

**EINFÜHRUNG IN DIE
NUMERISCHE MATHEMATIK I ¹**

Numerische Analysis

Prof. Dr. Hans Babovsky

Institut für Mathematik

Technische Universität Ilmenau

¹Version vom Herbst 2009

Inhaltsverzeichnis

1	Nichtlineare Gleichungen	2
1.1	Die Ordnung konvergenter Folgen	2
1.2	Nullstellen skalarer Funktionen: Intervallschachtelungen	3
1.2.1	Die Intervallhalbierungsmethode	4
1.2.2	Die Methode der Regula Falsi	6
1.2.3	Die Sekantenmethode	7
1.3	Fixpunktgleichungen	9
1.4	Das Newton-Verfahren	11
1.4.1	Newton-Iteration, Konvergenz	11
2	Interpolation und Approximation	16
2.1	Polynominterpolation	16
2.1.1	Das einfache Interpolationsproblem	16
2.2	Lagrange-Interpolation	17
2.3	Das erweiterte Interpolationsproblem	23
2.4	Approximationsfehler	28
2.5	Tschebyscheff-Interpolation	30
3	Spline-Interpolation	34
3.1	Polynom-Splines	34
3.2	Kubische Splines	35
3.3	B-Splines	40
4	Approximation durch Orthogonalsysteme	46
5	Numerische Integration	51
5.1	Quadraturformeln	51
5.2	Newton-Cotes-Formeln	54
5.3	Gauß-Christoffel-Quadratur	60
5.4	Extrapolationsverfahren	64
5.4.1	Die Eulersche Summenformel	64
5.4.2	Trapezregel: Asymptotische Fehlerentwicklung	69
5.4.3	Das Romberg-Verfahren	71
5.4.4	Adaptive Verfahren	76

5.5	Die Idee; Fehlerschätzer	76
5.6	Die Idee der Schrittweitensteuerung	77
5.7	Die Idee der Mehrgitterverfahren	79

1 Nichtlineare Gleichungen

In diesem Abschnitt sei eine Funktion

$$f : \mathbb{R}^n \longrightarrow \mathbb{R}^n$$

gegeben. Ziel dieses Kapitels ist es, Nullstellen bzw. Fixpunkte von f numerisch zu bestimmen. Hierzu werden wir Iterationsfolgen $\mathbf{x}^{(k)}$ entwickeln, welche gegen die gesuchte Nullstelle bzw. den Fixpunkt konvergieren. Zur Beurteilung der Qualität eines Iterationsverfahrens benötigen wir den Begriff der Konvergenzordnung.

1.1 Die Ordnung konvergenter Folgen

Sei (X, d) ein metrischer Raum.

(1.1.1) Definition: (a) Es sei $\mathbf{x}^{(n)}$ eine konvergente Folge in X mit $\lim_{k \rightarrow \infty} \mathbf{x}^{(k)} = \mathbf{x}^*$. Die Folge hat die *Konvergenzordnung*

$p = 1$, falls es ein $C \in (0, 1)$ und ein $k_0 \in \mathbb{N}$ gibt mit $d(\mathbf{x}^{(k+1)}, \mathbf{x}^*) \leq C \cdot d(\mathbf{x}^{(k)}, \mathbf{x}^*)$ für alle $k \geq k_0$.

$p > 1$, wenn es eine Konstante $C > 0$ und ein $k_0 \in \mathbb{N}$ gibt mit $d(\mathbf{x}^{(k+1)}, \mathbf{x}^*) \leq C \cdot d(\mathbf{x}^{(k)}, \mathbf{x}^*)^p$ für $\ell = 1, 2, \dots$ für alle $k \geq k_0$.

Für $p = 1$ heißt die Folge *linear konvergent*, für $p > 1$ *superlinear konvergent* und für $p = 2$ *quadratisch konvergent*.

(b) Gegeben sei ein Iterationsverfahren zur Berechnung eines Vektors \mathbf{x}^* , welches zu einem Startvektor $\mathbf{x}^{(0)}$ eine Folge $\mathbf{x}_{n \in \mathbb{N}}^{(n)}$ erzeugt. Das Verfahren hat die *Ordnung* p , falls $\lim_{k \rightarrow \infty} \mathbf{x}^{(k)} = \mathbf{x}^*$, und falls die Folge $\mathbf{x}^{(k)}$ die Konvergenzordnung p hat.

