) = a_0 + a_1 e^{ix} + a_2 e^{2ix} + \cdots + a_n e^{nix}$.
- Kubische Spline-Interpolation: Es gilt $\Phi \in C^2[a,b]$ und für eine gegebene Unterteilung

  $$a = x_0 < x_1 < \cdots < x_n = b$$

  des Intervalls $[a,b]$ ist die Funktion Φ auf jedem Teilintervall $[x_i, x_{i+1}]$ ein Polynom dritten Grades. Da die Interpolationsfunktion zweimal stetig differenzierbar sein soll, schließen die Polynomstücke in den Endpunkten der Teilintervalle bis zur 2. Ableitung stetig aneinander an.

Beispiele für nichtlineare Interpolationsaufgaben

- Rationale Interpolation:

  $$\Phi(x) = \frac{a_0 + a_1 x + a_2 x^2 + \cdots + a_n x^n}{b_0 + b_1 x + b_2 x^2 + \cdots + b_m x^m}.$$

- Interpolation durch Exponentialsummen:

  $$\Phi(x) = a_0 e^{\lambda_0 x} + a_1 e^{\lambda_1 x} + a_2 e^{\lambda_2 x} + \cdots + a_n e^{\lambda_n x}.$$

  Parameter sind in diesem Falle $a_0, \ldots, a_n$ und $\lambda_0, \ldots, \lambda_n$.

2.2. Polynominterpolation

2.2.1. Existenz- und Eindeutigkeitssatz

Es sei

$$\Pi_n = \left\{ P \mid P(x) = a_0 + a_1 x + a_2 x^2 + \cdots + a_n x^n, a_i \in \mathbb{R}, i = 0, \ldots, n \right\}$$

die Menge aller reellen Polynome vom Grade höchstens n. Das LAGRANGEsche Interpolationsproblem besteht in folgendem:
Zu gegebenen Stützpunkten $(x_0, f_0), \ldots, (x_n, f_n)$ ist ein Polynom $P \in \Pi_n$ gesucht, das die Interpolationsbedingungen

$$P(x_i) = f_i \quad i = 0, \ldots, n$$

erfüllt.
Als erstes stellt sich die Frage nach der Existenz und Eindeutigkeit eines solchen Polynoms. Diese Frage beantwortet der folgenden Satz.

2.1. Satz: *Zu gegebenen Stützstellen $x_0,\ldots,x_n$ und Stützwerten $f_0,\ldots,f_n$ existiert genau ein Polynom vom Grade höchstens n ($P\in\Pi_n$) mit*

$$P(x_i)=f_i \quad i=0,\ldots,n.$$

Beweis: (i) Existenz:
Wir geben ein Polynom $P\in\Pi_n$ an, das die Interpolationsbedingungen erfüllt. Dazu definieren wir folgende Hilfspolynome:

$$\begin{aligned} L_i^{(n)}(x) &= \prod_{\substack{j=0\\ j\neq i}}^{n}\frac{x-x_j}{x_i-x_j}=\frac{(x-x_0)\cdots(x-x_{i-1})(x-x_{i+1})\cdots(x-x_n)}{(x_i-x_0)\cdots(x_i-x_{i-1})(x_i-x_{i+1})\cdots(x_i-x_n)} \\ &= \frac{\omega(x)}{(x-x_i)\omega'(x_i)}, \quad \omega(x)=(x-x_0)(x-x_1)\cdots(x-x_n). \end{aligned}$$

Das Polynom $L_i^{(n)}$ heißt LAGRANGE-**Polynom** zur Stützstelle x_i. Für diese Polynome gilt offensichtlich

$$\begin{aligned} L_i^{(n)} &\in \Pi_n, \quad i=0,\ldots,n \\ L_i^{(n)}(x_j) &= \delta_{ij}=\begin{cases} 1 \text{ für } i=j \\ 0 \text{ für } i\neq j \end{cases}. \end{aligned}$$

Damit erhalten wir die LAGRANGEsche Darstellung des Interpolationspolynoms:

$$P(x)=\sum_{i=0}^{n} f_i L_i^{(n)}(x).$$

Da Π_n ein Vektorraum über $\mathbb{R}$ ist, folgt sofort $P\in\Pi_n$. Außerdem gilt

$$P(x_j)=\sum_{i=0}^{n} f_i L_i^{(n)}(x_j)=\sum_{i=0}^{n} f_i\delta_{ij}=f_j \quad j=0,\ldots,n.$$

Somit haben wir ein Polynom gefunden, das die Interpolationsbedingungen erfüllt.

(ii) Eindeutigkeit:
Es seien $P_1, P_2 \in \Pi_n$ Polynome mit

$$\begin{aligned} P_1(x_i) &= f_i \quad i = 0, \ldots, n \\ P_2(x_i) &= f_i \quad i = 0, \ldots, n. \end{aligned}$$

Dann gilt für das Polynom $Q = P_1 - P_2$

$$Q \in \Pi_n, \quad Q(x_i) = 0 \quad i = 0, \ldots, n.$$

Das Polynom Q hat daher einerseits höchstens den Grad n, aber andererseits mehr als n Nullstellen; somit ist Q das Nullpolynom. Damit gilt $Q(x) \equiv 0$ und weiter $P_1(x) \equiv P_2(x)$. ❀

Eine Möglichkeit zum Berechnen des LAGRANGEschen Interpolationspolynoms ergibt sich durch folgende Betrachtung. Der Nenner der Funktion $L_i^{(n)}$ hängt nicht von x ab; wir setzen daher

$$a_i = \frac{1}{\prod\limits_{\substack{k=0 \\ k \neq i}}^{n} (x_i - x_k)}$$

und erhalten

$$P(x) = \sum_{i=0}^{n} f_i a_i \prod_{\substack{k=0 \\ k \neq i}}^{n} (x - x_k).$$

Wegen

$$1 = \sum_{i=0}^{n} L_i^{(n)}(x) = \sum_{i=0}^{n} a_i \prod_{\substack{k=0 \\ k \neq i}}^{n} (x - x_k) \quad \text{und} \quad \prod_{\substack{k=0 \\ k \neq i}}^{n} (x - x_k) = \frac{\prod\limits_{k=0}^{n} (x - x_k)}{x - x_i}$$

erhalten wir schließlich

$$P(x) = \frac{\sum\limits_{i=0}^{n} \frac{a_i}{x - x_i} f_i}{\sum\limits_{i=0}^{n} \frac{a_i}{x - x_i}}.$$

Diese Darstellung ist für $x \neq x_i$ definiert. Zusammen gilt somit

$$P(x) = \begin{cases} f_i & x = x_i \ (i = 0, \ldots, n) \\ \dfrac{\displaystyle\sum_{i=0}^{n} \frac{a_i}{x - x_i} f_i}{\displaystyle\sum_{i=0}^{n} \frac{a_i}{x - x_i}} & x \neq x_i \ (i = 0, \ldots, n) \end{cases}.$$

Dies nennt man **baryzentrische Darstellung** des Polynoms P; sie lässt sich gut numerisch auswerten. Dazu sind zunächst die Koeffizienten zu berechnen:

2.2. Berechnen der Koeffizienten für die baryzentrische Darstellung:
Gegeben seien Stützstellen $x_0, \ldots, x_n$.
Zu berechnen sind die Koeffizienten für die baryzentrische Darstellung des Interpolationspolynoms.

for $i = 0$ ***to*** n ***do***
  $t = 1$; $s = x_i$
  for $k = 0$ ***to*** $i - 1$ ***do***
    $t := t \cdot (s - x_k)$
  endfor
  for $k = i + 1$ ***to*** n ***do***
    $t := t \cdot (s - x_k)$
  endfor
  $a_i = 1/t$
endfor

Nun berechnet sich der Interpolationswert nach folgendem Algorithmus.

2.3. Berechnen des Interpolationswertes bei einer baryzentrischen Darstellung:
Gegeben sind die Koeffizienten $a_0, \ldots, a_n$ *der baryzentrischen Darstellung eines Interpolationspolynoms zu Stützstellen* $x_0, \ldots, x_n$ *und Stützwerten* $f_0, \ldots, f_n$.
Gesucht ist der Wert des Interpolationspolynoms an der Stelle x.

$u = 0$; $v = 0$
for $i = 0$ ***to*** n ***do***
  $t = \dfrac{a_i}{x - x_i}$; $u = u + t \cdot f_i$; $v = v + t$
endfor
$P = \dfrac{u}{v}$

Aufwand: $n+2$ *Divisionen,* $n+1$ *Multiplikationen und* $2(n+1)$ *Additionen.*

Aus numerischer Sicht ist anzumerken, dass man hier durch Reduzieren des Fehlerfortpflanzens das Resultat numerisch gutartig erhält.

2.2.2. Der Neville-Algorithmus

Das LAGRANGEsche Interpolationspolynom dient zum Lösen mathematischer Probleme. In diesen Problemen wird man das Polynom selbst benötigen, oder Funktionswerte an wenigen oder vielen Stellen. Wir behandeln zunächst den Fall, dass man Funktionswerte des Interpolationspolynoms an wenigen Stellen benötigt. Im Sinne der numerischen Mathematik fordert man daher Algorithmen, die mit möglichst geringem Aufwand ihren Zweck erfüllen.
Dazu definieren wir Polynome P_{ik} mit folgenden Eigenschaften

- $P_{ik} \in \Pi_k$,
- $P_{ik}(x_j) = f_j \quad j = i-k, i-k+1, \ldots, i.$

Das Polynom P_{ik} löst das Interpolationsproblem mit den Stützstellen $x_{i-k}, \ldots, x_i$ und den zugehörigen Stützwerten.

2.4. Satz: *Für die Polynome* P_{ik} *gilt*

$$\begin{aligned} P_{i0}(x) &\equiv f_i \quad i = 0,1,\ldots,n \\ P_{ik}(x) &= \frac{(x-x_{i-k})P_{i,k-1}(x) - (x-x_i)P_{i-1,k-1}(x)}{x_i - x_{i-k}} \\ &\qquad k = 1,2,\ldots,i \quad i = 1,2,\ldots,n. \end{aligned}$$

Beweis: $P_{i0}(x) \equiv f_i$ für $i = 0,1,\ldots,n$ ist offensichtlich. Es sei nun

$$\bar{P}_{ik} = \frac{(x-x_{i-k})P_{i,k-1}(x) - (x-x_i)P_{i-1,k-1}(x)}{x_i - x_{i-k}}.$$

Man erkennt sofort, dass $\bar{P}_{ik} \in \Pi_k$ falls $P_{i,k-1} \in \Pi_{k-1}$ und $P_{i-1,k-1} \in \Pi_{k-1}$. Wegen

$$\begin{aligned} P_{i,k-1}(x_j) &= f_j \quad j = i-k+1, i-k+2, \ldots, i-1, i \\ P_{i-1,k-1}(x_j) &= f_j \quad j = i-k, i-k+1, \ldots, i-2, i-1 \end{aligned}$$

folgt für $j = i-k+1,\ldots,i-1$

$$\begin{aligned}\bar{P}_{ik}(x_j) &= \frac{(x_j - x_{i-k})P_{i,k-1}(x_j) - (x_j - x_i)P_{i-1,k-1}(x_j)}{x_i - x_{i-k}} \\ &= \frac{(x_j - x_{i-k})f_j - (x_j - x_i)f_j}{x_i - x_{i-k}} = \frac{(x_j - x_{i-k}) - (x_j - x_i)}{x_i - x_{i-k}} f_j = f_j.\end{aligned}$$

Weiter gilt

$$\bar{P}_{ik}(x_i) = \frac{(x_i - x_{i-k})P_{i,k-1}(x_i) - (x_i - x_i)P_{i-1,k-1}(x_i)}{x_i - x_{i-k}} = f_i$$

und

$$\bar{P}_{ik}(x_{i-k}) = \frac{(x_{i-k} - x_{i-k})P_{i,k-1}(x_{i-k}) - (x_{i-k} - x_i)P_{i-1,k-1}(x_{i-k})}{x_i - x_{i-k}} = f_{i-k}.$$

Wegen der Eindeutigkeit der Polynominterpolation folgt dann $\bar{P}_{ik} \equiv P_{ik}$. ❀

2.5. NEVILLE**-Algorithmus:**
Gegeben seien Stützstellen $x_0,\ldots,x_n$, Stützwerte $f_0,\ldots,f_n$ und eine Stelle $\bar{x}$. Zu berechnen ist der Wert des Interpolationspolynoms an der Stelle $\bar{x}$.

for $i = 1$ ***to*** n ***do***
  $P_{i0} = f_i$
  for $k = 1$ ***to*** i ***do***

$$\begin{aligned}P_{ik} &= \frac{(\bar{x} - x_{i-k})P_{i,k-1} - (\bar{x} - x_i)P_{i-1,k-1}}{x_i - x_{i-k}} \\ &= P_{i,k-1} + \frac{\bar{x} - x_i}{\bar{x} - x_{i-k}}\left[P_{i,k-1} - P_{i-1,k-1}\right] \\ &= P_{i,k-1} + \frac{P_{i,k-1} - P_{i-1,k-1}}{\dfrac{\bar{x} - x_{i-k}}{\bar{x} - x_i} - 1}\end{aligned}$$

  endfor
endfor

Damit wird im NEVILLE-Algorithmus das folgende Schema berechnet.

	$k=0$	1	2	3	4	5	6	7
$\bar{x}-x_0$	P_{00}							
		P_{11}						
$\bar{x}-x_1$	P_{10}		P_{22}					
		P_{21}		P_{33}				
$\bar{x}-x_2$	P_{20}		P_{32}		P_{44}			
		P_{31}		P_{43}		P_{55}		
$\bar{x}-x_3$	P_{30}		P_{42}		P_{54}		P_{66}	
		P_{41}		P_{53}		P_{65}		P_{77}
$\bar{x}-x_4$	P_{40}		P_{52}		P_{64}		P_{76}	
		P_{51}		P_{63}		P_{75}		
$\bar{x}-x_5$	P_{50}		P_{62}		P_{74}			
		P_{61}		P_{73}				
$\bar{x}-x_6$	P_{60}		P_{72}					
		P_{61}						
$\bar{x}-x_7$	P_{70}							

Die Hinzunahme weiterer Stützpunkte ist unproblematisch. Für jeden neuen Stützpunkt braucht nur eine weitere Schrägzeile im obigen Schema berechnet zu werden. Wie wir später sehen werden, ist es nicht sinnvoll, mit Polynomen hohen Grades zu arbeiten. Dies liegt insbesondere daran, dass bei den zu bildenden Differenzen fast gleichgroße Operanden auftreten, wodurch es zu Auslöschungen kommt und sich dabei verstärkende Fehler mit zunehmender Schematiefe das Ergebnis verfälscht. Man sollte nicht beliebig viele Spalten des Schemas berechnen. Als günstig hat sich $k = 5 \ldots 7$ erwiesen. Der arithmetische Aufwand ist gegenüber dem Berechnen mittels der baryzentrischen Darstellung höher, falls man mehrere Werte mit dem gleichen Stützstellenschema zu berechnen hat. Der arithmetische Aufwand ist andererseits gegenüber der baryzentrischen Darstellung geringer, falls zum Erreichen der gewünschten Genauigkeit nicht das gesamte Schema berechnet wird.

2.2.3. Die Newton'sche Interpolationsformel

Zum direkten Berechnen des Interpolationspolynoms oder zur Auswertung des Interpolationspolynoms an vielen Stellen ist der NEVILLE-Algorithmus nicht gut geeignet. Hier verwendet man besser die NEWTONsche Interpolationsformel. Wir machen dabei für das Interpolationspolynom den Ansatz

$$P(x) = a_0 + a_1(x-x_0) + \cdots + a_n(x-x_0)(x-x_1)\cdots(x-x_{n-1}).$$

Dieser Ansatz hat den Vorteil, dass sich der Polynomwert nach folgendem Schema berechnet:

$$P(x) = ((\ldots(a_n(x-x_{n-1})+a_{n-1})(x-x_{n-2})+a_{n-2})\ldots+a_1)(x-x_0)+a_0.$$

Die Auswertung des Polynoms P an einer bestimmten Stelle $\bar{x}$ vollzieht sich effektiv nach einem HORNERartigen Algorithmus.

2.6. Auswertung NEWTONsches Interpolationspolynoms:
Gegeben seien Stützstellen $x_0,\ldots,x_n$ und die Koeffizienten $a_0,\ldots,a_n$ des NEWTONschen Interpolationspolynoms. Zu berechnen ist der Wert des Interpolationspolynoms an der Stelle $\bar{x}$.

$P = a_n$
for $i = n-1$ ***to*** 0 ***step*** -1 ***do***
  $P = a_i + (\bar{x}-x_i)\cdot P$
endfor

Aufwand: n Multiplikationen und $2n$ Additionen.

Die Koeffizienten $a_0, a_1, \ldots, a_n$ werden nun so bestimmt, dass die Interpolationsbedingungen erfüllt sind. Wir erhalten folgendes Gleichungssystem:

$$\begin{aligned}
f_0 = P(x_0) &= a_0 \\
f_1 = P(x_1) &= a_0 + a_1(x_1-x_0) \\
f_2 = P(x_2) &= a_0 + a_1(x_2-x_0) + a_2(x_2-x_0)(x_2-x_1) \\
\vdots \quad &\quad \vdots \quad\quad \vdots \\
f_n = P(x_n) &= a_0 + a_1(x_n-x_0) + \cdots + a_n(x_n-x_0)(x_n-x_1)\cdots(x_n-x_{n-1}).
\end{aligned}$$

Aus diesem Gleichungssystem könnte man rekursiv die Koeffizienten $a_0, a_1, \ldots, a_n$ berechnen. In der Praxis verwendet man aber besser die sogenannten dividierten Differenzen, die sich numerisch stabiler berechnen lassen.
Es seien Stützstellen $x_0,\ldots,x_n$ und Stützwerte $f_0,\ldots,f_n$ gegeben. Die **k-te dividierte Differenz** $f[x_i, x_{i+1}, \ldots, x_{i+k}]$ ist rekursiv definiert durch:
Für $i = 0, 1, \ldots, n$ gilt

$$\begin{aligned}
f[x_i] &= f_i, \\
f[x_i, x_{i+1}, \ldots, x_{i+k}] &= \frac{f[x_{i+1}, x_{i+2}, \ldots, x_{i+k}] - f[x_i, x_{i+1}, \ldots, x_{i+k-1}]}{x_{i+k} - x_i}.
\end{aligned}$$

Um die dividierten Differenzen zu berechnen, wendet man ein Schema an, das dem NEVILLEschen ähnlich ist.

	$k=0$	1	2	3	4
x_0	$f_0 = f[x_0]$				
		$f[x_0,x_1]$			
x_1	$f_1 = f[x_1]$		$f[x_0,x_1,x_2]$		
		$f[x_1,x_2]$		$f[x_0,x_1,x_2,x_3]$	
x_2	$f_2 = f[x_2]$		$f[x_1,x_2,x_3]$		$f[x_0,x_1,x_2,x_3,x_4]$
		$f[x_2,x_3]$		$f[x_1,x_2,x_3,x_4]$	
x_3	$f_3 = f[x_3]$		$f[x_2,x_3,x_4]$		
		$f[x_3,x_4]$			
x_4	$f_4 = f[x_4]$				

Der Zusammenhang zwischen den dividierten Differenzen und der Interpolationsaufgabe wird in folgendem Satz beschrieben.

2.7. Satz: *Für gegebene Stützpunkte* $(x_0,f_0),\ldots,(x_n,f_n)$ *gilt*

$$\begin{aligned} P_{i+k,k}(x) &= f[x_i] + f[x_i,x_{i+1}](x-x_i) \\ &\quad + f[x_i,x_{i+1},x_{i+2}](x-x_i)(x-x_{i+1}) + \cdots \\ &\quad + f[x_i,\ldots,x_{i+k}](x-x_i)\cdots(x-x_{i+k-1}) \end{aligned}$$

und somit

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

($P_{i+k,k}$ bezeichnet dabei wie bisher das Polynom vom Grade höchstens k, das die Interpolationsbedingungen $P_{i+k,k}(x_j) = f_j$ für $j = i, i+1, \ldots, i+k$ erfüllt.)

Beweis: Wir beweisen den Satz mittels vollständiger Induktion über k. Offensichtlich gilt für $k=0$

$$P_{i,0}(x) \equiv f[x_i] = f_i, \quad i = 0,\ldots,n.$$

Damit ist der Induktionsanfang gesichert. Wir nehmen nun an, dass

$$\begin{aligned} P_{i+k,k}(x) &= f[x_i] + f[x_i,x_{i+1}](x-x_i) \\ &\quad + f[x_i,x_{i+1},x_{i+2}](x-x_i)(x-x_{i+1}) + \cdots \\ &\quad + f[x_i,\ldots,x_{i+k}](x-x_i)\cdots(x-x_{i+k-1}) \end{aligned}$$

für $i = 0, \ldots, n-k$ gilt. Das Polynom $P_{i+k+1,k+1}$ lässt sich dann in der Form

$$\begin{aligned} P_{i+k+1,k+1}(x) &= P_{i+k,k}(x) + a(x-x_i)\cdots(x-x_{i+k-1})(x-x_{i+k}) \\ &= a \cdot x^{k+1} + \cdots \end{aligned}$$

darstellen. Es ist zu zeigen, dass der Koeffizient a mit der entsprechenden $(k+1)$-ten dividierten Differenz übereinstimmt. Dazu verwenden wir die NEVILLEsche Rekursionsformel. Es gilt

$$P_{i+k+1,k+1}(x) = \frac{(x-x_i)P_{i+k+1,k}(x) - (x-x_{i+k+1})P_{i+k,k}(x)}{x_{i+k+1} - x_i}.$$

Multipliziert man in den Darstellungen der Polynome alle Terme aus, so erkennt man, dass $P_{i+k+1,k}$ und $P_{i+k,k}$ die Darstellung

$$P_{i+k+1,k}(x) = f[x_{i+1}, \ldots, x_{i+k+1}]x^k + \cdots$$

bzw.

$$P_{i+k,k}(x) = f[x_i, \ldots, x_{i+k}]x^k + \cdots$$

besitzen. Setzt man diese Darstellungen in die NEVILLEsche Rekursionsformel ein, so ergibt sich

$$P_{i+k+1,k+1}(x) = \frac{f[x_{i+1}, \ldots, x_{i+k+1}] - f[x_i, \ldots, x_{i+k}]}{x_{i+k+1} - x_i} x^{k+1} + \cdots.$$

Damit folgt

$$a = \frac{f[x_{i+1}, \ldots, x_{i+k+1}] - f[x_i, \ldots, x_{i+k}]}{x_{i+k+1} - x_i}.$$

❀

Die Koeffizienten der NEWTONschen Interpolationsformel sind daher durch die obere Schrägzeile im Schema der dividierten Differenzen gegeben. Die Auswertung der baryzentrischen Darstellung des LAGRANGEschen Interpolationspolynoms benötigt mehr arithmetische Operationen als die Auswertung des NEWTONschen Interpolationspolynoms; anderseits lässt sich die baryzentrische Darstellung mit höherer Genauigkeit auswerten, da hier ein Skalarprodukt und die Summe von $n+1$ Elementen zu bilden sind.

2.2.4. Fehler und Konvergenz der Polynominterpolation

Nimmt man an, dass die Werte f_i, $i = 0,\ldots,n$, von einer Funktion f stammen: $f_i = f(x_i)$ für $i = 0,\ldots,n$, dann stellt sich die Frage, wie gut das Interpolationspolynom mit der Funktion f übereinstimmt. Stellt man an die Funktion f keine weiteren Bedingungen, so wird diese Abweichung i. a. beliebig groß. Erfüllt die Funktion f jedoch gewisse Differenzierbarkeitsanforderungen, so lässt sich der Interpolationsfehler abschätzen. Es gilt der folgende Satz.

2.8. Satz: *Zu einer Funktion $f \in C^{n+1}[a,b]$ seien Stützpunkte $(x_0,f_0),\ldots,(x_n,f_n)$ mit $x_i \in [a,b]$, $f_i = f(x_i)$, $i = 0,\ldots,n$ gegeben; ferner sei $P \in \Pi_n$ das interpolierende Polynom. Dann existiert zu jedem $\bar{x} \in [a,b]$ ein*

$$\xi \in I = [\min\{\bar{x},x_0,\ldots,x_n\},\max\{\bar{x},x_0,\ldots,x_n\}]$$

mit

$$f(\bar{x}) - P(\bar{x}) = \frac{f^{(n+1)}(\xi)}{(n+1)!}(\bar{x}-x_0)(\bar{x}-x_1)\cdots(\bar{x}-x_n) = \frac{f^{(n+1)}(\xi)}{(n+1)!}\omega(\bar{x}).$$

Beweis: Für $\bar{x} \in \{x_0,\ldots,x_n\}$ ist die Behauptung trivial. Wir nehmen an, dass $\bar{x} \neq x_i$ für $i = 0,\ldots,n$ gilt. Die Funktion

$$F(x) = f(x) - P(x) - K\omega(x)$$

besitzt die Nullstellen $x_0,\ldots,x_n$. Wir bestimmen die Konstante K so, dass auch $\bar{x}$ Nullstelle von F ist. F hat dann $n+2$ Nullstellen im Intervall $[a,b]$. Nach dem Satz von ROLLE liegt zwischen zwei Nullstellen einer stetig differenzierbaren Funktion eine Nullstelle ihrer ersten Ableitung. Demnach hat die Funktion F' $n+1$ Nullstellen im Intervall

$$I = [\min\{\bar{x},x_0,\ldots,x_n\},\max\{\bar{x},x_0,\ldots,x_n\}].$$

Eine wiederholte Anwendung des Satzes von ROLLE liefert, dass $F^{(n+1)}$ mindestens eine Nullstelle $\xi \in I$ besitzt. Es gilt dann

$$0 = F^{(n+1)}(\xi) = f^{(n+1)}(\xi) - P^{(n+1)}(\xi) - K\omega^{(n+1)}(\xi) = f^{(n+1)}(\xi) - K(n+1)!\,.$$

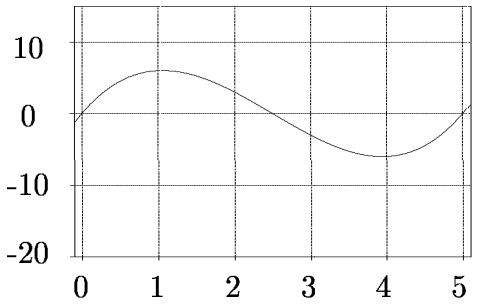
Damit folgt

$$K = \frac{f^{(n+1)}(\xi)}{(n+1)!} \text{ und } f(\bar{x}) - P(\bar{x}) = \frac{f^{(n+1)}(\xi)}{(n+1)!}\omega(\bar{x}).$$

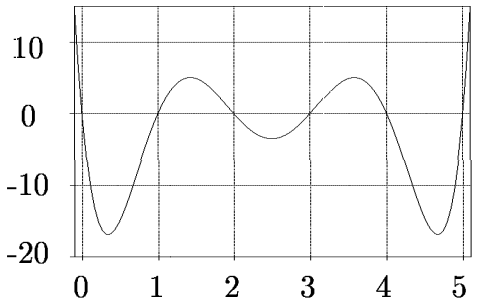
Bemerkung: Aus Satz 2.8 folgt die Abschätzung

$$|f(\bar{x}) - P(\bar{x})| \leqq M|\omega(\bar{x})|, \quad M = \max\left\{ \frac{|f^{(n+1)}(\xi)|}{(n+1)!} \;\middle|\; \xi \in [a,b] \right\}.$$

Während für eine gegebene Funktion f und ein festes Intervall $[a,b]$ die Größe M festliegt, wird man den Interpolationsfehler über $\omega(\bar{x})$ durch die Wahl der Stützstellen beeinflussen. Im Falle äquidistanter Stützstellen zeigt $\omega(x)$ etwa folgenden Verlauf.



$\omega(x) = x(x-2.5)(x-5)$

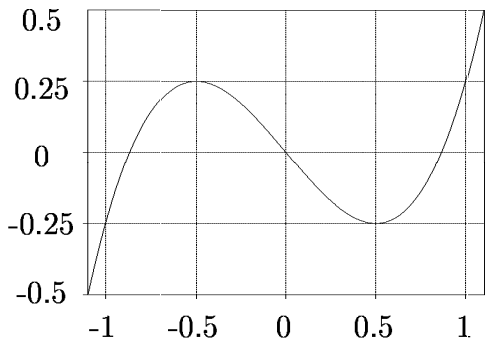


$\omega(x) = x(x-1)(x-2)(x-3)(x-4)(x-5)$

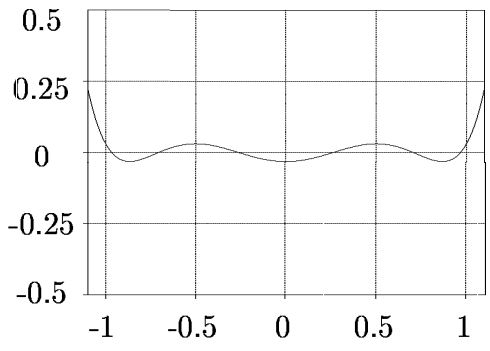
Bei äquidistanten Stützstellen erkennt man am Verlauf von ω, dass der Interpolationsfehler in den äußeren Intervallen größer wird als in den inneren. Die Extrema von ω nehmen zu den Randintervallen hin betragsmäßig zu. Außerhalb des kleinsten Intervalls, in dem alle Stützstellen liegen, wird der Interpolationsfehler i. a. beliebig groß. Durch geschickte Wahl der Stützstellen lässt sich erreichen, dass die Extrema von ω in allen Intervallen betragsmäßig gleich sind. In diesem Falle wird die Größe $\max\{\,|\omega(t)| \mid t \in [a,b]\,\}$ minimal. Bezogen auf das Intervall $[a,b] = [-1,1]$ erreicht man das durch die Wahl

$$x_i = \cos\left(\frac{2(n-i)+1}{2(n+1)}\pi\right), \quad i = 0,1,\ldots,n.$$

Das sind die Nullstellen des n-ten TSCHEBYSCHEFF-Polynoms T_n. Die Funktion ω hat dann für 3 bzw. 6 Stützstellen folgende Form.



$\omega(x) = T_2(x)$



$\omega(x) = T_5(x)$

Transformiert man sie auf ein beliebiges endliches Intervall $[a,b]$, so erhält man

$$x_i = \frac{a+b}{2} + \frac{b-a}{2}\cos\left(\frac{2(n-i)+1}{2(n+1)}\pi\right), \quad i = 0,1,\ldots,n.$$

Neben der Frage nach der Größe des Interpolationsfehlers für eine fixierte Stützstellenwahl interessiert noch die Konvergenz von Folgen von Interpolationspolynomen. Dazu nehmen wir an, wir hätten eine Funktion

$$f : [a,b] \longrightarrow \mathbb{R}$$

gegeben. Weiterhin betrachten wir eine Folge $\{S_n\}_{n\in\mathbb{N}_0}$ von Stützstellen

$$S_n = \{x_0^{(n)} < \cdots < x_n^{(n)}\}.$$

Jedem Folgeglied S_n entspricht dann in eindeutiger Weise ein Interpolationspolynom $P_n \in \Pi_n$, das den Bedingungen

$$P_n(x_i^{(n)}) = f(x_i^{(n)}) \quad i = 0,1,\ldots,n$$

genügt. Man hofft nun, dass die Folge der Interpolationspolynome zumindest punktweise (besser wäre aber gleichmäßig) gegen die Funktion f konvergiert, wenn die Glieder der Stützstellenfolge immer feiner werden, d. h.

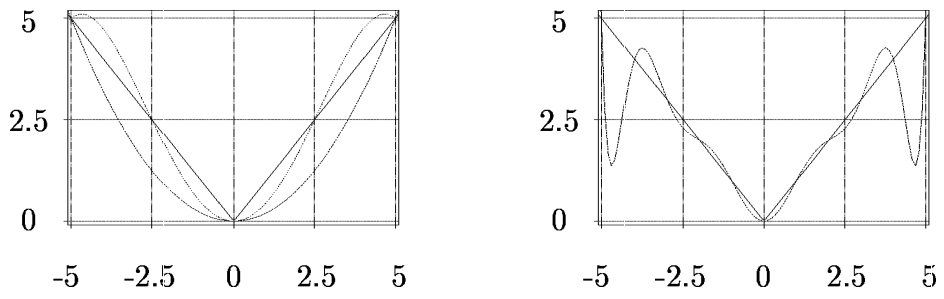
$$\lim_{n\to\infty} \max_{i=0,\ldots,n-1} \{x_{i+1}^{(n)} - x_i^{(n)}\} = 0$$

gilt. Aber schon einfache Beispiele zeigen, dass das nicht immer der Fall zu sein braucht.

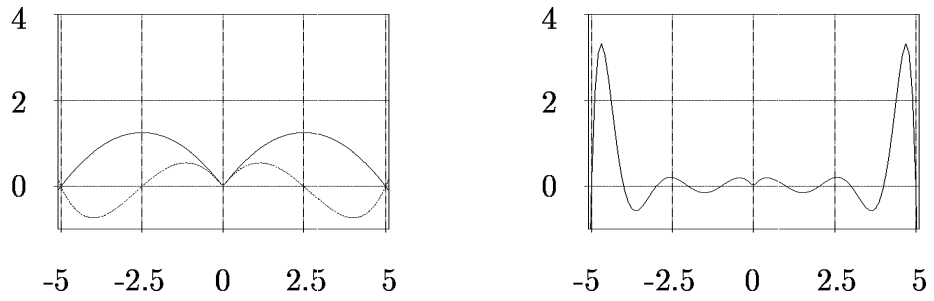
2.9. Beispiel: Wir betrachten die Funktion $f(x) = |x|$ im Intervall $[-1,1]$ mit einer Folge äquidistanter Stützstellen. Es gelte daher

$$x_i^{(n)} = -1 + \frac{2i}{n}.$$

Hier lässt sich zeigen,dass die zugehörige Folge von Interpolationspolynomen für alle x mit $0 < |x| < 1$ divergiert. Für $n = 2$ und 5 bzw. 10 sind die Interpolationspolynome in den folgenden Bildern dargestellt.



Betrachtet man nur den Interpolationsfehler, so ergeben sich folgende Bilder.



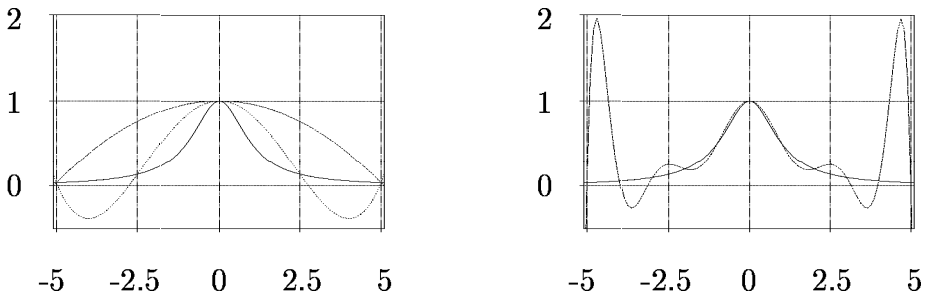
Man erkennt, dass bei den Interpolationspolynomen höheren Grades der Interpolationsfehler zum Rand des Intervalls groß wird. ♡

Nun könnte man vermuten, dass das schlechte Verhalten der Interpolationspolynome in diesem Falle damit zusammenhängt, dass die Funktion $f(x) = |x|$ im Punkt $x = 0$ nicht differenzierbar ist. Das ist aber nur bedingt richtig. Auch bei Funktionen, die beliebig oft differenzierbar sind, tritt ein ähnliches Verhalten auf.

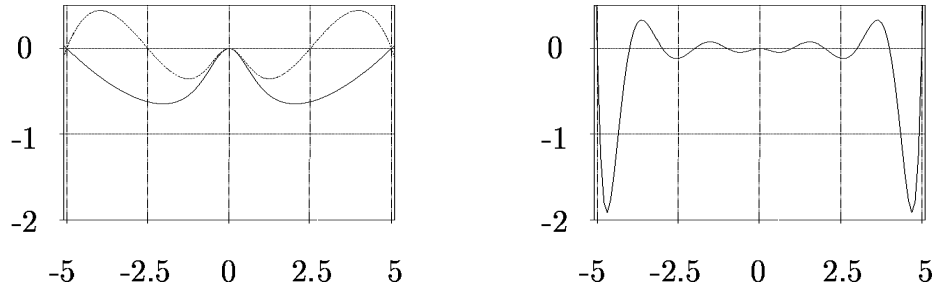
2.10. Beispiel: Wir betrachten die Funktion g mit

$$g(x) = \frac{1}{1+x^2}$$

im Intervall $[-5,5]$. In den folgenden zwei Bildern sind die Funktion und die Interpolationspolynome der Ordnung 2 und 5 bzw. 10 zu äquidistanten Stützstellen dargestellt.



Man erkennt wieder, dass besonders die Interpolationspolynome höheren Grades zum Rand des Intervalls hin stark von der zu interpolierenden Funktion abweichen. Noch deutlicher wird dies, falls wir wieder nur den Interpolationsfehler betrachten. Es ergeben sich folgende Bilder.



Dieses Beispiel stammt von C. RUNGE. Er konnte 1901 zeigen, dass die Folge der Interpolationspolynome nur für $|x| \leqq 3.63$ und $|x| = 5$ konvergiert und sonst divergiert. Der Grund für dieses Verhalten ist darin zu suchen, dass g zwar im Reellen analytisch ist, aber im Komplexen die Polstellen $x_{1,2} = \pm i$ besitzt. ♡

Ein weiteres Beispiel zeigt ein wiederum anderes Verhalten.

2.11. Beispiel: Auf dem Intervall $[0,1]$ betrachten wir die stetige Funktion h mit

$$h(x) = \begin{cases} x \sin\left(\frac{\pi}{x}\right) & \text{für} \quad x \in (0,1] \\ 0 & \text{für} \quad x = 0 \end{cases} .$$

Als Stützstellen für das Polynom $p_n \in \Pi_n$ wählen wir $x_i^{(n)} = 1/(i+1)$ für $i = 0, \ldots, n$. Dann gilt immer $h(x_i^{(n)}) = 0$. Damit ist $p_n(x) \equiv 0$ für beliebiges n. Die Folge der Interpolationspolynome konvergiert daher gleichmäßig, aber leider nicht gegen die Funktion $h(x)$. ♡

Die Beispiele zeigen, dass das Konvergenzverhalten von Interpolationspolynomfolgen unterschiedlich ist. Wir wollen ohne Beweis drei Sätze angeben, die das Verhalten von Polynomenfolgen genauer beleuchten. Dazu betrachten wir zu einer Funktion f jeweils ein Stützstellenschema

$$S: \quad \begin{array}{cccc} x_0^{(0)} & & & \\ x_0^{(1)} & x_1^{(1)} & & \\ \vdots & \vdots & \ddots & \\ x_0^{(n)} & x_1^{(n)} & \ldots & x_n^{(n)} \\ \vdots & \vdots & & \vdots \end{array}$$

und Interpolationspolynome $P_n(x)$, die durch

$$P_n \in \Pi_n, \quad P_n\left(x_i^{(n)}\right) = f\left(x_i^{(n)}\right), \quad i = 0, \ldots, n$$

definiert sind. Zuerst geben wir eine Klasse von Funktionen an, für die die Polynominterpolation problemlos ist.

2.12. Satz: *Für eine ganze, reelle Funktion f konvergiert die Folge $\{P_n\}_{n\in\mathbb{N}}$ der Interpolationspolynome bei beliebigem Stützstellenschema gleichmäßig gegen f.*

Für beliebige stetige Funktionen gilt der folgende Satz.

2.13. MARCINKIEWICZ: *Zu jeder Funktion $f \in C[a,b]$ gibt es ein Stützstellenschema S mit $x_i^{(n)} \in [a,b]$ für $n = 0,1,\ldots$ und $i = 0,1,\ldots,n$, so dass die Folge $\{P_n\}_{n\in\mathbb{N}}$ der Interpolationspolynome gleichmäßig gegen f konvergiert.*

Der Satz von MARCINKIEWICZ garantiert zwar für eine beliebige stetige Funktion die Existenz eines Stützstellenschemas, für das gleichmäßige Konvergenz eintritt, ein brauchbares Konstruktionsverfahren ist nicht bekannt. Wie der folgende Satz behauptet, gibt es kein Stützstellenschema, das für jede stetige Funktion gleichmäßig konvergente Folgen von Interpolationspolynomen liefert.

2.14. FABER: *Zu jedem Stützstellenschema S mit $x_i^{(n)} \in [a,b]$ für $n = 0,1,\ldots$ und $i = 0,1,\ldots,n$ gibt es eine Funktion $f \in C[a,b]$, so dass die Folge $\{P_n(x)\}_{n\in\mathbb{N}}$ der Interpolationspolynome nicht gleichmäßig gegen f konvergiert.*

2.3. Rationale Interpolation

2.3.1. Aufgabenstellung und grundlegende Begriffe

Gegeben seien wieder Stützpunkte $(x_0,f_0),\ldots,(x_{n+m},f_{n+m})$. Es ist eine rationale Funktion

$$\Phi^{m,n}(x) \equiv \frac{P^{m,n}(x)}{Q^{m,n}(x)} \equiv \frac{a_0 + a_1x + a_2x^2 + \cdots + a_mx^m}{b_0 + b_1x + b_2x^2 + \cdots + b_nx^n}$$

so zu bestimmen, dass die Interpolationsbedingungen

$$\Phi^{m,n}(x_i) = f_i \quad i = 0,1,\ldots,m+n \tag{2.1}$$

erfüllt sind. Da Zähler- und Nennerpolynom ohnehin durch einen konstanten Faktor kürzbar oder erweiterbar ist, reichen die $m+n+1$ Interpolationsbedingungen aus, um $\Phi^{m,n}$ zu bestimmen. Es liegt nun nahe, die Interpolationsbedingungen in ein lineares Gleichungssystem zu überführen:

$$P^{m,n}(x_i) - f_iQ^{m,n}(x_i) = 0 \quad i = 0,1,\ldots,m+n. \tag{2.2}$$

Das ist ein homogenes Gleichungssystem mit $m+n+1$ Gleichungen für $m+n+2$ Unbekannte. Es existieren daher immer nichttriviale Lösungen. Auf den ersten Blick scheint der Übergang von den eigentlichen Interpolationsbedingungen zu diesem homogenen Gleichungssystem problemlos zu sein. Leider trifft dies nicht zu.