(1.1.2) Bemerkungen: (a) Ist $(\mathbf{x}^{(k)})$ linear konvergent, so folgt die Existenz eines $\lambda > 0$ derart, dass

$$d(\mathbf{x}^{(k)}, \mathbf{x}^*) \leq \lambda \cdot C^k \rightarrow 0 \quad \text{für } k \rightarrow \infty \quad (1.1)$$

Wir erweitern die Definition (1.1.1)(a) und bezeichnen eine Folge auch dann als linear konvergent, wenn die (schwächere) Bedingung (1.1) für alle k gilt.

(b) Bei Konvergenzordnung $p > 1$ gilt für $k \geq k_0$

$$d(\mathbf{x}^{(k+1)}, \mathbf{x}^*) / d(\mathbf{x}^{(k)}, \mathbf{x}^*) \leq C \cdot (d(\mathbf{x}^{(k)}, \mathbf{x}^*))^{p-1}$$

Ist

$$\mu := C \cdot d(\mathbf{x}^{(k_0)}, \mathbf{x}^*)^{p-1} < 1$$

so folgt die Monotonie der Folge $d(\mathbf{x}^{(k)}, \mathbf{x}^*)$ sowie mit geeignetem $\lambda > 0$ die Abschätzung

$$d(\mathbf{x}^{(k)}, \mathbf{x}^*) \leq \lambda \mu^k \rightarrow 0$$

Konvergenz liegt also dann vor, wenn eines der Folgenglieder hinreichend nahe bei \mathbf{x}^* liegt.

(1.1.3) Beispiel: (a) Gegeben sei die Zahl π in Dezimaldarstellung,

$$\pi = 3.14159\dots = \sum_{i=0}^{\infty} a_i \cdot 10^{-i} \quad , \quad (1.2)$$

mit $a_i \in \{0, \dots, 9\}$. Welche Konvergenzordnung hat die Folge $s_k = \sum_{i=0}^k a_i \cdot 10^{-i}$? Offenbar ist

$$|s_k - \pi| \leq \sum_{i=k+1}^{\infty} 9 \cdot 10^{-i} = 10^{-k}.$$

Nach Bemerkung (1.1.2)(a) die Folge linear konvergent. Das stärkere Kriterium aus Definition (1.1.1) ist dagegen nicht erfüllt, da im Fall $a_k = 0$ gilt $|s_{k-1} - \pi| = |s_k - \pi|$.

(b) Die Folge $\mathbf{x}^{(k)}$ in \mathbb{R} sei quadratisch konvergent mit $C = 1$. Approximiert das Folgenglied $\mathbf{x}^{(k)}$ den Grenzwert \mathbf{x}^* auf n Nachkommastellen genau, so ist

$$|\mathbf{x}^{(k)} - \mathbf{x}^*| < 0.5 \cdot 10^{-n}$$

Nach der Definition der quadratischen Konvergenz folgt

$$|\mathbf{x}^{(k)} - \mathbf{x}^*| < 0.25 \cdot 10^{-2n}$$

d.h. die Anzahl der exakten Nachkommastellen von $\mathbf{x}^{(k+1)}$ ist mindestens $2n$ und hat sich verdoppelt.

1.2 Nullstellen skalarer Funktionen: Intervallschachtelungen

Gegeben sei eine stetige Funktion $f : [a, b] \rightarrow \mathbb{R}$. Gesucht ist eine Nullstelle. Die folgenden Verfahren beziehen sich auf Funktionen $f : \mathbb{R} \rightarrow \mathbb{R}$, welche stetig sind. Zusätzlich

setzen wir zunächst voraus, dass zwei Punkte a und b bekannt sind, an denen die Funktionswerte unterschiedliche Vorzeichen haben (solche Punkte kann man häufig durch eine grobe Kurvendiskussion bestimmen), für die also gilt

$$f(a) \cdot f(b) < 0. \quad (1.3)$$

Da f stetig ist, muß zwischen a und b (mindestens) eine Nullstelle liegen². Im folgenden soll versucht werden, durch Einschachtelungsverfahren den Ort einer Nullstelle immer weiter einzugrenzen und so ihre Lage mit beliebiger Genauigkeit zu bestimmen.

1.2.1 Die Intervallhalbierungsmethode

Bei dieser Methode wird in jedem Iterationsschritt das untersuchte Intervall in zwei gleich große Teilintervalle eingeteilt; in einem dieser beiden Teilintervalle muss sich eine Nullstelle befinden. Im nächsten Schritt wird die Suche auf dieses Teilintervall eingeschränkt.

Gegeben sei das Intervall $[a_0, b_0] \subseteq [a, b]$, $a_0 < b_0$, und es gelte

$$f(a_0)f(b_0) < 0$$

Wegen der Stetigkeit von f muss sich im Intervall $[a_0, b_0]$ eine Nullstelle von f befinden. Wir definieren eine Folge von Intervallen $[a_k, b_k]$ mit der Inklusionsbedingung

$$[a_{k+1}, b_{k+1}] \subseteq [a_k, b_k]$$

und eine Folge x_k wie folgt.

(1.2.4) Rekursionsschritt (Intervallhalbierung): Gegeben sei $[a_k, b_k]$.

²Dies folgt aus dem Zwischenwertsatz; vgl. Analysis-Vorlesung

(S1) Definiere x_{k+1} als den Intervallmittelpunkt:

$$x_{k+1} := 0.5 \cdot (a_k + b_k)$$

(S2) Definiere $[a_{k+1}, b_{k+1}]$ durch

$$[a_{k+1}, b_{k+1}] := \begin{cases} [a_k, x_{k+1}] & \text{falls } f(a_k)f(x_{k+1}) < 0 \\ [x_{k+1}, b_k] & \text{falls } f(x_{k+1})f(b_k) < 0 \\ [x_{k+1}, x_{k+1}] & \text{falls } f(x_{k+1}) = 0 \end{cases}$$

Die Iterationsvorschrift für $[a_{k+1}, b_{k+1}]$ stellt sicher, dass sich im neuen Intervall wieder eine Nullstelle von f befindet. Ist $f(x_{k+1}) = 0$ für ein k , so ist nach endlich vielen Iterationsschritten eine Nullstelle gefunden. (Dieser Fall wird in der Regel nicht eintreten.) Im andern Fall wird eine Nullstelle auf ein immer kleineres Intervall eingegrenzt. Die Folgen a_k und b_k sind beschränkte monotone Folgen mit Grenzwerten a_∞ und b_∞ . Wegen

$$b_{k+1} - a_{k+1} = 0.5(b_k - a_k)$$

und wegen $a_k \leq x_{k+1} \leq b_k$ folgt

$$a_\infty = \lim_{k \rightarrow \infty} x_k = b_\infty$$

Damit konvergiert die Folge x_k gegen eine Nullstelle von f .

Wir bestimmen die Konvergenzordnung für den typischen Fall, dass eine Nullstelle nicht nach endlich vielen Schritten gefunden wird.

Ist $H = b - a$ die Intervalllänge des Eingabeintervalls, und tritt das Ereignis $f(x^{(k)}) = 0$ nicht auf, so ist nach k Schritten die Intervalllänge gleich $b - a = H/2^k$; außerdem ist der Approximationsfehler kleiner als die halbe Intervalllänge, also

$$|x^{(k)} - x^*| < H/2^{k+1}.$$

(1.2.5) Bemerkung: Das Intervallhalbierungsverfahren ist ein **linear konvergentes**

Verfahren.

Für $\epsilon > 0$ bricht das Verfahren nach endlich vielen Schritten mit einem Näherungswert ab, welcher von x^* um höchstens ϵ abweicht. Da sich die Länge des untersuchten Teilintervalls mit jedem Schritt halbiert, muss der Schritt *S2* k -mal durchlaufen werden mit

$$k \approx \ln(H/\epsilon)/\ln(2) - 1.$$

1.2.2 Die Methode der Regula Falsi

Im Unterschied zur Intervallhalbierung wird bei der Methode der Regula falsi die Iterierte x^{k+1} nicht als Mittelpunkt des aktuellen Intervalls definiert, sondern als Nullstelle der *Sekante* bezüglich der Knoten a und b , also der Geraden durch die Punkte $(a, f(a))$ und $(b, f(b))$. Diese Gerade ist gegeben durch

$$s(x) = f(a) + (x - a) \cdot \frac{f(b) - f(a)}{b - a}$$

und hat die Nullstelle

$$x_0 = a - f(a) \cdot \frac{b - a}{f(b) - f(a)} = \frac{a \cdot f(b) - b \cdot f(a)}{f(b) - f(a)}. \quad (1.4)$$

Zur Implementierung dieses Verfahrens sind im Algorithmus zur Intervallhalbierung lediglich die Anweisungen $x^{(\cdot)} = (a + b)/2$ zu ersetzen durch

$$x^{(\cdot)} = \frac{a \cdot f(b) - b \cdot f(a)}{f(b) - f(a)}. \quad (1.5)$$

(1.2.6) Rekursionsschritt (*Regula falsi*): Ersetze in (1.2.4) den Schritt (S1) durch

$$x^{(\cdot)} = \frac{a \cdot f(b) - b \cdot f(a)}{f(b) - f(a)}. \quad (1.6)$$

Wir setzen voraus, dass f zweimal stetig differenzierbar ist. Ist x^* die gesuchte Nullstelle, und ist $f'(x^*) \neq 0$, sowie $f''(x^*) \neq 0$, so haben die erste und zweite Ableitung von f in einer Umgebung von x^* ein festes Vorzeichen. Wir untersuchen das Verhalten der