2.15. Beispiel: Zu den Stützpunkten $(0,1),(1,2)$ und $(2,2)$ ist eine rationale Interpolationsfunktion

$$\Phi^{1,1}(x) = \frac{a_0 + a_1 x}{b_0 + b_1 x}$$

zu bestimmen. Es ergibt sich das folgende Gleichungssystem zur Bestimmung der Parameter a_0, a_1, b_0 und b_1:

$$\begin{array}{lll} a_0 & -b_0 & = 0 \\ a_0 + a_1 & -2(b_0 + b_1) & = 0 \\ a_0 + 2a_1 & -2(b_0 + 2b_1) & = 0 \end{array}$$

Als Lösung erhält man (bis auf einen gemeinsamen Faktor)

$$a_0 = b_0 = 0 \quad a_1 = 2 \quad b_1 = 1,$$

also

$$\Phi^{1,1}(x) = \frac{2x}{x}.$$

Für $x = 0$ ist das ein unbestimmter Ausdruck. Kürzt man aber durch x, so erhält man $\tilde{\Phi}^{1,1}(x) \equiv 2$. Wegen $\tilde{\Phi}^{1,1}(0) = 2 \neq 1$ löst diese Funktion aber nicht die Interpolationsaufgabe. ♡

Offensichtlich ist durch das Gleichungssystem 2.2 eine notwendige Bedingung für eine Lösung des Interpolationsproblems 2.1 gegeben. Das Beispiel zeigt, dass nicht jedes rationale Interpolationsproblem lösbar ist. Außerdem sehen wir, dass nicht jede Lösung von 2.2 einer Lösung von 2.1 entspricht. Wir wollen dieses Problem genauer untersuchen. Dazu zunächst zwei Begriffe.
Zwei rationale Funktionen Φ_1 und Φ_2 mit

$$\Phi_1(x) = \frac{P_1(x)}{Q_1(x)} \quad \Phi_2(x) = \frac{P_2(x)}{Q_2(x)}$$

heißen **gleich**, in Zeichen $\Phi_1 \equiv \Phi_2$, falls sie durch Kürzen mit einer Konstanten $a \neq 0$ auseinander hervorgehen:

$$P_1(x) = aP_2(x) \quad Q_1(x) = aQ_2(x).$$

Wir nennen Φ_1 und Φ_2 **äquivalent**, in Zeichen $\Phi_1 \sim \Phi_2$, falls sie sich durch Kürzen ineinander überführen lassen:

$$P_1 Q_2 \equiv P_2 Q_1.$$

$\tilde{\Phi}(x)$ bezeichne den rationalen Ausdruck, der durch maximales Kürzen aus $\Phi(x)$ entsteht. Offenbar ist durch diese Definition eine Äquivalenzrelation gegeben; in jeder Äquivalenzklasse befindet sich ein maximal gekürztes Element.

2.16. Satz:

(i) Das homogene Gleichungssystem 2.2 hat stets nichttriviale Lösungen

$$\Phi^{m,n}(x) \equiv P^{m,n}(x)/Q^{m,n}(x), \quad Q^{m,n}(x) \not\equiv 0.$$

(ii) Sind $\Phi_1^{m,n}(x)$ und $\Phi_2^{m,n}(x)$ Lösungen von 2.2, so gilt $\Phi_1^{m,n}(x) \sim \Phi_2^{m,n}(x)$.

Beweis: (i) 2.2 besitzt als homogenes Gleichungssystem mit $m+n+1$ Gleichungen für $m+n+2$ Unbekannte stets nichttriviale Lösungen

$$(a_0, a_1, \ldots, a_m, b_0, b_1, \ldots, b_n) \neq (0, 0, \ldots, 0).$$

Gilt nun für eine Lösung

$$Q^{m,n}(x) \equiv b_0 + b_1 x + \cdots + b_n x^n \equiv 0,$$

so folgt für das Zählerpolynom

$$P^{m,n}(x_i) = 0, \quad i = 0, 1, \ldots, m+n.$$

Damit hat $P^{m,n}(x)$ $m+n+1 \geqq m+1$ Nullstellen. Da aber der Grad von $P^{m,n}(x)$ höchstens m ist, folgt $P^{m,n}(x) \equiv 0$ und damit der Widerspruch

$$(a_0, a_1, \ldots, a_m, b_0, b_1, \ldots, b_n) = (0, 0, \ldots, 0).$$

Die Annahme $Q^{m,n}(x) \equiv 0$ muss daher falsch sein.
(ii) Es seien

$$\Phi_1^{m,n}(x) = \frac{P_1^{m,n}(x)}{Q_1^{m,n}(x)}, \quad \Phi_2^{m,n}(x) = \frac{P_2^{m,n}(x)}{Q_2^{m,n}(x)}$$

zwei Lösungen von 2.2. Für das Polynom

$$R(x) = P_1^{m,n}(x) Q_2^{m,n}(x) - P_2^{m,n}(x) Q_1^{m,n}(x)$$

folgt dann für $i = 0, 1, \ldots, n$

$$\begin{aligned} R(x_i) &= P_1^{m,n}(x_i)Q_2^{m,n}(x_i) - P_2^{m,n}(x_i)Q_1^{m,n}(x_i) \\ &= f_i Q_1^{m,n}(x_i)Q_2^{m,n}(x_i) - f_i Q_2^{m,n}(x_i)Q_1^{m,n}(x_i) \\ &= 0. \end{aligned}$$

Da aber der Grad des Polynoms $R(x)$ nicht größer als $m+n$ ist, folgt sofort $R(x) \equiv 0$ und damit $\Phi_1^{m,n}(x) \sim \Phi_2^{m,n}(x)$. ❀

Ist $\Phi^{m,n}(x) \equiv P^{m,n}(x)/Q^{m,n}(x)$ Lösung von 2.2, so treten für ein $i \in \{0, 1, \ldots, m+n\}$ zwei Fälle ein.

1. Es gilt $Q^{m,n}(x_i) \neq 0$. Dann folgt sofort $\Phi^{m,n}(x_i) = f_i$; die entsprechende Interpolationsbedingung ist daher erfüllt.

2. Es gilt $Q^{m,n}(x_i) = 0$. Dann gilt auch $P^{m,n}(x_i) = 0$, und $\Phi^{m,n}(x)$ ist durch $x - x_i$ kürzbar. In diesem Falle ist $\Phi^{m,n}(x)$ für eine beliebige Wahl von f_i Lösung von 2.2. Ist speziell $f_i \neq \tilde{\Phi}^{m,n}(x_i)$, so besitzt 2.2 immer noch die Lösung $\Phi^{m,n}(x)$, diese ist aber nicht mehr Lösung von 2.1. Man nennt (x_i, f_i) dann einen **unerreichbaren Punkt**.

Falls das Gleichungssystem 2.2 höchstens den Rang $m+n+1$ hat, gilt das folgende Kriterium für das Auftreten unerreichbarer Punkte.

2.17. Satz: *Der Rang der Koeffizientenmatrix des Gleichungssystems 2.2 sei gleich $m+n+1$ und $\Phi^{m,n}(x)$ sei eine Lösung dieses Systems. Dann gilt:*

(i) Ist $\Phi^{m,n}(x) \equiv \tilde{\Phi}^{m,n}(x)$, so treten keine unerreichbaren Punkte auf und $\Phi^{m,n}(x)$ ist Lösung von 2.1.

(ii) Ist $\Phi^{m,n}(x) \not\equiv \tilde{\Phi}^{m,n}(x)$, so ist $\tilde{\Phi}^{m,n}(x)$ weder eine Lösung von 2.2 noch von 2.1. Es treten unerreichbare Punkte auf.

Beweis: (i) ist offensichtlich.
(ii) Da der Rang der Koeffizientenmatrix des Gleichungssystems gleich $m+n+1$ ist, ist die Dimension des Lösungsraums gleich 1. Damit sind alle Lösungen $\Phi^{m,n}(x)$ von 2.2 im Sinne obiger Definition gleich. Gilt aber

$$\tilde{\Phi}^{m,n}(x) \not\equiv \Phi^{m,n}(x),$$

so ist $\tilde{\Phi}^{m,n}(x)$ nicht Lösung von 2.2 und damit auch nicht Lösung von 2.1 sein. Es treten unerreichbare Punkte auf. ❀

2.3.2. Der Stoer-Algorithmus

Wir wollen in diesem Abschnitt einen NEVILLEartigen Algorithmus zur Auswertung einer rationalen Interpolationsfunktion an einer Stelle $\bar{x}$ angeben. Dazu setzen wir voraus, dass beim Lösen der Aufgabe keine Ausartungsfälle vorliegen, d. h. keine unerreichbaren Punkte auftreten. Analog zur Herleitung der NEVILLEschen Rekursionsformel bezeichnen wir mit

$$\Phi_s^{k,l}(x) \equiv \frac{P_s^{k,l}(x)}{Q_s^{k,l}(x)}$$

den rationalen Ausdruck mit

$$P_s^{k,l}(x) \in \Pi_k, \quad Q_s^{k,l}(x) \in \Pi_l \quad \Phi_s^{k,l}(x_i) = f_i \quad i = s, s+1, \ldots, s+k+l.$$

Durch $p_s^{k,l}$ und $q_s^{k,l}$ seien die Höchstkoeffizienten des Zähler- beziehungsweise Nennerpolynoms gegeben. Es gelte daher

$$\begin{aligned} P_s^{k,l}(x) &= p_s^{k,l} x^k + \cdots \\ Q_s^{k,l}(x) &= q_s^{k,l} x^l + \cdots \end{aligned}$$

Weiterhin definieren wir

$$\alpha_i = x - x_i \text{ und } T_s^{k,l}(x,y) = P_s^{k,l}(x) - yQ_s^{k,l}(x).$$

Dann gilt

$$T_s^{k,l}(x_i, f_i) = 0 \quad i = s, s+1, \ldots, s+k+l.$$

2.18. Satz: *Mit den Startwerten*

$$P_s^{0,0}(x) = f_s, \quad Q_s^{0,0}(x) = 1$$

gelten die folgenden Rekursionsformeln

- *Übergang* $(k-1,l) \longrightarrow (k,l)$ *für* $k \geqq 1$*:*

$$\begin{aligned} P_s^{k,l}(x) &= \alpha_s q_s^{k-1,l} P_{s+1}^{k-1,l}(x) - \alpha_{s+k+l} q_{s+1}^{k-1,l} P_s^{k-1,l}(x), && (2.3) \\ Q_s^{k,l}(x) &= \alpha_s q_s^{k-1,l} Q_{s+1}^{k-1,l}(x) - \alpha_{s+k+l} q_{s+1}^{k-1,l} Q_s^{k-1,l}(x). && (2.4) \end{aligned}$$

- *Übergang* $(k,l-1) \longrightarrow (k,l)$ *für* $l \geqq 1$*:*

$$P_s^{k,l}(x) = \alpha_s p_s^{k,l-1} P_{s+1}^{k,l-1}(x) - \alpha_{s+k+l} p_{s+1}^{k,l-1} P_s^{k,l-1}(x), \tag{2.5}$$

$$Q_s^{k,l}(x) = \alpha_s p_s^{k,l-1} Q_{s+1}^{k,l-1}(x) - \alpha_{s+k+l} p_{s+1}^{k,l-1} Q_s^{k,l-1}(x). \tag{2.6}$$

Beweis: Wir beweisen die Formeln für den Übergang $(k-1,l) \longrightarrow (k,l)$. Dazu nehmen wir an, dass $\Phi_s^{k-1,l}(x)$ und $\Phi_{s+1}^{k-1,l}(x)$ die entsprechenden Interpolationsbedingungen erfüllen. Es gelte daher

$$\begin{aligned} P_s^{k-1,l}(x), P_{s+1}^{k-1,l}(x) &\in \Pi_{k-1}, \\ Q_s^{k-1,l}(x), Q_{s+1}^{k-1,l}(x) &\in \Pi_l, \\ T_s^{k-1,l}(x_i,f_i) &= 0, \quad i = s, s+1, \ldots, s+k+l-1, \\ T_{s+1}^{k-1,l}(x_i,f_i) &= 0, \quad i = s+1, s+2, \ldots, s+k+l. \end{aligned}$$

Aus der ersten Rekursionsformel folgt sofort $P_s^{k,l}(x) \in \Pi_k$. Für die zweite Rekursionsformel überlegt man sich leicht, dass der Koeffizient von x^{l+1} gleich Null ist. Damit gilt $Q_s^{k,l}(x) \in \Pi_l$. Aus den Rekursionsformeln für die Zähler- und Nennerpolynome erhält man

$$T_s^{k,l}(x,y) = \alpha_s q_s^{k-1,l} T_{s+1}^{k-1,l}(x,y) - \alpha_{s+k+l} q_{s+1}^{k-1,l} T_s^{k-1,l}(x,y).$$

Daraus folgt

$$T_s^{k,l}(x_i,f_i) = 0, \quad i = s, s+1, \ldots, s+k+l.$$

Falls keine unerreichbaren Punkte auftreten, erhalten wir tatsächlich nach 2.3 und 2.4 Zähler- und Nennerpolynom von $\Phi_s^{k,l}(x)$. ❀

Will man die Rekursionsformeln aus Satz 2.18 anwenden, um wie beim NEVILLE-Algorithmus einzelne Werte von rationalen Interpolationsfunktionen rekursiv zu berechnen, so "'stören"' die unbekannten Koeffizienten $q_s^{k-1,l}, q_{s+1}^{k-1,l}, p_s^{k,l-1}$ und $p_{s+1}^{k,l-1}$. Mit dem folgenden Satzes sind sie aber aus den Rekursionsformeln eliminierbar.

2.19. Satz: *Für* $k,l \geqq 1$ *gilt*

$$\Phi_s^{k-1,l}(x) - \Phi_{s+1}^{k-1,l-1}(x) = -p_{s+1}^{k-1,l-1} q_s^{k-1,l} \frac{(x-x_{s+1}) \cdots (x-x_{s+k+l-1})}{Q_s^{k-1,l}(x) Q_{s+1}^{k-1,l-1}(x)}$$

und

$$\Phi_{s+1}^{k-1,l}(x) - \Phi_{s+1}^{k-1,l-1}(x) = -p_{s+1}^{k-1,l-1} q_{s+1}^{k-1,l} \frac{(x-x_{s+1}) \cdots (x-x_{s+k+l-1})}{Q_{s+1}^{k-1,l}(x) Q_{s+1}^{k-1,l-1}(x)}.$$

Beweis: Wir zeigen nur die erste Identität. Die Gültigkeit der zweiten Gleichung lässt sich analog beweisen.
Es sei

$$\begin{aligned} Z(x) &= P_s^{k-1,l}(x)Q_{s+1}^{k-1,l-1}(x) - P_{s+1}^{k-1,l-1}(x)Q_s^{k-1,l}(x) \\ &= \left[\Phi_s^{k-1,l}(x) - \Phi_{s+1}^{k-1,l-1}(x)\right] Q_s^{k-1,l}(x)Q_{s+1}^{k-1,l-1}(x). \end{aligned}$$

Dann ist $Z(x) \in \Pi_{k+l-1}$ und der Koeffizient von x^{k+l-1} ist $-p_{s+1}^{k-1,l-1}q_s^{k-1,l}$.
Für $i = s+1,\ldots,s+k+l-1$ gilt weiterhin

$$\begin{aligned} Z(x_i) &= \left[\Phi_s^{k-1,l}(x_i) - \Phi_{s+1}^{k-1,l-1}(x_i)\right] Q_s^{k-1,l}(x_i)Q_{s+1}^{k-1,l-1}(x_i) \\ &= [f_i - f_i]Q_s^{k-1,l}(x_i)Q_{s+1}^{k-1,l-1}(x_i) = 0. \end{aligned}$$

Daraus folgt

$$Z(x) = -p_{s+1}^{k-1,l-1}q_s^{k-1,l}(x-x_{s+1})\cdots(x-x_{s+k+l-1}),$$

daher

$$\begin{aligned} &\left[\Phi_s^{k-1,l}(x) - \Phi_{s+1}^{k-1,l-1}(x)\right] Q_s^{k-1,l}(x)Q_{s+1}^{k-1,l-1}(x) \\ &= -p_{s+1}^{k-1,l-1}q_s^{k-1,l}(x-x_{s+1})\cdots(x-x_{s+k+l-1}) \end{aligned}$$

und zuletzt

$$\Phi_s^{k-1,l}(x) - \Phi_{s+1}^{k-1,l-1}(x) = -p_{s+1}^{k-1,l-1}q_s^{k-1,l}\frac{(x-x_{s+1})\cdots(x-x_{s+k+l-1})}{Q_s^{k-1,l}(x)Q_{s+1}^{k-1,l-1}(x)}.$$

❀

Nun beseitigen wir die störenden Koeffizienten aus den Rekursionsformeln. Für den Übergang $(k-1,l) \longrightarrow (k,l)$ gilt nach Satz 2.18:

$$\Phi_s^{k,l}(x) = \frac{P_s^{k,l}(x)}{Q_s^{k,l}(x)} = \frac{\alpha_s q_s^{k-1,l}P_{s+1}^{k-1,l}(x) - \alpha_{s+k+l}q_{s+1}^{k-1,l}P_s^{k-1,l}(x)}{\alpha_s q_s^{k-1,l}Q_{s+1}^{k-1,l}(x) - \alpha_{s+k+l}q_{s+1}^{k-1,l}Q_s^{k-1,l}(x)}.$$

Aus Satz 2.19 erhält man

$$\frac{q_s^{k-1,l}}{q_{s+1}^{k-1,l}} = \frac{\Phi_s^{k-1,l}(x) - \Phi_{s+1}^{k-1,l-1}(x)}{\Phi_{s+1}^{k-1,l}(x) - \Phi_{s+1}^{k-1,l-1}(x)}\frac{Q_s^{k-1,l}(x)}{Q_{s+1}^{k-1,l}(x)}.$$

Setzt man die letzte Gleichung in die vorletzte ein, so ergibt sich

$$\begin{aligned}
\Phi_s^{k,l}(x) &= \frac{\alpha_s \frac{\Phi_s^{k-1,l}(x)-\Phi_{s+1}^{k-1,l-1}(x)}{\Phi_{s+1}^{k-1,l}(x)-\Phi_{s+1}^{k-1,l-1}(x)} \cdot \frac{Q_s^{k-1,l}(x)}{Q_{s+1}^{k-1,l}(x)} P_{s+1}^{k-1,l}(x) - \alpha_{s+k+l} P_s^{k-1,l}(x)}{\alpha_s \frac{\Phi_s^{k-1,l}(x)-\Phi_{s+1}^{k-1,l-1}(x)}{\Phi_{s+1}^{k-1,l}(x)-\Phi_{s+1}^{k-1,l-1}(x)} \cdot \frac{Q_s^{k-1,l}(x)}{Q_{s+1}^{k-1,l}(x)} Q_{s+1}^{k-1,l}(x) - \alpha_{s+k+l} Q_s^{k-1,l}(x)} \\
&= \frac{\alpha_s \frac{\Phi_s^{k-1,l}(x)-\Phi_{s+1}^{k-1,l-1}(x)}{\Phi_{s+1}^{k-1,l}(x)-\Phi_{s+1}^{k-1,l-1}(x)} Q_s^{k-1,l}(x)\ \Phi_{s+1}^{k-1,l}(x) - \alpha_{s+k+l} P_s^{k-1,l}(x)}{\alpha_s \frac{\Phi_s^{k-1,l}(x)-\Phi_{s+1}^{k-1,l-1}(x)}{\Phi_{s+1}^{k-1,l}(x)-\Phi_{s+1}^{k-1,l-1}(x)} Q_s^{k-1,l}(x) - \alpha_{s+k+l} Q_s^{k-1,l}(x)} \\
&= \frac{\alpha_s \frac{\Phi_s^{k-1,l}(x)-\Phi_{s+1}^{k-1,l-1}(x)}{\Phi_{s+1}^{k-1,l}(x)-\Phi_{s+1}^{k-1,l-1}(x)} \Phi_{s+1}^{k-1,l}(x) - \alpha_{s+k+l} \Phi_s^{k-1,l}(x)}{\alpha_s \frac{\Phi_s^{k-1,l}(x)-\Phi_{s+1}^{k-1,l-1}(x)}{\Phi_{s+1}^{k-1,l}(x)-\Phi_{s+1}^{k-1,l-1}(x)} - \alpha_{s+k+l}} \\
&= \Phi_{s+1}^{k-1,l}(x) + \frac{\Phi_{s+1}^{k-1,l}(x) - \Phi_s^{k-1,l}(x)}{\frac{\alpha_s}{\alpha_{s+k+l}} \cdot \frac{\Phi_s^{k-1,l}(x)-\Phi_{s+1}^{k-1,l-1}(x)}{\Phi_{s+1}^{k-1,l}(x)-\Phi_{s+1}^{k-1,l-1}(x)} - 1}
\end{aligned}$$

Für den Übergang $(k,l-1) \longrightarrow (k,l)$ gilt eine entsprechende Rekursionsformel. Damit haben wir den folgenden Satz bewiesen.

2.20. Satz: *Für $k,l \geqq 1$ gelten die folgenden Rekursionsformeln:*

$$\Phi_s^{k,l}(x) = \Phi_{s+1}^{k-1,l}(x) + \frac{\Phi_{s+1}^{k-1,l}(x) - \Phi_s^{k-1,l}(x)}{\frac{x-x_s}{x-x_{s+k+l}} \left[1 - \frac{\Phi_{s+1}^{k-1,l}(x)-\Phi_s^{k-1,l}(x)}{\Phi_{s+1}^{k-1,l}(x)-\Phi_{s+1}^{k-1,l-1}(x)} \right] - 1}$$

und

$$\Phi_s^{k,l}(x) = \Phi_{s+1}^{k,l-1}(x) + \frac{\Phi_{s+1}^{k,l-1}(x) - \Phi_s^{k,l-1}(x)}{\frac{x-x_s}{x-x_{s+k+l}} \left[1 - \frac{\Phi_{s+1}^{k,l-1}(x)-\Phi_s^{k,l-1}(x)}{\Phi_{s+1}^{k,l-1}(x)-\Phi_{s+1}^{k-1,l-1}(x)} \right] - 1}.$$

Es bleibt noch zu klären, welche Rekursionsformeln im Falle $k=0$ oder $l=0$ anzuwenden ist. Der Fall $l=0$ entspricht der reinen Polynominterpolation. Wir wenden die NEVILLEsche Rekursionsformel an. Es gilt

$$\begin{aligned}
\Phi_s^{0,0}(x) &= f_s, \\
\Phi_s^{k,0}(x) &= \Phi_{s+1}^{k-1,0}(x) + \frac{\Phi_{s+1}^{k-1,0}(x) - \Phi_s^{k-1,0}(x)}{\frac{x-x_s}{x-x_{s+k}} - 1}.
\end{aligned}$$

Diese Formel ist in der ersten Rekursionsformel aus Satz 2.20 enthalten, falls man die eckige Klammer im Nenner gleich 1 setzt. Das entspricht auch

$$\Phi_s^{k,-1}(x) = \infty.$$

Der Fall $k = 0$ lässt sich auf Polynominterpolation mit den Stützpunkten $(x_i, 1/f_i)$ zurückführen. Man erhält dann ebenfalls mit der NEVILLEschen Rekursionsformel

$$\begin{aligned}
\frac{1}{\Phi_s^{0,0}(x)} &= \frac{1}{f_s}, \\
\frac{1}{\Phi_s^{0,l}(x)} &= \frac{1}{\Phi_{s+1}^{0,l-1}(x)} + \frac{\frac{1}{\Phi_{s+1}^{0,l-1}(x)} - \frac{1}{\Phi_s^{0,l-1}(x)}}{\frac{x-x_s}{x-x_{s+l}} - 1}.
\end{aligned}$$

Daraus ergibt sich

$$\begin{aligned}
\Phi_s^{0,0}(x) &= f_s, \\
\Phi_s^{0,l}(x) &= \Phi_{s+1}^{0,l-1}(x) + \frac{\Phi_{s+1}^{0,l-1}(x) - \Phi_s^{0,l-1}(x)}{\frac{x-x_s}{x-x_{s+l}} \cdot \frac{\Phi_s^{0,l-1}(x)}{\Phi_{s+1}^{0,l-1}(x)} - 1}.
\end{aligned}$$

Diese Formel entspricht der zweiten Rekursionsformel aus Satz 2.20, falls man dort $\Phi_s^{-1,l}(x) = 0$ setzt.
Somit lassen sich mit den Rekursionsformeln aus Satz 2.20 beliebige rationale Ausdrücke $\Phi^{k,l}(x)$ mit $\Phi^{k,l}(x_i) = f_i$ berechnen, falls man

$$\Phi_s^{0,0}(x) = f_s, \quad s = 0, 1, \ldots, k+l,$$

$$\Phi_s^{-1,l}(x) = \infty$$

und

$$\Phi_s^{k,-1}(x) = 0$$

setzt. Der gesuchte rationale Ausdruck ist dann

$$\Phi^{k,l}(x) = \Phi_0^{k,l}(x).$$

Solange keine Ausartungen (d.h. keine unerreichbaren Punkte) auftreten, ist dies entlang eines Zick-Zack-Weges in der (k,l)-Ebene möglich. Es wird also abwechselnd der Zähler- und der Nennergrad des rationalen Ausdrucks erhöht. Bewährt hat sich die Folge

$$(k,l) : (0,0) \longrightarrow (0,1) \longrightarrow (1,1) \longrightarrow (1,2) \longrightarrow (2,2) \longrightarrow (2,3) \longrightarrow \cdots$$

Bei dieser Sequenz genügt die Angabe von $k+l$ statt k und l. Setzt man $i=s+k+l$ und $j=k+l$, so gilt

$$s=i-j, \quad k=[j/2], \quad l=[(j+1)/2].$$

Dabei bezeichnet $[r]$ den ganzen Teil der reellen Zahl r, d. h. die größte ganze Zahl, die nicht größer als r ist. Für die Größen

$$T_{ij}=\Phi_s^{k,l}(x)=\Phi_{i-j}^{[j/2],[(j+1)/2]}(x)$$

ergibt sich dann folgende einheitliche Rekursionsformel:

$$T_{i0}=f_i, \quad T_{i,-1}=0 \quad i=0,1,2\ldots,m+n$$

und für $j=1,2,\ldots,i,\ i=0,1,2\ldots,m+n$

$$T_{ij}=T_{i,j-1}+\frac{T_{i,j-1}-T_{i-1,j-1}}{\frac{x-x_{i-j}}{x-x_i}\left[1-\frac{T_{i,j-1}-T_{i-1,j-1}}{T_{i,j-1}-T_{i-1,j-2}}\right]-1}.$$

Diese Rekursionsformel unterscheidet sich von der im NEVILLE-Algorithmus nur durch die eckige Klammer im Nenner. Wir erhalten so den folgenden Algorithmus.

2.21. STOER**-Algorithmus:**
Gegeben seien die Stützpunkte $(x_0,f_0),\ldots,(x_n,f_n)$.
Zu berechnen ist der Wert einer interpolierenden rationalen Funktion an der Stelle $\bar{x}$.

for $i=1$ ***to*** n ***do***
  $T_{i0}=f_i$
endfor
for $k=1$ ***to*** i ***do***

$$T_{ik}=T_{i,k-1}+\frac{T_{i,k-1}-T_{i-1,k-1}}{\frac{\bar{x}-x_{i-k}}{\bar{x}-x_i}\left[1-\frac{T_{i,k-1}-T_{i-1,k-1}}{T_{i,k-1}-T_{i-1,k-2}}\right]-1}$$

endfor

Das Berechnen erfolgt wieder in einem Schema ähnlich zum NEVILLE-Schema.

	$j=0$	1	2	3	4	5
	$f_0 = T_{00}$					
$0 = T_{0,-1}$		T_{11}				
	$f_1 = T_{10}$		T_{22}			
$0 = T_{1,-1}$		T_{21}		T_{33}		
	$f_2 = T_{20}$		T_{32}		T_{44}	
$0 = T_{2,-1}$		T_{31}		T_{43}		T_{55}
	$f_3 = T_{30}$		T_{42}		T_{54}	
$0 = T_{3,-1}$		T_{41}		T_{53}		
	$f_4 = T_{40}$		T_{52}			
$0 = T_{4,-1}$		T_{51}				
	$f_5 = T_{50}$					

Dieser Algorithmus ist wie der NEVILLE-Algorithmus bei der Polynominterpolation besonders gut geeignet, falls man die rationale Interpolationsfunktion nur an einer Stelle auswerten möchte. Die Hinzunahme weiterer Stützstellen ist unproblematisch. Es ist für jede Stützstelle nur eine zusätzliche Schrägzeile an das Schema anzufügen.

2.3.3. Der Thiele'sche Kettenbruch

Will man die rationale Interpolationsfunktion selbst berechnen, oder sie an vielen Stellen auswerten, so ist der Algorithmus von STOER nicht gut geeignet. Hier geht man ähnlich wie beim NEWTONschen Ansatz für die Polynominterpolation vor. Der Ansatz erfolgt in Form eines Kettenbruchs. Wir beschränken uns jetzt auf rationale Ausdrücke $\Phi^{k,l}$ mit $k = l$ oder $k = l+1$. Betrachten wir zuerst den Fall

$$\Phi^{n,n}(x) = \frac{P^n(x)}{Q^n(x)}.$$

Die Interpolationsbedingungen lauten

$$\Phi^{n,n}(x_i) = f_i, \quad i = 0,1,2,\ldots,2n.$$

Es folgt

$$\Phi^{n,n}(x) = \frac{P^n(x)}{Q^n(x)} = f_0 + \frac{P^n(x)}{Q^n(x)} - \frac{P^n(x_0)}{Q^n(x_0)} = f_0 + \frac{P^n(x) - \frac{P^n(x_0)}{Q^n(x_0)} Q^n(x)}{Q^n(x)}$$

Für das Zählerpolynom folgt

$$\begin{aligned} Z^n(x) &= P^n(x) - \frac{P^n(x_0)}{Q^n(x_0)} Q^n(x) \in \Pi_n \\ Z^n(x_0) &= P^n(x_0) - \frac{P^n(x_0)}{Q^n(x_0)} Q^n(x_0) = 0 \end{aligned}$$

Damit gilt

$$Z^n(x) = (x - x_0) P^{n-1}(x), \quad P^{n-1} \in \Pi_{n-1}.$$

Die rationale Interpolationsfunktion lässt sich daher in der Form

$$\Phi^{n,n}(x) = f_0 + (x - x_0) \frac{P^{n-1}(x)}{Q^n(x)} = f_0 + \frac{x - x_0}{\frac{Q^n(x)}{P^{n-1}(x)}}$$

darstellen. Für den rationalen Ausdruck im Nenner folgt aus den Interpolationsbedingungen

$$\frac{Q^n(x_i)}{P^{n-1}(x_i)} = \frac{x_i - x_0}{f_i - f_0} = \varphi(x_0, x_i) \quad i = 1, 2, \ldots, 2n.$$

Damit lässt sich $Q^n(x)/P^{n-1}(x)$ wieder zerlegen. Es folgt

$$\begin{aligned} \frac{Q^n(x)}{P^{n-1}(x)} &= \varphi(x_0, x_1) + \frac{Q^n(x)}{P^{n-1}(x)} - \frac{Q^n(x_1)}{P^{n-1}(x_1)} \\ &= \varphi(x_0, x_1) + \frac{Q^n(x) - \frac{Q^n(x_1)}{P^{n-1}(x_1)} P^{n-1}(x)}{P^{n-1}(x)}. \end{aligned}$$

Für das Zählerpolynom erhalten wir

$$Z^{n-1}(x) = (x - x_1) Q^{n-1}(x), \quad Q^{n-1} \in \Pi_{n-1}$$

und es gilt

$$\frac{Q^n(x)}{P^{n-1}(x)} = \varphi(x_0, x_1) + \frac{x - x_1}{\frac{P^{n-1}(x)}{Q^{n-1}(x)}}.$$

Der rationale Ausdruck im Nenner erfüllt die Bedingungen

$$\frac{P^{n-1}(x_i)}{Q^{n-1}(x_i)} = \frac{x_i - x_1}{\varphi(x_0, x_i) - \varphi(x_0, x_1)} = \varphi(x_0, x_1, x_i), \quad i = 2, 3, \ldots, 2n.$$

Setzt man diesen Prozess fort, so erhält man

$$\begin{aligned}\Phi^{n,n}(x) &= f_0 + \frac{x-x_0}{\frac{Q^n(x)}{P^{n-1}(x)}} \\ &= f_0 + \frac{x-x_0}{\varphi(x_0,x_1) + \frac{x-x_1}{\frac{P^{n-1}(x)}{Q^{n-1}(x)}}} \\ &= f_0 + \frac{x-x_0}{\varphi(x_0,x_1) + \dfrac{x-x_1}{\varphi(x_0,x_1,x_2) + \dfrac{x-x_2}{\varphi(x_0,x_1,x_2,x_3) + \cdots \dfrac{x-x_{2n-1}}{\varphi(x_0,x_1,x_2,\ldots,x_{2n})}}}}.\end{aligned}$$

Damit haben wir eine Kettenbruchdarstellung für eine rationale Interpolationsfunktion gefunden. Abkürzend verwendet man für den Kettenbruch die Schreibweise

$$\begin{aligned}\Phi^{n,n}(x) =& f_0 + \underline{x-x_0} \Big/ \varphi(x_0,x_1) + \underline{x-x_1} \Big/ \varphi(x_0,x_1,x_2) + \cdots \\ & \cdots + \underline{x-x_{2n-1}} \Big/ \varphi(x_0,x_1,x_2,\ldots,x_{2n}).\end{aligned}$$

Die in der Darstellung auftretenden Koeffizienten $\varphi(x_{i_1},\ldots,x_{i_k})$ heißen **inverse Differenzen**. Sie werden rekursiv definiert:

$$\varphi(x_i,x_j) = \frac{x_i - x_j}{f_i - f_j}, \quad i,j = 0,1,\ldots,2n, \quad i \neq j$$

und

$$\varphi(x_i,\ldots,x_l,x_m,x_n) = \frac{x_m - x_n}{\varphi(x_i,\ldots,x_l,x_m) - \varphi(x_i,\ldots,x_l,x_n)}.$$

Ähnlich wie bei den dividierten Differenzen verwendet man wieder ein Schema zum Berechnen der inversen Differenzen:

	$k=0$	1	2	3	4
x_0	f_0				
x_1	f_1	$\varphi(x_0,x_1)$			
x_2	f_2	$\varphi(x_1,x_2)$	$\varphi(x_0,x_1,x_2)$		
x_3	f_3	$\varphi(x_2,x_3)$	$\varphi(x_1,x_2,x_3)$	$\varphi(x_0,x_1,x_2,x_3)$	
x_4	f_4	$\varphi(x_3,x_4)$	$\varphi(x_2,x_3,x_4)$	$\varphi(x_1,x_2,x_3,x_4)$	$\varphi(x_0,x_1,x_2,x_3,x_4)$

Betrachtet man die Teilbrüche des Kettenbruchs, so erkennt man, dass sie ebenfalls rationale Interpolationsfunktionen darstellen. Es gilt

$$\begin{aligned} f_0 &= \Phi_0^{0,0}(x) \\ f_0 + \underline{x - x_0 / \varphi(x_0, x_1)} &= \Phi_0^{1,0}(x) \\ f_0 + \underline{x - x_0 / \varphi(x_0, x_1)} + \underline{x - x_1 / \varphi(x_0, x_1, x_2)} &= \Phi_0^{1,1}(x) \\ \vdots &\quad \vdots \end{aligned}$$

Der Kettenbruch lässt sich wieder ähnlich wie die NEWTONsche Interpolationsformel auswerten.

2.22. Auswertung eines Kettenbruchs:
Gegeben seien die Stützpunkte $(x_0, f_0), \ldots, (x_n, f_n)$.
Zu berechnen ist der Wert des Kettenbruchs

$$\Phi(x) = a_0 + \underline{x - x_0 / a_1} + \underline{x - x_1 / a_2} + \cdots + \underline{x - x_{n-1} / a_n}$$

an der Stelle $\bar{x}$.

$\Phi = a_n$
for $i = n-1$ ***to*** 0 ***step*** -1 ***do***
  $\Phi = a_i + (\bar{x} - x_i)/\Phi$
endfor

Aufwand: n *Divisionen und* $2n$ *Additionen.*

Da die inversen Differenzen keine symmetrischen Funktionen ihrer Argumente sind, verwendet man besser die **reziproken Differenzen**. Diese sind folgendendermaßen definiert:

$$\begin{aligned} \rho(-) &= 0, \\ \rho(x_i) &= f_i, \quad i = 0, 1, 2, \ldots, \\ \rho(x_i, \ldots, x_{i+k}) &= \frac{x_i - x_k}{\rho(x_i, \ldots, x_{i+k-1}) - \rho(x_{i+1}, \ldots, x_{i+k})} + \rho(x_{i+1}, \ldots, x_{i+k-1}) \\ &\qquad i = 0, 1, 2, \ldots, \quad k = 1, 2, \ldots \end{aligned}$$

Von diesen reziproken Differenzen lässt sich zeigen, dass sie symmetrische Funktionen ihrer Argumente sind. Der Zusammenhang zwischen den reziproken und inversen Differenzen ist durch

$$\varphi(x_0, \ldots, x_i) = \rho(x_0, \ldots, x_i) - \rho(x_0, \ldots, x_{i-2})$$

gegeben. Diese Beziehung lässt sich leicht mittels vollständiger Induktion beweisen, falls man ausnutzt, dass die reziproken Differenzen invariant gegenüber

Vertauschungen ihrer Argumente sind.
Verwendet man im obigen Kettenbruch die reziproken Differenzen, so erhält man die endgültige Form, den THIELEschen Kettenbruch:

$$\begin{aligned}\Phi^{n,n}(x) &= \rho(x_0) + \underline{x-x_0}\Big/\rho(x_0,x_1) + \underline{x-x_1}\Big/\rho(x_0,x_1,x_2)-\rho(x_0) + \cdots \\ &\quad + \underline{x-x_{2n-1}}\Big/\rho(x_0,x_1,x_2,\ldots,x_{2n})-\rho(x_0,x_1,x_2,\ldots,x_{2n-2}).\end{aligned}$$

2.3.4. Fehler bei der Rationalen Interpolation

Bei rationaler Interpolation ist nicht auszuschließen, dass zwischen den Stützstellen Pole der rationalen Interpolationsfunktion auftreten. In der Nähe eines Pols wird der Interpolationsfehler im Allgemeinen beliebig groß werden. Daher lässt sich der Fehler nicht ohne Kenntnis der Interpolationsfunktion abschätzen. Es gilt der folgende Satz.

2.23. Satz: *Zu einer Funktion $f \in C^{n+1}[a,b]$ seien Stützpunkte*

$$(x_0,f_0),\ldots,(x_n,f_n) \in [a,b], \quad f_i = f(x_i), \quad i = 0,\ldots,n$$

gegeben. Ferner sei $\Phi(x) = P(x)/Q(x)$ eine rationale Funktion, für die $\Phi(x_i) = f_i$ für $i = 0,\ldots,n$ gilt. Dann existiert zu jedem $\bar{x} \in [a,b]$ ein

$$\xi \in I = [\min\{\bar{x},x_0,\ldots,x_n\}, \max\{\bar{x},x_0,\ldots,x_n\}]$$

mit

$$\begin{aligned}f(\bar{x}) - \Phi(\bar{x}) &= \frac{(\bar{x}-x_0)(\bar{x}-x_1)\cdots(\bar{x}-x_n)}{(n+1)!\,Q(\bar{x})}\frac{d^{n+1}}{dx^{n+1}}\left(Q(x)f(x)\right)\Big|_{x=\xi} \\ &= \frac{\omega(\bar{x})}{(n+1)!\,Q(\bar{x})}\frac{d^{n+1}}{dx^{n+1}}\left(Q(x)f(x)\right)\Big|_{x=\xi}\end{aligned}$$

Beweis: Für $\bar{x} \in \{x_0,\ldots,x_n\}$ ist die Behauptung trivial. Es sei daher $\bar{x} \neq x_i$ für $i = 0,\ldots,n$. Wir definieren die Funktion

$$F(x) = f(x) - \Phi(x) - K\frac{\omega(x)}{Q(x)}.$$

Offensichtlich gilt

$$F(x_i) = 0, \quad i = 0,1,\ldots,n.$$

Wir bestimmen die Konstante K so, dass auch $F(\bar{x}) = 0$ gilt. Dann hat die Funktion $F(x)$ $n+2$ Nullstellen im Intervall I. Das gilt ebenfalls für die Funktion

$$Q(x)F(x) = Q(x)f(x) - P(x) - K\omega(x).$$

Mehrmalige Anwendung des Satzes von ROLLE liefert, dass die $(n+1)$-te Ableitung der Funktion $Q(x)F(x)$ eine Nullstelle ξ im Intervall I hat. Es gilt dann

$$0 = \frac{d^{n+1}}{dx^{n+1}} \left(Q(x)F(x)\right)\Big|_{x=\xi} = \frac{d^{n+1}}{dx^{n+1}} \left(Q(x)f(x)\right)\Big|_{x=\xi} - K \cdot (n+1)!$$

wegen

$$\frac{d^{n+1}}{dx^{n+1}}\omega(x) = (n+1)!, \quad \frac{d^{n+1}}{dx^{n+1}}P(x) = 0.$$

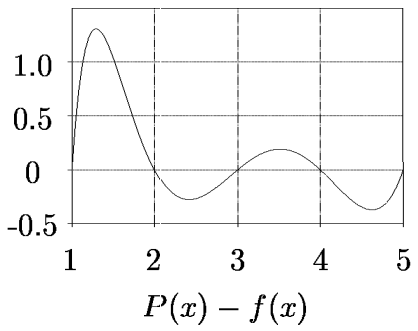
Damit folgt

$$K = K(\bar{x}) = \frac{1}{(n+1)!}\frac{d^{n+1}}{dx^{n+1}} \left(Q(x)f(x)\right)\Big|_{x=\xi}.$$

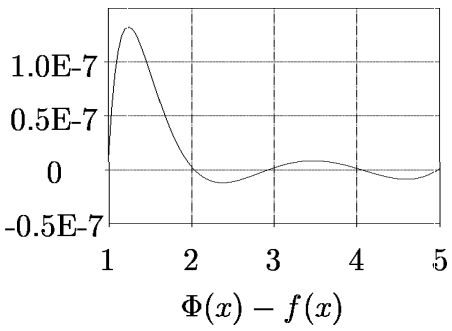
♣

Die Rationale Interpolation bietet Vorteile, falls man Funktionen interpolieren will, die Polstellen besitzen. In der Nähe eines Pols wird man mit Hilfe der Polynominterpolation nur ungenaue Resultate erhalten, wohingegen bei der Rationalen Interpolation eine gute Übereinstimmung zwischen der zu interpolierenden Funktion und der rationalen Interpolationsfunktion zu erwarten ist. Das folgende Beispiel zeigt diesen Unterschied.