Iterierten für den Spezialfall einer konvexen, streng monoton wachsenden Funktion; es sei also

$$f'(x) > 0 \text{ und } f''(x) > 0 \quad \text{für } x \in [a, b].$$

(Die anderen Fälle können entsprechend untersucht werden.) Man überzeugt sich leicht, dass in diesem Fall der rechte Rand b unverändert bleibt, während sich $x^{(k)}$ und damit der linke Rand a von links an die Nullstelle x^* annähern. Damit ist $b - a > b - x^*$, und das Intervall $[a, b]$ schrumpft nicht zusammen auf $\{x^*\}$. Aus

$$x^{(k+1)} = \frac{x^{(k)} \cdot f(b) - b \cdot f(x^{(k)})}{f(b) - f(x^{(k)})}$$

folgt

$$x^* - x^{(k+1)} = \frac{(x^* - x^{(k)}) \cdot f(b) + (b - x^{(k)}) \cdot f(x^{(k)})}{f(b) - f(x^{(k)})}.$$

Nach der Taylorformel ist

$$f(x^{(k)}) = f(x^*) - (x^* - x^{(k)}) \cdot f'(\zeta) = -(x^* - x^{(k)}) \cdot f'(\zeta)$$

mit einer Zwischenstelle $\zeta \in (x^{(k)}, x^*)$. Hieraus folgt

$$|x^* - x^{(k+1)}| = |x^* - x^{(k)}| \cdot \frac{|f(b) - (b - x^{(k)}) \cdot f'(\zeta)|}{|f(b) - f(x^{(k)})|} \leq C \cdot |x^* - x^{(k)}|$$

mit einer geeigneten Konstante C . Es handelt sich also auch hier um ein **linear konvergentes Verfahren**.

1.2.3 Die Sekantenmethode

Bei der Sekantenmethode wird – ausgehend von zwei Startwerten $x^{(0)}$ und $x^{(1)}$ – die Iterierte $x^{(k+1)}$ definiert als Nullstelle der Sekante bezüglich der Knoten $x^{(k-1)}$ und $x^{(k)}$. Diese Nullstelle wird analog zur Formel (5.2) berechnet, und es folgt

$$x^{(k+1)} = \frac{x^{(k-1)} \cdot f(x^{(k)}) - x^{(k)} \cdot f(x^{(k-1)})}{f(x^{(k)}) - f(x^{(k-1)})}. \quad (1.7)$$

Dieses Verfahren ist eng verwandt mit den in A und B beschriebenen, es handelt sich hier aber nicht um eine Intervallschachtelung; so wird beispielsweise für die Startwerte nicht gefordert, dass $f(x^{(0)}) \cdot f(x^{(1)}) < 0$. Ist dies nicht erfüllt, so liegt $x^{(2)}$ nicht innerhalb des durch $x^{(0)}$ und $x^{(1)}$ beschriebenen Intervalls. Entsprechend ist die Konvergenz des

Verfahrens auch nicht gesichert.

Für die folgende Fehleranalyse gehen wir davon aus, dass $x^* := \lim_{k \rightarrow \infty} x^{(k)}$ existiert. In diesem Fall ist x^* Nullstelle von f . (Begründung?) Wir definieren

$$\underline{x} := \inf_{k \in \mathbb{N}} x^{(k)} \quad \text{und} \quad \bar{x} := \sup_{k \in \mathbb{N}} x^{(k)}$$

und setzen wie in B eine in $[\underline{x}, \bar{x}]$ zweimal stetig differenzierbare Funktion f voraus, für welche

$$f'(x) \neq 0 \quad \text{und} \quad f''(x) \neq 0 \quad \text{für} \quad x \in [\underline{x}, \bar{x}].$$

Bezeichnet $\epsilon_k := x^{(k)} - x^*$ den Fehler der k -ten Iterierten, so folgt aus (5.4)

$$\begin{aligned} \epsilon_{k+1} &= \frac{(x^{(k-1)} - x^*) \cdot f(x^{(k)}) - (x^{(k)} - x^*) \cdot f(x^{(k-1)})}{f(x^{(k)}) - f(x^{(k-1)})} \\ &= \frac{\epsilon_{k-1} \cdot f(x^{(k)}) - \epsilon_k \cdot f(x^{(k-1)})}{f(x^{(k)}) - f(x^{(k-1)})} =: \frac{z_{k+1}}{n_{k+1}}. \end{aligned} \quad (1.8)$$