2.24. Beispiel: Für die Funktion $f(x) = \cot(\pi x/180)$ wurde zu den Stützstellen $x_i = i$ mit $i = 1,2,3,4,5$ das Interpolationspolynom $P(x)$ und eine rationale Interpolationsfunktion $\Phi(x)$ in Form des THIELEschen Kettenbruchs berechnet. Es sind die Interpolationsfehler der Polynominterpolation (links) und der Rationalen Interpolation (rechts) dargestellt. Man beachte den unterschiedlichen Maßstab auf der y-Achse.



$P(x) - f(x)$



$\Phi(x) - f(x)$

♡

2.4. Spline-Interpolation

2.4.1. Eigenschaften von Splinefunktionen

Wie wir in den letzten Abschnitten gesehen haben, wird der Fehler der Polynominterpolation i. a. selbst bei glatten Funktionen zwischen den Stützstellen beliebig groß. Ähnliches gilt für rationale Interpolationsfunktionen, wenn sie auch für bestimmte Aufgaben (Interpolation von Funktionen mit Polstellen) gewisse Vorteile haben. Andererseits ist bei der rationalen Interpolationsaufgabe die Existenz einer Lösung nicht gesichert. Wir wollen nun eine andere Klasse von Funktionen zur Interpolation verwenden. Dazu zunächst eine Definition.
Es sei

$$\Delta_n : \{x_0, x_1, \ldots, x_n\}, \quad a = x_0 < x_1 < \cdots < x_n = b$$

eine Zerlegung des Intervalls $[a,b]$ und $m \in \mathbb{N}$. Dann heißt die Funktion

$$s : [a,b] \longrightarrow \mathbb{R}$$

ein **(Polynom-) Spline** vom Grad m zur Zerlegung Δ_n, falls

1. $s \in C^{m-1}[a,b]$ und
2. $s|_{[x_j,x_{j+1}]} \in \Pi_m$ für $j = 0,1,\ldots,n-1$, d. h. die Einschränkung der Funktion s auf jedes der Intervalle $[x_j,x_{j+1}]$ ein Polynom vom Grade höchstens m ist.

Mit

$$S_m(\Delta_n) = \left\{ s \in C^{m-1}[a,b] \;\middle|\; s|_{[x_j,x_{j+1}]} \in \Pi_m \quad j = 0,1,\ldots,n-1 \right\}$$

bezeichnen wir die Menge aller Splinefunktionen vom Grad m zur Zerlegung Δ_n. Offensichtlich ist $S_m(\Delta_n)$ ein Vektorraum über dem Körper der reellen Zahlen. Im folgenden Satz wird eine Aussage über eine Basis und die Dimension von $S_m(\Delta_n)$ gemacht.

2.25. Satz: *Durch die Funktionen*

$$\{(x-x_0)^0, (x-x_0)^1, \ldots, (x-x_0)^m, (x-x_1)_+^m, (x-x_2)_+^m, \ldots, (x-x_{n-1})_+^m\}$$

ist eine Basis des $S_m(\Delta_n)$ gegeben. Die Funktionen $(x-x_j)_+^m : \mathbb{R} \longrightarrow \mathbb{R}$ sind dabei durch

$$(x-x_j)_+^m = \begin{cases} (x-x_j)^m & \text{für } x \geq x_j, \\ 0 & \text{für } x < x_j \end{cases}$$

definiert. Jedes $s \in S_m(\Delta_n)$ besitzt die eindeutige Darstellung

$$s(x) = \sum_{i=0}^{m} \frac{s^{(i)}(x_0)}{i!}(x-x_0)^i + \sum_{j=1}^{n-1} \frac{s^{(m)}(x_j+0) - s^{(m)}(x_j-0)}{m!}(x-x_j)_+^m,$$

wobei mit $s^{(m)}(x_j+0)$ bzw. $s^{(m)}(x_j-0)$ der rechts- bzw. linksseitige Grenzwert von $s^{(m)}$ an der Stelle x_j gemeint ist.
Damit gilt

$$\dim S_m(\Delta_n) = m+n.$$

Beweis: Offensichtlich gilt

$$(x-x_0)^0, (x-x_0)^1, \ldots, (x-x_0)^m \in S_m(\Delta_n),$$
$$(x-x_1)_+^m, (x-x_2)_+^m, \ldots, (x-x_{n-1})_+^m \in S_m(\Delta_n).$$

Es ist somit nur noch zu zeigen, dass jeder beliebige Spline $s \in S_m(\Delta_n)$ die angegebene Darstellung besitzt. Dazu definieren wir zu einem Spline $s \in S_m(\Delta_n)$ den Spline $t \in S_m(\Delta_n)$, gemäß

$$t(x) = \sum_{i=0}^{m} \frac{s^{(i)}(x_0)}{i!}(x-x_0)^i + \sum_{j=1}^{n-1} \frac{s^{(m)}(x_j+0) - s^{(m)}(x_j-0)}{m!}(x-x_j)_+^m.$$

Dann gilt für den Spline $(s-t)(x)$

$$(s-t)^{(m)}(x_j+0) = (s-t)^{(m)}(x_j-0), \quad j = 1,2,\ldots,n-1.$$

Damit ist $s-t \in C^m[a,b]$. Da aber $s-t$ stückweise aus Polynomen vom Höchstgrad m zusammengesetzt ist, gilt $s-t \in \Pi_m$. Weiterhin gilt $(s-t)^{(i)}(x_0) = 0$ für $i = 0,1,\ldots,m$. Daraus folgt $(s-t)(x) \equiv 0$ und somit $s = t$. Jeder Spline $s \in S_m(\Delta_n)$ besitzt demnach die angegebene Darstellung. Die Eindeutigkeit der Darstellung ist offensichtlich. daher ist

$$\{(x-x_0)^0, (x-x_0)^1, \ldots, (x-x_0)^m, (x-x_1)_+^m, \ldots, (x-x_{n-1})_+^m\}$$

eine Basis des Vektorraumes aller Splinefunktionen vom Grad m zur Zerlegung Δ_n, und es gilt

$$\dim S_m(\Delta_n) = m+n.$$

Wir erinnern uns nun daran, dass wir Splinefunktionen zur Interpolation nutzen wollen. Die entsprechende Interpolationsaufgabe lautet dann:
Für eine Zerlegung Δ_n des Intervalls $[a,b]$ und vorgegebene Werte $f_0, f_1, \ldots, f_n$ ist eine Splinefunktion $s \in S_m(\Delta_n)$ so zu bestimmen, dass die Interpolationsbedingungen

$$s(x_i) = f_i, \quad i = 0,1,\ldots,n$$

erfüllt sind. Nach Satz 2.25 benötigen wir aber $m+n$ Bedingungen, um eine Splinefunktion aus $S_m(\Delta_n)$ eindeutig festzulegen. Es liegt nahe, die fehlenden $m-1$ Bedingungen an den Randpunkten a und b festzulegen. Aus Symmetriegründen wäre es wünschenswert, dass die Anzahl der zusätzlichen Bedingungen gerade ist, damit sie gleichmäßig auf die Randpunkte des Intervalls $[a,b]$ aufteilbar sind. Das ist genau dann der Fall, wenn m ungerade ist. Aus diesem Grund werden wir uns im folgenden auch nur mit Splines ungeraden Grades beschäftigen. Wir werden daher immer $m = 2r+1$ voraussetzen. Splines ungeraden Grades zeichnen sich weiterhin durch eine Extremaleigenschaft aus, die wir als erstes beweisen wollen. Dazu definieren wir auf dem Raum $C[a,b]$ aller stetigen Funktionen über dem Intervall $[a,b]$ die Norm $\|\circ\|$ gemäß

$$\|f\| = \left(\int_a^b f^2(x)\,dx \right)^{1/2}.$$

Für den eigentlichen Beweis der Extremaleigenschaft von Splines ungeraden Grades benötigen wir folgenden Hilfssatz.

2.26. Satz: *Es sei Δ_n eine Zerlegung des Intervalls $[a,b]$. Weiterhin seien eine Splinefunktion $s \in S_{2r+1}(\Delta_n)$ und eine Funktion $g \in C^{r+1}[a,b]$ gegeben, die die Bedingungen*

$$s(x_i) = g(x_i) = f_i, \quad i = 0,1,\ldots,n$$

erfüllen. Dann gilt

$$\|g^{(r+1)} - s^{(r+1)}\|^2 = \|g^{(r+1)}\|^2 - \|s^{(r+1)}\|^2 - 2I(g,s)$$

mit

$$I(g,s) = \sum_{i=0}^{r-1} (-1)^i \left[g^{(r-i)}(x) - s^{(r-i)}(x) \right] s^{(r+1+i)}(x) \Big|_a^b.$$

Beweis: Es gilt

$$
\begin{aligned}
&\|g^{(r+1)} - s^{(r+1)}\|^2 = \\
&= \int_a^b \left[g^{(r+1)}(x) - s^{(r+1)}(x)\right]^2 dx \\
&= \int_a^b \left[\left(g^{(r+1)}(x)\right)^2 - 2g^{(r+1)}(x)s^{(r+1)}(x) + \left(s^{(r+1)}(x)\right)^2\right] dx \\
&= \int_a^b \left[\left(g^{(r+1)}(x)\right)^2 - 2\left(g^{(r+1)}(x) - s^{(r+1)}(x)\right)s^{(r+1)}(x) - \left(s^{(r+1)}(x)\right)^2\right] dx \\
&= \|g^{(r+1)}\|^2 - 2\int_a^b \left(g^{(r+1)}(x) - s^{(r+1)}(x)\right)s^{(r+1)}(x)\,dx - \|s^{(r+1)}\|^2.
\end{aligned}
$$

Für das verbleibende Integral ergibt sich nach mehrmaliger partieller Integration

$$
\begin{aligned}
&\int_a^b \left(g^{(r+1)}(x) - s^{(r+1)}(x)\right)s^{(r+1)}(x)\,dx = \\
&= \left(g^{(r)}(x) - s^{(r)}(x)\right)s^{(r+1)}(x)\Big|_a^b - \int_a^b \left(g^{(r)}(x) - s^{(r)}(x)\right)s^{(r+2)}(x)\,dx \\
&= \left(g^{(r)}(x) - s^{(r)}(x)\right)s^{(r+1)}(x)\Big|_a^b \\
&\qquad - \left(g^{(r-1)}(x) - s^{(r-1)}(x)\right)s^{(r+2)}(x)\Big|_a^b \\
&\qquad + \int_a^b \left(g^{(r-1)}(x) - s^{(r-1)}(x)\right)s^{(r+3)}(x)\,dx \\
&\ \vdots \\
&= \sum_{i=0}^{r-1}(-1)^i \left(g^{(r-i)}(x) - s^{(r-i)}(x)\right)s^{(r+1+i)}(x)\Big|_a^b \\
&\qquad + (-1)^r \int_a^b \left(g'(x) - s'(x)\right)s^{(2r+1)}(x)\,dx
\end{aligned}
$$

Wegen $s\Big|_{[x_j,x_{j+1}]} \in \Pi_{2r+1}$ ist $s^{(2r+1)}$ stückweise konstant. Damit gilt

$$\begin{aligned}
&\int_a^b \big(g'(x)-s'(x)\big)\, s^{(2r+1)}(x)\,dx = \sum_{j=0}^{n-1} \big(g(x)-s(x)\big)\, s^{(2r+1)}(x)\Big|_{x_j+0}^{x_{j+1}-0} \\
&= \sum_{j=0}^{n-1} \big(g(x_{j+1})-s(x_{j+1})\big)\, s^{(2r+1)}(x_{j+1}-0) - \\
&\qquad - \sum_{j=0}^{n-1} \big(g(x_j)-s(x_j)\big)\, s^{(2r+1)}(x_j+0) \\
&= \sum_{j=0}^{n-1} \big(f_{j+1}-f_{j+1}\big)\, s^{(2r+1)}(x_{j+1}-0) - \sum_{j=0}^{n-1} \big(f_j-f_j\big)\, s^{(2r+1)}(x_j+0) \\
&= 0.
\end{aligned}$$

❀

Wählt man nun zusätzliche Bedingungen so, dass $I(g,s)$ verschwindet, so gilt

$$0 \leqq \|g^{(r+1)} - s^{(r+1)}\|^2 = \|g^{(r+1)}\|^2 - \|s^{(r+1)}\|^2$$

daher

$$\|s^{(r+1)}\| \leqq \|g^{(r+1)}\|.$$

In diesem Falle wird das Funktional $\|\circ\|$ in gewissem Sinne durch eine Splinefunktion minimiert. Der verbleibende Term

$$I(g,s) = \sum_{i=0}^{r-1} (-1)^i \left[g^{(r-i)}(x) - s^{(r-i)}(x)\right] s^{(r+1+i)}(x)\Big|_a^b$$

verschwindet offensichtlich, falls

- $g^{(j)}(a) = s^{(j)}(a)$ und $g^{(j)}(b) = s^{(j)}(b)$ für $j = 1,\ldots,r$ oder
- $s^{(j)}(a) = s^{(j)}(b) = 0$ für $j = r+1,\ldots,2r$ oder
- $\left[g^{(j)}(a) - s^{(j)}(a)\right] s^{(r+j)}(a) = \left[g^{(j)}(b) - s^{(j)}(b)\right] s^{(r+j)}(b)$ für $j = 1,\ldots,r$

gilt. Daraus ergeben sich jeweils die fehlenden $2r$ Bedingungen zum Lösen des Interpolationsproblems. Wir erhalten folgende drei Typen von Interpolationsaufgaben:

(H) Interpolation mit HERMITE-Randbedingungen:
Für gegebene Werte $f_0, \ldots, f_n$ und Werte $f_0', \ldots, f_0^{(r)}, f_n', \ldots, f_n^{(r)}$ ist ein interpolierender Spline $s_f \in S_{2r+1}(\Delta_n)$ so zu bestimmen, dass

$$s_f^{(i)}(a) = f_0^{(i)}, \quad s_f^{(i)}(b) = f_n^{(i)}, \quad i = 1, \ldots, r$$

gilt. Ist eine Funktion $f \in C^r[a,b]$ zu interpolieren, so wird man natürlich

$$f_0^{(i)} = f^{(i)}(a), \quad f_n^{(i)} = f^{(i)}(b), \quad i = 1, \ldots, r$$

wählen.

(N) Interpolation mit natürlichen Randbedingungen:
Für gegebene Werte $f_0, \ldots, f_n$ ist ein interpolierender Spline $s_f \in S_{2r+1}(\Delta_n)$ zu bestimmen, so dass

$$s_f^{(i)}(a) = s_f^{(i)}(b) = 0, \quad i = r+1, \ldots, 2r$$

gilt. Zusätzlich sei vorausgesetzt, dass $n \geqq r$.

(P) Interpolation mit periodischen Randbedingungen:
Für gegebene Werte $f_0, \ldots, f_n$ mit $f_0 = f_n$ ist ein interpolierender Spline $s_f \in S_{2r+1}(\Delta_n)$ so zu bestimmen, dass

$$s_f^{(i)}(a) = s_f^{(i)}(b) \quad i = 1, \ldots, 2r$$

gilt.

Offensichtlich ist in jedem der drei Fälle die Gesamtzahl der Bedingungen gleich $2r+1+n$, d. h. gleich der Dimension von $S_{2r+1}(\Delta_n)$. Im folgenden Satz werden wir zeigen, dass jedes der Probleme eine eindeutige Lösung besitzt.

2.27. Satz: *Jede der Interpolationsaufgaben (H), (N) beziehungsweise (P) besitzt eine eindeutige Lösung $s_f \in S_{2r+1}(\Delta_n)$. Ist $g \in C^{r+1}[a,b]$ eine Funktion mit $g(x_i) = f_i$ für $i = 0, 1, \ldots, n$ und*

- *$g^{(j)}(a) = f_0^{(j)}$ und $g^{(j)}(b) = f_n^{(j)}$ für $j = 1, \ldots, r$ für die Aufgabe (H),*
- *$g^{(j)}(a) = g^{(j)}(b)$ für $j = 0, 1, \ldots, r$ für die Aufgabe (P),*

so gilt

$$\|g^{(r+1)}\| \geqq \|s_f^{(r+1)}\|$$

und im Falle $g \neq s_f$

$$\|g^{(r+1)}\| > \|s_f^{(r+1)}\|.$$

Beweis: Nach Satz 2.25 lässt sich die Splinefunktion $s_f \in S_{2r+1}(\Delta_n)$ in eindeutiger Weise als Linearkombination der dort angegebenen Basisfunktionen ausdrücken. Setzt man einen Ansatz

$$s_f(x) = \sum_{i=0}^{2r+1} \alpha_i (x-x_0)^i + \sum_{j=1}^{n-1} \beta_j (x-x_j)_+^m$$

in die Interpolationsbedingungen ein, so erhält man ein lineares Gleichungssystem von $2r+1+n$ Gleichungen zur Bestimmung der $2r+1+n$ Parameter

$$\alpha_0, \ldots, \alpha_{2r+1}, \beta_0, \ldots, \beta_{n-1}.$$

Wenn sich zeigen lässt, dass das homogene Problem eine eindeutige Lösung hat, so ist die Koeffizientenmatrix des Gleichungssystems regulär und jedes Interpolationsproblem hat eine eindeutige Lösung. Vom homogenen Problem wissen wir, dass es zumindestens die triviale Lösung besitzt. Allgemeiner nehmen wir an, dass s_f eine Lösung des jeweiligen Interpolationsproblems ist. Wir zeigen die Eindeutigkeit von s_f. Nach Satz 2.26 gilt

$$0 \leqq \|g^{(r+1)} - s_f^{(r+1)}\| = \|g^{(r+1)}\| - \|s_f^{(r+1)}\| - I(g, s_f)$$

mit

$$I(g, s_f) = \sum_{i=0}^{r-1} (-1)^i \left[g^{(r-i)}(x) - s^{(r-i)}(x) \right] s^{(r+1+i)}(x) \Big|_a^b$$

Aufgrund der zusätzlichen Bedingungen verschwindet dieser Ausdruck, so dass sofort $\|g^{(r+1)}\| \geqq \|s_f^{(r+1)}\|$ folgt. Ist nun

$$0 = \|g^{(r+1)}\| - \|s_f^{(r+1)}\| = \|g^{(r+1)} - s_f^{(r+1)}\| = \|(g-s_f)^{(r+1)}\|,$$

so folgt $g - s_f \in \Pi_r$. Wir unterscheiden nun die drei Fälle.

- Für HERMITE-Randbedingungen ist $(g-s_f)^{(j)}(a) = 0$ für $j = 0, 1, \ldots, r$. Daher ist a eine $(r+1)$-fache Nullstelle von $(g-s_f)$. Damit ist $g - s_f \equiv 0$, daher $g = s_f$.

- Für natürliche Randbedingungen ist $(g-s_f)(x_i)=0$ für $i=0,1,\ldots,n$. Da hier $n \geqq r$ vorausgesetzt wurde, folgt ebenfalls $g=s_f$.
- Bei periodischen Randbedingungen ist

$$(g-s_f)^{(j)}(a)=(g-s_f)^{(j)}(b), \quad j=0,1,\ldots,r.$$

Wegen $g-s_f \in \mathbf{\Pi}_r$ folgt daraus ebenfalls $g=s_f$.

Damit gilt $\|g^{(r+1)}\| > \|s_f^{(r+1)}\|$ für $g \neq s_f$. Es sei nun $\bar{s}_f \neq s_f$ eine weitere Splinefunktion, die das Problem löst. Dann gilt nach dem eben bewiesenen einerseits $\|\bar{s}_f^{(r+1)}\| > \|s_f^{(r+1)}\|$, andererseits minimiert auch $\bar{s}_f$ das Funktional $\|\circ\|$; es muss daher $\|\bar{s}_f^{(r+1)}\| = \|s_f^{(r+1)}\|$ gelten. Das ist ein Widerspruch. Es gilt somit $\bar{s}_f = s_f$. Die Lösung des Interpolationsproblems ist eindeutig bestimmt. Nach dem oben gesagten folgt aus der Eindeutigkeit aber auch die Existenz der interpolierenden Splinefunktion. ❀

2.4.2. Berechnen der interpolierenden kubischen Splines

Von besonderer Bedeutung sind die kubischen Splinefunktionen ($m=3$). Bei ihnen wird das Funktional

$$\|y\| = \sqrt{\int_a^b (y''(x))^2 dx}$$

als Gesamtkrümmung der Splinefunktion im Intervall $[a,b]$ interpretiert. [1] Dies erklärt auch den Namen Splinefunktion. Im Englischen versteht man unter "'spline"' eine Art biegsames Kurvenlineal, das in einigen Punkten fest gelagert wird, um eine Kurve möglichst geringer Krümmung zu erhalten.
Es sei nun wieder durch $\Delta_n : \{x_0, x_1, \ldots, x_n\}$ mit $a = x_0 < x_1 < \cdots < x_n = b$ eine Zerlegung des Intervalls $[a,b]$ gegeben. Es soll eine kubische Splinefunktion $s_f \in S_3(\Delta_n)$ berechnet werden, die in den Punkten $x_0, x_1, \ldots, x_n$ gegebene Werte $f_0, f_1, \ldots, f_n$ annimmt und zusätzlich eine der Bedingungen (N), (H) oder (P)

[1] Die Krümmung einer Kurve in einem Punkt ist durch $y''/(1+y'^2)^{3/2}$ gegeben. Für kleines $|y'|$ ist dann

$$\|y\| = \sqrt{\int_a^b (f''(x))^2 dx}$$

ein Maß für die Gesamtkrümmung von $y=f(x)$ im Intervall $[a,b]$.

erfüllt. Auf jedem Intervall $[x_i, x_{i+1}]$ für $i = 0, 1, \ldots, n-1$ ist die Funktion s_f durch ein kubisches Polynom gegeben. Wir machen darum folgenden Ansatz:

$$s_f\Big|_{[x_i,x_{i+1}]}(x) = \alpha_i + \beta_i(x-x_i) + \gamma_i(x-x_i)^2 + \delta_i(x-x_i)^3 \quad i = 0, 1, \ldots, n-1.$$

Wie schon im Beweis zu Satz 2.27 angedeutet, läuft das Berechnen der interpolierenden kubischen Splinefunktion $s_f(x)$ auf das Lösen eines linearen Gleichungssystems zur Bestimmung der Parameter $\alpha_i, \beta_i, \gamma_i, \delta_i$ für $i = 0, 1, \ldots, n-1$ hinaus. Das wäre ein System der Dimension $4n$. Wir werden aber sehen, dass man mit dem Lösen eines Gleichungssystems der Dimension $\approx n$ auskommt.
Wir führen zunächst die Momente M_i, $i = 0, 1, \ldots, n$, das sind die Werte der zweiten Ableitung von s_f an den Stellen x_i, $i = 0, 1, \ldots, n$, ein. Da s_f stückweise aus Polynomen dritten Grades besteht, ist s_f'' stückweise linear. Wegen der Stetigkeit der zweiten Ableitung gilt dann

$$s_f''\Big|_{[x_i,x_{i+1}]}(x) = M_i + \frac{M_{i+1} - M_i}{x_{i+1} - x_i}(x - x_i), \quad i = 0, 1, \ldots, n-1.$$

Zweimalige Integration liefert

$$s_f\Big|_{[x_i,x_{i+1}]}(x) = \alpha_i + \beta_i(x-x_i) + \frac{M_i}{2}(x-x_i)^2 + \frac{M_{i+1} - M_i}{6(x_{i+1} - x_i)}(x-x_i)^3.$$

Aus

$$s_f\Big|_{[x_i,x_{i+1}]}(x_i) = f_i$$

folgt

$$\alpha_i = f_i,$$

und aus

$$s_f\Big|_{[x_i,x_{i+1}]}(x_{i+1}) = f_{i+1}$$

folgt

$$\begin{aligned} \beta_i &= \frac{f_{i+1} - f_i}{x_{i+1} - x_i} - \frac{M_i}{2}(x_{i+1} - x_i) - \frac{M_{i+1} - M_i}{6}(x_{i+1} - x_i) \\ &= \frac{f_{i+1} - f_i}{x_{i+1} - x_i} - \frac{M_{i+1} + 2M_i}{6}(x_{i+1} - x_i). \end{aligned}$$

Mit der Abkürzung $h_i = x_{i+1} - x_i$ erhalten wir damit folgende Darstellung der Parameter der kubischen Polynome durch die Momente der Splinefunktion:

$$\alpha_i = f_i, \tag{2.7}$$

$$\beta_i = \frac{f_{i+1} - f_i}{h_i} - \frac{M_{i+1} + 2M_i}{6} h_i, \tag{2.8}$$

$$\gamma_i = \frac{M_i}{2}, \tag{2.9}$$

$$\delta_i = \frac{M_{i+1} - M_i}{6h_i}. \tag{2.10}$$

für $i = 0, 1, \ldots, n-1$. Wegen der Stetigkeit der ersten Ableitung von s_f gilt:

$$s'_f \Big|_{[x_i, x_{i+1}]}(x_{i+1}) = s'_f \Big|_{[x_{i+1}, x_{i+2}]}(x_{i+1}), \quad i = 0, 1, \ldots, n-2,$$

und damit

$$\beta_i + 2\gamma_i \cdot h_i + 3\delta_i \cdot h_i = \beta_{i+1}, \quad i = 0, 1, \ldots, n-2.$$

Setzt man nun die Ausdrücke 2.8 bis 2.10 in die letzte Gleichung ein, so erhält man $n-1$ Gleichungen zur Bestimmung der $n+1$ Momente $M_0, M_1, \ldots, M_n$:

$$\frac{h_i}{6} M_i + \frac{h_i + h_{i+1}}{3} M_{i+1} + \frac{h_{i+1}}{6} M_{i+2} = \frac{f_{i+2} - f_{i+1}}{h_{i+1}} - \frac{f_{i+1} - f_i}{h_i}, \quad i = 0, 1, \ldots, n-2. \tag{2.11}$$

Diesem System soll eine Form gegeben werden, in der man den Zusammenhang zwischen den Elementen der Koeffizientenmatrix besser erkennt. Dazu multiplizieren wir die Gleichungen jeweils mit $6/(h_i + h_{i+1})$, ersetzen i durch $i-1$ und erhalten

$$\frac{h_{i-1}}{h_{i-1} + h_i} M_{i-1} + 2M_i + \frac{h_i}{h_{i-1} + h_i} M_{i+1} = \frac{6}{h_{i-1} + h_i} \left[\frac{f_{i+1} - f_i}{h_i} - \frac{f_i - f_{i-1}}{h_{i-1}} \right] \tag{2.12}$$

für $i = 1, \ldots, n-1$.
Die fehlenden zwei Gleichungen ergeben sich aus den zusätzlichen Randbedingungen der Probleme (H), (N) bzw. (P). Speziell folgt für die einzelnen Probleme:

$$\mathsf{H}: \qquad s'_f(x_0) = f'_0 \Longrightarrow 2M_0 + M_1 = \frac{6}{h_0} \left[\frac{f_1 - f_0}{h_0} - f'_0 \right],$$

$$s'_f(x_n) = f'_n \Longrightarrow M_{n-1} + 2M_n = \frac{6}{h_n} \left[f'_n - \frac{f_n - f_{n-1}}{h_{n-1}} \right].$$

$$\mathsf{N}: \qquad \begin{aligned} s_f''(x_0) = 0 &\Longrightarrow M_0 = 0, \\ s_f''(x_n) = 0 &\Longrightarrow M_n = 0. \end{aligned}$$

$$\mathsf{P}: \qquad \begin{aligned} s_f(x_0) = s_f(x_n) &\Longrightarrow f_0 = f_n \\ s_f''(x_0) = s_f''(x_n) &\Longrightarrow M_0 = M_n, \\ s_f'(x_0) = s_f'(x_n) &\Longrightarrow \frac{h_{n-1}}{h_{n-1}+h_0} M_{n-1} + 2M_n + \frac{h_0}{h_{n-1}+h_0} M_1 \\ &\qquad = \frac{6}{h_{n-1}+h_0} \left[\frac{f_1 - f_n}{h_0} - \frac{f_n - f_{n-1}}{h_{n-1}} \right]. \end{aligned}$$

Damit sind die Momente für die Probleme (H), (N) bzw. (P) jeweils durch ein lineares Gleichungssystem der Form $Am = d$ festgelegt. Dabei gilt

$$A = \begin{pmatrix} 2 & \lambda_0 & 0 & \dots & 0 \\ \mu_1 & 2 & \lambda_1 & \ddots & \vdots \\ 0 & \ddots & \ddots & \ddots & 0 \\ \vdots & \ddots & \mu_{n-1} & 2 & \lambda_{n-1} \\ 0 & \dots & 0 & \mu_n & 2 \end{pmatrix}, \quad m = \begin{pmatrix} M_0 \\ M_1 \\ \vdots \\ M_{n-1} \\ M_n \end{pmatrix}, \quad d = \begin{pmatrix} d_0 \\ d_1 \\ \vdots \\ d_{n-1} \\ d_n \end{pmatrix}$$

mit

$$\left.\begin{aligned} \mu_i &= \frac{h_{i-1}}{h_i + h_{i-1}}, \\ \lambda_i &= \frac{h_i}{h_i + h_{i-1}} = 1 - \mu_i, \\ d_i &= \frac{6}{h_i + h_{i-1}} \left[\frac{f_{i+1} - f_i}{h_i} - \frac{f_i - f_{i-1}}{h_{i-1}} \right] \end{aligned}\right\} \quad i = 1, \dots, n-1$$

und

$$\left.\begin{aligned} \mu_n &= \lambda_0 = 1, \\ d_1 &= \frac{6}{h_0} \left[\frac{f_1 - f_0}{h_0} - f_0' \right], \\ d_n &= \frac{6}{h_{n-1}} \left[f_n' - \frac{f_n - f_{n-1}}{h_{n-1}} \right] \end{aligned}\right\} \quad \text{für das Problem (H)}$$

bzw.

$$\left.\begin{array}{l} \mu_n = \lambda_0 = 0 \\ \\ d_0 = d_n = 0 \end{array}\right\} \quad \text{für das Problem } (\mathsf{N}).$$

Für diese beiden Problemstellungen ergeben sich tridiagonale Koeffizientenmatrizen. Das macht das Lösen der entsprechenden Gleichungssysteme besonders einfach.
Im Falle periodischer Randbedingungen ergibt sich das Gleichungssystem

$$Am = d$$

mit

$$A = \begin{pmatrix} 2 & \lambda_1 & 0 & \dots & 0 & \mu_1 \\ \mu_2 & 2 & \lambda_2 & \ddots & & 0 \\ 0 & \mu_3 & 2 & \ddots & \ddots & \vdots \\ \vdots & \ddots & \ddots & \ddots & \ddots & 0 \\ \vdots & & \ddots & \mu_{n-1} & 2 & \lambda_{n-1} \\ \lambda_n & 0 & \dots & 0 & \mu_n & 2 \end{pmatrix}, \quad m = \begin{pmatrix} M_1 \\ M_2 \\ \vdots \\ M_{n-1} \\ M_n \end{pmatrix}, \quad d = \begin{pmatrix} d_1 \\ d_2 \\ \vdots \\ d_{n-1} \\ d_n \end{pmatrix}.$$

Dabei sind μ_i, λ_i und d_i für $i = 1, \dots, n-1$ wie bei den Problemen (H) und (N) festgelegt. Außerdem gilt

$$\mu_n = \frac{h_{n-1}}{h_0 + h_{n-1}},$$

$$\lambda_n = \frac{h_0}{h_0 + h_{n-1}} = 1 - \mu_n,$$

$$d_n = \frac{6}{h_0 + h_{n-1}} \left[\frac{f_1 - f_n}{h_0} - \frac{f_n - f_{n-1}}{h_{n-1}} \right].$$

Die Koeffizientenmatrizen sind jeweils zeilendiagonaldominant, der Betrag des Diagonalelementes jeder Zeile ist größer als die Summe der Beträge der Nichtdiagonalelemente. Wie wir später sehen werden, sind solche Matrizen stets regulär, so dass auch auf diese konstruktive Weise die Existenz- und Eindeutigkeit der Lösungen der Probleme (H), (N) und (P) folgt.

2.4.3. Fehler und Konvergenz der Spline-Interpolation

In Abschnitt 2.2.4. haben wir gesehen, dass Interpolationspolynome nur unter starken Voraussetzungen gegen die Funktion konvergieren, die interpoliert werden soll. Selbst für immer feiner werdende Zerlegungen liegt im allgemeinen keine Konvergenz vor. Dagegen ist das Konvergenzverhalten der Spline-Funktionen bedeutend angenehmer. Wir werden zeigen, dass nicht nur die Splinefunktionen selbst gleichmäßig gegen die zu interpolierenden Funktionen konvergieren, sondern auch noch Ableitungen der Splinefunktion gegen entsprechende Ableitungen der zu interpolierenden Funktion.

2.28. Satz: *Es sei $f \in C^{r+1}[a,b]$ und $s_f \in S_{2r+1}(\Delta_n)$ die eindeutige Lösung eines der Probleme (H), (N) oder (P). Weiterhin sei*

$$h = \max_{j=0,\ldots,n-1} \{x_{j+1} - x_j\}.$$

Dann gilt

$$\|f^{(j)} - s_f^{(j)}\|_\infty \leqq \frac{(r+1)!}{\sqrt{r+1}\, j!} h^{r+1/2-j} \|f^{(r+1)}\|_2, \quad j = 0,1,\ldots,r.$$

Beweis: Die Funktion $q(x) = f(x) - s_f(x)$ hat die Nullstellen $x_0, x_1, \ldots, x_n$ im Intervall $[a,b]$. Nach dem Satz von ROLLE besitzt dann q' n Nullstellen in den Intervallen $[x_i, x_{i+1}]$, $i = 1,\ldots,n-1$, q'' $n-1$ Nullstellen in den Intervallen $[x_i, x_{i+2}]$, $i = 1,\ldots,n-2$, usw. Allgemein besitzt $q^{(j)}$ für $j = 1,\ldots,r+1$ mindestens $n-j+1$ Nullstellen, die jeweils in den Intervallen $[x_i, x_{i+j}]$, $i = 1,\ldots,n-j$ liegen. Der Abstand zweier aufeinanderfolgender Nullstellen von $q^{(j)}$ ist damit höchstens $(j+1)h$. Für ein beliebiges $t \in [a,b]$ und $j \in \{0,\ldots,r\}$ existiert damit ein

$$t_j \in \{t-(j+1)h, t+(j+1)h\} \cap [a,b] \text{ mit } q^{(j)}(t_j) = 0.$$

Dann ist

$$\begin{aligned} \left|q^{(j)}(t)\right| &= \left| \int_{t_j}^{t} q^{(j+1)}(x)\, dx \right| \\ &\leqq (j+1)h \, \|q^{(j+1)}\|_\infty, \quad j = 0,\ldots,r-1. \end{aligned}$$

Daraus folgt

$$\|q^{(j)}\|_\infty \leqq (j+1)h \, \|q^{(j+1)}\|_\infty, \quad j = 0,\ldots,r-1.$$

Mehrmalige Anwendung dieser Ungleichung liefert

$$\begin{aligned}\|q^{(j)}\|_\infty &\leqq (j+1)h\,\|q^{(j+1)}\|_\infty \\ &\leqq (j+1)(j+2)h^2\|q^{(j+2)}\|_\infty \\ &\vdots \\ &\leqq \frac{r!}{j!}h^{r-j}\|q^{(r)}\|_\infty.\end{aligned}$$

Eine Abschätzung von $\|q^{(r)}\|_\infty$ erhält man mittels der CAUCHY-SCHWARZschen Ungleichung. Es gilt für beliebiges $t \in [a,b]$

$$\begin{aligned}|q^{(r)}(t)| &= \left|\int_{t_r}^{t} q^{(r+1)}(x)\,dx\right| \\ &\leqq \sqrt{\int_{t_r}^{t} 1^2\,dx \cdot \int_{t_r}^{t} \left(q^{(r+1)}(x)\right)^2 dx} \\ &= \sqrt{|t-t_r|}\,\|q^{(r+1)}\|_2 \\ &\leqq \sqrt{(r+1)h}\,\|q^{(r+1)}\|_2.\end{aligned}$$

Damit erhalten wir zunächst

$$\|f^{(j)} - s_f^{(j)}\|_\infty \leqq \frac{(r+1)!}{\sqrt{r+1}\,j!}\sqrt{h}h^{r-j}\|f^{(r+1)} - s_f^{(r+1)}\|_2.$$

Nach Satz 2.27 gilt aber

$$\|f^{(r+1)} - s_f^{(r+1)}\|_2^2 = \|f^{(r+1)}\|_2^2 - \|s_f^{(r+1)}\|_2^2 \leqq \|f^{(r+1)}\|_2^2.$$

Damit folgt die Behauptung. ❀

Aus diesem Satz ergibt sich aber auch sofort die angekündigte Konvergenzeigenschaft der interpolierenden Splinefunktionen.

2.29. Satz: *Es seien* $\{\Delta_n\}_{n\in\mathbb{N}}$ *mit*

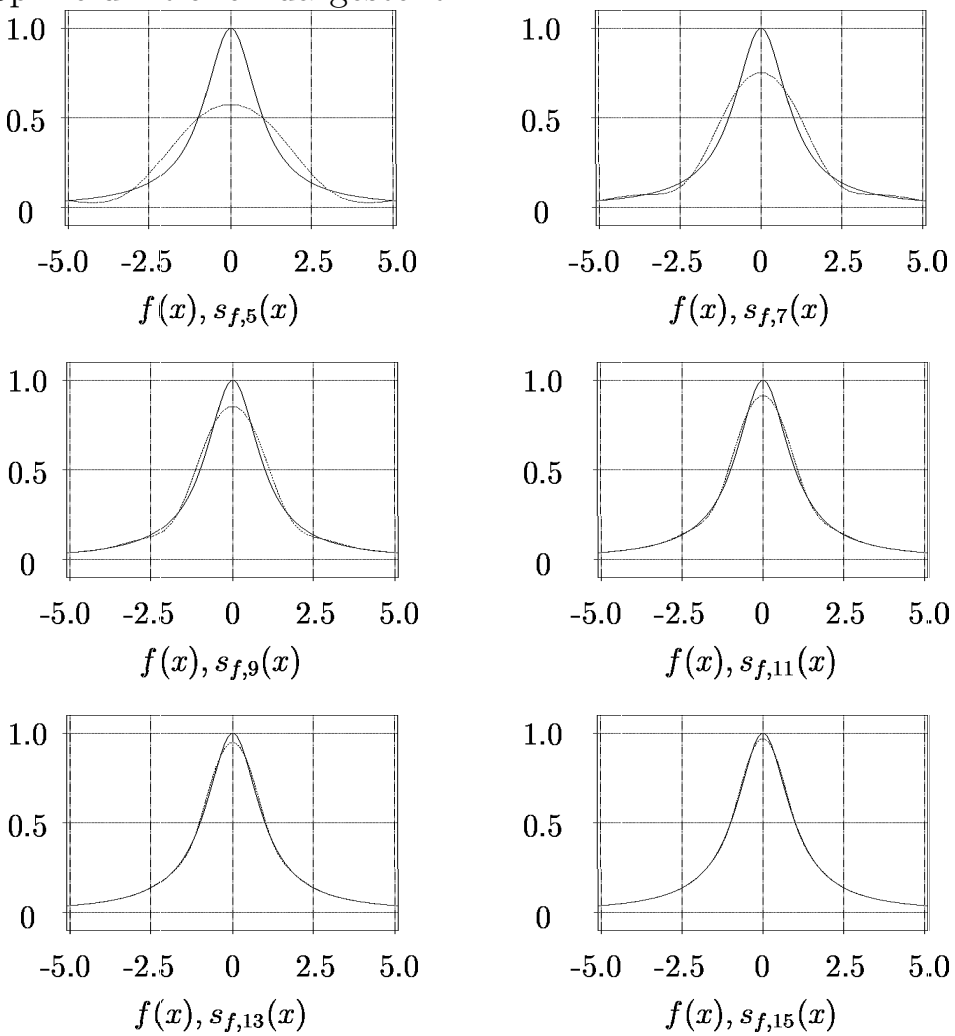
$$\Delta_n = \{a = x_0^{(n)} < x_1^{(n)} < \cdots < x_{n-1}^{(n)} < x_n^{(n)} = b\}$$

eine Folge von Zerlegungen des Intervalls $[a,b]$ *und* $f \in C^{r+1}[a,b]$.
Die Funktion $s_{n,f} \in S_{2r+1}(\Delta_n)$ *sei die Lösung der entsprechenden Interpolationsprobleme (H), (N) bzw. (P). Weiterhin sei*

$$h_n = \max_{j=0,1,\ldots,n-1} \{x_{j+1}^{(n)} - x_j^{(n)}\}$$

die Feinheit der Zerlegung Δ_n. *Dann gilt: Für alle Folgen* $\{\Delta_n\}_{n\in\mathbb{N}}$, *deren Feinheiten eine Nullfolge bilden, konvergiert die Folge der zugehörigen interpolierenden Splinefunktionen* $\{s_{n,f}\}_{n\in\mathbb{N}}$ *auf dem Intervall* $[a,b]$ *gleichmäßig gegen die Funktion* f. *Außerdem konvergieren alle Ableitungen bis zur Ordnung* r *von* $s_{n,f}$ *gleichmäßig gegen die entsprechenden Ableitungen von* f.

2.30. Beispiel: Für die Funktion $f(x) = 1/(1+x^2)$ wurde für $n = 5,7,9,11,13,15$ äquidistanten Stützstellen die jeweilige interpolierende Splinefunktion mit natürlichen Randbedingungen berechnet. Es sind die Funktion $f(x)$ und die entsprechenden Splinefunktionen dargestellt.



♡

2.5. Aufgaben

1. Zu den Stützstellen $x_0,\ldots,x_n$ seien L_i, $i=0,\ldots,n$ die LAGRANGE-Polynome. Man zeige:

   (a) $$\sum_{i=0}^{n} c_i x_i^j = \begin{cases} 1 & \text{für } j=0 \\ 0 & \text{für } j=1,2,\ldots,n \\ (-1)^n x_0\ldots x_n & \text{für } j=n+1 \end{cases} \qquad c_i = L_i(0)$$

   (b) $$\sum_{i=0}^{n} L_i(x) \equiv 1.$$

2. Die BESSEL-Funktion nullter Ordnung

   $$I_0(x) = \frac{1}{\pi}\int_0^{\pi} \cos(x\cdot\sin(t))\,dt$$

   soll an äquidistanten Stützstellen $x_i = x_0 + ih$, $i=0,1,\ldots$ tabelliert werden. Welche Schrittweite h ist zu wählen, falls bei

   (a) linearer Interpolation,

   (b) quadratischer Interpolation

   mit Hilfe dieser Tafel der absolute Fehler $|I_0(x) - P_n(x)|$ kleiner als 10^{-6} ausfallen soll?