Aus der Taylorformel folgt mit geeigneten Zwischenwerten $\zeta^{(k-1)}$ und $\zeta^{(k)}$

$$\begin{aligned} f(x^{(k-1)}) &= \epsilon_{k-1} \cdot f'(x^*) + \frac{\epsilon_{k-1}^2}{2} f''(\zeta^{(k-1)}), \\ f(x^{(k)}) &= \epsilon_k \cdot f'(x^*) + \frac{\epsilon_k^2}{2} f''(\zeta^{(k)}). \end{aligned}$$

Aus der Konvergenz $x^{(k)} \rightarrow x^*$ folgt $\zeta^{(k)} \rightarrow x^*$. Daher gilt für Zähler und Nenner von (5.5)

$$\begin{aligned} z_{k+1} &= \frac{1}{2} \epsilon_{k-1} \epsilon_k \cdot (\epsilon_k f''(\zeta_k) - \epsilon_{k-1} f''(\zeta_{k-1})) \rightarrow \frac{1}{2} \epsilon_{k-1} \epsilon_k \cdot (\epsilon_k - \epsilon_{k-1}) \cdot f''(x^*), \\ n_{k+1} &\rightarrow (\epsilon_k - \epsilon_{k-1}) \cdot \left(f'(x^*) + \frac{\epsilon_{k-1} + \epsilon_k}{2} \cdot f''(x^*) \right). \end{aligned}$$

Hieraus folgt mit geeigneten Konstanten $\underline{C}, \bar{C} > 0$

$$\underline{C} \cdot |\epsilon_{k-1}| \cdot |\epsilon_k| \leq |\epsilon_{k+1}| \leq \bar{C} \cdot |\epsilon_{k-1}| \cdot |\epsilon_k|.$$

Um hieraus auf das Konvergenzverhalten der Sekantenmethode zu schließen, untersuchen wir das Vergleichsproblem

$$a_{k+1} = C \cdot a_{k-1} \cdot a_k. \quad (1.9)$$

mit dem Ansatz $a_k = \alpha \cdot a_{k-1}^p$. Einsetzen ergibt

$$a_{k+1} = \alpha^{p+1} a_{k-1}^{p^2} = C \alpha a_{k-1}^{p+1},$$

also $C = \alpha^p$, und p Nullstelle von $p^2 - p - 1$ und damit $p = \frac{1}{2}(1 \pm \sqrt{5})$. a_k ist Nullfolge höchstens dann (falls $\alpha \neq 0$), wenn $p \geq 1^3$, also für $p = \frac{1}{2}(1 + \sqrt{5}) = 1.618$. Hieraus kann man schließen, dass im Falle $|\epsilon_1| \leq \alpha \cdot |\epsilon_0|^p$ das Sekantenverfahren **superlinear konvergent** ist mit $p = 1.618$.

1.3 Fixpunktgleichungen

In diesem Abschnitt bezeichne $D \subseteq \mathbb{R}^n$ eine abgeschlossene Menge und

$$F : D \longrightarrow D$$

eine stetige Funktion. Ein Punkt $x^* \in D$ heißt **Fixpunkt von F**, falls gilt

$$x^* = F(x^*).$$

Wir untersuchen die Möglichkeit, einen Fixpunkt x^* von F durch eine Folge $x^{(k)}$ zu approximieren, welche für $k \rightarrow \infty$ gegen den gesuchten Fixpunkt konvergiert. Ein häufig benutztes Verfahren ist die Fixpunktiteration, welche wie folgt beschrieben ist.

(1.3.1) Algorithmus (Fixpunktiteration):

- S1 Bestimme einen geeigneten Startwert $x^{(0)}$ (z.B. einen Schätzwert von x^*).
- S2 Löse das folgende Iterationsverfahren für $k = 0, 1, 2, \dots$:
Ist $x^{(k)}$ gegeben, so bestimme $x^{(k+1)}$ durch die Gleichung

$$x^{(k+1)} := F(x^{(k)}).$$

Aus der Stetigkeit von F folgt unmittelbar:

(1.3.2) Lemma: Konvergiert das Iterationsverfahren für $k \rightarrow \infty$ gegen einen Vektor \tilde{x} , so ist \tilde{x} ein Fixpunkt von F .

³Beweisen Sie dies, indem Sie überprüfen, unter welchen Voraussetzungen gilt $a_{k+1} \leq a_k$.

Ein hinreichendes Kriterium, unter welchem F einen (eindeutig bestimmten) Fixpunkt hat, liefert der folgende Satz.