3. Man schreibe ein Programm zur Polynominterploation, das die baryzentrische Darstellung verwendet.

4. Die auf dem abgeschlossenen Intervall $I=[-1,1]$ zweimal stetig differenzierbare Funktion f werde durch ein lineares Polynom zu den Stützpunkten $(x_0, f(x_0))$ und $(x_1, f(x_1))$ mit $x_0, x_1 \in I$ interpoliert. Dann ist

   $$\alpha = \frac{1}{2}\max_{\xi\in I}|f''(\xi)|\max_{x\in I}|(x-x_0)(x-x_1)|$$

   eine obere Schranke für den maximalen, absoluten Interpolationsfehler auf I. Wie hat man x_0 und x_1 zu wählen, damit α möglichst klein wird? Welcher Zusammenhang besteht dann zwischen $(x-x_0)(x-x_1)$ und $\cos(2\cdot\arccos(x))$?

5. Man berechne $\cot(x)$ für $x=\frac{\pi}{12}$, falls $\cot(x)$ an folgenden Stützstellen bekannt ist:

x_i	$\frac{\pi}{6}$	$\frac{\pi}{4}$	$\frac{\pi}{3}$	$\frac{\pi}{2}$
$\cot(x_i)$	$\sqrt{3}$	1	$\frac{\sqrt{3}}{3}$	0

(a) mit Hilfe der Polynominterpolation,

(b) mit Hilfe der rationalen Interpolation.

Man vergleiche die Ergebnisse mit dem exakten Wert $2+\sqrt{3}$.

6. Man programmiere den rekursiven Algorithmus von STOER zur Auswertung einer rationalen Interpolationsfunktion so, dass man neben den Feldern der Länge n+1 für die x_i bzw. f_i nur noch ein Feld der Länge n+1 für die $T_{i,k}$ und zwei reelle Hilfsvariablen benötigt.

7. Es sei $\Phi^{m,n}$ die rationale Interpolationsfunktion zu den Stützpunkten

$$(x_0, f_0), \ldots, (x_{m+n}, f_{m+n})$$

. Man zeige, dass $\Phi^{m,n}$ sich wie folgt darstellen lässt:

$$\Phi^{m,n}(x) = \frac{|f_k, x_k - x, \ldots, (x_k - x)^m, (x_k - x)f_k, \ldots, (x_k - x)^n f_k|_{k=0}^{m+n}}{|1, x_k - x, \ldots, (x_k - x)^m, (x_k - x)f_k, \ldots, (x_k - x)^n f_k|_{k=0}^{m+n}}.$$

Dabei steht $|\alpha_k, \ldots, \xi_k|_{k=0}^{m+n}$ für

$$det \begin{bmatrix} \alpha_0 & \cdots & \xi_0 \\ \vdots & & \vdots \\ \alpha_{m+n} & \cdots & \xi_{m+n} \end{bmatrix}$$

Hinweise:

(a) Es ist zu zeigen, dass $\Phi^{m,n}(x_i) = f_i$ für $i = 0, 1, \ldots, m+n$.

(b) Es ist zu zeigen, dass der Grad des Zähler- bzw. Nennerpolynoms von $\Phi^{m,n}$ höchstens m bzw. n ist.

8. Man berechne für die Stützpunkte

x_i	0	1	-1	2	-2
f_i	1	3	$\frac{3}{5}$	3	$\frac{3}{5}$

die inversen oder reziproken Differenzen und bestimme damit den rationalen Ausdruck $\Phi^{2,2}(x)$ mit Zählergrad = Nennergrad = 2 und $\Phi^{2,2}(x_i) = f_i$ in der Form eines Kettenbruches. Man berechne daraus auch das Zähler- und das Nennerpolynom.

9. $E_{\Delta,f}(x)$ bezeichne denjenigen Spline, der zu einer Zerlegung

$$\Delta = \{a = x_0 < x_1 < \cdots < x_{N-1} < x_N = b\}$$

des Intervalls $[a,b]$ und zu gegebenen $\lambda_i, i = 0,\ldots,N-1$ das Funktional

$$F[y] = \sum_{i=0}^{N-1} \int_{x_i}^{x_{i+1}} \left[y''(x)^2 + \lambda_i^2 y'(x)^2 \right] dx$$

über $C^2[a,b]$ minimiert.

(a) Man zeige:
$E_{\Delta,f}$ hat intervallweise die Darstellung

$$E_{\Delta,f}(x) = \alpha_i + \beta_i(x-x_i) + \gamma_i \psi_i(x-x_i) + \delta_i \varphi_i(x-x_i)$$

für $x_i \leq x \leq x_{i+1},\ i = 0,\ldots,N-1$,

$$\psi_i(x) = \frac{2\left[\cosh(\lambda_i x) - 1\right]}{\lambda_i^2},$$

$$\varphi_i(x) = \frac{6\left[\sinh(\lambda_i x) - \lambda_i x\right]}{\lambda_i^3},$$

$\alpha_i,\ \beta_i, \gamma_i$ und δ_i Konstanten. $E_{\Delta,f}$ bezeichnet man als Exponentialspline.

(b) Was erhält man im Grenzfall $\lambda_i \longrightarrow 0$?

Hinweis: Man gehe ähnlich wie im Beweis der entsprechenden Extremaleigenschaft von Polynomialsplines ungeraden Grades vor und überlege sich, wie die fehlenden Bedingungen zu wählen sind.

10. Für das Intervall $[a,b]$ sei durch $h = (b-a)/n$ und $x_\nu = a + \nu h$ für $\nu = -3,-2,\ldots,n+3$ eine Zerlegung gegeben. B_ν $(\nu = -3,\ldots,n-1)$ bezeichne die kubische Spline-Funktion mit folgenden Eigenschaften:

(a)

$$B_\nu(x) \equiv 0 \qquad \text{für} \qquad x \leq x_\nu \quad \text{und} \quad x \geq x_{\nu+4}$$

(b)

$$\int_{x_\nu}^{x_{\nu+4}} B_\nu(x)\, dx = 1$$

Man berechne $B_\nu(x)$!

Chapter 3

Integration

3.1. Einführung

3.1.1. Aufgabenstellung und grundlegende Begriffe

Wir wollen uns in diesem Abschnitt mit dem näherungsweisen Berechnen bestimmter Integrale der Form

$$I(f) = \int_a^b \omega(x) f(x)\, dx$$

mit endlichen Grenzen a, b beschäftigen. Für die Gewichtsfunktion $\omega(x)$ gelte:

1. $\omega : [a,b] \longrightarrow \mathbb{R}_+ = \left\{ x \in \mathbb{R} \mid x \geqq 0 \right\}$,
2. $\omega \in L^1[a,b]$,
3. ω besitzt auf $[a,b]$ nur endlich viele Nullstellen.

Man benötigt numerische Integrationsverfahren nicht nur in den Fällen, wo die Funktionen nicht elementar integrierbar sind (z.B. $\omega(x) \equiv 1$ und $f(x) = e^x/x$ oder $f(x) = e^{-x^2}$ oder $f(x) = 1/\ln x$ oder ähnliches). Oft ist die Funktion f selbst nicht als analytischer Ausdruck, sondern nur punktweise gegeben. In diesem Falle bleibt nur die numerische Methode zur Integralberechnung. Ziel wird es sein, unter Verwendung von möglichst wenig Funktionswerten möglichst gute Näherungen von $I(f)$ in der Form

$$I(f) \approx \sum_{i=0}^{n} w_i f(x_i)$$

zu erhalten. Hier und im folgenden sei

$$\Delta^n = \{x_0, \ldots, x_n\}, \quad a = x_0 < x_1 < \cdots < x_n = b$$

eine Zerlegung des Intervalls $[a,b]$. Eine Näherungsfunktion Q_n für das bestimmte Integral $I(f)$ in der Form

$$Q_n(f) = \sum_{i=0}^{n} w_i f(x_i)$$

bezeichnen wir als **Quadraturformel**, die Daten w_i, $i = 0,1,\ldots,n$, heißen **Gewichte**. Die Aufgabe besteht darin, Stützstellen und Gewichte so zu bestimmen, dass $Q_n(f)$ eine gute Näherung von $I(f)$ für möglichst viele Funktionen f ist. Dabei dürfen die Stützstellen vorgegeben sein (z.B. äquidistant), um die Gewichte zu ermitteln. Zu festen Gewichten können geeignete Stützstellen gesucht werden; oder man bestimmt Stützstellen und Gewichte simultan so, dass Quadraturformeln hoher Genauigkeit entstehen. Dazu ist zunächst ein Maß für die Genauigkeit einer Quadraturformel festzulegen.
Eine Quadraturformel Q_n

$$Q_n(f) = \sum_{i=0}^{n} w_i f(x_i)$$

zum näherungsweisen Berechnen des bestimmten Integrals $I(f)$ heißt **exakt** vom Grade m (m-exakt), falls die Quadraturformel Q_n alle Polynome aus Π_m exakt integriert. Die größte dieser Zahlen m heißt **Exaktheitsgrad** der Quadraturformel. Man erkennt sofort, dass der Exaktheitsgrad einer Quadraturformel beschränkt ist.

3.1. Satz: *Der Exaktheitsgrad einer Quadraturformel Q_n ist höchstens $2n+1$.*

Beweis: Es genügt, ein Polynom vom Grad $2n+2$ anzugeben, das von der Quadraturformel Q_n nicht exakt integriert wird. Es sei

$$p(x) = (x-x_0)^2(x-x_1)^2\cdots(x-x_n)^2 \in \Pi_{2n+2}.$$

Dann gilt offensichtlich

$$I(p) = \int_a^b \omega(x)p(x)\,dx > 0,$$

aber

$$Q_n(p) = \sum_{i=0}^{n} w_i p(x_i) = 0,$$

daher $I(p) \neq Q_n(p)$. ❀

Mit der GAUSSschen Integrationsmethode werden wir im Abschnitt 3.3. maximal exakte Quadraturformeln kennenlernen.

3.1.2. Die Peano'sche Darstellung des Quadraturfehlers

Neben dem Exaktheitsgrad interessiert für eine konkrete Funktion f die Größe des Fehlers

$$R_n(f) = I(f) - Q_n(f).$$

Der folgende Satz liefert uns eine Möglichkeit, Darstellungen für diesen Fehler zu finden, falls der Exaktheitsgrad der Quadraturformel bekannt ist.

3.2. PEANO: *Es sei* $f \in C^{l+1}[a,b]$ *mit* $0 \leq l \leq m$ *und* Q_n *mit*

$$Q_n(f) = \sum_{i=0}^{n} w_i f(x_i)$$

eine m*-exakte Quadraturformel zum näherungsweisen Berechnen des bestimmten Integrals* $I(f)$*. Dann gilt*

$$R_n(f) = \int_a^b f^{(l+1)}(t) K_l(t)\, dt$$

mit

$$K_l(t) = \frac{1}{l!} R_n\left[(x-t)_+^l\right] = \frac{1}{l!}\left[\int_a^b \omega(t)(x-t)_+^l\, dt - \sum_{i=0}^{n} w_i (x_i - t)_+^l\right].$$

Beweis: Die Funktion $K_l(t)$ heißt PEANO-Kern von R_n. Die TAYLOR-Entwicklung der Funktion f lautet

$$f(x) = \sum_{i=0}^{l} \frac{f^{(i)}(a)}{i!}(x-a)^i + r_l(x)$$

mit dem Restglied

$$r_l(x) = \frac{1}{l!}\int_a^x f^{(l+1)}(t)(x-t)^l\,dt = \frac{1}{l!}\int_a^b f^{(l+1)}(t)(x-t)_+^l\,dt.$$

Wendet man das Funktional R_n auf die TAYLOR-Entwicklung von f an, so erhält man

$$\begin{aligned}
R_n(f) &= R_n(r_l)\\
&= \frac{1}{l!}R_n\left(\int_a^b f^{(l+1)}(t)(x-t)_+^l\,dt\right)\\
&= \frac{1}{l!}\left[\int_a^b \omega(x)\int_a^b f^{(l+1)}(t)(x-t)_+^l\,dt\,dx - \sum_{i=0}^n w_i\int_a^b f^{(l+1)}(t)(x_i-t)_+^l\,dt\right]\\
&= \frac{1}{l!}\int_a^b f^{(l+1)}(t)\left[\int_a^b \omega(x)(x-t)_+^l\,dx - \sum_{i=0}^n w_i(x_i-t)_+^l\right]dt\\
&= \int_a^b f^{(l+1)}(t)\frac{1}{l!}R_n\left[(x-t)_+^l\right]dt = \int_a^b f^{(l+1)}(t)K_l(t)\,dt.
\end{aligned}$$

❀

Mit Hilfe dieses Satzes erhalten wir sofort eine Abschätzung für den Quadraturfehler

$$|R_n(f)| \leqq \max_{x\in[a,b]}|f^{(l+1)}(x)|\cdot\int_a^b |K_l(t)|\,dt.$$

Hat der PEANO-Kern K_m auf $[a,b]$ ein konstantes Vorzeichen, so lässt sich auf die PEANOsche Restglieddarstellung der Mittelwertsatz der Integralrechnung anwenden. Man erhält

$$R_n(f) = f^{(m+1)}(\xi)\int_a^b K_m(t)\,dt, \quad \xi\in[a,b].$$

Setzt man speziell $f(x) = x^{m+1}$, so folgt

$$R_n(x^{m+1}) = (m+1)!\int_a^b K_m(t)\,dt,$$

daher

$$\int_a^b K_m(t)\,dt = \frac{R_n(x^{m+1})}{(m+1)!}.$$

Damit gilt der folgende Satz.

3.3. Satz: *Es seien $f \in C^{m+1}[a,b]$ und Q_n eine m-exakte Quadraturformel mit einem* PEANO*-Kern, der ein konstantes Vorzeichen auf dem Intervall $[a,b]$ besitzt.*
Dann existiert ein $\xi \in [a,b]$ mit

$$R_n(f) = \frac{f^{(m+1)}(\xi)}{(m+1)!} R_n(x^{m+1}).$$

Der Quadraturfehler ist daher beschränkt, falls die Funktion f hinreichend oft differenzierbar ist.

3.4. Beispiel: Wir betrachten die folgende einfache Quadraturformel:

$$I(f) = \int_a^b f(x)\,dx \approx (b-a) f\left(\frac{a+b}{2}\right) = Q(f).$$

Das ist die sogenannte Mittelpunkt- oder Rechteckregel. Zuerst wollen wir den Exaktheitsgrad dieser Quadraturformel ermitteln. Dafür setzen wir nacheinander Polynome $x^0, x^1, x^2, \ldots$ in die Quadraturformel ein und überprüfen, ob sich das exakte Integral ergibt.

- $f(x) = x^0 = 1$

$$R(1) = I(1) - Q(1) = \int_a^b 1\,dx - (b-a)\cdot 1 = (b-a) - (b-a) = 0.$$

- $f(x) = x^1 = x$

$$R(x) = I(x) - Q(x) = \int_a^b x\,dx - (b-a)\frac{b+a}{2} = \frac{b^2-a^2}{2} - \frac{b^2-a^2}{2} = 0.$$

- $f(x) = x^2$

$$\begin{aligned} R(x^2) &= I(x^2) - Q(x^2) = \int_a^b x^2\,dx - (b-a)\left(\frac{b+a}{2}\right)^2 \\ &= \frac{b^3-a^3}{3} - (b-a)\frac{b^2+2ab+a^2}{4} = \frac{b^3-a^3}{3} - \frac{b^3+ab^2-a^2b-a^3}{4} \\ &= \frac{b^3-3ab^2+3a^2b-a^3}{12} = \frac{(b-a)^3}{12} \neq 0. \end{aligned}$$

Der Exaktheitsgrad der Rechteckregel ist also $m = 1$.
Nach Satz 3.2 erhalten wir für das Restglied der Rechteckregel die Darstellung

$$R(f) = \int_a^b f''(t) K_1(t)\,dt,$$

die für alle Funktionen $f \in C^2[a,b]$ gilt. Für den PEANO-Kern K_1 ergibt sich

$$\begin{aligned} K_1(t) &= R[(x-t)_+] \\ &= \int_a^b (x-t)_+\,dx - (b-a)\left(\frac{a+b}{2} - t\right)_+ \\ &= \int_t^b (x-t)\,dx - (b-a)\left(\frac{a+b}{2} - t\right)_+ \\ &= \frac{1}{2}(x-t)^2\Big|_t^b - (b-a)\left(\frac{a+b}{2} - t\right)_+ \\ &= \frac{1}{2}(b-t)^2 - (b-a)\left(\frac{a+b}{2} - t\right)_+. \end{aligned}$$

Wir haben zwei Fälle zu unterscheiden:

- $a \leqq t \leqq \frac{a+b}{2}$

$$\begin{aligned} K_1(t) &= \frac{1}{2}(b-t)^2 - (b-a)\left(\frac{a+b}{2} - t\right) \\ &= \frac{1}{2}\left[b^2 - 2bt + t^2 - b^2 + a^2 + 2bt - 2at\right] \\ &= \frac{1}{2}\left[a^2 - 2at + t^2\right] = \frac{(a-t)^2}{2} \geqq 0 \end{aligned}$$

- $\frac{a+b}{2} < t \leqq b$

$$K_1(t) = \frac{1}{2}(b-t)^2 \geqq 0.$$

Der PEANO-Kern ändert daher nicht sein Vorzeichen auf dem Intervall $[a,b]$. Damit gilt

$$R(f) = \frac{f''(\xi)}{2} R(x^2) = \frac{f''(\xi)}{2} \cdot \frac{(b-a)^3}{12} = f''(\xi)\frac{(b-a)^3}{24}.$$

♡

3.1.3. Asymptotische Exaktheit von Quadraturformeln

Neben Fehlerabschätzungen für Quadraturformeln interessiert auch das Verhalten des Quadraturfehlers bei zunehmender Anzahl von Stützstellen. Dazu sind einige Begriffe einzuführen.
Es seien $\{\Delta^n\}_{n\in\mathbb{N}} = \{\{x_0^{(n)}, \ldots, x_n^{(n)}\}\}_{n\in\mathbb{N}}$ eine Zerlegungsfolge des Intervalls $[a,b]$, $\{w^{(n)}\}_{n\in\mathbb{N}} = \{\{w_0^{(n)}, \ldots, w_n^{(n)}\}\}$ eine Gewichtsfolge und $\{Q_n\}_{n\in\mathbb{N}}$ eine Folge von Quadraturformeln mit

$$Q_n(f) = \sum_{i=0}^{n} w_i^{(n)} f(x_i^{(n)}).$$

Die Folge $\{Q_n\}_{n\in\mathbb{N}}$ heißt **Quadraturverfahren**. Eine Quadraturformel Q_n heißt **asymptotisch exakt**, falls das entsprechende Quadraturverfahren $\{Q_n\}_{n\in\mathbb{N}}$ für jede Funktion $f \in C[a,b]$ gegen $I(f)$ konvergiert:

$$\lim_{n\to\infty} Q_n(f) = I(f).$$

Ziel ist es nun, notwendige und hinreichende Bedingung für die asymptotische Exaktheit von Quadraturformeln anzugeben. Als Hilfe benötigen wir dazu den folgenden Satz.

3.5. WEIERSTRASS**scher Approximationssatz:** *Es sei $f \in C[a,b]$. Dann existiert zu jedem $\varepsilon > 0$ ein $n \in \mathbb{N}$ und ein Polynom $p \in \Pi_n$, so dass*

$$\|f-p\|_\infty = \max_{x\in[a,b]} |f(x)-p(x)| < \varepsilon$$

ist.

Beweis: Da jedes endliche Intervall $[a,b]$ mittels der linearen Transformation $x = a + (b-a)\xi$ auf das Intervall $[0,1]$ abbildbar ist, beschränken wir uns im Beweis auf die Betrachtung des Intervalls $[0,1]$. Wir zeigen nun, dass die Folge der BERNSTEIN-Polynome

$$(B_n f)(x) = \sum_{i=0}^{n} f\left(\frac{i}{n}\right) \binom{n}{i} x^i (1-x)^{n-i}, \quad n = 0,1,\ldots$$

auf dem Intervall $[0,1]$ gleichmäßig gegen die Funktion f konvergiert.
Zuerst stellen wir fest, dass $(B_n f)(0) = 0$ und $(B_n f)(1) = 1$ für alle $n \in \mathbb{N}$ gilt. Weiterhin ist für alle n

$$1 = [x + (1-x)]^n = \sum_{i=0}^{n} \binom{n}{i} x^i (1-x)^{n-i} = \sum_{i=0}^{n} q_{ni}(x).$$

Damit folgt

$$\begin{aligned} f(x) - (B_n f)(x) &= f(x) \sum_{i=0}^{n} q_{ni}(x) - \sum_{i=0}^{n} f\left(\frac{i}{n}\right) q_{ni}(x) \\ &= \sum_{i=0}^{n} \left[f(x) - f\left(\frac{i}{n}\right) \right] q_{ni}(x) \end{aligned}$$

Wir erhalten so für alle $x \in [0,1]$ die Abschätzung

$$|f(x) - (B_n f)(x)| \leqq \sum_{i=0}^{n} \left| f(x) - f\left(\frac{i}{n}\right) \right| q_{ni}(x).$$

Aus der Stetigkeit von f folgt, dass zu gegebenem $\varepsilon > 0$ ein $\delta > 0$ derart existiert, dass

$$\left| f(x) - f\left(\frac{i}{n}\right) \right| \leqq \frac{\varepsilon}{2}$$

für alle i gilt, die $\left|x - \frac{i}{n}\right| \leqq \delta$ erfüllen. Wir definieren nun die Indexmengen

$$\begin{aligned} N_1(x) &= \left\{ i \in \{0,1,\ldots,n\} \;\middle|\; \left| x - \frac{i}{n} \right| < \delta \right\} \\ N_2(x) &= \left\{ i \in \{0,1,\ldots,n\} \;\middle|\; \left| x - \frac{i}{n} \right| \geqq \delta \right\} = \{0,1,\ldots,n\} \setminus N_1(x). \end{aligned}$$

Bezüglich dieser Indexmengen zerlegen wir die obige Summe in zwei Teilsummen:

$$\sum_{i=0}^{n}\left|f(x)-f\left(\frac{i}{n}\right)\right|q_{ni}(x)=\sum_{i\in N_1(x)}\left|f(x)-f\left(\frac{i}{n}\right)\right|q_{ni}(x)+$$

$$+\sum_{i\in N_2(x)}\left|f(x)-f\left(\frac{i}{n}\right)\right|q_{ni}(x).$$

Für die erste Summe gilt

$$\sum_{i\in N_1(x)}\left|f(x)-f\left(\frac{i}{n}\right)\right|q_{ni}(x)<\frac{\varepsilon}{2}\sum_{i\in N_1(x)}q_{ni}(x)\leq\frac{\varepsilon}{2}\sum_{i=0}^{n}q_{ni}(x)=\frac{\varepsilon}{2}.$$

Für die zweite Summe erhalten wir

$$\begin{aligned}\sum_{i\in N_2(x)}\left|f(x)-f\left(\frac{i}{n}\right)\right|q_{ni}(x) &\leq \sum_{i\in N_2(x)}\left|f(x)-f\left(\frac{i}{n}\right)\right|q_{ni}(x)\frac{(x-\frac{i}{n})^2}{\delta^2}\\ &\leq \frac{1}{\delta^2}\sum_{i\in N_2(x)}\left[|f(x)|+\left|f\left(\frac{i}{n}\right)\right|\right]q_{ni}(x)(x-\frac{i}{n})^2\\ &\leq \frac{2M}{\delta^2}\sum_{i\in N_2(x)}q_{ni}(x)(x-\frac{i}{n})^2\\ &\leq \frac{2M}{\delta^2}\sum_{i=0}^{n}q_{ni}(x)(x-\frac{i}{n})^2\end{aligned}$$

mit $M=\max\limits_{x\in[0,1]}|f(x)|$.

Die letzte Summe in der Abschätzung lässt sich weiter umformen:

$$\sum_{i=0}^{n}q_{ni}(x)(x-\frac{i}{n})^2=x^2\sum_{i=0}^{n}q_{ni}(x)-2x\sum_{i=0}^{n}\frac{i}{n}q_{ni}(x)+\sum_{i=0}^{n}\frac{i^2}{n^2}q_{ni}(x).$$

Im Einzelnen ergibt sich

$$x^2\sum_{i=0}^{n}q_{ni}(x) \;=\; x^2.$$

$$\begin{aligned}
2x\sum_{i=0}^{n}\frac{i}{n}q_{ni}(x) &= 2x\sum_{i=0}^{n}\frac{i}{n}\binom{n}{i}x^i(1-x)^{n-i}\\
&= 2x\sum_{i=0}^{n}\frac{i}{n}\frac{n!}{i!(n-i)!}x^i(1-x)^{n-i}\\
&= 2x^2\sum_{i=1}^{n}\frac{(n-1)!}{(i-1)!(n-i)!}x^{i-1}(1-x)^{(n-1)-(i-1)}\\
&= 2x^2\sum_{i=0}^{n-1}\binom{n-1}{i}x^i(1-x)^{(n-1)-i} = 2x^2.
\end{aligned}$$

$$\begin{aligned}
\sum_{i=0}^{n}\frac{i^2}{n^2}q_{ni}(x) &= \sum_{i=0}^{n}\frac{i^2}{n^2}\binom{n}{i}x^i(1-x)^{n-i}\\
&= \sum_{i=0}^{n}\frac{i^2}{n^2}\frac{n!}{i!(n-i)!}x^i(1-x)^{n-i}\\
&= x\sum_{i=1}^{n}\frac{i}{n}\frac{(n-1)!}{(i-1)!(n-i)!}x^{i-1}(1-x)^{(n-1)-(i-1)}\\
&= \frac{x}{n}\left[\sum_{i=2}^{n}(i-1)\frac{(n-1)!}{(i-1)!(n-i)!}x^{i-1}(1-x)^{(n-1)-(i-1)}+\right.\\
&\qquad\left.+\sum_{i=1}^{n}\binom{n-1}{i-1}x^{i-1}(1-x)^{(n-1)-(i-1)}\right]\\
&= \frac{x^2}{n}\sum_{i=2}^{n}\frac{(n-1)(n-2)!}{(i-2)!(n-i)!}x^{i-2}(1-x)^{(n-2)-(i-2)}+\\
&\qquad+\frac{x}{n}\sum_{i=0}^{n-1}\binom{n-1}{i}x^i(1-x)^{(n-1)-i}\\
&= \frac{(n-1)x^2}{n}\sum_{i=0}^{n-2}\binom{n-2}{i}x^i(1-x)^{(n-2)-(i-2)}+\frac{x}{n}\\
&= x^2(1-\frac{1}{n})+\frac{x}{n} = x^2+\frac{x(1-x)}{n}.
\end{aligned}$$

Somit ergibt sich

$$\sum_{i=0}^{n}q_{ni}(x)(x-\frac{i}{n})^2 = x^2-2x^2+x^2+\frac{x(1-x)}{n} = \frac{x(1-x)}{n} \leq \frac{1}{4n}$$

und weiter

$$\sum_{i\in N_2(x)} \left| f(x) - f\left(\frac{i}{n}\right) \right| q_{ni}(x) \leqq \frac{2M}{\delta^2}\frac{1}{4n} = \frac{M}{2n\delta^2} < \frac{\varepsilon}{2}$$

falls $n > M/(\varepsilon\delta^2)$ gewählt wird. Damit existiert zu jedem $\varepsilon > 0$ ein $n_0 (= [M/(\varepsilon\delta^2)])$ derart, dass

$$|f(x) - (B_n f)(x)| < \frac{\varepsilon}{2} + \frac{\varepsilon}{2} = \varepsilon$$

für alle $n > n_0$ gilt. Die BERNSTEIN-Polynome $B_n f$ konvergieren damit auf dem Intervall $[0,1]$ gleichmäßig gegen die Funktion f. ❀

Nun sind wir in der Lage, eine Aussage über die asymptotische Exaktheit von Quadraturverfahren zu machen.

3.6. Satz: *Es sei $f \in C[a,b]$ und Q_n mit*

$$Q_n(f) = \sum_{i=0}^{n} w_i^{(n)} f(x_i^{(n)})$$

eine Quadraturformel zum näherungsweisen Berechnen des bestimmten Integrals

$$I(f) = \int_a^b \omega(x) f(x)\, dx.$$

Die Quadraturformel ist asymptotisch exakt, falls sie für alle Polynome exakt ist und die absolute Gewichtssummenfolge gleichmäßig beschränkt ist.

Beweis: Es sei $f \in C[a,b]$, $\varepsilon > 0$ und $W = \int_a^b \omega(x)\, dx$. Nach Satz 3.5 existiert ein Polynom p mit

$$\|f - p\|_\infty \leqq \frac{\varepsilon}{2(K+W)}.$$

Wegen der Voraussetzung gibt es zu p ein $n_0(\varepsilon)$ mit

$$|Q_n(p) - I(p)| < \frac{\varepsilon}{2} \quad \forall n \geqq n_0(\varepsilon).$$

Damit gilt für alle $n \geqq n_0(\varepsilon)$

$$\begin{aligned}
|Q_n(f)-I(f)| &= |Q_n(f)-Q_n(p)+Q_n(p)-I(p)+I(p)-I(f)| \\
&\leqq |Q_n(f)-Q_n(p)|+|Q_n(p)-I(p)|+|I(p)-I(f)| \\
&\leqq \sum_{i=0}^{n} |w_i^{(n)}| \left| f(x_i^{(n)})-p(x_i^{(n)}) \right| + \frac{\varepsilon}{2} + \int_a^b \omega(x)|p(x)-f(x)|\,dx \\
&\leqq \frac{\varepsilon}{2(K+W)} \sum_{i=0}^{n} |w_i^{(n)}| + \frac{\varepsilon}{2} + \frac{\varepsilon}{2(K+W)} \int_a^b \omega(x)\,dx \\
&\leqq \frac{\varepsilon K}{2(K+W)} + \frac{\varepsilon}{2} + \frac{\varepsilon W}{2(K+W)} = \varepsilon.
\end{aligned}$$

❀

Bemerkung: Die Bedingungen des Satzes sind auch notwendig für asymptotische Exaktheit von Quadraturformeln. Als Folgerung aus Satz 3.6 erhalten wir sofort den folgenden Satz.

3.7. Satz: *Es sei $f \in C[a,b]$ und Q_n mit*

$$Q_n(f) = \sum_{i=0}^{n} w_i^{(n)} f(x_i^{(n)})$$

eine Quadraturformel zum näherungsweisen Berechnen des bestimmten Integrals

$$I(f) = \int_a^b \omega(x) f(x)\,dx$$

gegeben. Die Quadraturformel ist asymptotisch exakt, falls sie für alle Polynome exakt ist und alle Gewichte nichtnegativ sind.

Beweis: Es gilt

$$Q_n(1) = \sum_{i=0}^{n} w_i^{(n)} = \sum_{i=0}^{n} |w_i^{(n)}|.$$

Da nach Voraussetzung die Folge $\{Q_n(1)\}_{n\in\mathbb{N}}$ konvergiert, ist sie auch beschränkt. Damit gilt

$$\exists K > 0\, \forall n \in \mathbb{N}\, K > Q_n(1) = \sum_{i=0}^{n} |w_i^{(n)}|.$$

Mit Satz 3.6 folgt sofort die Exaktheit der Quadraturformel für alle $f \in C[a,b]$. ❀

Dieser Satz erklärt die besondere Bedeutung von Quadraturformeln mit positiven Gewichten.

3.2. Die Newton-Cotes-Formeln

3.2.1. Die geschlossenen Newton-Cotes-Formeln

Betrachten wir nun die Aufgabe, zu vorgegebenen Stützstellen, eine Quadraturformel mit möglichst hohem Exaktheitsgrad zu konstruieren. Für den Fall $\omega(x) \equiv 1$ und äquidistante Stützstellen erhält man die sogenannten NEWTON-COTES-Formeln. Die Vorgehensweise zu ihrer Konstruktion ist dabei einfach und einleuchtend. Wir ersetzen den Integranden f durch ein Interpolationspolynom, das an den Stützstellen x_i die Werte $f_i = f(x_i)$ annimmt. Das bestimmte Integral über dieses Interpolationspolynom liefert uns dann eine Näherung für das zu berechnende Integral. Mit Hilfe der LAGRANGEschen Darstellung des Interpolationspolynoms erhält man

$$f(x) = \sum_{i=0}^{n} f_i L_i^{(n)}(x) + r_n(x)$$

mit

$$L_i^{(n)}(x) = \prod_{\substack{j=0 \\ j \neq i}}^{n} \frac{x - x_j}{x_i - x_j}, \quad i = 0, 1, \ldots, n$$

und einem Restglied $r_n(x)$. Die bestimmte Integration über das Intervall $[a,b]$ liefert

$$\int_a^b f(x)\,dx = \int_a^b \sum_{i=0}^{n} f_i L_i^{(n)}(x)\,dx + \int_a^b r_n(x)\,dx = \sum_{i=0}^{n} w_i f_i + R_n(f)$$

$$\text{mit} \quad w_i = \int_a^b L_i^{(n)}(x)\,dx, \quad i = 0, 1, \ldots, n; \quad R_n(f) = \int_a^b r_n(x)\,dx.$$

Für die Gewichte w_i wollen wir noch eine etwas günstigere Darstellung finden. Mit den Substitutionen $x = a + sh$, $x_i = a + ih$ und $x_j = a + jh$ erhalten wir

$$\begin{aligned} w_i &= \int_a^b L_i^{(n)}(x)\,dx = \int_a^b \prod_{\substack{j=0 \\ j\neq i}}^{n} \frac{x - x_j}{x_i - x_j}\,dx \\ &= h \int_0^n \prod_{\substack{j=0 \\ j\neq i}}^{n} \frac{(a+sh) - (a+jh)}{(a+ih) - (a+jh)}\,ds = \frac{b-a}{n} \int_0^n \prod_{\substack{j=0 \\ j\neq i}}^{n} \frac{s-j}{i-j}\,ds. \end{aligned}$$

Die durch diese äquidistanten Stützstellen und die entsprechenden Gewichte gegebenen Quadraturformeln werden als geschlossene NEWTON-COTES-Formeln bezeichnet. Üblicherweise werden sie in der Form

$$I(f) = (b-a) \sum_{i=0}^{n} \sigma_i^{(n)} f(x_i^{(n)}) + R_n(f) = (b-a) \sum_{i=0}^{n} \sigma_i^{(n)} f_i^{(n)} + R_n(f)$$

mit

$$x_i^{(n)} = a + i\frac{b-a}{n} = a + ih$$

und

$$\sigma_i^{(n)} = \frac{1}{n} \int_0^n \prod_{\substack{j=0 \\ j\neq i}}^{n} \frac{s-j}{i-j}\,ds, \quad i = 0, 1, \ldots, n$$

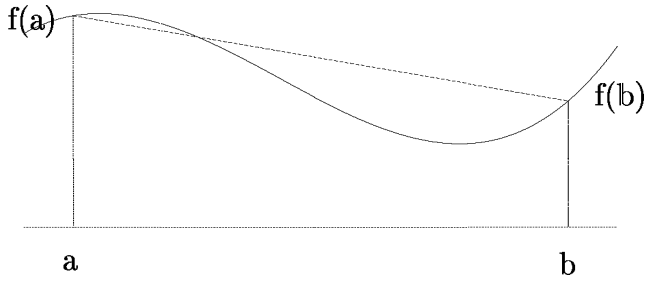
angegeben.

3.8. Beispiel: Betrachten wir den einfachsten Fall $n = 1$. Die Stützstellen sind dann durch $x_0 = a$ und $x_1 = b$ gegeben. Für die Gewichte erhalten wir

$$\begin{aligned} \sigma_0^{(1)} &= \int_0^1 \frac{s-1}{0-1}\,ds = -\left(\frac{s^2}{2} - s\right)\Big|_0^1 = \frac{1}{2}, \\ \sigma_1^{(1)} &= \int_0^1 \frac{s-0}{1-0}\,ds = \frac{s^2}{2}\Big|_0^1 = \frac{1}{2}. \end{aligned}$$

Das ist die sogenannte **Trapez-Regel**

$$Q_1(f) = (b-a)\left[\frac{f(a)}{2} + \frac{f(b)}{2}\right].$$



Statt der Fläche unter der Kurve wird die Trapezfläche berechnet. Offensichtlich erhält man für lineare Funktionen dabei das exakte Ergebnis. Für $f(x) = x^2$ dagegen ergibt sich

$$\begin{aligned} I(x^2) &= \int_a^b x^2\,dx = \frac{x^3}{3}\Big|_a^b = \frac{b^3-a^3}{3}, \\ Q_1(x^2) &= (b-a)\left[\frac{a^2}{2}+\frac{b^2}{2}\right] = \frac{b^3-ab^2+a^2b-a^3}{2}, \end{aligned}$$

also $Q_1(x^2) \neq I(x^2)$. Der Exaktheitsgrad der Trapez-Regel ist damit 1. Mit der PEANOschen Restglieddarstellung bestimmen wir nun den Quadraturfehler. Es gilt

$$R_1(f) = \int_a^b f(x)\,dx - \frac{b-a}{2}\left[f(a)+f(b)\right].$$

Damit erhalten wir für den PEANO-Kern

$$\begin{aligned} K_1(t) &= R_1((x-t)_+^1) = \int_a^b (x-t)_+\,dx - \frac{b-a}{2}\left[(a-t)_+ + (b-t)_+\right] \\ &= \int_t^b (x-t)\,dx - \frac{(b-a)(b-t)}{2} = \left(\frac{x^2}{2} - xt\right)_t^b - \frac{(b-a)(b-t)}{2} \\ &= \frac{b^2}{2} - bt - \frac{t^2}{2} + t^2 - \frac{b^2}{2} + \frac{ab}{2} + \frac{bt}{2} - \frac{at}{2} = \frac{t^2}{2} + \frac{ab}{2} - \frac{bt}{2} - \frac{at}{2} \\ &= \frac{1}{2}(a-t)(b-t). \end{aligned}$$

Auf dem Intervall $[a,b]$ gilt daher $K_1(t) \leqq 0$. Nach Satz 3.3 folgt dann für alle Funktionen $f \in C^2[a,b]$ die Existenz eines $\xi \in [a,b]$, mit

$$R_1(f) = \frac{f''(\xi)}{2!} R_1(x^2).$$

Weiter gilt

$$\begin{aligned} R_1(x^2) &= I(x^2) - Q_1(x^2) = \frac{b^3}{3} - \frac{a^3}{3} - \frac{b^3}{2} + \frac{ab^2}{2} - \frac{a^2b}{2} + \frac{a^3}{2} \\ &= -\frac{b^3 - 3ab^2 + 3a^2b - a^3}{6} = -\frac{(b-a)^3}{6}. \end{aligned}$$

Es ergibt sich damit die folgende Restglieddarstellung der Trapez-Regel

$$R_1(f) = -\frac{f''(\xi)}{12}(b-a)^3 = -\frac{f''(\xi)}{12}h^3.$$

♡

Die obige Tabelle enthält eine Zusammenstellung der ersten acht geschlossenen Newton-Cotes-Formeln. Man beachte dabei die Gültigkeit von $\sigma_{n-i}^{(n)} = \sigma_i^{(n)}$. Einige Formeln haben Namen:

n	$s\sigma_i^{(n)}$							s	$R_n(f)$
1	1	1						2	$-\frac{h^3}{12}f''(\xi)$
2	1	4	1					6	$-\frac{h^5}{90}f^{(4)}(\xi)$
3	1	3	3	1				8	$-\frac{3h^5}{80}f^{(4)}(\xi)$
4	7	32	12	32	7			90	$-\frac{8h^7}{945}f^{(6)}(\xi)$
5	19	75	50	50	75	19		288	$-\frac{275h^7}{12096}f^{(6)}(\xi)$
6	41	216	27	272	27	216	41	840	$-\frac{9h^9}{1400}f^{(8)}(\xi)$
7	751	3577	1323	2989	2989	1323	...	17280	$-\frac{8183h^9}{518400}f^{(8)}(\xi)$
8	989	5888	-928	10496	-4540	10496	...	28350	$-\frac{2368h^9}{467775}f^{(10)}(\xi)$

Table 3.1: Geschlossene Newton-Cotes-Formeln

Trapezregel ($n = 1$), Simpson-Regel ($n = 2$), 3/8-Regel ($n = 3$), Milne-Regel ($n = 4$) und Weddle-Regel ($n = 6$).
Ab $n = 8$ treten negative Gewichte auf. Es zeigt sich, dass für die Newton-Cotes-Formeln kein $K > 0$ mit

$$\sum_{i=0}^{n} |\sigma_i^{(n)}| < K$$

für alle $n \in \mathbb{N}$ existiert. Damit ist durch die NEWTON-COTES-Formeln keine asymptotisch exakte Quadraturformel gegeben. Meist verwendet man NEWTON-COTES-Formeln kleiner Ordnung (Trapezregel, SIMPSON-Regel) und wendet diese auf Teilintervalle von $[a,b]$ an. Wir werden noch erkennen, dass sich damit asymptotische exakte Quadraturformeln ergeben.

3.2.2. Die offenen Newton-Cotes-Formeln

Neben den geschlossenen NEWTON-COTES-Formeln gibt es die offenen NEWTON-COTES-Formeln. Diese entstehen, falls man bei der Interpolation des Integranden die Randpunkte nicht als Stützstellen nutzt. Man erhält so

$$I(f) = \sum_{i=1}^{n-1} f_i \int_a^b \bar{L}_i^{(n)}(x)\,dx + \bar{R}_n(f) = (b-a)\sum_{i=1}^{n-1} \bar{\sigma}_i^{(n)} f_i + \bar{R}_n(f)$$

mit

$$\bar{L}_i^{(n)}(x) = \prod_{\substack{j=1 \\ j\neq i}}^{n-1} \frac{x-x_j}{x_i-x_j}, \quad \bar{\sigma}_i^{(n)} = \frac{1}{n}\int_0^n \prod_{\substack{j=1 \\ j\neq i}}^{n-1} \frac{s-j}{i-j}\,ds.$$

In der folgenden Tabelle sind wieder die ersten Formeln angegeben.

n	$s\bar{\sigma}_i^{(n)}$							s	$\bar{R}_n(f)$
2	1							1	$\frac{h^3}{3}f''(\xi)$
3	1	1						2	$\frac{h^3}{4}f''(\xi)$
4	2	-1	2					3	$\frac{14h^5}{45}f^{(4)}(\xi)$
5	11	1	1	11				24	$\frac{95h^5}{144}f^{(4)}(\xi)$
6	11	-14	26	-14	11			20	$\frac{41h^7}{140}f^{(6)}(\xi)$
7	611	-453	562	562	-453	611		1440	$\frac{5267h^7}{8640}f^{(6)}(\xi)$
8	460	-954	2196	-2459	2196	-954	460	945	$\frac{3956h^9}{14175}f^{(8)}(\xi)$

Table 3.2: Offene NEWTON-COTES-Formeln

Hierin heißt die erste Methode Rechteck-Regel.

3.2.3. Zusammengesetzte Newton-Cotes-Formeln

Wie in 3.2.1. bemerkt wurde, sind die geschlossenen NEWTON-COTES-Formeln nicht asymptotisch exakt. Es ist günstiger, mit Formeln niedrigen Grades zu arbeiten, diese aber auf kleinere Teilintervalle anzuwenden. Wir teilen das Intervall $[a,b]$ in N äquidistante Teilintervalle ein. Dem entspricht eine Zerlegung des bestimmten Integrals $I(f)$ in eine Summe von Integralen über den einzelnen Teilintervallen:

$$I(f) = \int_a^b f(x)\,dx = \sum_{i=0}^{N-1} \int_{x_i}^{x_{i+1}} f(x)\,dx$$

mit

$$x_i = a + ih, \quad i = 0,1,\ldots,N, \quad h = \frac{b-a}{N}.$$

Zum Berechnen der Integrale auf $[x_i, x_{i+1}]$ wenden wir einfache NEWTON-COTES-Formeln an. Mit der Trapezregel erhält man für $f \in C^2[a,b]$

$$\begin{aligned} I(f) &= \sum_{i=0}^{N-1} \left\{ (x_{i+1}-x_i)\left[\frac{f(x_i)}{2} + \frac{f(x_{i+1})}{2}\right] - \frac{f''(\xi_i)}{12}(x_{i+1}-x_i)^3 \right\} \\ &= \frac{h}{2}\sum_{i=0}^{N-1}[f(x_i)+f(x_{i+1})] - \frac{h^3}{12}\sum_{i=0}^{N-1} f''(\xi_i) \\ &= \frac{h}{2}\left[f(a) + 2\sum_{i=1}^{N-1} f(x_i) + f(b)\right] - \frac{h^2(b-a)}{12}\frac{1}{N}\sum_{i=0}^{N-1} f''(\xi_i), \end{aligned}$$

wobei jeweils $\xi_i \in [x_i, x_{i+1}]$ gilt. Das Restglied lässt sich noch einfacher angeben. es gilt

$$\min_{0 \leqq i \leqq N-1} f''(\xi_i) \leqq \frac{1}{N}\sum_{i=0}^{N-1} f''(\xi_i) \leqq \max_{0 \leqq i \leqq N-1} f''(\xi_i).$$

Nach dem Zwischenwertsatz für stetige Funktionen existiert dann ein $\xi \in [a,b]$ mit

$$f''(\xi) = \frac{1}{N}\sum_{i=0}^{N-1} f''(\xi_i).$$

Wir erhalten so die **zusammengesetzte Trapezregel** oder **Trapezsumme**

$$I(f) = T(f;h) - \frac{h^2(b-a)}{12} f''(\xi),$$

mit

$$T(f;h) = h\left[\frac{1}{2}f(a) + f(a+h) + \cdots + f(b-h) + \frac{1}{2}f(b)\right].$$

Ist N gerade: $N = 2M$, und wendet man auf den Doppelintervallen $[x_{2i}, x_{2i+2}]$ für $i = 0, 1, \ldots, M-1$ jeweils die SIMPSON-Regel an, so erhält man für $f \in C^4[a,b]$

$$\begin{aligned}
I(f) &= \sum_{i=0}^{M-1} \int_{x_{2i}}^{x_{2i+2}} f(x)\,dx \\
&= \sum_{i=0}^{M-1} \left\{ (x_{2i+2} - x_{2i}) \left[\frac{f(x_{2i})}{6} + \frac{4f(x_{2i+1})}{6} + \frac{f(x_{2i+2})}{6}\right] - \frac{h^5}{90} f^{(4)}(\xi_i) \right\} \\
&= \frac{h}{3} \sum_{i=0}^{M-1} [f(x_{2i}) + 4f(x_{2i+1}) + f(x_{2i+2})] - \frac{h^4}{90} \frac{b-a}{2M} \sum_{i=0}^{M-1} f^{(4)}(\xi_i) \\
&= \frac{h}{3} \left[\sum_{i=0}^{M-1} f(x_{2i}) + 4 \sum_{i=0}^{M-1} f(x_{2i+1}) + \sum_{i=0}^{M-1} f(x_{2i+2}) \right] \\
&\qquad - \frac{(b-a)h^4}{180} \frac{1}{M} \sum_{i=0}^{M-1} f^{(4)}(\xi_i) \\
&= \frac{h}{3} \left[\sum_{i=0}^{M-1} f(x_{2i}) + 4 \sum_{i=0}^{M-1} f(x_{2i+1}) + \sum_{i=1}^{M} f(x_{2i}) \right] - \frac{(b-a)h^4}{180} f^{(4)}(\xi),
\end{aligned}$$

wobei aus der Stetigkeit von $f^{(4)}(x)$ wieder die Existenz eines $\xi \in [a,b]$ mit

$$f^{(4)}(\xi) = \frac{1}{M} \sum_{i=0}^{M-1} f^{(4)}(\xi_i)$$

folgt. Insgesamt ergibt sich die **zusammengesetzte** SIMPSON**-Regel**

$$I(f) = S(f;h) - \frac{(b-a)h^4}{180} f^{(4)}(\xi)$$

mit der SIMPSON**-Summe**

$$S(f;h) = \frac{h}{3}\left[f(a) + 2 \sum_{i=1}^{M-1} f(x_{2i}) + 4 \sum_{i=0}^{M-1} f(x_{2i+1}) + f(b)) \right].$$

(Man beachte, dass $h = (b-a)/N = (b-a)/(2M)$ gilt.)
Aus der Restglieddarstellung von $T(f;h)$ und $S(f;h)$ erkennt man sofort, dass für beliebige Polynome p

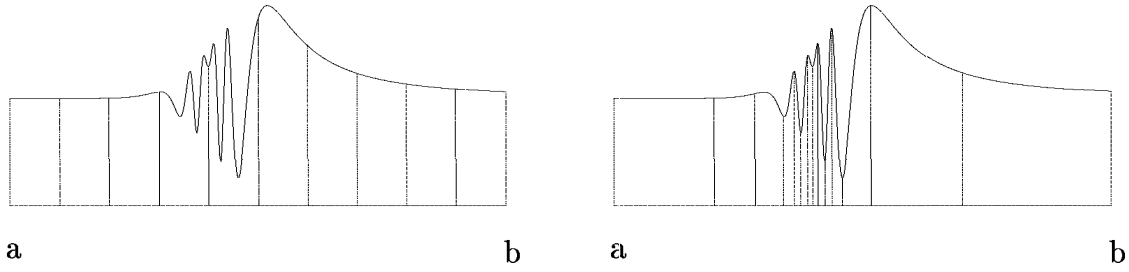
$$\lim_{N \to \infty} T(p;h) = I(p)$$

bzw.

$$\lim_{N\to\infty} S(p;h) = I(p)$$

gilt, denn p'' bzw. $p^{(4)}$ sind auf $[a,b]$ beschränkt. Damit sind wegen der Positivität der Gewichte alle Voraussetzungen von Satz 3.7 erfüllt. Die Trapezsummen bzw. SIMPSON-Summen stellen also jeweils asymptotisch exakte Quadraturformeln dar.

Bemerkung: Die äquidistante Intervalleinteilung berücksichtigt nicht den Verlauf des Integranden. Dies erkennt man an den folgenden Bildern.



In der linken Darstellung wurden äquidistante Intervalle gewählt. Im mittleren oszillierenden Bereich entsteht dadurch ein großer Integrationsfehler, während in den Randbereichen die Intervalle größer gewählt werden könnten. Eine angepasste Intervalleinteilung ist im rechten Bild zu sehen.
Im Abschnitt 6.2.5 werden wir Möglichkeiten zur Schrittweitensteuerung bei Einschrittverfahren zum Lösen von Anfangswertproblemen kennenlernen. Diese Methoden lassen sich so modifizieren, dass sie auf die numerische Integration anwendbar sind.

3.2.4. Quadraturformeln mit gleichen Gewichten

Falls das Berechnen von Funktionswerten des Integranden mit größeren Fehlern verbunden ist, möchte man, dass diese Fehler gleichmäßig in das Ergebnis eingehen, um es so gering wie möglich zu verfälschen. Dies führt auf die Forderung nach Quadraturformeln mit gleichen Gewichten. Wir suchen daher nach einer Integrationsmethode der Form

$$I(f) = \int_a^b f(x)\,dx = w^{(n)} \sum_{i=1}^{n} f(x_i) + R_n(f)$$

mit maximalem Exaktheitsgrad. Das Gewicht $w^{(n)}$ und die Stützstellen $x_1, x_2, \ldots, x_n$ sind zu bestimmen.

Die Forderung $R_n(1) = 0$ liefert

$$\int_a^b dx = b - a = w^{(n)} \sum_{i=1}^{n} 1 = n w^{(n)},$$

daher $w^{(n)} = \frac{b-a}{n}$. Bei einer Formel mit n Stützstellen ist mindestens der Exaktheitsgrad n zu erwarten. Das führt auf ein nichtlineares Gleichungssystem zur Bestimmung der x_i, $i = 1, 2, \ldots, n$:

$$\frac{b^{k+1} - a^{k+1}}{k+1} = \frac{b-a}{n} \sum_{i=1}^{n} x_i^k, \quad k = 1, 2, \ldots, n.$$

Dieses Gleichungssystem besitzt nur für $n = 1, 2, 3, 4, 5, 6, 7, 9$ ausschließlich reelle Lösungen. Für $n = 8$ und $n \geqq 10$ treten komplexe Lösungen auf. Die entsprechenden Formeln sind damit unbrauchbar.
Beschränkt man sich auf das Intervall $[a, b] = [-1, 1]$, so ergeben sich im Einzelnen folgende Stützstellen:

$$\begin{array}{lrcl}
n=1: & x_1 & = & 0, \\
n=2: & -x_1 = x_2 & = & 0.5773502692 = \frac{\sqrt{3}}{3}, \\
n=3: & -x_1 = x_3 & = & 0.7071067812 = \frac{\sqrt{2}}{2}, \\
 & x_2 & = & 0, \\
n=4: & -x_1 = x_4 & = & 0.7946544723 = \sqrt{\frac{5+2\sqrt{5}}{15}}, \\
 & -x_2 = x_3 & = & 0.1875924741 = \sqrt{\frac{5-2\sqrt{5}}{15}}, \\
n=5: & -x_1 = x_5 & = & 0.8324974870 = \sqrt{\frac{5+\sqrt{11}}{12}}, \\
 & -x_2 = x_4 & = & 0.3745414096 = \sqrt{\frac{5-\sqrt{11}}{12}}, \\
 & x_3 & = & 0, \\
n=6: & -x_1 = x_6 & = & 0.8662468181, \\
 & -x_2 = x_5 & = & 0.4225186538, \\
 & -x_3 = x_4 & = & 0.2666354015, \\
n=7: & -x_1 = x_7 & = & 0.8838617008, \\
 & -x_2 = x_6 & = & 0.5296567753, \\
 & -x_3 = x_5 & = & 0.3239118105, \\
 & x_4 & = & 0,
\end{array}$$

$$
\begin{array}{rrcccl}
n=9 & : & -x_1 & = & x_9 & = 0.9115893077, \\
 & & -x_2 & = & x_8 & = 0.6010186554, \\
 & & -x_3 & = & x_7 & = 0.5287617831, \\
 & & -x_4 & = & x_6 & = 0.1679061842, \\
 & & x_5 & = & 0. &
\end{array}
$$

Durch eine lineare Transformation lassen sich diese Stützstellen leicht auf ein beliebiges endliches Intervall $[a,b]$ übertragen.

3.3. Die Gauß'sche Integrationsmethode

In diesem Abschnitt wollen wir maximal exakte Quadraturformeln zum Berechnen des bestimmten Integrals

$$I(f) = \int_a^b \omega(x) f(x)\, dx$$

konstruieren. Wie wir später zeigen werden, sind die Stützstellen dieser Quadraturformeln gerade durch die Nullstellen gewisser Polynome gegeben. Mit diesen Polynomen werden wir uns zunächst im folgenden Unterabschnitt beschäftigen.

3.3.1. Orthogonalpolynome

Es sei

$$\tilde{\Pi}_k = \left\{ p \;\middle|\; p(x) = x^k + a_{k-1}x^{k-1} + \cdots + a_1 x + a_0 \right\}$$

die Menge aller normierten Polynome vom Grad k. Weiterhin führen wir auf dem Vektorraum $C[a,b]$ folgendes Skalarprodukt $(\cdot\,,\cdot)_\omega$ ein:

$$(f,g)_\omega = \int_a^b \omega(x) f(x) g(x)\, dx.$$

Dann gilt der folgende Satz.

3.9. Satz: *Es gibt eindeutig bestimmte Polynome $p_j \in \tilde{\Pi}_j$, $j = 0,1,2,\ldots$ mit*

$$(p_j, p_k)_\omega = 0, \quad j \neq k.$$

Die Polynome genügen der Rekursionsformel

1.

$$p_0(x) \equiv 1,$$

2.

$$p_{i+1}(x) = (x - \delta_{i+1})p_i(x) - \gamma_{i+1}^2 p_{i-1}(x), \quad i \geqq 0$$

mit

$$\begin{aligned} p_{-1} &= 0, \\ \delta_{i+1} &= \frac{(x \cdot p_i, p_i)_\omega}{(p_i, p_i)_\omega}, \\ \gamma_{i+1}^2 &= \begin{cases} 0 & \text{für } i = 0 \\ \frac{(p_i, p_i)_\omega}{(p_{i-1}, p_{i-1})_\omega} & \text{für } i \geqq 1 \end{cases}. \end{aligned}$$

Diese Polynome werden als zur Gewichtsfunktion ω gehörende **Orthogonalpolynome** *bezeichnet.*

Beweis: $p_0(x) \equiv 1$ ist offensichtlich. Wir nehmen an, dass wir schon Polynome $p_j \in \tilde{\Pi}_j$ für $j = 0, 1, \ldots, i$ konstruiert hätten. Es ist zu zeigen, dass ein eindeutig bestimmtes Polynom $p_{i+1} \in \tilde{\Pi}_{i+1}$ mit

$$(p_{i+1}, p_j)_\omega = 0, \quad j = 0, 1, \ldots, i$$

existiert, und dass dieses Polynom den angegebenen Rekursionsformeln genügt. Dazu machen wir für p_{i+1} den Ansatz

$$p_{i+1}(x) = (x - c_i)p_i(x) + c_{i-1}p_{i-1}(x) + \cdots + c_1 p_1(x) + c_0 p_0(x).$$

Damit gilt zunächst $p_{i+1} \in \tilde{\Pi}_{i+1}$. Die Parameter c_j, $j = 0, 1, \ldots, i$ werden nun so bestimmt, dass die Orthogonalitätsbedingungen

$$0 = (p_{i+1}, p_j)_\omega = (x \cdot p_i, p_j)_\omega - c_i(p_i, p_j)_\omega + \sum_{k=0}^{i-1} c_k (p_k, p_j)_\omega$$

erfüllt sind. Wir erhalten

- $j = i$

$$\begin{aligned} 0 &= (x \cdot p_i, p_i)_\omega - c_i(p_i, p_i)_\omega \\ c_i &= \frac{(x \cdot p_i, p_i)_\omega}{(p_i, p_i)_\omega} = \delta_{i+1}, \end{aligned}$$

- $j < i$

$$\begin{aligned} 0 &= (x \cdot p_i, p_j)_\omega + c_j (p_j, p_j)_\omega \\ c_j &= -\frac{(x \cdot p_i, p_j)_\omega}{(p_j, p_j)_\omega}, \quad j = 0, 1, \ldots, i-1. \end{aligned}$$

Gilt für ein $j \leqq i$ die Rekursionformel

$$p_j(x) = (x - \delta_j) p_{j-1}(x) - \gamma_j^2 p_{j-2}(x),$$

so folgt $(p_j, p_i)_\omega = (x \cdot p_{j-1}, p_i)_\omega = (x \cdot p_i, p_{j-1})_\omega$ und damit

$$c_j = -\frac{(x \cdot p_i, p_j)_\omega}{(p_j, p_j)_\omega} = -\frac{(p_{j+1}, p_i)_\omega}{(p_j, p_j)_\omega},$$

weiter

$$c_{i-1} = -\frac{(p_i, p_i)_\omega}{(p_{i-1}, p_{i-1})_\omega}, \quad c_j = 0 \quad j = 0, 1, \ldots, i-2.$$

❀

Die Polynome $p_0, p_1, \ldots, p_k$ bilden somit eine orthogonale Basis des Vektorraums $\mathbf{\Pi}_k$. Jedes Polynom $p \in \mathbf{\Pi}_k$ lässt sich in eindeutiger Weise als Linearkombination dieser Basispolynome darstellen. Es gilt

$$p = \sum_{i=0}^{k} \alpha_i p_i, \quad \alpha_i = \frac{(p, p_i)_\omega}{(p_i, p_i)_\omega}.$$

Wie schon erwähnt, spielen die Nullstellen dieser Orthogonalpolynome eine wichtige Rolle bei der Konstruktion von maximal exakten Quadraturformeln. Im folgenden Satz wird eine wichtige Eigenschaft dieser Nullstellen gezeigt.

3.10. Satz: *Die Nullstellen $x_1, x_2, \ldots, x_n$ des n-ten Orthogonalpolynoms p_n sind reell und einfach. Sie liegen im offenen Intervall (a, b).*

Beweis: O.B.d.A. seien $x_1, x_2, \ldots, x_l$ die Nullstellen, die in (a, b) liegen und an denen p_n sein Vorzeichen wechselt. Wir definieren das Polynom

$$q(x) = (x - x_1)(x - x_2) \cdots (x - x_l) \in \tilde{\mathbf{\Pi}}_l.$$

Ist nun $l < n$, so folgt $(q, p_n)_\omega = 0$. Andererseits gilt aber für alle $x \in (a, b)$

$$\omega(x) q(x) p_n(x) \geqq 0,$$

und damit

$$(q, p_n)_\omega = \int_a^b \omega(x) q(x) p_n(x)\, dx > 0.$$

Das ist ein Widerspruch, woraus $l = n$ folgt. Alle Nullstellen von p_n liegen in (a,b) und sind einfach. ❀

Der nächste Satz beschreibt eine weitere Eigenschaft der Orthogonalpolynome, die wir im folgenden Abschnitt zum Berechnen von Gewichten der GAUSSschen Quadraturformeln benötigen werden.

3.11. Satz: *Für beliebige $t_1 < t_2 < \cdots < t_n$ ist die Matrix*

$$A(t) = \begin{pmatrix} p_0(t_1) & p_0(t_2) & \dots & p_0(t_n) \\ p_1(t_1) & p_1(t_2) & \dots & p_1(t_n) \\ \vdots & \vdots & \ddots & \vdots \\ p_{n-1}(t_1) & p_{n-1}(t_2) & \dots & p_{n-1}(t_n) \end{pmatrix}$$

regulär.

Beweis: Wir zeigen, dass das Gleichungssystem $A(t)^T c = 0$ nur die triviale Lösung besitzt. Es sei also $c \in \mathbb{R}^n$. Für das Polynom

$$q(x) = c_0 p_0(x) + c_1 p_1(x) + \cdots + c_{n-1} p_{n-1}(x) \in \Pi_{n-1}$$

folgen dann aus $A(t)^T c = 0$ die Gleichungen $q(t_i) = 0$ für $i = 1, 2, \dots, n$. Damit hat das Polynom q mindestens n verschiedene Nullstellen und muss somit das Nullpolynom sein, woraus dann $c_i = 0$ für $i = 0, 1, \dots, n-1$ folgt. ❀

Die für die numerische Integration wichtigsten Orthogonalpolynome sind in der folgenden Tabelle zusammengestellt.

3.3.2. Berechnen der Stützstellen und Gewichte

Nach diesen Vorbereitungen sind wir nun in der Lage, die eigentliche GAUSSsche Integrationsmethode zu entwickeln. Wir beweisen den folgenden Satz.

3.12. Satz: *Zu jedem $n \in \mathbb{N}$ gibt es eindeutig bestimmte Zahlen x_i, w_i, $i = 1, \dots, n$, so dass*

$$\int_a^b \omega(x) p(x)\, dx = \sum_{i=1}^n w_i p(x_i)$$

$[a,b]$	$\omega(x)$	Name der Orthogonalpolynome
$[-1,1]$	1	$P_n(x)$, LEGENDRE-Polynome
$[-1,1]$	$1/\sqrt{1-x^2}$	$T_n(x)$, TSCHEBYSCHEFF-Polynome 1.Art
$[0,\infty)$	e^{-x}	$L_n(x)$, LAGUERRE-Polynome
$(-\infty,\infty)$	e^{-x^2}	$H_n(x)$, HERMITE-Polynome

Table 3.3: Orthogonalpolynome

für alle Polynome $p \in \Pi_{2n-1}$ gilt. Die x_i sind die Nullstellen des entsprechenden n-ten Orthogonalpolynoms p_n. Die w_i ergeben sich als Lösung des linearen Gleichungssystems

$$A(x)w = (p_0, p_0)_\omega e_1.$$

Dabei gilt $x = (x_1, x_2, \ldots, x_n)^T$, $w = (w_1, w_2, \ldots, w_n)^T$ und $e_1 = (1, 0, \ldots, 0)^T \in \mathbb{R}^n$. Die Matrix $A(x)$ entspricht der in Satz 3.11 definierten Matrix. Es gilt $w_i > 0$ für $i = 1, \ldots, n$.

Beweis: Wir zeigen zunächst, dass durch die im Satz angegebenen Stützstellen und Gewichte eine maximal exakte Quadraturformel gegeben ist. Anschließend zeigen wir die Eindeutigkeit der Stützstellen und Gewichte sowie die Positivität der Gewichte.
Nach Satz 3.10 liegen alle Nullstellen von p_n im offenen Intervall (a,b) und sind paarweise verschieden. Nach Satz 3.11 ist die Matrix $A(x)$ regulär und das Gleichungssystem $A(x)w = (p_0, p_0)_\omega e_1$ besitzt genau eine Lösung. Es sei nun p ein beliebiges Polynom vom Höchstgrad $2n-1$. Es lässt sich in der Form

$$p(x) = q(x)p_n(x) + r(x)$$

mit $q, r \in \Pi_{n-1}$ zerlegen. Die Polynome q und r wiederum sind als Linearkombinationen der Orthogonalpolynome $p_0, p_1, \ldots p_{n-1}$ darstellbar:

$$q(x) = \sum_{k=0}^{n-1} \alpha_k p_k(x), \quad r(x) = \sum_{k=0}^{n-1} \beta_k p_k(x).$$

Damit folgt

$$\begin{aligned}
\int_a^b \omega(x)p(x)\,dx &= \int_a^b \omega(x)\,[q(x)p_n(x)+r(x)]\,dx \\
&= \int_a^b \omega(x)q(x)p_n(x)\,dx + \int_a^b \omega(x)r(x)\,dx \\
&= \int_a^b \omega(x)q(x)p_n(x)\,dx + \int_a^b \omega(x)r(x)p_0(x)\,dx \\
&= (q,p_n)_\omega + (r,p_0)_\omega \\
&= 0 + \sum_{k=0}^{n-1} \beta_k (p_k,p_0)_\omega = \beta_0 (p_0,p_0)_\omega .
\end{aligned}$$

Andererseits ist

$$\begin{aligned}
\sum_{i=1}^{n} w_i p(x_i) &= \sum_{i=1}^{n} w_i\,[q(x_i)p_n(x_i)+r(x_i)] = \sum_{i=1}^{n} w_i r(x_i) \\
&= \sum_{i=1}^{n} w_i \sum_{k=0}^{n-1} \beta_k p_k(x_i) = \sum_{k=0}^{n-1} \beta_k \sum_{i=1}^{n} w_i p_k(x_i) \\
&= \sum_{k=0}^{n-1} \beta_k (p_0,p_0)_\omega \delta_{k0} = \beta_0 (p_0,p_0)_\omega .
\end{aligned}$$

Für die beiden letzten Schritte beachte man, dass δ_{k0} das KRONECKER-Symbol darstellt: $\delta_{k0} = 1$ für $k = 0$ und $\delta_{k0} = 0$ für $k \neq 0$, und dass die Beziehung

$$\sum_{i=1}^{n} w_i p_k(x_i) = (p_0,p_0)_\omega \delta_{k0}$$

aus dem Gleichungssystem

$$A(x)w = (p_0,p_0)_\omega e_1$$

folgt.
Damit ist gezeigt, dass durch die obigen Stützstellen und Gewichte eine Quadraturformel mit dem maximalen Exaktheitsgrad $2n-1$ gegeben ist.
Wir wollen nun die Eindeutigkeit zeigen. Es seien durch

$$\bar{x} = (\bar{x}_1,\ldots,\bar{x}_n)^T \text{ und } \bar{w} = (\bar{w}_1,\ldots,\bar{w}_n)^T$$

Stützstellen und von Null verschiedene Gewichte einer weiteren maximal exakten Quadraturformel gegeben. Dann gilt

$$\int_a^b \omega(x) p_k(x)\,dx = (p_0, p_k)_\omega = \sum_{i=1}^{n} \bar{w}_i p_k(\bar{x}_i), \quad k = 0, 1, \ldots, n-1.$$

Die Gewichte und Stützstellen erfüllen somit das Gleichungssystem

$$A(\bar{x})\bar{w} = (p_0, p_0)_\omega e_1.$$

Setzt man nacheinander die Polynome $p_n(x)p_k(x)$, $k = 0, \ldots, n-1$, in die Quadraturformel ein, so erhält man

$$\int_a^b \omega(x) p_n(x) p_k(x)\,dx = (p_n, p_k)_\omega = 0 = \sum_{i=1}^{n} \bar{w}_i p_n(\bar{x}_i) p_k(\bar{x}_i), \quad k = 0, 1, \ldots, n-1.$$

Dies entspricht dem linearen Gleichungssystem $A(\bar{x})c = 0$ mit

$$c = (\bar{w}_1 p_n(\bar{x}_1), \ldots, \bar{w}_n p_n(\bar{x}_n))^T.$$

Da die Stützstellen paarweise verschieden sind, ist die Matrix $A(\bar{x})$ nach Satz 3.11 regulär. Damit folgt aus dem letzten homogenen Gleichungssystem $c = 0$. Da aber alle Gewichte von Null verschieden sind, folgt weiter $p_n(\bar{x}_i) = 0$ für $= 1, \ldots, n$. Die Stützstellen $\bar{x}_i$, $i = 1, \ldots, n$, sind also Nullstellen des n-ten Orthogonalpolynoms p_n. Es gilt damit $\bar{x}_i = x_i$ für $i = 1, \ldots, n$. Dann ist aber auch $A(\bar{x}) = A(x)$ und weiter $\bar{w} = w$. Damit ist die Eindeutigkeit der Quadraturformel gezeigt.
Die Positivität der Gewichte folgt, falls man nacheinander die Polynome

$$\bar{p}_k(x) = \prod_{\substack{j=1 \\ j \neq k}}^{n} (x - x_j)^2 \geqq 0$$

in die Quadraturformel einsetzt. Wegen $\bar{p}_k \in \Pi_{2n-2}$ gilt $I(\bar{p}_k) = Q_n(\bar{p}_k)$, also

$$0 < \int_a^b \omega(x) \bar{p}_k(x)\,dx = \sum_{i=1}^{n} w_i \bar{p}_k(x_i) = w_k \bar{p}_k(x_k).$$

Aus $\bar{p}_k(x_k) > 0$ folgt $w_k > 0$ für $k = 1, \ldots, n$. ❀

Die in diesem Satz beschriebenen Quadraturformeln werden als GAUSSsche Quadraturformeln bezeichnet. Je nach verwendeter Gewichtsfunktion $\omega(x)$ und Integrationsintervall $[a,b]$ spricht man im einzelnen von

- GAUSS-LEGENDRE-Quadratur:

$$\omega(x) \equiv 1, \quad [a,b] = [-1,1],$$

- GAUSS-TSCHEBYSCHEFF-Quadratur:

$$\omega(x) = 1/\sqrt{(1-x^2)}, \quad [a,b] = [-1,1],$$

- GAUSS-LAGUERRE-Quadratur:

$$\omega(x) = e^{-x}, \quad [a,b] = [0,\infty),$$

- GAUSS-HERMITE-Quadratur:

$$\omega(x) = e^{-x^2}, \quad [a,b] = (-\infty,\infty),$$

um nur die wichtigsten zu nennen. Die zugehörigen Stützstellen und Gewichte sind tabelliert. Man findet sie zum Beispiel in ABRAMOWITZ/STEGUN "'Handbook of mathematical functions"'.
Es lässt sich zeigen, dass der PEANO-Kern bei GAUSSschen Quadraturformeln ein konstantes Vorzeichen auf $[a,b]$ besitzt. Damit erhält man folgende Restglieddarstellung:

$$\begin{aligned} R_n(f) &= \frac{f^{(2n)}(\xi)}{(2n)!} R_n(x^{2n}) = \frac{f^{(2n)}(\xi)}{(2n)!} R_n(p_n(x)^2) \\ &= \frac{f^{(2n)}(\xi)}{(2n)!} \left[I(p_n(x)^2) - Q_n(p_n(x)^2) \right] \\ &= \frac{f^{(2n)}(\xi)}{(2n)!} \left[\int_a^b \omega(x) p_n(x)^2 \, dx - \sum_{i=1}^{n} w_i p_n(x_i)^2 \right] = \frac{f^{(2n)}(\xi)}{(2n)!} (p_n, p_n)_\omega \end{aligned}$$

mit einem $\xi \in [a,b]$.

3.4. Das Romberg-Verfahren

3.4.1. Die Euler-MacLaurin'sche Summenformel

Wesentliche Grundlage der Integrationsmethode, die in diesem Abschnitt behandelt werden soll, ist die EULER-MACLAURINsche Summenformel. Zur Herleitung dieser Formel benötigen wir die BERNOULLI-Polynome und BERNOULLI-Zahlen. Die BERNOULLI-**Polynome** $B_k \in \Pi_k$ sind rekursiv definiert durch

1. $B_0(t) \equiv 1$,
2. $B_k'(t) = B_{k-1}(t), \quad \int_0^1 B_k(t)\,dt = 0$ für $k = 1, \ldots$

Die Zahlen $B_k = k!B_k(0)$ heißen BERNOULLI-**Zahlen**. BERNOULLI-Polynome und BERNOULLI-Zahlen haben interessante Eigenschaften.

3.13. Satz: *Es gilt*

1. $B_k(0) = B_k(1)$ *für* $k = 2, 3, \ldots,$
2. $B_k(t) = (-1)^k B_k(1-t)$ *für* $k = 0, 1, \ldots,$
3. $B_{2k+1}(0) = B_{2k+1}(1/2) = B_{2k+1}(1) = 0$ *für* $k = 1, 2, \ldots,$
4. *(a)* $B_{2k}(t) - B_{2k}(0)$ *besitzt für* $k = 1, 2, \ldots$ *auf dem Intervall* $[0,1]$ *nur die Nullstellen* $t = 0$ *und* $t = 1$,
   (b) $B_{2k+1}(t) - B_{2k+1}(0)$ *besitzt für* $k = 1, 2, \ldots$ *auf dem Intervall* $[0,1]$ *nur die Nullstellen* $t = 0$, $t = 1/2$.

Beweis:

1. Es gilt

$$\begin{aligned} B_k'(s) &= B_{k-1}(s) \\ \int_0^t B_k'(s)\,ds &= \int_0^t B_{k-1}(s)\,ds \\ B_k(t) - B_k(0) &= \int_0^t B_{k-1}(s)\,ds \\ B_k(1) - B_k(0) &= \int_0^1 B_{k-1}(s)\,ds = 0, \quad k \geqq 2. \end{aligned}$$

2. Es sei $C_k(t) = (-1)^k B_k(1-t)$. Wir zeigen, dass $C_k(t)$ denselben Rekursionsbeziehungen genügt wie $B_k(t)$. Es gilt

$$\begin{aligned} C_0(t) &= B_0(1-t) = 1 = B_0(t), \\ C_k'(t) &= (-1)^k \frac{d}{dt} B_k(1-t) \\ &= (-1)^k (-1) B_k'(1-t) \\ &= (-1)^{k-1} B_{k-1}(1-t) \\ &= C_{k-1}(t) \end{aligned}$$

und

$$\begin{aligned} \int_0^1 C_k(t)\,dt &= (-1)^k \int_0^1 B_k(1-t)\,dt \\ &= (-1)^k \int_1^0 B_k(s)\,(-ds) \\ &= (-1)^k \int_0^1 B_k(s)\,ds \\ &= 0. \end{aligned}$$

3. Wegen 1. gilt

$$B_{2k+1}(0) = B_{2k+1}(1), \quad k = 1, 2, \ldots.$$

Wegen 2. gilt

$$B_{2k+1}(0) = -B_{2k+1}(1), \quad k = 0, 1, \ldots.$$

Damit folgt

$$B_{2k+1}(0) = B_{2k+1}(1) = 0, \quad k = 1, 2, \ldots.$$

Wegen 2. gilt

$$B_{2k+1}(1/2) = -B_{2k+1}(1/2), \quad k = 0, 1, \ldots,$$

also

$$B_{2k+1}(1/2) = 0, \quad k = 0, 1, \ldots.$$

4. Wir beweisen die Aussagen mittels vollständiger Induktion.
Induktionsanfang: Für $k = 1$ gilt

$$\begin{aligned} B_2(t) &= \frac{1}{2}t^2 - \frac{1}{2}t + \frac{1}{12}, \\ B_3(t) &= \frac{1}{6}t^3 - \frac{1}{4}t^2 + \frac{1}{12}t, \\ B_2(t) - B_2(0) &= \frac{1}{2}t^2 - \frac{1}{2}t = \frac{1}{2}t(t-1), \\ B_3(t) - B_3(0) &= \frac{1}{6}t(t-1)(t-\frac{1}{2}). \end{aligned}$$

Induktionsvoraussetzung: Es gelte für ein gewisses k

(a) $B_{2k}(t) - B_{2k}(0)$ besitzt auf dem Intervall $[0,1]$ nur die Nullstellen $t = 0$ und $t = 1$,

(b) $B_{2k+1}(t) - B_{2k+1}(0)$ besitzt auf dem Intervall $[0,1]$ nur die Nullstellen $t = 0$, $t = 1/2$ und $t = 1$.

Induktionsschritt: Besitzt nun $B_{2k+2}(t) - B_{2k+2}(0)$ außer 0 und 1 eine weitere Nullstelle $\xi \in [0,1]$, so dürfen wir wegen der Symmetrie von $B_{2k+2}(t) - B_{2k+2}(0)$ annehmen, dass $\xi \in (0,1/2]$ gilt. Nach dem Satz von ROLLE hat dann aber

$$B'_{2k+2}(t) = B_{2k+1}(t) = B_{2k+1}(t) - B_{2k+1}(0)$$

eine Nullstelle im Intervall $(0,1/2)$ im Widerspruch zur Behauptung 4b. Besitzt $B_{2k+3}(t) - B_{2k+3}$ außer 0, 1/2 und 1 eine weitere Nullstelle $\xi \in [0,1]$, so dürfen wir wegen der Symmetrie von $B_{2k+3}(t) - B_{2k+3}(0)$ wieder annehmen, dass $\xi \in (0,1/2)$ gilt. Nach dem Satz von ROLLE hat dann aber $B'_{2k+3}(t) = B_{2k+2}(t)$ ebenfalls eine Nullstelle in $(0,1/2)$ und genauso

$$B''_{2k+3}(t) = B_{2k+1}(t) = B_{2k+1}(t) - B_{2k+1}(0).$$

Das ist wieder ein Widerspruch zu 4b. Damit gilt die Aussage 4. für beliebige $k \geqq 1$.

❀

Nun beweisen wir die EULER-MACLAURINsche Summenformel.

3.14. Satz: *Es seien $n, m \in \mathbb{N}$, $f \in C^{2m}[a,b]$ und*

$$T(f;h) = h\left[\frac{1}{2}f(a) + \sum_{j=1}^{n-1} f(x_j) + \frac{1}{2}f(b)\right]$$

mit $h = \dfrac{b-a}{n}, \quad x_j = a + jh, \quad j = 0,1,\ldots,n.$

Dann gilt

$$\int_a^b f(x)\,dx = T(f;h) - \sum_{i=1}^{m-1} \frac{B_{2i}}{(2i)!}\left[f^{(2i-1)}(b) - f^{(2i-1)}(a)\right] h^{2i} - \frac{(b-a)B_{2m}}{(2m)!} f^{(2m)}(\xi) h^{2m}$$

für ein $\xi \in (a,b)$.

Beweis: Es gilt $B_1(t) = t - 1/2$ und damit $B_1(0) = -1/2$ und $B_1(1) = 1/2$. Aus Satz 3.13 wissen wir weiter, dass

$$B_{2i}(0) = B_{2i}(1), \quad B_{2i+1}(0) = B_{2i+1}(1) = 0, \quad i = 1,2,\ldots$$

gilt. Für eine Funktion $g \in C^{2m}[0,1]$ folgt dann durch fortgesetzte partielle Integration

$$\begin{aligned}
&\int_0^1 g(t)\,dt = \int_0^1 B_0(t)g(t)\,dt = \int_0^1 B_1'(t)g(t)\,dt \\
&= B_1(t)g(t)\big|_0^1 - \int_0^1 B_1(t)g'(t)\,dt = B_1(t)g(t)\big|_0^1 - \int_0^1 B_2'(t)g'(t)\,dt \\
&= B_1(t)g(t)\big|_0^1 - B_2(t)g'(t)\big|_0^1 + \int_0^1 B_2(t)g''(t)\,dt \\
&= B_1(t)g(t)\big|_0^1 - B_2(t)g'(t)\big|_0^1 + \int_0^1 B_3'(t)g''(t)\,dt \\
&= B_1(t)g(t)\big|_0^1 - B_2(t)g'(t)\big|_0^1 + B_3(t)g''(t)\big|_0^1 - \int_0^1 B_3(t)g'''(t)\,dt \\
&= B_1(t)g(t)\big|_0^1 - B_2(t)g'(t)\big|_0^1 - \int_0^1 B_4'(t)g'''(t)\,dt \\
&\vdots \qquad \vdots \\
&= B_1(t)g(t)\big|_0^1 - \textstyle\sum_{i=1}^{m-1} B_{2i}(t)g^{(2i-1)}(t)\Big|_0^1 - \int_0^1 B_{2m}'(t)g^{(2m-1)}(t)\,dt \\
&= B_1(t)g(t)\big|_0^1 - \textstyle\sum_{i=1}^{m-1} B_{2i}(t)g^{(2i-1)}(t)\Big|_0^1 - \int_0^1 [B_{2m}(t) - B_{2m}(0)]' g^{(2m-1)}(t)\,dt \\
&= B_1(t)g(t)\big|_0^1 - \textstyle\sum_{i=1}^{m-1} B_{2i}(t)g^{(2i-1)}(t)\Big|_0^1 - [B_{2m}(t) - B_{2m}(0)]\, g^{(2m-1)}(t)\Big|_0^1 +
\end{aligned}$$

$$
\begin{aligned}
&\quad + \int_0^1 [B_{2m}(t) - B_{2m}(0)]\, g^{(2m)}(t)\, dt \\
&= B_1(t)g(t)|_0^1 - \textstyle\sum_{i=1}^{m-1} B_{2i}(t) g^{(2i-1)}(t) \Big|_0^1 + \int_0^1 [B_{2m}(t) - B_{2m}(0)]\, g^{(2m)}(t)\, dt.
\end{aligned}
$$

Nun wechselt nach Satz 3.13 der Term $B_{2m}(t) - B_{2m}(0)$ auf dem Intervall $[0,1]$ nicht sein Vorzeichen. Wir setzen die Integrationsgrenzen ein und wenden den verallgemeinerten Mittelwertsatz der Integralrechnung anwenden: Es existiert ein $\xi \in (0,1)$ mit

$$
\begin{aligned}
\int_0^1 [B_{2m}(t) - B_{2m}(0)]\, g^{(2m)}(t)\, dt &= g^{(2m)}(\xi) \int_0^1 [B_{2m}(t) - B_{2m}(0)]\, dt \\
&= -B_{2m}(0) g^{(2m)}(\xi).
\end{aligned}
$$

Damit gilt

$$
\begin{aligned}
&\int_0^1 g(t)\, dt = \\
&\frac{1}{2}[g(0) + g(1)] - \sum_{i=1}^{m-1} B_{2i}(0) \left[g^{(2i-1)}(1) - g^{(2i-1)}(0) \right] - B_{2m}(0) g^{(2m)}(\xi).
\end{aligned}
$$

Wir setzen nun $g(t) = f(x_j + ht)$. Dann gilt $g^{(k)}(t) = h^k f^{(k)}(x_j + ht)$. Einsetzen in die linke bzw. rechte Seite der letzten Gleichung liefert

$$
\int_0^1 g(t)\, dt = \int_0^1 f(x_j + ht)\, dt = \frac{1}{h} \int_{x_j}^{x_{j+1}} f(x)\, dx
$$

und

$$
\begin{aligned}
&\frac{1}{2}[g(0) + g(1)] - \sum_{i=1}^{m-1} B_{2i}(0) \left[g^{(2i-1)}(1) - g^{(2i-1)}(0) \right] - B_{2m}(0) g^{(2m)}(\xi) \\
&= \frac{1}{2} \left[f(x_j) + f(x_{j+1}) \right] - \sum_{i=1}^{m-1} B_{2i}(0) \left[f^{(2i-1)}(x_{j+1}) - f^{(2i-1)}(x_j) \right] h^{2i-1} \\
&\qquad - B_{2m}(0) f^{(2m)}(\xi_j) h^{2m}
\end{aligned}
$$

mit einem $\xi_j \in (x_j, x_{j+1})$. Zusammengefasst ergibt das

$$\begin{aligned}\int_{x_j}^{x_{j+1}} f(x)\,dx =& \frac{h}{2}\left[f(x_j)+f(x_{j+1})\right] \\ &- \sum_{i=1}^{m-1} B_{2i}(0)\left[f^{(2i-1)}(x_{j+1}) - f^{(2i-1)}(x_j)\right] h^{2i} \\ &- B_{2m}(0) f^{(2m)}(\xi_j) h^{2m+1}.\end{aligned}$$

Nun brauchen wir nur noch über alle Intervalle zu summieren. Wir erhalten

$$\begin{aligned}\int_a^b f(x)\,dx =& \sum_{j=0}^{n-1} \int_{x_j}^{x_{j+1}} f(x)\,dx \\ =& \sum_{j=0}^{n-1} \frac{h}{2}\left[f(x_j)+f(x_{j+1})\right] \\ &- \sum_{j=0}^{n-1}\sum_{i=1}^{m-1} B_{2i}(0)\left[f^{(2i-1)}(x_{j+1}) - f^{(2i-1)}(x_j)\right] h^{2i} \\ &- B_{2m}(0) h^{2m+1} \sum_{j=0}^{n-1} f^{(2m)}(\xi_j) \\ =& T(f;h) - \sum_{i=1}^{m-1} B_{2i}(0) h^{2i} \sum_{j=0}^{n-1}\left[f^{(2i-1)}(x_{j+1}) - f^{(2i-1)}(x_j)\right] \\ &- B_{2m}(0) h^{2m} \frac{b-a}{n} \sum_{j=0}^{n-1} f^{(2m)}(\xi_j) \\ =& T(f;h) - \sum_{i=1}^{m-1} B_{2i}(0) h^{2i}\left[f^{(2i-1)}(b) - f^{(2i-1)}(a)\right] \\ &- B_{2m}(0) h^{2m} (b-a) f^{(2m)}(\xi)\end{aligned}$$

mit einem $\xi \in (a,b)$. Setzt man jeweils $B_{2i}(0) = \frac{B_{2i}}{(2i)!}$, so ergibt sich die EULER-MACLAURINsche Summenformel. ❀

Bemerkungen: (i) Für $f \in C^4[a,b]$ erhält man aus der EULER-MACLAURINschen Summenformel eine wesentliche Verbesserung der Trapezsumme. Mit $B_2 = 1/6$

und $B_4 = -1/30$ ergibt sich

$$\int_a^b f(x)\,dx = T(f;h) - \frac{h^2}{12}\left[f'(b) - f'(a)\right] + \frac{(b-a)h^4}{720} f^{(4)}(\xi), \qquad \xi \in (a,b).$$

Falls die Werte der 1. Ableitung von f an den Randpunkten zur Verfügung stehen, erhöht sich so der Exaktheitsgrad durch diesen Korrekturterm ohne wesentlichen Mehraufwand um 2.
(ii) Weiterhin liest man aus Satz 3.14 ab, dass die Trapezsumme hervorragend zur Integration von periodischen Funktionen über eine ganze Periode geeignet ist. Ist nämlich $f \in C^{2m}(\mathbb{R})$ eine $(b-a)$-periodische Funktion, so gilt $f^{(i)}(a) = f^{(i)}(b)$ für $i = 0, 1, \ldots, 2m$. Damit folgt aus der EULER-MACLAURINsche Summenformel

$$\int_a^b f(x)\,dx = T(f;h) - \frac{(b-a)B_{2m}h^{2m}}{(2m)!} f^{(2m)}(\xi), \qquad \xi \in (a,b).$$

Ist f $(b-a)$-periodisch und beliebig oft differenzierbar, so ist die numerische Integration mittels der Trapezsumme über eine volle Periode exakt.