(1.3.3) Banachscher Fixpunktsatz: Die Abbildung F sei **Lipschitz-stetig**, d.h. es gebe eine Konstante $\gamma \geq 0$ ("Lipschitz-Konstante") mit der Eigenschaft

$$\|F(x) - F(y)\| \leq \gamma \|x - y\|.$$

Für die Lipschitz-Konstante gelte $\gamma < 1$.⁴

Dann besitzt F genau einen Fixpunkt x^* , und die Fixpunkt-Iteration konvergiert für jeden beliebigen Startwert $x^{(0)}$ linear gegen x^* . Es gelten die folgenden Abschätzungen.

(a) Für die Abweichung der k -ten Iterierten $x^{(k)}$ vom gesuchten Punkt gilt

$$\|x^{(k)} - x^*\| \leq \gamma^k \|x^{(0)} - x^*\|.$$

(b) Der Abstand zum gesuchten Fixpunkt kann durch den Abstand zweier aufeinanderfolgender Iterierter abgeschätzt werden durch

$$\|x^{(k)} - x^*\| \leq \frac{\gamma}{1-\gamma} \|x^{(k)} - x^{(k-1)}\| \leq \frac{\gamma^2}{1-\gamma} \|x^{(k-1)} - x^{(k-2)}\| \leq \dots \leq \frac{\gamma^k}{1-\gamma} \|x^{(1)} - x^{(0)}\|.$$

(1.3.4) Bemerkung: Eine Lipschitzkonstante für f kann man häufig durch Abschätzen der ersten Ableitung von f erhalten. Betrachten wir zunächst den skalaren Fall. Nach einem Mittelwertsatz der Analysis gilt für $x < y$ mit einer geeigneten Zwischenstelle $\xi \in (x, y)$ $f(y) = f(x) + f'(\xi) \cdot (y - x)$, also

$$|f(y) - f(x)| \leq \sup_{\xi \in (x,y)} |f'(\xi)| \cdot |y - x|. \quad (1.10)$$

Eine entsprechende Aussage gilt auch für Systeme. Hierbei ist $Df(\xi)$ die Funktionalmatrix von f in ξ , und es ist

$$\|f(\mathbf{y}) - f(\mathbf{x})\| \leq \sup_{\xi \in D} \|Df(\xi)\| \cdot \|\mathbf{y} - \mathbf{x}\| \quad (1.11)$$

für beliebige kompatible Normpaare.

(1.3.5) Beispiele/Übungen: **(a)** Gesucht ist eine Nullstelle $x^* > 0$ von $f(x) =$

⁴In diesem Fall heißt F auch **Kontraktion**.

$\tan(x) - x$. Bei der Umformulierung in ein Fixpunktproblem ist die Wahl von $F(x) := \tan(x)$ wegen $F'(x) = 1/\cos^2(x) = 1 + \tan^2(x) > 1$ ungeeignet. Dagegen führt $F(x) := \arctan(x)$ wegen $F'(x) = 1/(1+x^2) < 1$ zu einem konvergenten Verfahren. Die exakte Lösung liegt in der Nähe von $x = 4.5$; dort ist $F(x) \approx 0.047$. In der Nähe der Lösung ist die Lipschitz-Konstante somit durch 0.05 beschränkt; das Fixpunktverfahren konvergiert also bei hinreichend guter Wahl des Startpunkts schnell.

(b) Wandeln Sie die Suche einer Nullstelle von $f: \mathbb{R}_+ \rightarrow \mathbb{R}$, $f(x) = \exp(x) - 2 - x$, in ein konvergentes Fixpunktproblem um.

1.4 Das Newton-Verfahren

1.4.1 Newton-Iteration, Konvergenz

Gegeben sei $\mathbf{f}: \mathbb{R}^n \rightarrow \mathbb{R}^n$. Gesucht ist eine Nullstelle \mathbf{x}^* . Wir setzen voraus

- (i) \mathbf{f} ist zweimal stetig differenzierbar;
- (ii) die Funktionalmatrix $D\mathbf{f}(\mathbf{x}^*)$ ist regulär.

Aus (i) folgt, dass \mathbf{f} lokal gut durch eine lineare Abbildung beschrieben werden kann:

$$\mathbf{f}(\mathbf{x} + \mathbf{h}) \approx \mathbf{f}(\mathbf{x}) + D\mathbf{f}(\mathbf{x}) \cdot \mathbf{h}, \quad \text{falls } \|\mathbf{h}\| \ll 1.$$

Aus (ii) folgt, dass $D\mathbf{f}(\mathbf{x})$ regulär ist, falls \mathbf{x} hinreichend nahe an \mathbf{x}^* liegt. In diesem Fall ist die Tangente

$$T(\mathbf{y}) := \mathbf{f}(\mathbf{x}) + D\mathbf{f}(\mathbf{x}) \cdot (\mathbf{y} - \mathbf{x})$$

invertierbar und besitzt insbesondere eine eindeutig bestimmte Nullstelle

$$\overset{\circ}{\mathbf{y}} = \mathbf{x} - (D\mathbf{f}(\mathbf{x}))^{-1} \mathbf{f}(\mathbf{x}).$$