3.4.2. Konstruktion des Romberg-Verfahrens

Nun kommen wir zu einer Anwendung der EULER-MACLAURINsche Summenformel. Sie besteht in der Konstruktion von Extrapolationsverfahren. Nach Satz 3.14 gilt für eine Funktion $f \in C^{2m+2}[a,b]$ und eine Schrittweite der Form $h = (b-a)/n$, $n \in \mathbb{N}$,

$$T(f;h) = \tau_0 + \tau_1 h^2 + \cdots + \tau_m h^{2m} + \alpha_{m+1}(f;h)h^{2m+2}$$

mit

$$\begin{aligned}
\tau_0 &= \int_a^b f(x)\,dx, \\
\tau_i &= \frac{B_{2i}}{(2i)!}\left[f^{(2i-1)}(b) - f^{(2i-1)}(a)\right] \quad i = 1, \ldots, m, \\
\alpha_{m+1}(f;h) &= \frac{B_{2m+2}}{(2m+2)!} f^{(2m)}(\xi) \quad \xi \in (a,b)
\end{aligned}$$

und

$$|\alpha_{m+1}(f;h)| \leqq M_{m+1} = \frac{B_{2m+2}}{(2m+2)!} \max_{\xi \in (a,b)} |f^{(2m)}(\xi)| < \infty.$$

Für die Trapezsumme $T(f;h)$ existiert somit eine sogenannte asymptotisch Entwicklung. Das Restglied $\alpha_{m+1}(f;h)h^{2m+2}$ ist beschränkt. Normalerweise sind die Koeffizienten τ_i nicht bekannt. Es interessiert nur der Koeffizient τ_0. Hier setzt nun die eigentliche Idee des Verfahrens ein. Wir sind in der Lage, für verschiedene Schrittweiten $h_j = (b-a)/n_j$ mittels

$$T(f;h_j) = h_j \left[\frac{1}{2} f(a) + f(a+h_j) + \cdots + f(b-h_j) + \frac{1}{2} f(b) \right]$$

verschiedene Werte von $T(f;h)$ zu berechnen. Wir wissen aus der asymptotischen Entwicklung von $T(f;h)$, dass sich $T(f;h)$ für kleine Schrittweiten h wie ein Polynom in h^2 verhält. Von diesem Polynom interessiert nur der Wert an der Stelle $h = 0$. Nun liegt es nahe, zu den Wertepaaren $(h_j^2, T(f;h_j))$, $j = 0,1,\ldots,k$ ein Interpolationspolynom zu berechnen, und den Wert dieses Interpolationspolynoms an der Stelle $h = 0$ als Näherungswert für τ_0, also für das zu berechnende bestimmte Integral, zu nehmen. Wir machen für das Interpolationspolynom den Ansatz

$$\tilde{T}(h) = a_0 + a_1 h^2 + \cdots + a_k h^{2k}.$$

Dieses Polynom soll die Interpolationsbedingungen

$$\tilde{T}(h_j) = T(f;h_j) = T_{j0}, \quad j = 1,\ldots,k$$

erfüllen. Von dem Polynom ist $a_0 = \tilde{T}(0)$ zu berechnen. Das Polynom selbst interessiert nicht. Zum Lösen dieses Problems bietet sich der NEVILLE-Algorithmus an. Das so entstehende Verfahren wird ROMBERG-Integration genannt.

3.15. ROMBERG**-Integration:**

Wähle Schrittweitenfolge $\{h_k\}$ *und Extrapolationstiefe* m.
for $k = 0$ ***to*** m ***do***
  {*Berechne die Trapezsumme* T_{k0}}
  $$T_{k0} = h_k \left[\frac{1}{2} f(a) + f(a+h_k) + \cdots + f(b-h_k) + \frac{1}{2} f(b) \right]$$
  for $i = 1$ ***to*** k ***do***
    $$T_{ki} = T_{k,i-1} + \frac{T_{k,i-1} - T_{k-1,i-1}}{\left(\frac{h_{k-i}}{h_k}\right)^2 - 1}.$$
  endfor
endfor

Bemerkungen: (i) Als Schrittweitenfolge wurde von ROMBERG die Folge

$$h_0 = b-a, \quad h_1 = \frac{h_0}{2}, \quad h_2 = \frac{h_1}{2}, \quad h_3 = \frac{h_2}{2}, \ldots$$

gewählt. Hier verdoppelt sich in jedem Schritt die Anzahl der Stützstellen. Der Rechenaufwand wächst schnell an. Darum wurde von BULIRSCH die langsamer fallende Schrittweitenfolge

$$h_0 = b-a,\ h_1 = \frac{h_0}{2},\ h_2 = \frac{h_0}{3},\ h_3 = \frac{h_1}{2},\ h_4 = \frac{h_2}{2},\ h_5 = \frac{h_3}{2},\ h_6 = \frac{h_4}{2}, \ldots$$

vorgeschlagen. Möglich ist auch die RUTISHAUSER-Folge

$$h_0 = b-a,\ h_1 = \frac{h_0}{2},\ h_2 = \frac{h_0}{3},\ h_3 = \frac{h_1}{3},\ h_4 = \frac{h_2}{3},\ h_5 = \frac{h_3}{3},\ h_6 = \frac{h_4}{2}, \ldots$$

(ii) Es fehlt noch eine geeignete Abbruchbedingung. Da jeder Wert T_{ki} im Interpolationsschema eine Näherung für das gesuchte bestimmte Integral darstellt, könnte man abbrechen, falls sich die T_{ki} nicht mehr stark voneinander unterscheiden, also eine Bedingung der Form $|T_{ki} - T_{k,i-1}| \leqq \varepsilon$ erfüllt ist. Als ε wird $\varepsilon = K\text{eps}|T_{k0}|$ verwendet. Um zufällige Effekte auszuschließen, sollte man aber erst dann abbrechen, wenn die obige Bedingung mehrmals hintereinander erfüllt ist (etwa dreimal).
(iii) Die i-Schleife sollte man bei großem k nicht zu weit laufen lassen. Wie wir aus Abschnitt 2.2.4. wissen, ist Interpolation mit Polynomen hohen Grades nicht sinnvoll. Vernünftig ist es, etwa 5 bis 7 Spalten im NEVILLE-Schema zu berechnen. Die Laufanweisung müsste also zu
for $i = 1$ **to** $\min\{k, 5..7\}$ **do**
abgeändert werden.
(iv) Praktische Erfahrungen haben gezeigt, dass man oft bessere Resultate erhält, falls man statt Polynominterpolation Rationale Interpolation anwenden. Dazu wäre die Rekursionsformel durch

$$T_{ki} = T_{k,i-1} + \frac{T_{k,i-1} - T_{k-1,i-1}}{\left(\frac{h_{k-i}}{h_k}\right)\left[1 - \frac{T_{k,i-1}-T_{k-1,i-1}}{T_{k,i-1}-T_{k-1,i-2}}\right] - 1}$$

zu ersetzen. Dabei wird wieder $T_{k,-1} = 0$ für $k = 0, 1, \ldots$ gesetzt.

3.4.3. Fehlerabschätzungen und Konvergenz

Zuerst wollen wir das ROMBERG-Verfahren mit der Schrittweitenfolge $h_k = (b-a)/2^k$ betrachten. Hier gilt der folgende

3.16. Satz: *Es sei T_{k0} die Trapezsumme zur Schrittweite $h_k = (b-a)/2^k$. Die T_{ki}, $i = 1,\ldots,k$, seien durch das* ROMBERG-*Verfahren erzeugt. Es gilt*

$$T_{ki} = T_{k,i-1} + \frac{T_{k,i-1} - T_{k-1,i-1}}{4^i - 1}$$

für $i = 1,\ldots,k$.
Dann ist T_{ki} eine Quadraturformel mit nichtnegativen Gewichten.

Beweis: Durch vollständige Induktion über i zeigen wir

1. Für $k = i-1, i, \ldots$ ist $T_{k,i-1}$ eine Quadraturformel mit nichtnegativen Gewichten.
2. Für $k = i, i+1, \ldots$ ist $Q_{ki} = 4^i T_{k,i-1} - 2T_{k-1,i-1}$ eine Quadraturformel mit nichtnegativen Gewichten.

Induktionsanfang: $i = 1$
T_{k0} ist als Trapezsumme eine Quadraturformel mit nichtnegativen Gewichten.
Für Q_{k1} gilt

$$\begin{aligned}
Q_{k1} &= 4T_{k0} - 2T_{k-1,0} \\
&= 4\frac{b-a}{2^k}\left[\frac{1}{2}f(a) + \frac{1}{2}f(b) + \sum_{j=1}^{2^k-1} f\left(a + j\frac{b-a}{2^k}\right)\right] - 2T_{k-1,0} \\
&= 4\frac{b-a}{2^k}\left[\frac{1}{2}f(a) + \frac{1}{2}f(b) + \sum_{j=1}^{2^{k-1}-1} f\left(a + 2j\frac{b-a}{2^k}\right)\right. \\
&\qquad\qquad \left. + \sum_{j=1}^{2^{k-1}} f\left(a + (2j-1)\frac{b-a}{2^k}\right)\right] - 2T_{k-1,0} \\
&= 4\frac{1}{2}T_{k-1,0} + 4\frac{b-a}{2^k}\sum_{j=1}^{2^{k-1}} f\left(a + (2j-1)\frac{b-a}{2^k}\right) - 2T_{k-1,0} \\
&= 4\frac{b-a}{2^k}\sum_{j=1}^{2^{k-1}} f\left(a + (2j-1)\frac{b-a}{2^k}\right).
\end{aligned}$$

Damit ist Q_{k1} eine Quadraturformel mit nichtnegativen Gewichten.
Induktionsvoraussetzung: Es gelten 1. und 2. für ein gewisses i.
Induktionsschritt: Wir zeigen die Gültigkeit von 1. und 2. für $i+1$. Betrachten wir zuerst T_{ki}.

$$T_{ki} = T_{k,i-1} + \frac{T_{k,i-1} - T_{k-1,i-1}}{4^i - 1} = \frac{4^i T_{k,i-1} - T_{k-1,i-1}}{4^i - 1} = \frac{Q_{k,i} + T_{k-1,i-1}}{4^i - 1}.$$

Als positive Summe von Quadraturformel mit nichtnegativen Gewichten ist T_{ki} eine Quadraturformel mit nichtnegativen Gewichten. Für $Q_{k,i+1}$ folgt

$$\begin{aligned} Q_{k,i+1} &= 4^{i+1}T_{k,i} - 2T_{k-1,i} = 4^{i+1}\frac{Q_{k,i}+T_{k-1,i-1}}{4^i-1} - 2\frac{4^iT_{k-1,i-1}-T_{k-2,i-1}}{4^i-1} \\ &= \frac{24^iT_{k-1,i-1}+4^{i+1}Q_{k,i}+2T_{k-2,i-1}}{4^i-1}. \end{aligned}$$

Damit ist $Q_{k,i+1}$ ebenfalls eine Quadraturformel mit nichtnegativen Gewichten. ❀

Da die Trapezsummen selbst für beliebige Polynome gegen das entsprechende bestimmte Integral konvergieren, ist zu erwarten, dass auch alle T_{ik} für Polynome konvergieren. Nach Satz 3.7 folgt damit aber auch die Konvergenz für alle stetigen Funktionen f. Dies wollen wir nun genauer zeigen. Dazu benötigen wir einige Eigenschaften der LAGRANGEschen Darstellung der Interpolationspolynome.

3.17. Satz: *Für die Folge $\{t_j\}_{j\in\mathbb{N}}$ sei $P_i^{(k)} \in \Pi_k$ das Interpolationspolynom zu den Stützpunkten $(t_j, f(t_j))$, $j = i,\ldots,i+k$. In der* LAGRANGE*schen Darstellung gilt*

$$P_i^{(k)}(t) = \sum_{j=i}^{i+k} f(t_j)L_{ij}^{(k)}(t)$$

mit

$$L_{ij}^{(k)}(t) = \prod_{\substack{l=i \\ l\neq j}}^{i+k} \frac{t-t_l}{t_j-t_l}.$$

Dann gilt

1. $$\sum_{j=i}^{i+k} L_{ij}^{(k)}(0)t_j^\nu = \begin{cases} 1 & \text{für} \quad \nu = 0 \\ 0 & \text{für} \quad \nu = 1,\ldots,k \\ (-1)^k t_i t_{i+1}\cdots t_{i+k-1}t_{i+k} & \text{für} \quad \nu = k+1 \end{cases}.$$

2. $$\lim_{k\to\infty} L_{ij}^{(k)}(0) = 0.$$

3. Fällt die Folge $\{t_j\}_{j\in\mathbb{N}}$ *im Sinne von*

$$\frac{t_j}{t_{j+1}} \geqq c > 1,$$

hinreichend schnell, so existiert eine Konstante Λ*, so dass für beliebige* i *und* k *gilt*

$$\sum_{j=i}^{i+k} |L_{ij}^{(k)}(0)| < \Lambda.$$

Beweis: 1. Das Interpolationspolynom $P_i^{(k)}$ stimmt für $f \in \Pi_k$ mit der Funktion f exakt überein. Setzt man für f nacheinander die Polynome $1, t, \ldots, t^k$ ein, so erhält man die ersten beiden Aussagen, falls man $f(t) = P_i^{(k)}(t)$ an der Stelle $t = 0$ betrachtet. Die dritte Aussage ergibt sich aus der Darstellung des Interpolationsfehlers im Falle $f(t) = t^{k+1}$. Nach Satz 2.8 gilt

$$f(t) - P_i^{(k)}(t) = \frac{f^{(k+1)}(\xi)}{(k+1)!}(t-t_i)(t-t_{i+1})\cdots(t-t_{i+k-1})(t-t_{i+k}).$$

Mit $f^{(k+1)}(t) \equiv (k+1)!$ folgt

$$\begin{aligned} f(0) - P_i^{(k)}(0) &= (-t_i)(-t_{i+1})\cdots(-t_{i+k-1})(-t_{i+k}) \\ P_i^{(k)}(0) &= (-1)^k t_i t_{i+1}\cdots t_{i+k-1} t_{i+k}. \end{aligned}$$

Beachtet man noch, dass wegen $f(t) = t^{k+1}$ an der Stelle $t = 0$

$$P_i^{(k)}(0) = \sum_{j=i}^{i+k} t_j^{k+1} L_{ij}^{(k)}(0)$$

gilt, so ergibt sich die dritte Behauptung.
2. Aus

$$\sum_{j=i}^{i+k} L_{ij}^{(k)}(0) = 1$$

für beliebige k folgt die Konvergenz der Reihe

$$\lim_{k\to\infty} \sum_{j=i}^{i+k} L_{ij}^{(k)}(0).$$

Das notwendige Konvergenzkriterium für Reihen liefert dann die Behauptung.
3. Für diese Abschätzung schreiben wir die Summe über die $|L_{ij}^{(k)}(0)|$ etwas um. Es gilt

$$\begin{aligned}\sum_{j=i}^{i+k} |L_{ij}^{(k)}(0)| &= \sum_{j=i}^{i+k} \left| \prod_{\substack{l=i \\ l\neq j}}^{i+k} \frac{t_l}{t_l - t_j} \right| \\ &= \sum_{j=i}^{i+k} \prod_{\substack{l=i \\ l\neq j}}^{i+k} \left| 1 - \frac{t_j}{t_l} \right|^{-1} = \sum_{j=i}^{i+k} \prod_{l=i}^{j-1} \left| 1 - \frac{t_j}{t_l} \right|^{-1} \prod_{l=j+1}^{i+k} \left| 1 - \frac{t_j}{t_l} \right|^{-1}.\end{aligned}$$

Aus der Voraussetzung

$$\frac{t_j}{t_{j+1}} \geqq c > 1$$

für alle j folgt

$$\prod_{l=i}^{j-1} \left| 1 - \frac{t_j}{t_l} \right|^{-1} \leqq \prod_{l=i}^{j-1} \left| 1 - \left(\frac{1}{c}\right)^{j-l} \right|^{-1} = \prod_{l=1}^{j-i} \left| 1 - \left(\frac{1}{c}\right)^{l} \right|^{-1}.$$

Wir zeigen die Konvergenz des unendlichen Produktes

$$\prod_{l=1}^{\infty} \left| 1 - \left(\frac{1}{c}\right)^{l} \right|^{-1}.$$

Es gilt

$$\begin{aligned}\prod_{l=1}^{k} \left| 1 - \left(\frac{1}{c}\right)^{l} \right|^{-1} &= \prod_{l=1}^{k} \left| \frac{c^l}{c^l - 1} \right| = \prod_{l=1}^{k} \left(1 + \frac{1}{c^l - 1} \right) \\ &= \exp\left(\sum_{l=1}^{k} \ln\left(1 + \frac{1}{c^l - 1} \right) \right).\end{aligned}$$

Wegen $\ln(1+x) < x$ für $x > 0$ folgt daraus

$$\prod_{l=1}^{k} \left| 1 - \left(\frac{1}{c}\right)^{l} \right|^{-1} < \exp\left(\sum_{l=1}^{k} \frac{1}{c^l - 1} \right).$$

Das Quotientenkriterium zeigt die Konvergenz der Reihe

$$\sum_{l=1}^{\infty} \frac{1}{c^l - 1}.$$

Damit gilt

$$\lim_{k\to\infty} \prod_{l=1}^{k} \left| 1 - \left(\frac{1}{c}\right)^l \right|^{-1} \leqq \exp\left(\lim_{k\to\infty} \sum_{l=1}^{k} \frac{1}{c^l - 1} \right) = \Sigma.$$

Wegen $c > 1$ ist aber die Folge

$$\left\{ \prod_{l=1}^{k} \left| 1 - \left(\frac{1}{c}\right)^l \right|^{-1} \right\}_{k\in\mathbb{N}}$$

monoton wachsend. Damit ist die Konvergenz des unendlichen Produktes bewiesen. Es existiert somit eine Konstante Λ', so dass für beliebige i, j, k

$$\prod_{l=i}^{j-1} \left| 1 - \frac{t_j}{t_l} \right|^{-1} \leqq \Lambda'$$

gilt. Weiterhin gilt

$$\prod_{l=j+1}^{i+k} \left| 1 - \frac{t_j}{t_l} \right|^{-1} = \prod_{l=j+1}^{i+k} \left(\frac{t_j}{t_l} - 1 \right)^{-1} = \prod_{l=j+1}^{i+k} \frac{1}{\frac{t_j}{t_l} - 1} \leqq \prod_{l=j+1}^{i+k} \frac{1}{c^{l-j} - 1}.$$

Damit erhalten wir

$$\begin{aligned} \sum_{j=i}^{i+k} |L_{ij}^{(k)}(0)| &\leqq \Lambda' \sum_{j=i}^{i+k} \prod_{l=j+1}^{i+k} \frac{1}{c^{l-j} - 1} = \Lambda' \sum_{j=0}^{k} \prod_{l=i+k-j+1}^{i+k} \frac{1}{c^{l-i-k+j} - 1} \\ &= \Lambda' \sum_{j=0}^{k} \prod_{l=1}^{j} \frac{1}{c^l - 1} = \Lambda' \left(1 + \sum_{j=1}^{k} \prod_{l=1}^{j} \frac{1}{c^l - 1} \right). \end{aligned}$$

Mit Hilfe des Quotientenkriteriums wird nun wieder die Konvergenz der Reihe

$$\sum_{j=1}^{\infty} \prod_{l=1}^{j} \frac{1}{c^l - 1}$$

gezeigt. Es gilt also

$$1 + \lim_{k\to\infty} \sum_{j=1}^{k} \prod_{l=1}^{j} \frac{1}{c^l - 1} = \Sigma'$$

und damit

$$\sum_{j=i}^{i+k} |L_{ij}^{(k)}(0)| < \Lambda'\Sigma' = \Lambda$$

für beliebige i, j, k. ❀

Nun beweisen wir den eigentlichen Konvergenzsatz für das ROMBERG-Verfahren.

3.18. Satz: *Es sei $\{h_j\}_{j\in\mathbb{N}}$ eine Schrittweitenfolge mit*

$$\frac{h_j}{h_{j+1}} < c$$

für alle $j = 0, 1, \ldots$. Die Größen

$$T_{ik}, \quad 0 \leqq k \leqq i, \quad i = 0, 1, \ldots$$

seien durch das ROMBERG*-Verfahren berechnete Näherungen für $\int_a^b f(x)\,dx$. Dann gilt*

1. *Ist $f \in \Pi_{2k+1}$, so ist*

$$T_{ik} = \int_a^b f(x)\,dx.$$

Die k-te Spalte im Interpolationsschema liefert somit für Polynome bis zum Grad $2k+1$ exakte Werte.

2. *Ist $f \in C^{2k+2}[a,b]$, so gilt*

$$T_{ik} = \int_a^b f(x)\,dx + O(h_{i-k}^{2k+2}).$$

Folglich konvergiert die k-te Spalte im Interpolationsschema mit der Ordnung $2k+2$ gegen das bestimmte Integral, falls der Integrand $2k+1$-mal stetig differenzierbar auf $[a,b]$ ist.

Beweis: Die T_{ik} sind Werte von Interpolationspolynomen $\tilde{T}_{ik}$ in h^2 an der Stelle $h=0$. Die Polynome $\tilde{T}_{ik}$ erfüllen die Interpolationsbedingungen

$$\tilde{T}_{ik}(h_j) = T_{j0} = T(f;h_j), \quad j = i-k,\ldots,i.$$

In der LAGRANGEschen Darstellung gilt

$$\tilde{T}_{ik}(h) = \sum_{j=i-k}^{i} T_{j0} \prod_{\substack{l=i-k \\ l\neq j}}^{i} \frac{h^2-h_l^2}{h_j^2-h_l^2} = \sum_{j=i-k}^{i} T_{j0} L_{i-k,j}^{(k)}(h^2),$$

und damit

$$T_{ik} = \tilde{T}_{ik}(0) = \sum_{j=i-k}^{i} T_{j0} L_{i-k,j}^{(k)}(0) = \sum_{j=i-k}^{i} c_{i-k,j}^{(k)} T_{j0}$$

mit

$$c_{i-k,j}^{(k)} = L_{i-k,j}^{(k)}(0) = \prod_{\substack{l=i-k \\ l\neq j}}^{i} \frac{h_l^2}{h_l^2-h_j^2}.$$

Aus Satz 3.14 wissen wir, dass für $f \in C^{2k+2}[a,b]$ die Darstellung

$$\int_a^b f(x)\,dx = T(f;h_j) + \sum_{l=1}^{k} \beta_l h_j^{2l} + \frac{(b-a)B_{2k+2}}{(2k+2)!} f^{(2k+2)}(\xi_j) h_j^{2k+2}$$

mit gewissen Koeffizienten β_l und einem $\xi_j \in (a,b)$ gültig ist. Multipliziert man diese Gleichung nacheinander mit den $c_{i-k,j}^{(k)}$ und summiert anschließend von $j=i-k$ bis $j=i$, so ergibt sich

$$\begin{aligned}\sum_{j=i-k}^{i} c_{i-k,j}^{(k)} \int_a^b f(x)\,dx \;&=\; \sum_{j=i-k}^{i} c_{i-k,j}^{(k)} T_{j0} + \sum_{j=i-k}^{i} c_{i-k,j}^{(k)} \sum_{l=1}^{k} \beta_l h_j^{2l} + \\ &\quad + \frac{(b-a)B_{2k+2}}{(2k+2)!} \sum_{j=i-k}^{i} c_{i-k,j}^{(k)} f^{(2k+2)}(\xi_j) h_j^{2k+2}.\end{aligned}$$

Nach Satz 3.17 gilt

$$\sum_{j=i-k}^{i} c_{i-k,j}^{(k)} = 1$$

und

$$\sum_{j=i-k}^{i} c_{i-k,j}^{(k)} h_j^{\nu} = 0$$

für $\nu = 1, \ldots, k$. Damit folgt

$$\begin{aligned}\int_a^b f(x)\,dx &= T_{ik} + \sum_{l=1}^{k} \beta_l \sum_{j=i-k}^{i} c_{i-k,j}^{(k)} h_j^{2l} \\ &\quad + \frac{(b-a)B_{2k+2}}{(2k+2)!} \sum_{j=i-k}^{i} c_{i-k,j}^{(k)} f^{(2k+2)}(\xi_j) h_j^{2k+2} \\ &= T_{ik} + \frac{(b-a)B_{2k+2}}{(2k+2)!} \sum_{j=i-k}^{i} c_{i-k,j}^{(k)} f^{(2k+2)}(\xi_j) h_j^{2k+2}.\end{aligned}$$

Damit ist schon die erste Behauptung gezeigt, denn für $f \in \Pi_{2k+1}$ ist $f^{(2k+2)}(x) \equiv 0$.
Weiterhin gilt

$$\begin{aligned}\left| \int_a^b f(x)\,dx - T_{ik} \right| &= (b-a)\frac{|B_{2k+2}|}{(2k+2)!} \left| \sum_{j=i-k}^{i} c_{i-k,j}^{(k)} f^{(2k+2)}(\xi_j) h_j^{2k+2} \right| \\ &\leq (b-a)\frac{|B_{2k+2}|}{(2k+2)!} \sum_{j=i-k}^{i} \left| c_{i-k,j}^{(k)} \right| \left| f^{(2k+2)}(\xi_j) \right| h_j^{2k+2} \\ &\leq (b-a)\frac{|B_{2k+2}|}{(2k+2)!} \|f^{(2k+2)}\|_\infty \sum_{j=i-k}^{i} \left| c_{i-k,j}^{(k)} \right| h_j^{2k+2} \\ &\leq (b-a)\frac{|B_{2k+2}|}{(2k+2)!} \|f^{(2k+2)}\|_\infty h_{i-k}^{2k+2} \sum_{j=i-k}^{i} \left| c_{i-k,j}^{(k)} \right|.\end{aligned}$$

Nach Satz 3.17 existiert eine Konstante Λ mit

$$\sum_{j=i-k}^{i} \left| c_{i-k,j}^{(k)} \right| \leq \Lambda.$$

Damit folgt

$$\left| \int_a^b f(x)\,dx - T_{ik} \right| \leq \Lambda(b-a)\frac{|B_{2k+2}|}{(2k+2)!} \|f^{(2k+2)}\|_\infty h_{i-k}^{2k+2},$$

und letztendlich

$$\int_a^b f(x)\,dx = T_{ik} + O(h_{i-k}^{2k+2}).$$

❀

Bemerkungen: (i) Mit diesem Satz und Satz 3.16 folgt die Konvergenz des Verfahrens für beliebige stetige Funktionen, falls man die ROMBERG-Folge

$$h_0 = b-a, \quad h_1 = \frac{h_0}{2}, \quad h_2 = \frac{h_1}{2}, \ldots$$

verwendet. Darüber hinaus lässt sich zeigen, dass das ROMBERG-Verfahren für eine beliebige RIEMANN-integrierbare Funktion konvergiert, falls man eine hinreichend schnell fallende Schrittweitenfolge verwendet ($h_j/h_{j+1} \geqq c > 1$). Unter diesen schwachen Voraussetzungen lässt sich nichts über die Konvergenzgeschwindigkeit aussagen. Diese hängt, wie der letzte Satz zeigte, von den analytischen Eigenschaften der Funktion f ab.
(ii) Für die ROMBERG-, BULIRSCH- bzw. RUTISHAUSER-Folgen ist die Bedingung

$$\frac{h_j}{h_{j+1}} \geqq c > 1$$

mit $c = 2$ (ROMBERG), $c = 4/3$ (BULIRSCH) bzw. $c = 3/2$ (RUTISHAUSER) erfüllt.
(iii) Verwendet man die ROMBERG-Folge, so lässt sich sogar eine Darstellung des Quadraturfehlers angeben. Falls $f \in C^{2k+2}[a,b]$ gilt

$$\int_a^b f(x)\,dx - T_{ik} = (b-a)(-1)^{k+1} h_{i-k}^2 h_{i-k+1}^2 \cdots h_{i-1}^2 h_i^2 \frac{B_{2k+2}}{(2k+2)!} f^{(2k+2)}(\xi)$$

mit einem $\xi \in (a,b)$.

3.5. Aufgaben

1. Man zeige: Eine NEWTON-COTES-Formel vom Grade $n = 2k$ integriert auch Polynome vom Grade $2k+1$ exakt.

2. Es sei eine geschlossene NEWTON-COTES-Formel gegeben:

$$\int_a^b f(x)\,dx = (b-a)\sum_{i=0}^{n} \sigma_i f(x_i) + R_n(f).$$

Man zeige: $\sigma_i = \sigma_{n-i}$ für $i = 0,\ldots,n$.

3. Man zeige:

(a) Für $f \in C^2[a,b]$ gilt für die Trapezsumme $T(h)$:

$$T(h) - I[f] = \frac{(b-a)h^2 f''(\xi)}{12} \text{ für ein } \xi \in [a,b].$$

(b) Für $f \in C^4[a,b]$ gilt für die SIMPSON-Summe $S(h)$:

$$S(h) - I[f] = \frac{(b-a)h^4 f^{(4)}(\xi)}{180} \text{ für ein } \xi \in [a,b].$$

$$I[f] = \int_b^a f(x)\,dx.$$

4. Man zeige:

(a) Der Ausdruck T_{11} bei der ROMBERG-Integration mit Polynominterpolation liefert für die Schrittweitenfolge

$$h_0 = b-a\,, \quad h_1 = \frac{h_0}{2}$$

die SIMPSON-Regel.

(b) Der Ausdruck T_{22} bei der ROMBERG-Integration mit Polynominterpolation liefert für die Schrittweitenfolge

$$h_0 = b-a\,, \quad h_1 = \frac{h_0}{2}\,, \quad h_2 = \frac{h_1}{2}$$

die MILNE-Regel.

5. (a) Man zeige, dass die Quadraturformel

$$\int_{-\infty}^{\infty} e^{-x^2} f(x)\,dx \approx \frac{\sqrt{\pi}}{6}\left[f\left(-\sqrt{\frac{3}{2}}\right) + 4f(0) + f\left(\sqrt{\frac{3}{2}}\right)\right]$$

den Exaktheitsgrad 5 hat.

(b) Man zeige, dass die Quadraturformel

$$\int_0^\infty e^{-x} f(x)\,dx \approx \frac{2+\sqrt{2}}{4} f(2-\sqrt{2}) + \frac{2-\sqrt{2}}{4} f(2+\sqrt{2})$$

den Exaktheitsgrad 3 hat.

6. Für die folgenden Quadraturformeln Q_n zum näherungsweisen Berechnen von $I(f)$ bestimme man den Exaktheitsgrad m. Weiterhin leite man mit Hilfe des PEANO-Kerns eine Restglieddarstellung her, die für alle $f \in C^{m+1}[a,b]$ gilt.

(a)

$$I(f) = \int_a^b f(x)\,dx \qquad Q_1(f) = \frac{b-a}{2}[f(a)+f(b)],$$

(b)

$$I(f) = \int_a^b f(x)\,dx \qquad Q_2(f) = \frac{b-a}{6}[f(a)+4f(\frac{a+b}{2})+f(b)],$$

(c)

$$I(f) = \int_{-1}^1 f(x)\,dx \qquad Q_1(f) = f(-\frac{1}{\sqrt{3}})+f(\frac{1}{\sqrt{3}}).$$

7. Für $f \in C^1[0,\infty)$ folgt aus der PEANOschen Restglieddarstellung für beliebige $x_{i+1}, x_i \in [0,\infty)$

$$\int_{x_i}^{x_{i+1}} f(x)\,dx - \frac{x_{i+1}-x_i}{2}[f(x_i)+f(x_{i+1})] = \int_{x_i}^{x_{i+1}} K_0(t) f'(t)\,dt$$

mit

$$K_0(t) = \int_{x_i}^{x_{i+1}} (x-t)_+^0\,dx - \frac{x_{i+1}-x_i}{2}[(x_i-t)_+^0 + (x_{i+1}-t)_+^0].$$

Damit zeige man:

$$\sum_{i=0}^{n} f(i) = \frac{1}{2}[f(0)+f(n)] + \int_0^n f(x)\,dx + \sum_{i=1}^{n} \int_{i-1}^{i} (x-i+\frac{1}{2}) f'(x)\,dx$$

und folgere weiter die Existenz der EULERschen Konstanten

$$C = \lim_{n\to\infty} (\sum_{i=1}^{n} \frac{1}{i} - \ln(n))$$

und die Gleichheit

$$C = \frac{1}{2} - \int_0^{\infty} \frac{x-[x]-\frac{1}{2}}{(1+x)^2}\,dx.$$

Hinweise:

(a) $(x-t)_+^0 = 1$ für $t \leqq x$ und 0 für $t > x$,

(b) $[x]$ - ganzer Teil von x (größte ganze Zahl, die kleiner gleich x),

(c) Man zeige zuerst die Integraldarstellung für C und folgere aus einer Abschätzung des Integrals die Existenz des Grenzwertes.

8. Man zeige mit Hilfe der PEANOschen Restglieddarstellung, dass der Fehler für die Trapezregel im Falle $f \in C^2[a,b]$ durch

$$\int_a^b f(x)\,dx - \frac{b-a}{2}(f(a)+f(b)) = -\frac{h^3}{12} f''(\xi) \quad \xi \in (a,b)$$

gegeben ist.

9. Zu den n Stützstellen $x_1,\ldots,x_n$ sei durch

$$Q_n = \sum_{i=1}^{n} w_i f(x_i)$$

eine interpolatorische Quadraturformel zum näherungsweisen Berechnen von

$$I(f) = \int_a^b \omega(x) f(x)\,dx$$

mit der zulässigen Gewichtsfunktion $\omega(x)$ gegeben. Man zeige, dass der Exaktheitsgrad von Q_n dann $2n-1$ ist, falls mit

$$q_n = \prod_{k=1}^{n}(x-x_k) \quad \text{gilt:} \int_a^b \omega(x) q_n(x) p(x)\,dx = 0 \quad \text{für alle} \quad p \in \Pi_{n-1}.$$

Bemerkung: Eine Quadraturformel Q_n heißt interpolatorisch, falls ihr Exaktheitsgrad mindestens gleich $n-1$ ist.

10. Aus der EULER-MACLAURINschen Summenformel erhält man die Quadraturformel von CHEVILLIET

$$\int_a^b f(x)\,dx \approx \frac{1}{2}(b-a)\left[f(a)+f(b)\right] - \frac{1}{12}(b-a)^2\left[f'(b)-f'(a)\right].$$

Man zeige, dass diese Quadraturformel den Exaktheitsgrad $m=3$ hat und gebe mit Hilfe des PEANO-Kerns K_3 eine Darstellung des Restgliedes an, die für alle Funktionen $f \in C^4[a,b]$ gültig ist.

11. Man bestimme zu den äquidistanten Stützstellen x_0, x_1, x_2, x_3 die Gewichte w_0, w_1, w_2 der Quadraturformel

$$\int_{x_2}^{x_3} f(x)\,dx = \sum_{i=0}^{2} w_i f(x_i) + R(f),$$

die durch Integration des Polynoms entsteht, das f in x_0, x_1, x_2 interpoliert.

Chapter 4

Differentiation

4.1. Interpolatorische Differentiationsformeln

Gegeben sei eine reelle differenzierbare Funktion f. In diesem Kapitel werden Methoden untersuchen, mit denen sich die Ableitung von f an einer Stelle $\bar{x}$ näherungsweise bestimmen lässt. Die prinzipielle Vorgehensweise entspricht dabei der Vorgehensweise bei der Herleitung von Quadraturformeln zur numerischen Integration. Wir ersetzen die Funktion f durch eine andere Funktion φ (ein Polynom), deren Ableitung sich leichter bestimmen lässt. Es sei also

$$f(x) = \varphi(x) + R(x).$$

Dabei ist die Funktion φ so zu wählen, dass das Restglied $R(x)$ in einer Umgebung der Stelle $\bar{x}$ hinreichend klein wird. Bei der numerischen Integration folgte aus der Kleinheit des Restgliedes die Kleinheit des Integrationsfehlers:

$$\forall x \in [a,b] : |R(x)| \leqq M \Longrightarrow \left| \int_a^b f(x)\,dx - \int_a^b \varphi(x)\,dx \right| \leqq (b-a)M.$$

Wie das folgende Beispiel zeigt, gilt dies bei der numerischen Differentiation nicht.

4.1. Beispiel: Es sei $\varphi \in C^1[a,b]$ und

$$f(x) = \varphi(x) + \frac{1}{n}\sin(n^2 x).$$

Es gilt

$$\|f(x) - \varphi(x)\| = \max_{x\in[a,b]} |f(x) - \varphi(x)| = \max_{x\in[a,b]} \frac{1}{n} |\sin(n^2 x)| \leqq \frac{1}{n}.$$

Für hinreichend großes n wird das Restglied also beliebig klein. Betrachten wir nun die erste Ableitung von f. Hier gilt

$$f'(x) = \varphi'(x) + n\cos(n^2 x).$$

Daraus folgt

$$\|f'(x) - \varphi'(x)\| = \max_{x\in[a,b]} |f'(x) - \varphi'(x)| = \max_{x\in[a,b]} n|\cos(n^2 x)|.$$

Der Unterschied der Ableitungen von f und φ wird mit n beliebig groß. ♡

Man könnte nun annehmen, die Schwierigkeiten liegen in dem speziellen Beispiel. Dem ist aber nicht so. Bei der Realisierung einer Funktion auf einem Rechner werden die Funktionswerte durch Rundungs- und Verfahrensfehler gerade so verfälscht, dass sie um die wahren Funktionswerte schwanken. Die Situation entspricht also in gewisser Weise der Beziehung zwischen den Funktionen φ und f in unserem Beispiel. Dies zeigt eine prinzipielle Schwierigkeit des numerischen Differenzierens. Kleine Änderungen der Eingabedaten (Funktionswerte) bewirken i. a. große Änderungen in den Ausgabedaten (Ableitungswerte). Numerisches Differenzieren ist damit eine **inkorrekt gestellte Aufgabe**. Trotzdem ist es in vielen Fällen notwendig, Ableitungen von Funktionen zu approximieren, so dass wir uns mit dieser Aufgabe beschäftigen. Die einfachste Möglichkeit, zu Approximationen der ersten Ableitung einer Funktion f zu kommen, ist die, f durch ein Polynom φ zu ersetzen. Dabei wird φ durch gewisse Interpolationsbedingungen $\varphi(x_i) = f(x_i) = f_i$ festgelegt. Betrachten wir den Fall der linearen und den der quadratischen Interpolation. Es sei eine einmal stetig differenzierbare Funktion f gegeben. Die Ableitung ist an der Stelle $\bar{x}$ zu bestimmen.
Fall 1: (**Lineare Interpolation**) $x_0 = \bar{x},\ x_1 = \bar{x} + h,\ \varphi \in \Pi_1$. In der LAGRANGEschen Darstellung erhalten wir das Interpolationspolynom

$$\begin{aligned}\varphi(x) &= f(\bar{x})\frac{x-(\bar{x}+h)}{\bar{x}-(\bar{x}+h)} + f(\bar{x}+h)\frac{x-\bar{x}}{(\bar{x}+h)-\bar{x}} \\ &= \frac{f(\bar{x}+h)-f(\bar{x})}{h}x + \frac{(\bar{x}+h)f(\bar{x}) - \bar{x}f(\bar{x}+h)}{h}.\end{aligned}$$

Differentiation liefert den bekannten Differenzenquotienten

$$\varphi'(x) = \frac{f(\bar{x}+h)-f(\bar{x})}{h} = \varphi'(\bar{x}).$$

Fall 2: (**Quadratische Interpolation**) $x_0 = \bar{x}-h$, $x_1 = \bar{x}$, $x_2 = \bar{x}+h$, $\varphi \in \Pi_2$. In der LAGRANGEschen Darstellung erhalten wir das Interpolationspolynom

$$\begin{aligned}
\varphi(x) &= f(\bar{x}-h)\frac{(x-\bar{x})(x-(\bar{x}+h))}{((\bar{x}-h)-\bar{x})((\bar{x}-h)-(\bar{x}+h))} \\
&\quad + f(\bar{x})\frac{(x-(\bar{x}-h))(x-(\bar{x}+h))}{(\bar{x}-(\bar{x}-h))(\bar{x}-(\bar{x}+h))} \\
&\quad + f(\bar{x}+h)\frac{(x-(\bar{x}-h))(x-\bar{x})}{((\bar{x}+h)-(\bar{x}-h))((\bar{x}+h)-\bar{x})} \\
&= f(\bar{x}-h)\frac{x^2-x(2\bar{x}+h)+\bar{x}(\bar{x}+h)}{2h^2} + f(\bar{x})\frac{x^2-2\bar{x}x+\bar{x}^2-h^2}{h^2} \\
&\quad + f(\bar{x}+h)\frac{x^2-x(2\bar{x}-h)+\bar{x}(\bar{x}-h)}{2h^2} \\
&= \frac{f(\bar{x}-h)-2f(\bar{x})+f(\bar{x}+h)}{2h^2}x^2 \\
&\quad - \frac{(2\bar{x}+h)f(\bar{x}-h)-4\bar{x}f(\bar{x})+(2\bar{x}-h)f(\bar{x}+h)}{2h^2}x \\
&\quad + \frac{(\bar{x}^2+\bar{x}h)f(\bar{x}-h)-2(\bar{x}^2-h^2)f(\bar{x})+(\bar{x}^2-\bar{x}h)f(\bar{x}+h)}{2h^2}.
\end{aligned}$$

Differentiation und Auswertung an der Stelle $x = \bar{x}$ liefert den zentralen Differenzenquotienten:

$$\begin{aligned}
\varphi'(x) &= \frac{f(\bar{x}-h)-2f(\bar{x})+f(\bar{x}+h)}{h^2}x - \\
&\quad - \frac{(2\bar{x}+h)f(\bar{x}-h)-4\bar{x}f(\bar{x})+(2\bar{x}-h)f(\bar{x}+h)}{2h^2} \\
\varphi'(\bar{x}) &= \frac{-hf(\bar{x}-h)+hf(\bar{x}+h)}{2h^2} = \frac{f(\bar{x}+h)-f(\bar{x}-h)}{2h}.
\end{aligned}$$

Verwendet man mehr Stützstellen, erhält man entsprechende Differentiationsformeln aus

$$\begin{aligned}
f(x) &= \sum_{i=0}^{n} f_i L_i^{(n)}(x) + R_n(x) \\
\frac{d}{dx}f(x) &= \sum_{i=0}^{n} f_i \frac{d}{dx}L_i^{(n)}(x) + \frac{d}{dx}R_n(x).
\end{aligned}$$

Üblich sind äquidistante Stützstellen, die symmetrisch zu der Stelle liegen, an der die Ableitung berechnet werden soll. Es ergeben sich Formeln der Art

$$f'(\bar{x}) = \frac{1}{h}\sum_{i=-k}^{k} \beta_i^{(k)} f_i + r_{2k}(f;\bar{x})$$

mit

$$x_i = \bar{x} + ih, \quad f_i = f(x_i), \quad i = -k, \ldots, k.$$

In der folgenden Tabelle sind die einfachsten symmetrischen Differentiationsformeln angegeben. Auf demselben Weg lassen sich auch höherer Ableitungen

2k	$s\beta_i^{(k)}$	s	$r_{2k}(f;\bar{x})$
2	-1 0 1	2	$-\frac{h^2}{6}f'''(\xi)$
4	1 -8 0 8 -1	12	$\frac{h^4}{30}f^{(5)}(\xi)$
6	-1 9 -45 0 45 -9 1	60	$-\frac{h^6}{140}f^{(7)}(\xi)$

Table 4.1: Symmetrische Differentiationsformeln für die 1.Ableitung

approximieren. Für die zweite Ableitung ergeben sich Formeln der Art

$$f''(\bar{x}) = \frac{1}{h^2}\sum_{i=-k}^{k} \gamma_i^{(k)} f_i + \bar{r}_{2k}(f;\bar{x}).$$

Die beiden einfachsten symmetrischen Formeln sind in der folgenden Tabelle angegeben.

2k	$s\gamma_i^{(k)}$	s	$\bar{r}_{2k}(f;\bar{x})$
2	1 -2 1	1	$-\frac{h^2}{12}f^{(4)}(\xi)$
4	-1 16 -30 16 1	12	$\frac{h^4}{90}f^{(6)}(\xi)$

Table 4.2: Symmetrische Differentiationsformeln für die 2.Ableitung

4.2. Der Fehler bei interpolatorischer Differentiation

Zuerst wollen wir ein Restglieddarstellung für den Differentiationsfehler herleiten, die eine Verallgemeinerung der Restglieddarstellung der Polynominterpolation darstellt.

4.2. Satz: *Es sei* $f \in C^{n+1}[a,b]$ *und* $a \leqq x_0 < x_1 < \cdots < x_n \leqq b$. $P_n \in \Pi_n$ *sei das Interpolationspolynom, das die Bedingungen*

$$P_n(x_i) = f(x_i), \quad i = 0,1,\ldots,n$$

erfüllt. Dann gilt für jedes $k \leqq n$

$$R_n^{(k)}(x) = f^{(k)}(x) - P_n^{(k)}(x) = \prod_{j=0}^{n-k}(x-\xi_j^{(k)})\frac{f^{(n+1)}(\eta_k)}{(n+1-k)!},$$

wobei die $n+1-k$ *Punkte* $\xi_j^{(k)}$, $j = 0,1,\ldots,n-k$, *nicht von* x *abhängen. Es gilt*

$$\xi_j^{(k)} \in (x_j, x_{j+k}), \quad j = 0,1,\ldots,n-k,$$
$$\eta_k = \eta_k(x) \in (\min\{x,x_0,\ldots,x_n\}, \max\{x,x_0,\ldots,x_n\}).$$

Beweis: Das Restglied

$$R_n(x) = f(x) - P_n(x)$$

ist $(n+1)$-mal stetig differenzierbar. Weiterhin gilt $R_n(x_i) = 0$ für $i = 0,\ldots,n$. Eine k-malige Anwendung des Satzes von ROLLE liefert für die Lage der Nullstellen höherer Ableitungen folgende Aussagen. Die Nullstellen von $R_n^{(1)}$ liegen in den Intervallen (x_i, x_{i+1}), $i = 0,\ldots,n-1$, die Nullstellen von $R_n^{(2)}$ liegen in den Intervallen (x_i, x_{i+2}), $i = 0,\ldots,n-2$, usw. Die Nullstellen $\xi_j^{(k)}$, $j = 0,\ldots,n-k$, der k-ten Ableitung von R_n liegen damit in den Intervallen (x_i, x_{i+k}), $i = 0,\ldots,n-k$. Weiterhin erkennt man, dass sie nur von der Funktion f und von den Stützstellen abhängen. Es sei nun

$$F_k(x) = R_n^{(k)}(x) - \alpha_k \prod_{j=0}^{n-k}(x-\xi_j^{(k)}).$$

$F_k(x)$ hat die Nullstellen $\xi_j^{(k)}$, $j = 0,\ldots,n-k$. Wir bestimmen nun α_k so, dass auch $F_k(\bar{x}) = 0$ gilt. Dann hat $F_k(x)$ $n-k+2$ Nullstellen im Intervall

$$I = (\min\{x,x_0,\ldots,x_n\}, \max\{x,x_0,\ldots,x_n\}).$$

Außerdem ist $F_k(x)$ $(n-k+1)$-mal stetig differenzierbar. $(n-k+1)$-malige Anwendung des Satzes von ROLLE zeigt, dass dann $F_k^{(n-k+1)}$ eine Nullstelle η_k

in I besitzt. Es folgt

$$\begin{aligned} 0 &= F_k^{(n-k+1)}(\eta_k) = R_n^{(n+1)}(\eta_k) - \alpha_k(n-k+1)! \\ &= f^{(n+1)}(\eta_k) - \alpha_k(n-k+1)! \end{aligned}$$

Damit ergibt sich

$$\alpha_k = \frac{f^{(n+1)}(\eta_k)}{(n-k+1)!}$$

und

$$R_n^{(k)}(x) = \frac{f^{(n+1)}(\eta_k)}{(n-k+1)!} \prod_{j=0}^{n-k}(x-\xi_j^{(k)}).$$

❀

Eine andere Fehlerdarstellung erhält man, falls man analog zur PEANOschen Darstellung des Integrationsfehlers vorgeht. Schaut man sich den Beweis des Satzes 3.2 an, so erkennt man, dass der wesentliche Beweisschritt im Vertauschen von Integration und Anwendung des Funktionals $I(f)-Q_n(f)$ besteht. Damit gilt aber ein entsprechender Satz für jedes Funktional F, das mit der Integration vertauschbar ist. Fassen wir die Differentiationsformel als Funktional auf, so ergibt sich der folgenden Satz.

4.3. Satz: *Es sei*

$$D(f;h)(x) = \frac{1}{h}\sum_{i=0}^{n}\beta_i f(x_i)$$

eine Differentiationsformel zur Approximation der 1.Ableitung einer Funktion f an der Stelle x. Weiterhin gelte für Polynome $p \in \Pi_m$

$$D(p;h) = p'(x)$$

für beliebige x und beliebige Schrittweiten $h>0$. Dann gilt für $f \in C^{l+1}[a,b]$ mit $0 \leqq l \leqq m$ die Restglieddarstellung

$$R_n(f) = f'(x) - D(f;h)(x) = \int_a^b f^{(l+1)}(t)K_l(t)\,dt$$

mit

$$\begin{aligned} K_l(t) &= \frac{1}{l!} R_n\left[(x-t)_+^l\right] \\ &= \frac{1}{l!}\left[\frac{d}{dx}(x-t)_+^l - \sum_{i=0}^{n} \beta_i (x_i - t)_+^l\right]. \end{aligned}$$

Dabei ist $[a,b] = [\min\{x,x_0,\ldots,x_n\}, \max\{x,x_0,\ldots,x_n\}]$. *Die Funktion* $K_l(t)$ *heißt* PEANO-*Kern des Restgliedes* R_n.

Bemerkung: Falls der PEANO-Kern ein konstantes Vorzeichen auf dem Intervall $[a,b]$ besitzt, so führt die Anwendung des verallgemeinerten Mittelwertsatzes der Integralrechnung wieder auf ein Restglied der Form

$$R_n(f) = \frac{f^{(l+1)}(\xi)}{(l+1)!} R_n(x^{l+1}).$$

4.3. Extrapolationsverfahren

Auch bei der numerischen Differentiation lassen sich mittels Extrapolationsverfahren aus bekannten Näherungen für die Ableitung an einer Stelle bessere Näherungen berechnen. Wir wollen die Vorgehensweise an Hand des zentralen Differenzenquotienten demonstrieren. Er ist durch

$$D(f;h)(x) = \frac{f(x+h) - f(x-h)}{2h}$$

gegeben. Es sei $f \in C^{2m+3}[x-a, x+a]$ und $|h| < a$. Eine TAYLOR-Entwicklung von f um den Punkt x liefert

$$\begin{aligned} f(x+h) &= f(x) + hf'(x) + \frac{h^2}{2} f''(x) + \cdots + \frac{h^{2m+2}}{(2m+2)!} f^{(2m+2)}(x) \\ &\quad + \frac{h^{2m+3}}{(2m+3)!} f^{(2m+3)}(\xi_+) \\ f(x-h) &= f(x) - hf'(x) + \frac{h^2}{2} f''(x) - \cdots + \frac{h^{2m+2}}{(2m+2)!} f^{(2m+2)}(x) \\ &\quad - \frac{h^{2m+3}}{(2m+3)!} f^{(2m+3)}(\xi_-) \end{aligned}$$

mit $\xi_+ \in (x, x+h)$ und $\xi_- \in (x-h, x)$. Setzt man diese Entwicklungen in den zentralen Differenzenquotienten ein, so erhält man

$$\begin{aligned} D(f;h)(x) &= f'(x) + \frac{h^2}{3!}f'''(x) + \frac{h^4}{5!}f^{(5)}(x) + \cdots + \frac{h^{2m}}{(2m+1)!}f^{(2m+1)}(x) \\ &\quad + \frac{h^{2m+2}}{(2m+3)!}\frac{f^{(2m+3)}(\xi_+) + f^{(2m+3)}(\xi_-)}{2} \\ &= f'(x) + \frac{h^2}{3!}f'''(x) + \frac{h^4}{5!}f^{(5)}(x) + \cdots + \frac{h^{2m}}{(2m+1)!}f^{(2m+1)}(x) \\ &\quad + \frac{h^{2m+2}}{(2m+3)!}f^{(2m+3)}(\xi) \end{aligned}$$

mit $\xi \in (x-h, x+h)$. Wir erhalten also wieder eine Entwicklung in h^2, wie wir sie schon beim ROMBERG-Verfahren kennengelernt hatten. Damit gehen wir hier analog vor.

4.4. Extrapolationsverfahren zur numerischen Differentiation:

Wähle Schrittweitenfolge $\{h_k\}$ *und Extrapolationstiefe* m.
for $k = 0$ ***to*** m ***do***
  $D_{k0} = \dfrac{f(x+h_k) - f(x-h_k)}{2h_k}$
  for $i = 1$ ***to*** k ***do***
    $D_{ki} = D_{k,i-1} + \dfrac{D_{k,i-1} - D_{k-1,i-1}}{\left(\dfrac{h_{k-i}}{h_k}\right)^2 - 1}$
  endfor
endfor

Die Bemerkungen und Fehlerabschätzungen vom ROMBERG-Verfahren übertragen sich in analoger Weise. Statt des zentralen Differenzenquotienten darf man auch eine andere Differentiationsformel anwenden. Man beachte aber, dass man eine symmetrische Formel anwenden sollte, damit man Entwicklungen in h^2 erhält. Weitere Differentiationsformeln lassen sich auch mit anderen Interpolationsarten, wie zum Beispiel der Spline-Interpolation, gewinnen.

4.4. Aufgaben

1. Man leite mit Hilfe der Spline-Interpolation eine Formel zur numerischen Differentiation her, die die Stützstellen $x-2h, x-h, x, x+h$ und $x+2h$ verwendet. Die Ableitung soll an der Stelle x berechnet werden, h sei eine

beliebige Schrittweite. Man verwende kubische Splines mit natürlichen Randbedingungen.

2. Die erste Ableitung einer Funktion f soll an der Stelle x näherungsweise berechnet werden. Wie groß ist der relative Rundungsfehler, falls eine der folgenden Differentiationsformeln angewendet werden?

   (a)

   $$f'(x) \approx \frac{f(x+h) - f(x)}{h}$$

   (b)

   $$f'(x) \approx \frac{f(x+h) - f(x-h)}{2h}$$

   Dabei sei h eine Zweierpotenz, so dass Multiplikation mit h und Division durch h oder $2h$ exakt ausgeführt werden.
   Was ergibt sich für $f(x) = ax + b$, falls a und b Maschinenzahlen sind?
   Was ergibt sich für $h \longrightarrow 0$?

Chapter 5

Eindimensionale Nullstellen

5.1. Einfache Iterationsverfahren

Wir betrachten die folgende Aufgabe:

> Gegeben sei eine Funktion $f : D \subseteq \mathbb{R} \longrightarrow \mathbb{R}$.
> Man finde einen Punkt $x^* \in D$, der Nullstelle der Funktion ist:
>
> $$f(x^*) = 0.$$

Es ist klar, dass die Aufgabe in dieser Allgemeinheit keine Lösung hat. Wegen der großen Allgemeinheit des Funktionsbegriffes gibt es auch keine notwendig und hinreichenden Bedingungen für die Lösbarkeit der Aufgabe. Der folgende Satz liefert hinreichende Bedingungen für die Lösbarkeit.

5.1. Satz: *Es sei* $f \in C[a,b]$ *und* $f(a)f(b) < 0$. *Dann existiert ein* $x^* \in (a,b)$ *mit* $f(x^*) = 0$.

Beweis: Nach dem Zwischenwertsatz für stetige Funktionen nimmt f auf $[a,b]$ alle Werte zwischen $f(a)$ und $f(b)$ an. Da laut Voraussetzung $f(a)$ und $f(b)$ von unterschiedlichem Vorzeichen sind, nimmt f auf (a,b) auch den Wert Null an. ❀

Wir erhalten als Anwendung dieses Satzes sofort ein einfaches Verfahren zur Nullstellenbestimmung einer stetigen Funktion.

5.2. Bisektionsverfahren:
Es sei $f \in C[a,b]$, $f(a) < 0$ *und* $f(b) > 0$.

S0 *Setze* $a_0 = a$, $b_0 = b$ *und* $k = 0$.

S1 *Berechne*

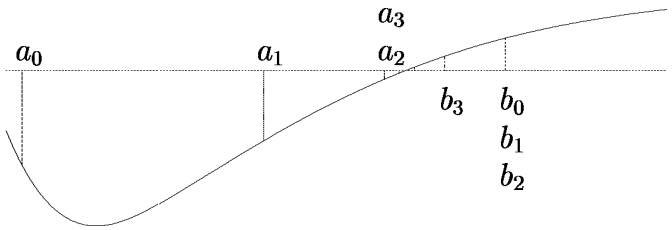
$$\xi = \frac{a_k + b_k}{2}, \quad \eta = f(\xi).$$

S2 *Für*

$$\eta \begin{cases} > 0 & a_{k+1} = a_k, \quad b_{k+1} = \xi, \\ = 0 & x^* = \xi, \quad \textit{STOPP}, \\ < 0 & a_{k+1} = \xi, \quad b_{k+1} = b_k. \end{cases}$$

S3 *Setze* $k = k+1$ *und gehe zu Schritt* **S1**.

Im folgenden Bild ist das Verfahren an einem Beispiel dargestellt.



Das Verfahren bricht ab, falls zufällig ξ Nullstelle von f ist. Ansonsten benötigt man noch eine geeignete Abbruchbedingung, wie etwa $|\frac{b_k - a_k}{a_k}| \leq \text{eps}$. Das Bisektionsverfahren liefert im nichtabbrechenden Falle unter den angegebenen Voraussetzungen eine Folge von ineinander liegenden Intervallen, wobei deren Intervall-Längenfolge eine Nullfolge ist; daher erhält man im Grenzfall mit Sicherheit eine Nullstelle der Funktion f. Diese Eigenschaft bezeichnen wir als **globale Konvergenz**. Wie wir später sehen werden, ist die Konvergenzgeschwindigkeit bei diesem Verfahren aber nicht besonders hoch.
Andere Verfahren erhält man, wenn man die Funktion f in der Nähe einer Nullstelle in geeigneter Weise durch eine einfachere Funktion ersetzt. Es sei dazu f in einer Umgebung einer Nullstelle x^* k-mal stetig differenzierbar. Dann gibt es für die Funktion f in einem beliebigen Punkte x_0 aus dieser Umgebung eine TAYLOR-Entwicklung. Man erhält

$$\begin{aligned} f(x) &= f(x_0 + (x - x_0)) \\ &= f(x_0) + (x - x_0) f'(x_0) + \frac{(x - x_0)^2}{2} f''(x_0) + \cdots \\ &\quad + \frac{(x - x_0)^k}{k!} f^{(k)}(x_0 + \theta(x - x_0)) \end{aligned}$$

mit $\theta \in (0,1)$. Vernachlässigen wir nun höhere Potenzen von $(x - x_0)$ und bestimmen eine Nullstelle $\bar{x}$ dieser so entstehenden Ersatzfunktion, so ist diese

Nullstelle im allgemeinen eine bessere Näherung für x^* als x_0. Ein Abbruch nach dem linearen Glied liefert

$$0 = f(x_0) + (\bar{x} - x_0) f'(x_0)$$

und damit die neue Näherung

$$\bar{x} = x_0 - \frac{f(x_0)}{f'(x_0)}.$$

Nimmt man noch den quadratischen Term der TAYLOR-Entwicklung mit, so erhält man

$$0 = f(x_0) + (\bar{x} - x_0) f'(x_0) + \frac{(\bar{x} - x_0)^2}{2} f''(x_0)$$

und

$$\bar{x} = x_0 - \frac{f'(x_0) \pm \sqrt{(f'(x_0))^2 - 2f(x_0)f''(x_0)}}{f''(x_0)}.$$

Die obigen Formeln sind als Iterationsverfahren aufzufassen. Man erhält im ersten Falle das NEWTON-RAPHSON-Verfahren 1. Art (oder kurz NEWTON-Verfahren).

5.3. NEWTON-RAPHSON**-Verfahren 1. Art:**
Es sei $f \in C^1[a,b]$.

S0 *Wähle ein* x_0 *und setze* $k = 0$.

S1 *Berechne*

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

S2 *Setze* $k = k+1$ *und gehe zu Schritt* **S1**.

Im zweiten Falle ergeben sich die NEWTON-RAPHSON-Verfahren 2. Art.

5.4. NEWTON-RAPHSON**-Verfahren 2. Art:**
Es sei $f \in C^2[a,b]$.

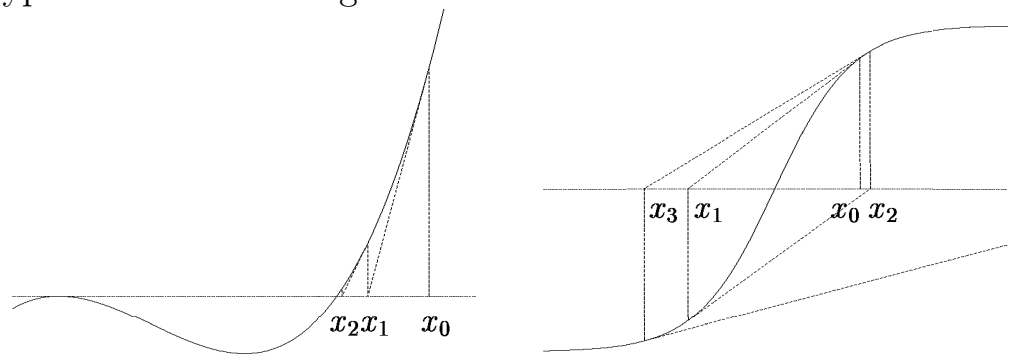
S0 *Wähle ein* x_0 *und setze* $k = 0$.

S1 *Berechne*

$$x_{k+1} = x_k - \frac{f'(x_k) \pm \sqrt{(f'(x_k))^2 - 2f(x_k)f''(x_k)}}{f''(x_k)}.$$

S2 *Setze* $k = k+1$ *und gehe zu Schritt* **S1**.

In beiden Algorithmen fehlen noch geeignete Abbruchbedingungen. Dazu hat man $|x_{k+1} - x_k|$ und $f(x_{k+1})$ zu untersuchen.
Beim NEWTON-RAPHSON-Verfahren 1. Art ersetzen wir die Funktion f durch die Tangente im Iterationspunkt $(x_k, f(x_k))$. Die beiden folgenden Bilder zeigen das typisch lokale Konvergenzverhalten.



Im linken Bild liegt der Startpunkt nahe genug an der Nullstelle. Das Verfahren konvergiert. Im rechten Bild entfernen sich die Iterationspunkte immer mehr von der Nullstelle. Das Verfahren divergiert. Im nächsten Abschnitt werden wir das Konvergenzverhalten des NEWTON-Verfahrens genauer untersuchen.
Verwendet man statt der Tangente im Punkt $(\bar{x}, f(\bar{x}))$ die Sekante durch zwei Punkte $(\bar{x}, f(\bar{x}))$ und $(\tilde{x}, f(\tilde{x}))$ zur Bestimmung einer nächsten Näherung, so erhält man ein Verfahren, das als *regula falsi* bezeichnet wird. Es gibt drei Varianten dieses Verfahrens.
Es sei $f \in C[a,b]$ und $f(a)f(b) < 0$. Die Sekante durch die Punkte $(a, f(a))$ und $(b, f(b))$ hat die Gleichung

$$\varphi(x) = f(a)\frac{x-b}{a-b} + f(b)\frac{x-a}{b-a}.$$

Als Nullstelle dieser Geraden erhält man

$$\xi = \frac{af(b) - bf(a)}{f(b) - f(a)}.$$

Offensichtlich gilt $a < \xi < b$. Wie beim Bisektionsverfahren wählt man nun wieder von den beiden Intervallen $[a, \xi]$ und $[\xi, b]$ das Intervall aus, in dem ein Vorzeichenwechsel auftritt. Wir erhalten folgenden Algorithmus.

5.5. Regula falsi 1:
Es sei $f \in C[a,b]$, $f(a) < 0$ *und* $f(b) > 0$.
S0 *Setze* $a_0 = a$, $b_0 = b$ *und* $k = 0$.

S1 *Berechne*

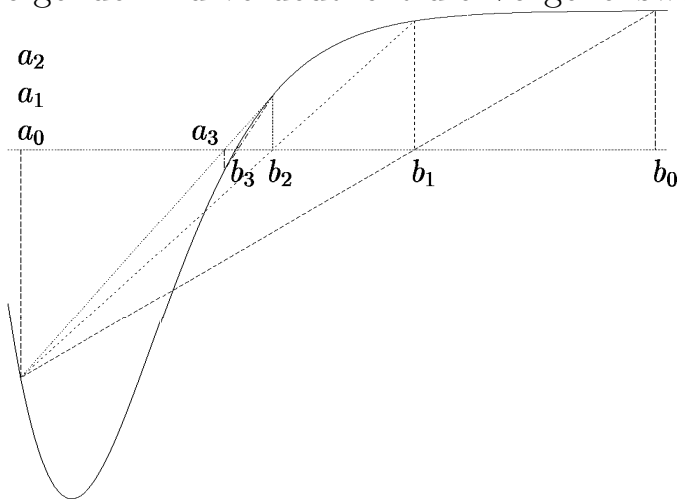
$$\xi = \frac{a_k f(b_k) - b_k f(a_k)}{f(b_k) - f(a_k)} = a_k - \frac{b_k - a_k}{f(b_k) - f(a_k)} f(a_k),$$
$$\eta = f(\xi).$$

S2 *Für*

$$\eta \begin{cases} > 0 & a_{k+1} = a_k, \quad b_{k+1} = \xi, \\ = 0 & x^* = \xi, \quad STOPP, \\ < 0 & a_{k+1} = \xi, \quad b_{k+1} = b_k. \end{cases}$$

S3 *Setze* $k = k+1$ *und gehe zu Schritt* **S1**.

In Bezug auf Abbruchbedingungen gilt das beim Bisektionsverfahren gesagte. Das folgende Bild verdeutlicht die Vorgehensweise.



Hält man eine der Intervallgrenzen fest und verzichtet im Schritt **S2** auf die Untersuchung des Vorzeichenwechsels, so erhält man eine zweite Variante der Regula falsi.

5.6. Regula falsi 2:

Es sei $f \in C[a,b]$, $f(a) < 0$ *und* $f(b) > 0$.

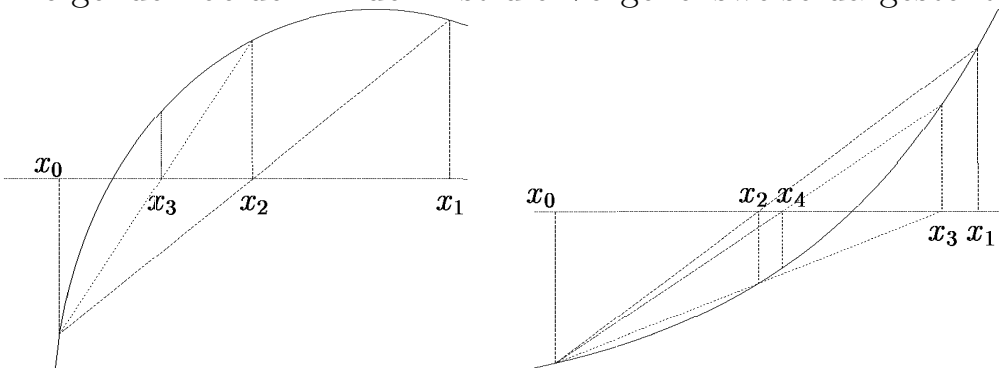
S0 *Setze* $x_0 = a$, $x_1 = b$ *und* $k = 1$.

S1 *Berechne*

$$x_{k+1} = x_k - \frac{x_0 - x_k}{f(x_0) - f(x_k)} f(x_k).$$

S2 *Setze* $k = k+1$ *und gehe zu Schritt* **S1**.

Das Verfahren konvergiert lokal und nur unter bestimmten Voraussetzungen. In den folgenden beiden Bildern ist die Vorgehensweise dargestellt.



Die dritte Variante der Regula falsi erhält man, falls man die Sekante jeweils durch die letzten beiden Iterationspunkte legt.

5.7. Regula falsi 3 oder Sekantenverfahren:
Es sei $f \in C[a,b]$.

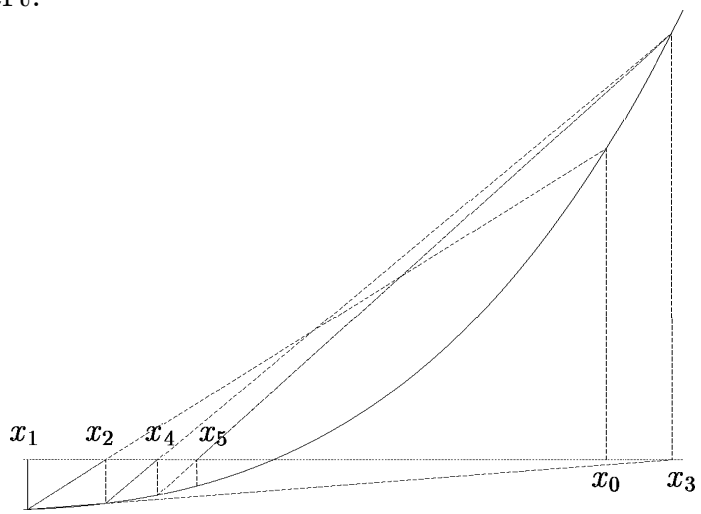
S0 *Wähle* $x_0 \in [a,b]$ *und* $x_1 \in [a,b]$ *und setze* $k = 1$.

S1 *Berechne*

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

S2 *Setze* $k = k+1$ *und gehe zu Schritt* **S1**.

Im folgenden Bild ist die Wirkungsweise des Verfahrens an einem Beispiel erläutert.



Auch dieses Verfahren konvergiert nur lokal. Abbruchbedingungen sind für die letzten beiden Verfahren noch zu formulieren.

5.2. Konvergenzbetrachtungen

In diesem Abschnitt wollen wir einige Konvergenzsätze behandeln, die für die Beurteilung verschiedener Verfahren wichtig sind. Dabei beschränken wir uns nicht auf den Fall reellwertiger Funktionen einer Veränderlichen, sondern wir betrachten eine Funktion

$$\Phi : \mathbb{R}^n \longrightarrow \mathbb{R}$$

und eine Folge

$$\{x_i\}_{i\in\mathbb{N}} \subset \mathbb{R}^n$$

mit

$$x_{i+1} = \Phi(x_i), \quad i = 0, 1, \ldots$$

Das sind Iterationsverfahren wie zum Beispiel das NEWTON-RAPHSON-Verfahren oder die Regula falsi 2 und 3.
Weiterhin sei $\|\circ\| : \mathbb{R}^n \longrightarrow \mathbb{R}^n_+$ eine beliebige Norm auf dem $\mathbb{R}^n$. Es seien $\{x_i\}_{i\in\mathbb{N}}$ eine durch die Iterationsfunktion Φ erzeugte Folge und $\xi \in \mathbb{R}^n$ ein Fixpunkt von Φ: $\Phi(\xi) = \xi$. Wenn eine Umgebung $U(\xi)$ und eine Zahl p existieren, so dass für einen beliebigen Startpunkt $x_0 \in U(\xi)$ die Ungleichung

$$\|x_{i+1} - \xi\| \leqq C\|x_i - \xi\|^p, \quad i = 0, 1, \ldots$$

(und $C < 1$ im Falle $p = 1$) gilt, so heißt das von Φ erzeugte Iterationsverfahren ein Verfahren von mindestens p-ter Ordnung. Für Verfahren der Ordnung p gilt der folgende Satz.

5.8. Satz: *Jedes Iterationsverfahren der Ordnung p zur Bestimmung eines Fixpunktes $\xi = \Phi(\xi)$ ist in folgendem Sinne lokal konvergent:*
Es existiert eine Umgebung $U(\xi)$, so dass für jeden Startpunkt $x_0 \in U(\xi)$, die durch Φ erzeugte Folge $\{x_i\}_{i\in\mathbb{N}}$ gegen ξ konvergiert.

Beweis: (i) $p = 1$
Es sei $U_0(\xi)$ eine Umgebung, in der für alle $x_0 \in U_0(\xi)$

$$\|x_{i+1} - \xi\| \leqq C\|x_i - \xi\|, \quad i = 0, 1, \ldots$$

mit $C < 1$ gilt. Dann gilt offensichtlich

$$\|x_i - \xi\| \leqq C^i\|x_0 - \xi\|, \quad i = 0, 1, \ldots$$

und damit

$$0 \leqq \lim_{i\to\infty} \|x_i - \xi\| \leqq \|x_0 - \xi\| \lim_{i\to\infty} C^i = 0.$$

Es folgt

$$\lim_{i\to\infty} x_i = \xi.$$

(ii) $p > 1$
Es sei $U_0(\xi)$ eine Umgebung, in der für alle $x_0 \in U_0(\xi)$

$$\|x_{i+1} - \xi\| \leqq C\|x_i - \xi\|^p, \quad i = 0, 1, \ldots$$

gilt. Weiterhin sei

$$U_1(\xi) = \left\{ x \;\middle|\; \|x - \xi\| \leqq C^{\frac{1}{1-p}} \right\}$$

und $U(\xi) = U_0(\xi) \cap U_1(\xi)$. Dann gilt für ein beliebiges $x_0 \in U(\xi)$

$$\begin{aligned}
\|x_i - \xi\| &\leqq C\|x_{i-1} - \xi\|^p \leqq C\,(C\|x_{i-2} - \xi\|^p)^p = C^{p+1}\|x_{i-2} - \xi\|^{p^2} \\
&\leqq C^{p^2+p+1}\|x_{i-3} - \xi\|^{p^3} \\
&\vdots \\
&\leqq C^{p^{i-1}+\cdots+p+1}\|x_0 - \xi\|^{p^i} \\
&= C^{\frac{p^i-1}{p-1}}\|x_0 - \xi\|^{p^i} \\
&= C^{\frac{1}{1-p}}\left(C^{\frac{1}{p-1}}\|x_0 - \xi\|\right)^{p^i}.
\end{aligned}$$

Wegen $x_0 \in U_1(\xi)$ folgt

$$C^{\frac{1}{p-1}}\|x_0 - \xi\| < 1$$

und damit

$$0 \leqq \lim_{i\to\infty} \|x_i - \xi\| \leqq C^{\frac{1}{1-p}} \lim_{i\to\infty} \left(C^{\frac{1}{p-1}}\|x_0 - \xi\|\right)^{p^i} = 0.$$

Es gilt also wieder

$$\lim_{i\to\infty} x_i = \xi.$$

❀

Im eindimensionalen Falle ($\Phi : \mathbb{R} \longrightarrow \mathbb{R}$) lässt sich die Ordnung des durch Φ erzeugten Verfahrens häufig leicht bestimmen.

5.9. Satz: *Es sei* $\Phi : \mathbb{R} \longrightarrow \mathbb{R}$ *und* $\Phi \in C^p(U(\xi))$. *Für die Ableitungen von* Φ *an der Stelle* ξ *gelte*

$$\Phi^{(i)}(\xi) = 0 \quad i = 1, \ldots, p-1, \quad \Phi^{(p)}(\xi) \neq 0.$$

Im Falle $p = 1$ *gelte zusätzlich* $|\Phi'(\xi)| < 1$. *Dann ist das durch* Φ *erzeugte Iterationsverfahren von* p*-ter Ordnung.*

Beweis: (i) $p > 1$
Für $x_i \in U(\xi)$ gilt

$$x_{i+1} = \Phi(x_i) = \Phi(\xi + (x_i - \xi)) = \Phi(\xi) + \frac{(x_i - \xi)^p}{p!}\Phi^{(p)}(\xi + \vartheta(x_i - \xi))$$

mit einem $\vartheta \in (0,1)$. Da ξ Fixpunkt von Φ ist folgt

$$|x_{i+1} - \xi| = \frac{\Phi^{(p)}(\xi + \vartheta(x_i - \xi))}{p!}|x_i - \xi|^p \leqq C|x_i - \xi|^p$$

mit

$$C = \frac{1}{p!} \sup_{x \in U(\xi)} |\Phi^{(p)}(x)|.$$

(ii) p=1
Hier gilt

$$|x_{i+1} - \xi| = |\Phi'(\xi + \vartheta(x_i - \xi))|\,|x_i - \xi|.$$

Aus $|\Phi'(\xi)| < 1$ und der Stetigkeit von Φ folgt, dass eine Umgebung $U_1(\xi)$ existiert, so dass für alle $x \in U_1(\xi)$ $|\Phi'(x) \leqq C < 1|$ gilt. Damit gilt für jedes $x_i \in U_1(\xi)$

$$|x_{i+1} - \xi| \leqq C|x_i - \xi|, \quad C < 1.$$

Konvergenz des Newton-Verfahrens

Betrachten wir zuerst das NEWTON-Verfahren für eine einfache Nullstelle der Funktion f. Es sei also $f(x^*) = 0$ und $f'(x^*) \neq 0$. Als Iterationsvorschrift erhalten wir

$$x_{i+1} = x_i - \frac{f(x_i)}{f'(x_i)}.$$

Wir haben also die Iterationsfunktion

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

zu untersuchen. Es gilt

$$\Phi'(x) = 1 - \frac{f'(x)}{f'(x)} + \frac{f(x)f''(x)}{f'(x)^2} = \frac{f(x)f''(x)}{f'(x)^2}$$

und damit

$$\Phi'(x^*) = 0.$$

Weiter ergibt sich

$$\Phi''(x) = \frac{f'(x)f''(x) - f(x)f'''(x)}{f'(x)^2} - 2\frac{f(x)f''(x)^2}{f'(x)^3}$$

und damit

$$\Phi''(x^*) = \frac{f''(x^*)}{f'(x^*)}.$$

Das NEWTON-Verfahren ist somit lokal mindestens quadratisch konvergent, falls x^* einfache Nullstelle der Funktion f ist.
Betrachten wir nun das NEWTON-Verfahren für eine m-fache Nullstelle der Funktion f $(m > 1)$. Es sei also

$$f^{(i)}(x^*) = 0, \quad i = 0, 1, \ldots, m-1$$

und

$$f^{(m)}(x^*) \neq 0.$$

Wir stellen f dann in der Form

$$f(x) = (x - x^*)^m g(x)$$

mit $g(x^*) \neq 0$ dar. Es folgt

$$f'(x) = m(x-x^*)^{m-1}g(x) + (x-x^*)^m g'(x)$$

und

$$f''(x) = m(m-1)(x-x^*)^{m-2}g(x) + 2m(x-x^*)^{m-1}g'(x) + (x-x^*)^m g''(x).$$

Damit gilt mit $h = x - x^*$

$$\begin{aligned}
\Phi'(x) &= \frac{f(x)f''(x)}{f'(x)^2} \\
&= \frac{h^m g(x)\left[m(m-1)h^{m-2}g(x) + 2mh^{m-1}g'(x) + h^m g''(x)\right]}{\left[mh^{m-1}g(x) + h^m g'(x)\right]^2} \\
&= \frac{h^{2m-2}g(x)\left[m(m-1)g(x) + 2mhg'(x) + h^2 g''(x)\right]}{h^{2m-2}\left[mg(x) + hg'(x)\right]^2} \\
&= \frac{g(x)\left[m(m-1)g(x) + 2mhg'(x) + h^2 g''(x)\right]}{\left[mg(x) + hg'(x)\right]^2}.
\end{aligned}$$

An der Stelle x^* erhalten wir

$$\Phi'(x^*) = \frac{g(x^*)m(m-1)g(x^*)}{m^2 g(x^*)^2} = \frac{m-1}{m} = 1 - \frac{1}{m}.$$

Für eine mehrfache Nullstelle ist das NEWTON-Verfahren also nur noch linear konvergent mit dem Faktor $C = 1 - \frac{1}{m}$. Das ist ein schlechteres Konvergenzverhalten als das des Bisektionsverfahrens.

Konvergenz des Bisektionsverfahrens

Das Bisektionsverfahren hat nicht die Form $x_{k+1} = \Phi(x_k)$. Trotzdem lässt sich auch hier die Konvergenzordnung bestimmen. Wir wissen aus der Konstruktion des Verfahrens, dass in jedem Intervall $[a_k, b_k]$ eine Nullstelle der Funktion f liegt. Diese Nullstelle sei x^*. Dann gilt

$$|a_{k+1} - x^*| \leqq |b_{k+1} - a_{k+1}| = \frac{1}{2}|b_k - a_k|$$

und

$$|b_{k+1} - x^*| \leqq |b_{k+1} - a_{k+1}| = \frac{1}{2}|b_k - a_k|.$$

Daraus folgt aber

$$|a_k - x^*| \leq \frac{1}{2^k}|b_0 - a_0| = \frac{b-a}{2^k}$$

und

$$|b_k - x^*| \leq \frac{1}{2^k}|b_0 - a_0| = \frac{b-a}{2^k}.$$

Die Intervallgrenzen konvergieren demnach linear gegen die Nullstelle x^* mit dem Faktor 1/2.