Das Newton-Verfahren ist ein Iterationsverfahren, welches bei gegebener Iterierter $\mathbf{x}^{(k)}$ die neue Iterierte $\mathbf{x}^{(k+1)}$ als Nullstelle der Tangente durch $\mathbf{x}^{(k)}$ berechnet. Nach den obigen Vorbemerkungen wird das Verfahren wie folgt beschrieben.

(1.4.1) Algorithmus (Newton-Verfahren): Gewählt sei ein hinreichend guter Startvektor $\mathbf{x}^{(0)}$. Ist $\mathbf{x}^{(k)}$ gegeben, so definiere $\mathbf{x}^{(k+1)}$ wie folgt.

k	0	1	2	3
$x^{(k)}$	1.0...	0.905...	0.90001...	0.90000000001...

Tabelle 1: Berechnung von $\sqrt{0.81}$ mit dem Newton-Verfahren

- S1 Berechne $\mathbf{f}(\mathbf{x}^{(k)})$ und die Funktionalmatrix $D\mathbf{f}(\mathbf{x}^{(k)})$.
 S2 Berechne die Lösung \mathbf{s} des linearen Gleichungssystems $D\mathbf{f}(\mathbf{x}^{(k)}) \cdot \mathbf{s} = -\mathbf{f}(\mathbf{x}^{(k)})$.
 S3 Definiere $\mathbf{x}^{(k+1)} := \mathbf{x}^{(k)} + \mathbf{s}$.

(1.4.2) Beispiele: (a) Gesucht ist eine numerische Näherung von \sqrt{a} , $a > 0$. Offenbar ist $x^* = \sqrt{a}$ eine Nullstelle der Funktion $f(x) = x^2 - a$. Das Newton-Verfahren liefert als Iterationsvorschrift:

$$x^{(k+1)} = x^{(k)} - \frac{(x^{(k)})^2 - a}{2x^{(k)}} = \frac{1}{2} \left(x^{(k)} + \frac{a}{x^{(k)}} \right).$$

Ein Zahlenbeispiel für $a = 0.81$ und $x^{(0)} = 1.0$ ist in Tabelle 2 angegeben.

(b) Gesucht ist eine Nullstelle x^* der Funktion

$$f(x, y) := \begin{pmatrix} x \exp(y) - 1 \\ \sin(x) - y \end{pmatrix}.$$

Die Jacobi-Matrix von f ist

$$Df(x, y) = \begin{pmatrix} \exp(y) & x \exp(y) \\ \cos(x) & -1 \end{pmatrix}.$$

Mit dem Startwert

$$\begin{pmatrix} x^{(0)} \\ y^{(0)} \end{pmatrix} := \begin{pmatrix} 0 \\ 0 \end{pmatrix}$$

liefert das Newton-Verfahren die folgenden Werte.

$$\begin{pmatrix} x^{(1)} \\ y^{(1)} \end{pmatrix} = \begin{pmatrix} 0 \\ 0 \end{pmatrix} - \begin{pmatrix} 1 & 0 \\ 1 & -1 \end{pmatrix}^{-1} \begin{pmatrix} -1 \\ 0 \end{pmatrix} = \begin{pmatrix} 1 \\ 1 \end{pmatrix},$$

$$\begin{pmatrix} x^{(2)} \\ y^{(2)} \end{pmatrix} = \begin{pmatrix} 1 \\ 1 \end{pmatrix} - \begin{pmatrix} \exp(1) & \exp(1) \\ \cos(1) & -1 \end{pmatrix}^{-1} \begin{pmatrix} \exp(1) - 1 \\ \sin(1) - 1 \end{pmatrix} = \begin{pmatrix} 0.693... \\ 0.675... \end{pmatrix},$$

$$\begin{pmatrix} x^{(5)} \\ y^{(5)} \end{pmatrix} = \dots = \begin{pmatrix} 0.578713\dots \\ 0.546947\dots \end{pmatrix}.$$

Dieser letzte Wert hat den Funktionswert $f(x^{(5)}, y^{(5)}) = (0 \quad -1.15\text{E-}7)^T$ und ist eine sehr gute Näherung der gesuchten Nullstelle.