Konvergenz der Regula falsi 1 und 2

Die Variante 1 der Regula falsi ist wegen $x^* \in (a_k, b_k)$ und $b_{k+1} - a_{k+1} < b_k - a_k$ numerisch stabil. Sie liefert immer kleinere Einschließungen der Nullstelle x^*. Betrachten wir eine einfache Nullstelle ($f(x^*) \neq 0$) und fordern zusätzlich $f''(x^*) \neq 0$, so existiert eine Umgebung $U(x^*)$, in der f'' ein konstantes Vorzeichen besitzt. Wir wollen uns hier auf den Fall $f''(x^*) > 0$ und $f(a_k) < 0 < f(b_k)$ beschränken. Die Iterationsvorschrift liefert

$$\xi = a_k - \frac{a_k - b_k}{f(a_k) - f(b_k)} f(b_k) = b_k - \frac{a_k - b_k}{f(a_k) - f(b_k)} f(a_k).$$

Wegen

$$\frac{a_k - b_k}{f(a_k) - f(b_k)} > 0$$

folgt daraus $a_k < \xi < b_k$. Aus der Konvexität von f ($f''(x^*) > 0$) folgt $f(\xi) < 0$. Damit gilt $a_{k+1} = \xi$ und $b_{k+1} = b_k$. Die rechte Intervallgrenze bleibt fest und die linke Intervallgrenze ändert sich in jedem Schritt. Damit geht die Variante 1 der Regula falsi in diesem Falle in die Variante 2 über. Für die Untersuchung des lokalen Konvergenzverhaltens genügt es also, die Folge $\{x_k\}_{k \in \mathbb{N}}$ mit $x_{k+1} = \Phi(x_k)$ und

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

zu betrachten. Diese Folge ist monoton wachsend und beschränkt. Damit ist sie konvergent. Es sei

$$\bar{x} = \lim_{k \to \infty} x_k.$$

Aus der Stetigkeit von f folgt sofort $f(\bar{x}) \leqq 0$. Wegen $f(\bar{x}) < 0 < f(b)$ und der Tatsache, dass $\bar{x}$ Fixpunkt von Φ ist, also

$$\bar{x} = \bar{x} - \frac{\bar{x}-b}{f(\bar{x})-f(b)} f(\bar{x}),$$

folgt $(\bar{x}-b)f(\bar{x}) = 0$. Da aber aus $f(\bar{x}) < f(b)$ $\bar{x} \neq b$ folgt, erhalten wir $f(\bar{x}) = 0$. Das Verfahren konvergiert also gegen eine Nullstelle $\bar{x} = x^*$ von f. Nun lässt sich mit Satz 5.9 die Konvergenzordnung des Verfahrens bestimmen. Es gilt

$$\Phi'(x) = 1 - \frac{f(x)}{f(x)-f(b)} - \frac{x-b}{f(x)-f(b)} f'(x) + \frac{x-b}{(f(x)-f(b))^2} f(x) f'(x).$$

Damit folgt

$$\Phi'(x^*) = 1 - \frac{x^*-b}{f(x^*)-f(b)} f'(x^*).$$

Es ist also lineare Konvergenz zu erwarten. Wir haben noch zu zeigen, dass die Ungleichung $|\Phi'(x^*)| < 1$ gilt. Nach dem Mittelwertsatz der Differentialrechnung gilt

$$\frac{f(x^*)-f(b)}{x^*-b} = f'(\eta_1), \quad x^* < \eta_1 < b$$

und

$$\frac{f(x_i)-f(x^*)}{x_i-x^*} = f'(\eta_2), \quad x_i < \eta_2 < x^*.$$

Da wir $f''(x^*) > 0$ vorausgesetzt haben, ist f' in einer Umgebung $U(x^*)$ streng monoton wachsend. Es gilt also

$$f`(\eta_2) < f'(x^*) < f'(\eta_1).$$

Außerdem folgt aus $f(x_i) < 0 = f(x^*)$ und $x_i < x^*$ $f'(\eta_2) > 0$. Es gilt also

$$0 < \frac{f'(x^*)}{f'(\eta_1)} = \frac{x^*-b}{f(x^*)-f(b)} f'(x^*) < 1$$

und damit $0 < \Phi'(x^*) < 1$. Die Regula falsi 2 (und damit auch die Regula falsi 1) ist unter den obigen Bedingungen linear konvergent mit einem Faktor, der i. a. nahe bei 1 liegt. Sie lässt im allgemeinen keine besseren Resultate als das Bisektionsverfahren erwarten.

Konvergenz des Sekantenverfahrens

Das Sekantenverfahren ist nur lokal konvergent. Wir betrachten das Verfahren in einer hinreichend kleinen Umgebung der Nullstelle x^*. Es gilt

$$x_{i+1} = x_i - \frac{x_i - x_{i-1}}{f(x_i) - f(x_{i-1})} f(x_i)$$

$$\begin{aligned}
x_{i+1} - x^* &= x_i - x^* - \frac{x_i - x_{i-1}}{f(x_i) - f(x_{i-1})}[f(x_i) - f(x^*)] \\
&= (x_i - x^*)\left[1 - \frac{\frac{f(x_i)-f(x^*)}{x_i - x^*}}{\frac{f(x_i)-f(x_{i-1})}{x_i - x_{i-1}}}\right] = (x_i - x^*)\left[1 - \frac{f[x_i, x^*]}{f[x_{i-1}, x_i]}\right] \\
&= (x_i - x^*)\frac{f[x_{i-1}, x_i] - f[x_i, x^*]}{f[x_{i-1}, x_i]} \frac{x_{i-1} - x^*}{x_{i-1} - x^*} \\
&= (x_i - x^*)(x_{i-1} - x^*)\frac{f[x_{i-1}, x_i, x^*]}{f[x_{i-1}, x_i]}.
\end{aligned}$$

Aus dem Mittelwertsatz der Differentialrechnung folgt für zweimal stetig differenzierbares f

$$f[x_{i-1}, x_i] = f'(\eta_1), \quad \eta_1 \in (\min\{x_{i-1}, x_i\}, \max\{x_{i-1}, x_i\})$$

und

$$f[x_{i-1}, x_i, x^*] = \frac{1}{2} f''(\eta_2), \quad \eta_2 \in (\min\{x_{i-1}, x_i, x^*\}, \max\{x_{i-1}, x_i, x^*\}).$$

Ist x^* eine einfache Nullstelle ($f'(x^*) \neq 0$), so ist der Quotient in einer hinreichend kleinen Umgebung $U(x^*)$ der Nullstelle beschränkt. Es gilt

$$\left|\frac{f[x_{i-1}, x_i, x^*]}{f[x_i, x_{i-1}]}\right| \leq M = \sup\left\{\left|\frac{f''(\eta_2)}{2f'(\eta_1)}\right|, \eta_1 \in U(x^*), \eta_2 \in U(x^*)\right\} < \infty.$$

Damit erhält man

$$|x_{i+1} - x^*| \leq M|x_i - x^*|\,|x_{i-1} - x^*|$$

und

$$M|x_{i+1} - x^*| \leq M|x_i - x^*| M|x_{i-1} - x^*|.$$

Mit den Bezeichnungen $e_i = M|x_i - x^*|$ und $\delta = \max\{e_0, e_1\}$ folgt dann

$$e_{i+1} \leq e_i e_{i-1}, \quad i = 1, 2, \ldots,$$

und weiter

$$\begin{aligned} e_0 &\leqq \delta \\ e_1 &\leqq \delta \\ e_2 &\leqq \delta \cdot \delta = \delta^2 \\ e_3 &\leqq \delta^2 \cdot \delta = \delta^3 \\ e_4 &\leqq \delta^3 \cdot \delta^2 = \delta^5 \\ &\vdots \\ e_k &\leqq \delta^{m_k} \end{aligned}$$

mit $m_0 = m_1 = 1$ und $m_{k+1} = m_k + m_{k-1}$. Die Folge $\{m_k\}_{k\in\mathbb{N}}$ ist die bekannte FIBONACCI-Folge. Das Glied m_k hat die allgemeine Darstellung

$$m_k = \frac{1}{\sqrt{5}} \left[\left(\frac{1+\sqrt{5}}{2} \right)^{k+1} - \left(\frac{1-\sqrt{5}}{2} \right)^{k+1} \right].$$

Für große k gilt dann

$$m_k \approx \frac{1}{\sqrt{5}} \left(\frac{1+\sqrt{5}}{2} \right)^{k+1} \approx 0.7236 \cdot 1.618^k.$$

Wir erhalten insgesamt

$$|x_k - x^*| \leqq \frac{\delta^{0.7236}}{M} \cdot \delta^{1.618^k}.$$

Vergleicht man diese Ungleichung mit den Abschätzungen aus dem Beweis von Satz 5.9, so erkennt man, dass das Sekantenverfahren die Konvergenzordnung

$$q = \frac{1}{2}(1+\sqrt{5}) \approx 1.618$$

besitzt. Damit liegt das Verfahren von der Konvergenzordnung her zwischen Bisektionsverfahren und NEWTON-Verfahren. Da aber das Sekantenverfahren pro Schritt nur eine Funktionswertberechnung benötigt, sind zwei Schritte des Sekantenverfahrens in etwa so aufwendig wie ein Schritt des NEWTON-Verfahrens. Will man das Sekantenverfahren also mit dem NEWTON-Verfahren vergleichen, so müßte man ein "‘ Zwei-Schritt-Sekantenverfahren"’ betrachten. Dieses hat aber die Konvergenzordnung $q^2 = 2.618$ und ist damit günstiger als das NEWTON-Verfahren. Man beachte aber, dass bei der Iterationsvorschrift des Sekantenverfahrens in der Nähe der Nullstelle Auslöschung auftritt.
Von besonderer Bedeutung für die Untersuchung von Iterationsverfahren ist der BANACHsche Fixpunktsatz.

5.10. BANACHscher Fixpunktsatz: *Es sei* $(\mathbf{X}, \|\circ\|)$ *ein* BANACH*-Raum und*

$$\Phi : \mathbf{X} \longrightarrow \mathbf{X}$$

eine kontrahierende Abbildung auf $\mathbf{X}$*. Es gilt also*

$$\exists C < 1 : \forall x, y \in \mathbf{X} : \|\Phi(x) - \Phi(y)\| \leqq C\|x - y\|.$$

Dann besitzt Φ *genau einen Fixpunkt* $\xi = \Phi(\xi)$ *in* $\mathbf{X}$*. Jede durch* Φ *erzeugte Iterationsfolge* $\{x_i\}_{i\in\mathbb{N}}$ *konvergiert für beliebige Startwerte* $x_0 \in \mathbf{X}$ *gegen* ξ*.*

Beweis: Wir zeigen, dass $\{x_i\}_{i\in\mathbb{N}}$ CAUCHY-Folge ist.

$$\begin{aligned} \|x_{k+1} - x_k\| &= \|\Phi(x_k) - \Phi(x_{k-1})\| \\ &\leqq C\|x_k - x_{k-1}\| \leqq C^2\|x_{k-1} - x_{k-2}\| \ldots \leqq C^k\|x_1 - x_0\|. \end{aligned}$$

Damit gilt für $l > k$

$$\begin{aligned} \|x_l - x_k\| &= \|x_l - x_{l-1} + x_{l-1} - \cdots - x_{k+1} + x_{k+1} - x_k\| \\ &\leqq \|x_l - x_{l-1}\| + \|x_{l-1} - x_{l-2}\| + \cdots + \|x_{k+1} - x_k\| \\ &\leqq (C^{l-1} + C^{l-2} + \cdots + C^k)\|x_1 - x_0\| \\ &= C^k \frac{1 - C^{l-k}}{1 - C}\|x_1 - x_0\| \leqq \frac{C^k}{1 - C}\|x_1 - x_0\|. \end{aligned}$$

Offensichtlich existiert zu jedem $\varepsilon > 0$ ein k_0 mit

$$\frac{C^{k_0}}{1 - C}\|x_1 - x_0\| \leqq \varepsilon.$$

Dann gilt aber

$$\forall k, l \geqq k_0 : \|x_l - x_k\| \leqq \varepsilon.$$

Damit ist $\{x_i\}_{i\in\mathbb{N}}$ CAUCHY-Folge und konvergiert wegen der Abgeschlossenheit von $\mathbf{X}$ gegen einen Grenzwert $\xi \in \mathbf{X}$. Wir zeigen nun, dass ξ Fixpunkt von Φ ist. Es gilt

$$\begin{aligned} 0 \leqq \|\xi - \Phi(\xi)\| &= \|\xi - x_k + x_k - \Phi(\xi)\| \\ &\leqq \|\xi - x_k\| + \|x_k - \Phi(\xi)\| \\ &= \|\xi - x_k\| + \|\Phi(x_{k-1}) - \Phi(\xi)\| \\ &\leqq \|\xi - x_k\| + C\|x_{k-1} - \xi\|. \end{aligned}$$

Damit folgt

$$0 \leqq \|\xi - \Phi(\xi)\| \leqq \lim_{k\to\infty} (\|\xi - x_k\| + C\|x_{k-1} - \xi\|) = 0,$$

also

$$\xi = \Phi(\xi).$$

Es ist noch die Eindeutigkeit des Fixpunktes zu zeigen. Es sei $\eta \in \mathbf{X}$ ein weiterer Fixpunkt von Φ. Dann gilt

$$\|\xi - \eta\| = \|\Phi(\xi) - \Phi(\eta)\| \leqq C\|\xi - \eta\|,$$

und damit wegen $C < 1$

$$\|\xi - \eta\| \leqq 0.$$

Wegen der Normeigenschaft gilt aber auch

$$\|\xi - \eta\| \geqq 0,$$

also $\xi = \eta$. ❀

Die in diesem Satz geforderte globale Kontraktivität von Φ lässt sich zu einer lokalen Kontraktivität abgeschwächen.

5.11. Satz: *Es sei* $(\mathbf{X}, \|\circ\|)$ *ein* BANACH*-Raum und* $\Phi : \mathbf{X} \longrightarrow \mathbf{X}$ *eine Abbildung von* $\mathbf{X}$ *in* $\mathbf{X}$. *Zu* $x_0 \in \mathbf{X}$ *existiere eine Umgebung*

$$S_r(x_0) = \{\, x \mid \|x - x_0\| < r \,\} \subseteq \mathbf{X},$$

in der Φ *kontrahierend ist. Es existiert somit eine Konstante* $C < 1$, *so dass*

$$\text{für alle } x, y \in S_r(x_0) : \ \|\Phi(x) - \Phi(y)\| \leqq \|x - y\|$$

gilt. Weiterhin gelte

$$\|x_1 - x_0\| = \|\Phi(x_0) - x_0\| \leqq (1 - C)r < r.$$

Dann besitzt Φ *genau einen Fixpunkt* $\xi \in \overline{S_r(x_0)}$, *und die durch* Φ *mit dem Startpunkt* x_0 *erzeugte Folge* $\{x_k\}_{k\in\mathbb{N}}$ *konvergiert gegen* ξ.

Beweis: Wir zeigen nur, dass $x_k \in S_r(x_0)$ für alle k gilt. Der Rest folgt aus Satz 5.10.
Zuerst einmal gilt $x_1 \in S_r(x_0)$. Weiter folgt

$$\begin{aligned}
\|x_k - x_0\| &= \|x_k - x_{k-1} + x_{k-1} - \cdots - x_1 + x_1 - x_0\| \\
&\leqq \|x_k - x_{k-1}\| + \|x_{k-1} - x_{k-2}\| + \cdots + \|x_1 - x_0\| \\
&\leqq (C^{k-1} + C^{k-2} + \cdots C + 1)\|x_1 - x_0\| \\
&= \frac{1-C^k}{1-C}\|x_1 - x_0\| \leqq \frac{1-C^k}{1-C}(1-C)r \leqq (1-C^k)r < r.
\end{aligned}$$

Damit gilt $x_k \in S_r(x_0)$ für alle k. ❀

Der letzte Satz sagt aus, dass ein Iterationsverfahren konvergiert, falls die Iterationsfunktion in einer Umgebung des Startpunktes (nicht des Fixpunktes!) kontraktiv ist, und falls der erste Schritt nicht zu weit vom Startpunkt wegführt.

5.3. Konvergenzbeschleunigung

Wir haben bisher verschiedene Verfahren zur Nullstellenbestimmung und ihre Konvergenzeigenschaften kennengelernt. In diesem Abschnitt geht es um Methoden zur Transformation einer Folge $\{x_k\}_{k\in\mathbb{N}} \subset \mathbb{R}$ in eine andere Folge $\{\bar{x}_k\}_{k\in\mathbb{N}} \subset \mathbb{R}$ mit besseren Konvergenzeigenschaften. Diese Methoden lassen sich natürlich auch auf Folgen anwenden, die von Verfahren zur Nullstellenbestimmung herrühren. Betrachten wir also eine Folge $\{x_k\}_{k\in\mathbb{N}} \subset \mathbb{R}$, die von einer Iterationsfunktion Φ erzeugt wird:

$$x_{k+1} = \Phi(x_k), \quad k = 0, 1, \ldots$$

Der Grenzwert ξ der Folge ist (falls er existiert) ein Fixpunkt der Iterationsfunktion Φ. Es gilt also $\xi = \Phi(\xi)$. Damit ist ξ Nullstelle der Funktion

$$g(x) = \Phi(x) - x.$$

Es seien nun x_k und $x_{k+1} = \Phi(x_k)$ Glieder der ursprünglichen Folge. Wir verwenden x_k und x_{k+1}, um mit der Funktion g einen Schritt des Sekantenverfahrens durchzuführen. Dabei bestimmen wir die Nullstelle der Geraden durch die

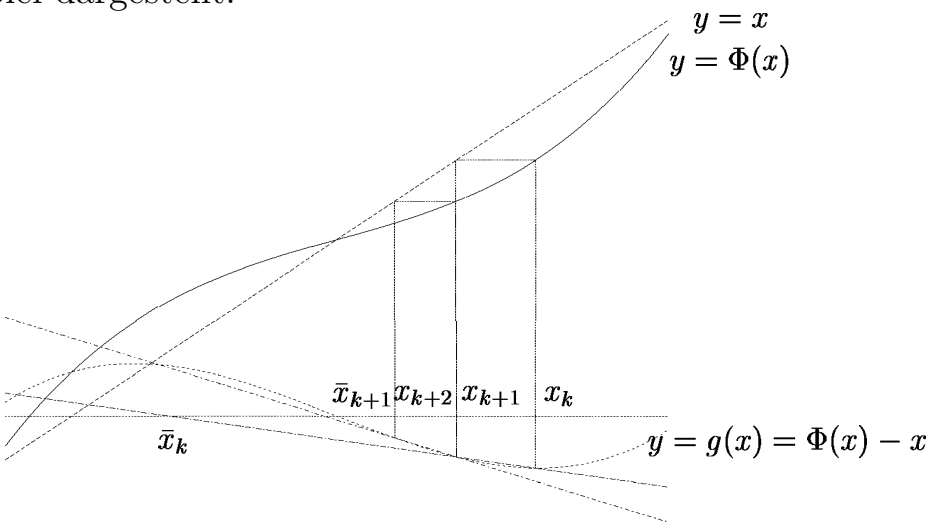
Punkte $(x_k, g(x_k))$ und $(x_{k+1}, g(x_{k+1}))$. Man erhält

$$\begin{aligned}
\bar{x}_k &= x_k - \frac{x_{k+1}-x_k}{g(x_{k+1})-g(x_k)} g(x_k) \\
&= x_k - \frac{x_{k+1}-x_k}{\Phi(x_{k+1})-x_{k+1}-(\Phi(x_k)-x_k)} (\Phi(x_k)-x_k) \\
&= x_k - \frac{x_{k+1}-x_k}{(x_{k+2}-x_{k+1})-(x_{k+1}-x_k)} (x_{k+1}-x_k) \\
&= x_k - \frac{(x_{k+1}-x_k)^2}{(x_{k+2}-x_{k+1})-(x_{k+1}-x_k)}.
\end{aligned}$$

Mit den Bezeichnungen $\Delta x_k = x_{k+1} - x_k$ und $\Delta^2 x_k = \Delta(\Delta x_k) = \Delta x_{k+1} - \Delta x_k$ lautet die obige Formel

$$\bar{x}_k = x_k - \frac{(\Delta x_k)^2}{\Delta^2 x_k}.$$

Diese Konstruktionsvorschrift der Folge $\{\bar{x}_k\}_{k\in\mathbb{N}}$ wird als **Δ^2-Methode von** AITKEN bezeichnet.
Im folgenden Bild ist die prinzipielle Vorgehensweise des Verfahrens an einem Beispiel dargestellt.



Es lässt sich zeigen, dass die Folge $\{\bar{x}_k\}_{k\in\mathbb{N}}$ schneller als $\{x_k\}_{k\in\mathbb{N}}$ konvergiert, falls sich die Folge $\{x_k\}_{k\in\mathbb{N}}$ asymptotisch wie eine geometrische Folge verhält. Trotzdem erscheint der Weg der Transformation der Folge $\{x_k\}_{k\in\mathbb{N}}$ in die Folge $\{\bar{x}_k\}_{k\in\mathbb{N}}$ nicht sinnvoll. Wenn man schon aus drei Folgegliedern x_k, x_{k+1} und x_{k+2} mittels der AITKENschen Δ^2-Methode eine neue Näherung $\bar{x}_k$ berechnet hat, so liegt es nahe, diese auch gleich wieder zu verwenden. Wir setzen also $x_{k+3} = \bar{x}_k$ und berechnen x_{k+4} aus x_{k+1}, x_{k+2} und x_{k+3}. Das führt auf die Methode von STEFFENSEN:

Es sei Φ eine Iterationsfunktion und x_k ein Iterationspunkt. Wir berechnen den nächsten Iterationspunkt gemäß

$$\begin{aligned} y_k &= \Phi(x_k), \\ z_k &= \Phi(y_k), \\ x_{k+1} &= x_k - \frac{(y_k - x_k)^2}{z_k - 2y_k + x_k}. \end{aligned}$$

Diesem Vorgehen entspricht eine neue Iterationsfunktion Ψ. Es gilt

$$x_{k+1} = \Psi(x_k)$$

mit

$$\Psi(x) = x - \frac{[\Phi(x) - x]^2}{\Phi(\Phi(x)) - 2\Phi(x) + x}.$$

Der Zusammenhang zwischen Φ und Ψ wird im folgenden Satz beschrieben.

5.12. Satz: *Es sei Φ eine Iterationsfunktion und Ψ sei nach der Methode von* Steffensen *aus Φ erzeugt, also*

$$\Psi(x) = x - \frac{[\Phi(x) - x]^2}{\Phi(\Phi(x)) - 2\Phi(x) + x}.$$

Dann gilt

1. *Ist ξ Fixpunkt von Ψ, so ist ξ auch Fixpunkt von Φ.*
2. *Ist ξ Fixpunkt von Φ und gilt in einer Umgebung von ξ*

   $$\lim_{x \to \xi} \left| \frac{\Phi(x) - x}{(x - \xi)^p} \right| = A < \infty$$

   mit einem $p \geqq 1$, so ist ξ auch Fixpunkt von Ψ.

Beweis:

1. Es sei $\Psi(\xi) = \xi$. Dann folgt

   $$\frac{[\Phi(\xi) - \xi]^2}{\Phi(\Phi(\xi)) - 2\Phi(\xi) + \xi} = 0$$

   und notwendigerweise $\Phi(\xi) = \xi$.

2. Es gilt

$$\begin{aligned}
\lim_{x\to\xi}\Psi(x) &= \lim_{x\to\xi}\left(x-\frac{[\Phi(x)-x]^2}{\Phi(\Phi(x))-2\Phi(x)+x}\right)\\
&= \xi-\lim_{x\to\xi}\frac{[\Phi(x)-x]^2}{\Phi(\Phi(x))-2\Phi(x)+x}\\
&= \xi-\lim_{x\to\xi}\frac{\left[\frac{\Phi(x)-x}{(x-\xi)^p}\right]^2(x-\xi)^p}{\frac{\Phi(\Phi(x))-\Phi(x)}{(x-\xi)^p}-\frac{\Phi(x)-x}{(x-\xi)^p}}\\
&= \xi-\lim_{x\to\xi}\frac{A^2(x-\xi)^p}{\frac{\Phi(\Phi(x))-\Phi(x)}{(\Phi(x)-\xi)^p}\left(\frac{\Phi(x)-\xi}{x-\xi}\right)^p-A}\\
&= \xi-\lim_{x\to\xi}\frac{A^2(x-\xi)^p}{A\left(1+\frac{\Phi(x)-x}{x-\xi}\right)^p-A}\\
&= \xi-\lim_{x\to\xi}\frac{A(x-\xi)^p}{\left(1+\frac{\Phi(x)-x}{x-\xi}\right)^p-1}.
\end{aligned}$$

Im Falle $p=1$ folgt sofort

$$\begin{aligned}
\lim_{x\to\xi}\Psi(x) &= \xi-\lim_{x\to\xi}\frac{A(x-\xi)}{\left(1+\frac{\Phi(x)-x}{x-\xi}\right)-1}\\
&= \xi-\lim_{x\to\xi}\frac{A(x-\xi)}{(1+A)-1}=\xi-\lim_{x\to\xi}(x-\xi)=\xi.
\end{aligned}$$

Für $p>1$ ergibt sich

$$\begin{aligned}
\lim_{x\to\xi}\Psi(x) &= \xi-\lim_{x\to\xi}\frac{A(x-\xi)^p}{\left(1+\frac{\Phi(x)-x}{x-\xi}\right)^p-1}\\
&= \xi-\lim_{x\to\xi}\frac{A(x-\xi)^p}{\left(1+\frac{\Phi(x)-x}{(x-\xi)^p}(x-\xi)^{p-1}\right)^p-1}\\
&= \xi-\lim_{x\to\xi}\frac{A(x-\xi)^p}{p\frac{\Phi(x)-x}{(x-\xi)^p}(x-\xi)^{p-1}+\binom{p}{2}\left[\frac{\Phi(x)-x}{(x-\xi)^p}\right]^2(x-\xi)^{2p-2}+\cdots}\\
&= \xi-\lim_{x\to\xi}\frac{A(x-\xi)}{pA+\binom{p}{2}A^2(x-\xi)^{p-1}+\cdots}\\
&= \xi.
\end{aligned}$$

❀

Der folgende Satz zeigt darüber hinaus einen Zusammenhang der Konvergenzordnungen von Φ und Ψ.

5.13. Satz: *Durch die Iterationsfunktion Φ sei ein Verfahren zur Bestimmung des Fixpunktes ξ gegeben. Es existiere ein $p \geqq 1$, so dass die Iterationsfunktion Φ $(p+1)$-mal stetig differenzierbar in einer Umgebung des Fixpunktes ξ ist und*

$$\Phi^{(k)}(\xi) = 0, \quad k = 1, \ldots, p-1$$

gilt. Im Falle $p = 1$ gelte zusätzlich $\Phi'(\xi) \neq 1$. Dann gilt für die Konvergenzordnung q des durch die Iterationsfunktion

$$\Psi(x) = x - \frac{(\Phi(x) - x)^2}{\Phi(\Phi(x)) - 2\Phi(x) + x}$$

erzeugten Verfahrens:

1. *$q = 2p - 1$ falls $p > 1$ bzw.*
2. *$q \geqq 2$ falls $p = 1$.*

Beweis: TAYLOR-Entwicklung von Φ in der Nähe von ξ liefert

$$\Phi(x) = \Phi(\xi) + \frac{(x-\xi)^p}{p!}\Phi^{(p)}(\xi) + \frac{(x-\xi)^{p+1}}{(p+1)!}\Phi^{(p+1)}(x + \vartheta(x-\xi)), \quad \vartheta \in (0,1)$$

und weiter

$$\Phi(x) = \xi + A(x-\xi)^p + B(x-\xi)^{p+1}$$

mit

$$A = \frac{1}{p!}\Phi^{(p)}(\xi), \quad B = B(x) = \frac{1}{(p+1)!}\Phi^{(p+1)}(x + \vartheta(x-\xi)), \quad \vartheta \in (0,1).$$

Wegen der Stetigkeit von $\Phi^{(p)}$ und $\Phi^{(p+1)}$ sind A und B beschränkt, und es existiert eine Umgebung von ξ, in der

$$\delta = \delta(x) = A(x-\xi)^p + B(x-\xi)^{p+1}$$

so klein ist, dass $\Phi(\Phi(x))$ ebenfalls in eine TAYLOR-Reihe entwickelbar ist. Es gilt

$$\Phi(\Phi(x)) = \Phi(\xi + \delta) = \xi + A\delta^p + \bar{B}\delta^{p+1}$$

mit

$$\bar{B} = \bar{B}(x) = \frac{1}{(p+1)!}\Phi^{(p+1)}(\xi + \bar{\vartheta}\delta), \quad \bar{\vartheta} \in (0,1).$$

Für $\Psi(x)$ gilt dann in der Nähe des Fixpunktes ξ

$$\begin{aligned}
\Psi(x) &= \frac{x\Phi(\Phi(x)) - (\varphi(x))^2}{\Phi(\Phi(x)) - 2\Phi(x) + x} \\
&= \frac{(\xi + (x-\xi))(\xi + A\delta^p + \bar{B}\delta^{p+1}) - (\xi+\delta)^2}{\xi + A\delta^p + \bar{B}\delta^{p+1} - 2(\xi+\delta) + \xi + (x-\xi)} \\
&= \frac{\xi^2 + \xi(A\delta^p + \bar{B}\delta^{p+1}) + (x-\xi)\xi + (x-\xi)(A\delta^p + \bar{B}\delta^{p+1}) - \xi^2 - 2\xi\delta - \delta^2}{(x-\xi) - 2\delta + A\delta^p + \bar{B}\delta^{p+1}} \\
&= \frac{\xi\left[(x-\xi) - 2\delta + A\delta^p + \bar{B}\delta^{p+1}\right] + (x-\xi)(A\delta^p + \bar{B}\delta^{p+1}) - \delta^2}{(x-\xi) - 2\delta + A\delta^p + \bar{B}\delta^{p+1}} \\
&= \xi - \frac{\delta^2 - A(x-\xi)\delta^p - \bar{B}(x-\xi)\delta^{p+1}}{(x-\xi) - 2\delta + A\delta^p + \bar{B}\delta^{p+1}}.
\end{aligned}$$

Setzen wir

$$\delta = A(x-\xi)^p - B(x-\xi)^{p+1} = C(x-\xi)^p, \quad C = A + B(x-\xi),$$

so erhalten wir

$$\Psi(x) = \xi - \frac{C^2(x-\xi)^{2p} - AC^p(x-\xi)^{p^2+1} - \bar{B}C^{p+1}(x-\xi)^{p^2+p+1}}{(x-\xi) - 2C(x-\xi)^p + AC^p(x-\xi)^{p^2} + \bar{B}C^{p+1}(x-\xi)^{p^2+p}}.$$

Für $p > 1$ folgt mit $h = x - \xi$

$$\Psi(x) = \xi - h^{2p-1}\frac{C^2 - AC^p h^{(p-1)^2} - \bar{B}C^{p+1}h^{p^2-p+1}}{1 - 2Ch^{p-1} + AC^p h^{p^2-1} + \bar{B}C^{p+1}h^{p^2+p-1}},$$

und es gilt

$$\lim_{x\to\xi} \frac{C^2 - AC^p h^{(p-1)^2} - \bar{B}C^{p+1}h^{p^2-p+1}}{1 - 2Ch^{p-1} + AC^p h^{p^2-1} + \bar{B}C^{p+1}h^{p^2+p-1}} = C^2 = A^2 \neq 0.$$

Damit besitzt Ψ die folgende Darstellung:

$$\Psi(x) = \xi - A^2(x-\xi)^{2p-1} + O(|x-\xi|^{2p}).$$

Das durch die Funktion Ψ erzeugte Iterationsverfahren hat also die Konvergenzordnung $q = 2p-1$.
Im Falle $p = 1$ erhält man

$$\begin{aligned}
\Psi(x) &= \xi - \frac{C^2(x-\xi)^2 - AC(x-\xi)^2 - \bar{B}C^2(x-\xi)^3}{(x-\xi) - 2C(x-\xi) + AC(x-\xi) + \bar{B}C^2(x-\xi)^2} \\
&= \xi - C(x-\xi)^2 \frac{C - A - \bar{B}C(x-\xi)}{(x-\xi)\,(1-2C+AC+\bar{B}C^2(x-\xi))} \\
&= \xi - C(x-\xi)^2 \frac{B(x-\xi) - \bar{B}C(x-\xi)}{(x-\xi)\,(1-2C+AC+\bar{B}C^2(x-\xi))} \\
&= \xi - C(x-\xi)^2 \frac{B - \bar{B}C}{1-2C+AC+\bar{B}C^2(x-\xi)}.
\end{aligned}$$

Hier gilt wegen

$$\lim_{x\to\xi} B = \lim_{x\to\xi} \bar{B} = \frac{1}{2}\Phi''(\xi), \quad \lim_{x\to\xi} C = A$$

$$\lim_{x\to\xi} \frac{B - \bar{B}C}{1-2C+AC+\bar{B}C^2(x-\xi)} = \frac{\frac{1}{2}\Phi''(\xi) - \frac{1}{2}\Phi''(\xi)A}{(1-A)^2} = \frac{\Phi''(\xi)}{2(1-A)}.$$

Für $A \neq 1$ erhalten wir dann

$$\Psi(x) = \xi - \frac{\Phi''(\xi)}{2(1-A)}(x-\xi)^2 + O(|x-\xi|^3).$$

Das durch die Funktion Ψ erzeugte Iterationsverfahren hat somit eine Konvergenzordnung $q \geqq 2$. ❀

Bemerkung: Im Falle $p = 1$ ist nur die Forderung $A = \Phi'(\xi) \neq 1$ für die Konvergenz des durch Ψ erzeugten Verfahrens notwendig. Man erhält auch im Falle $|\Phi'(\xi)| > 1$, also in einem Fall, bei dem das durch Φ erzeugte Verfahren divergiert, ein quadratisch konvergentes Iterationsverfahren.

5.14. Beispiel: Es sei $\Phi(x) = x^2$. Fixpunkte sind $\xi_1 = 0$ und $\xi_2 = 1$. Es gilt

$$\Phi'(\xi_1) = 0, \quad \Phi''(\xi_1) = 2 \Longrightarrow \text{quadratische Konvergenz}$$

und

$$\Phi'(\xi_2) = 2 \Longrightarrow \text{Divergenz}.$$

Wählt man einen Startpunkt $|x_0| < 1$, so ergeben sich konvergente Folgen mit dem Grenzwert $\xi_1 = 0$. Für einen Startpunkt $|x_0| > 1$ divergiert das durch Φ erzeugte Verfahren.
Mit der STEFFENSEN-Methode erhält man die neue Iterationsfunktion

$$\begin{aligned}\Psi &= x - \frac{(x^2-x)^2}{x^4-2x^2+x} = \frac{x^5-2x^3+x^2-x^4+2x^3-x^2}{x^4-2x^2+x} \\ &= \frac{x^4(x-1)}{x(x^3-2x+1)} = \frac{x^3}{x^2+x-1}.\end{aligned}$$

Für $|x_0| \leqq 1/2$ ist Ψ kontrahierend und die erzeugte Folge konvergiert gegen $\xi_1 = 0$. Die Konvergenz ist kubisch. Es gilt

$$x_{i+1} = \Psi(x_i) \approx -x_i^3.$$

Wendet man aber die Iterationsvorschrift

$$x_{i+1} = \Phi(\Phi(x_i)) = x_i^4$$

an, die einen geringeren Aufwand als das durch Ψ erzeugte Verfahren hat, so erhält man sogar ein Verfahren der Konvergenzordnung $q = 4$ zur Bestimmung des Fixpunktes ξ_1. In diesem Falle bringt die STEFFENSEN-Methode also eine Konvergenzverschlechterung.
Für $|x_0| \geqq (\sqrt{5}-1)/2$ ist Ψ ebenfalls kontrahierend und die erzeugte Folge konvergiert gegen $\xi_2 = 1$. Nach Satz 5.13 konvergiert die durch Ψ erzeugte Folge wegen $\Phi(\xi_2) = 2 \neq 1$ mindestens quadratisch. Dies erkennt man auch aus

$$\Psi'(\xi_2) = 0, \quad \Psi''(\xi_2) = 4 \neq 0.$$

In diesem Falle bringt die STEFFENSEN-Methode eine wirkliche Konvergenzverbesserung. ♡

Die Situation aus diesem Beispiel ist typisch. Bei Verfahren mit einer Konvergenzordnung $p > 1$ bringt das STEFFENSEN-Verfahren eine Konvergenzverschlechterung. Vergleicht man nämlich einen Schritt des durch Ψ erzeugten Verfahrens mit einem etwa gleich aufwendigen Doppelschritt des durch Φ erzeugten Verfahrens, so gilt

$$x_{i+1} = \Psi(x_i) \Longrightarrow \text{Konvergenzordnung } q = 2p-1,$$

$$x_{i+1} = \Phi(\Phi(x_i)) \Longrightarrow \text{Konvergenzordnung } q = p^2 > 2p-1.$$

Für $p > 1$ sollte man also besser das ursprüngliche Verfahren nutzen. Eine wirkliche Konvergenzverbesserung erreicht man nur für linear konvergente oder divergente Verfahren.
Wir betrachten das einfache Iterationsverfahren

$$\Phi(x) = x + f(x).$$

Jede Nullstelle von f ist Fixpunkt von Φ. Mit der Methode von STEFFENSEN erhalten wir die neue Iterationsfunktion

$$\begin{aligned}\Psi(x) &= x - \frac{(f(x))^2}{x + f(x) + f(x + f(x)) - 2(x + f(x)) + x} \\ &= x - \frac{(f(x))^2}{f(x + f(x)) - f(x)} = x - \frac{f(x)}{\frac{f(x+f(x))-f(x)}{f(x)}}.\end{aligned}$$

Ist $|f(x)| \ll 1$ (Das ist der Fall, falls der Punkt x nahe genug bei einer Nullstelle von f liegt.), so stellt der Nenner eine Näherung für $f'(x)$ dar. Man erkennt die Verwandtschaft zum NEWTON-Verfahren. Aus diesem Grund wird dieses Verfahren auch als **Quasi-NEWTON-Verfahren** bezeichnet. Ein anderes Quasi-NEWTON-Verfahren erhält man, falls man von der Iterationsfunktion

$$\Phi(x) = x - f(x)$$

ausgeht. Es ergibt sich

$$\Psi(x) = x - \frac{(f(x))^2}{f(x) - f(x - f(x))}.$$

Ist ξ eine einfache Nullstelle von f, so gilt $\Phi'(\xi) \neq 1$. Mit Satz 5.13 folgt dann die quadratische Konvergenz der Quasi-NEWTON-Verfahren. Man beachte aber, dass in der Nähe der Nullstelle beim Berechnen der Nenner Auslöschung auftritt.

5.4. Hybridverfahren

Es erscheint sinnvoll, global konvergente Verfahren mit lokal schneller konvergenten Verfahren zu koppeln. Das führt auf sogenannte **Hybridverfahren**. Wir geben ein Hybridverfahren an, dass das Bisektionsverfahren mit einem lokal schneller konvergentem Verfahren kombiniert.

5.15. Hybridverfahren:

Für eine Funktion f sei durch Φ ein Iterationsverfahren zur Nullstellenbestimmung mit einer Konvergenzordnung $p > 1$ gegeben.
Weiterhin sei $[a,b]$ ein Intervall mit $f(a)f(b) < 0$.
Die Funktion f erfülle alle Voraussetzungen des durch Φ gegebenen Iterationsverfahrens.
O.B.d.A. sei $f(a) < 0$ und $f(b) > 0$.

S0: *Setze*

$$a_0 = a, \quad b_0 = b, \quad x_0 = a, \quad k = 0.$$

S1: *Berechne $\mu = \Phi(x_k)$.*

S2: *Wenn $\mu \notin [a_k, b_k]$, so gehe zu* **S5**.

S3: *Berechne $\eta = f(\mu)$. Wenn*

$$\eta \begin{cases} > & 0, \ d = \mu - a_k, \\ = & 0, \ x^* = \mu, \ STOPP, \\ < & 0, \ d = b_k - \mu. \end{cases}$$

S4: *Wenn $d \leqq (b_k - a_k)/2$, so gehe zu* **S6**.

S5: *Setze $\mu = (a_k + b_k)/2$ und berechne $\eta = f(\mu)$.*

S6: *Wenn*

$$\eta \begin{cases} > & 0, \ a_{k+1} = a_k, & b_{k+1} = x_{k+1} = \mu, \\ = & 0, \ x^* = \mu, & STOPP, \\ < & 0, \ a_{k+1} = x_{k+1} = \mu, & b_{k+1} = b_k. \end{cases}$$

S7: *Setze $k = k+1$ und gehe zu* **S1**.

Falls durch Φ in einem Schritt ein Punkt μ außerhalb des aktuellen Iterationsintervalls erzeugt wird, wird ein zusätzlicher Bisektionsschritt ausgeführt. Das geschieht ebenfalls, falls das neue Iterationsintervall größer als die Hälfte des alten Iterationsintervalls werden würde. Dadurch wird globale lineare Konvergenz mit dem Faktor $1/2$ gesichert. Lokal sollten nur noch Schritte mit dem schneller konvergenten Verfahren ausgeführt werden.

5.5. Aufgaben

1. Man zeige, dass für $f \in C^2[a,b]$ gilt:
   Für beliebige $x,y,z \in [a,b]$ existiert ein $\xi \in I[x,y,z]$ mit

$$\frac{1}{2} f''(\xi) = f[x,y,z] \, .$$

$I[x,y,z]$ bezeichne das kleinste Intervall, das x,y und z enthält, $f[x,y,z]$ bezeichne die zweite dividierte Differenz.
Hinweis: O.B.d.A. sei $x \leqq y \leqq z$. Man entwickle f bis zum quadratischen Glied und wende den Zwischenwertsatz für stetige Funktionen an.

2. Man zeige:

$$\lim_{i \longrightarrow \infty} x_i = 2$$

für $x_0 = 0$ und $x_{i+1} = \sqrt{2+x_i}$, $i = 0,1,\ldots$

3. Man zeige, dass die Iteration

$$x_{k+1} = \cos x_k$$

für alle $x_0 \in R$ gegen den einzigen Fixpunkt ξ ($\xi = \cos\xi$) konvergiert.

4. $f : R \longrightarrow R$ habe die einzige Nullstelle ξ. Man zeige:
Wendet man auf die Iterationsfunktion

$$\Phi(x) = x - f(x)$$

die Methode von STEFFENSEN an, so erhält man die Iteration

$$x_{n+1} = x_n - \frac{f(x_n)^2}{f(x_n) - f(x_n - f(x_n))} \quad n = 0,1,2,\ldots$$

Man zeige weiterhin, dass dieses Verfahren für eine einfache Nullstelle ξ lokal mindestens quadratisch und für eine mehrfache Nullstelle ξ linear konvergiert.

5. Die Funktion $f : R \longrightarrow R$ sei für alle $x \in U(\xi) = \left\{ x \mid |x-\xi| \leqq r \right\}$ zweimal stetig differenzierbar. Dabei sei ξ eine einfache Nullstelle von f ($f(\xi) = 0$, $f'(\xi) \neq 0$). Man zeige:
Das Iterationsverfahren

$$\begin{aligned} y &= x_n - f'(x_n)^{-1} f(x_n) \\ x_{n+1} &= y - f'(x_n)^{-1} f(y) \end{aligned}$$

konvergiert von mindestens 3.Ordnung gegen ξ.

6. Man berechne iterativ $x = \frac{1}{a}$ für ein gegebenes $a > 0$ ohne Verwendung der Division. Für welche Startwerte x_0 konvergiert das Verfahren?

Index