(c) Gesucht ist das Maximum der Funktion $H : \mathbb{R}^2 \rightarrow \mathbb{R}$,

$$H(x, y) = -5x^4 + 2(x - 1)^3(y + 1) - 100y^2.$$

Eine notwendige Bedingung, dass das Maximum im Punkt $(x^*, y^*)^T$ angenommen wird, ist, dass die partiellen Ableitungen

$$\frac{\partial H(x^*, y^*)}{\partial x} \quad \text{und} \quad \frac{\partial H(x^*, y^*)}{\partial y}$$

verschwinden. Gesucht ist damit eine Nullstelle der Funktion

$$f(x, y) := \begin{pmatrix} \frac{\partial H(x^*, y^*)}{\partial x} \\ \frac{\partial H(x^*, y^*)}{\partial y} \end{pmatrix} = \begin{pmatrix} -20x^3 + 6(x - 1)^2(y + 1) \\ 2(x - 1)^3 - 200y \end{pmatrix}.$$

Diese kann mit Hilfe des Newton-Verfahrens berechnet werden.

Zur Konvergenz des Newton-Verfahrens veweisen wir das folgende Ergebnis im skalaren Fall ($n = 1$).

(1.4.3) Satz: $D = (a, b)$ sei ein nicht-leeres offenes Intervall auf \mathbb{R} . \overline{D} sei der Abschluss. Es sei $f : \overline{D} \rightarrow \mathbb{R}$ eine zweimal stetig differenzierbare Funktion mit $\inf_{x \in \overline{D}} |f'(x)| > 0$ und $\sup_{x \in \overline{D}} |f''(x)| < \infty$. Wir definieren

$$\omega := \frac{\sup_{\overline{D}} |f''|}{\inf_{\overline{D}} |f'|}. \quad (1.12)$$

$x^* \in D$ sei Nullstelle von f ; $\rho \in (0, 2/\omega)$ sei so gewählt, dass

$$B_\rho(x^*) = \{x \in \mathbb{R} : |x - x^*| < \rho\} \subseteq D. \quad (1.13)$$

Es seien $x^{(0)} \in B_\rho(x^*)$ und $x^{(k)}$ die Iterierten des Newton-Verfahrens. Dann gilt:

- (a) Für alle $k \in \mathbb{N}$ ist $x^{(k)} \in B_\rho(x^*)$.
- (b) Es ist $\lim_k x^{(k)} = x^*$.
- (c) Es gilt

$$|x^{(k+1)} - x^*| \leq \frac{\omega}{2} |x^{(k)} - x^*|^2. \quad (1.14)$$

- (d) x^* ist die einzige Nullstelle in $B_{2/\omega}(x^*)$.

Beweis: Induktionsbeweis zu (a), (b) und (c): Sei $x^{(k)} \in B_\rho(x^*)$. Nach Definition des Newton-Verfahrens ist

$$x^{(k+1)} - x^* = x^{(k)} - x^* - f(x^{(k)})/f'(x^{(k)}). \quad (1.15)$$

Die Taylorentwicklung von $f(x^*) = 0$ um $x^{(k)}$ ergibt mit einer geeigneten Zwischenstelle $\xi \in D$

$$f(x^*) = f(x^{(k)}) + (x^* - x^{(k)}) \cdot f'(x^{(k)}) + \frac{(x^* - x^{(k)})^2}{2} \cdot f''(\xi), \quad (1.16)$$

also

$$\frac{f(x^{(k)})}{f'(x^{(k)})} = (x^{(k)} - x^*) - \frac{f''(\xi)}{2f'(x^{(k)})} \cdot (x^{(k)} - x^*)^2. \quad (1.17)$$

Eingesetzt in das Newton-Verfahren folgt mit der Definition von ω

$$|x^{(k+1)} - x^*| \leq \frac{\omega}{2} |x^{(k)} - x^*|^2. \quad (1.18)$$

Damit ist der Induktionsschritt für (c) gezeigt. Mit $q := \rho\omega/2 < 1$ gilt außerdem

$$|x^{(k+1)} - x^*| \leq q \cdot |x^{(k)} - x^*|. \quad (1.19)$$

Also ist einerseits $x^{(k+1)} \in B_\rho(x^*)$ und andererseits $x^{(k)} - x^*$ eine Nullfolge, womit (a) und (b) gezeigt sind.

Zu (d): Sei $x^{**} \in B_{2/\omega}(x^*)$ eine weitere Nullstelle von f . Ersetzen wir in allen obigen Rechnungen $x^{(k)}$ und $x^{(k+1)}$ durch x^{**} , (warum ist das sinnvoll?) so folgt

$$|x^{**} - x^*| \leq q \cdot |x^{**} - x^*| \quad (1.20)$$

mit $q < 1$, also $x^{**} = x^*$. $\quad \bigcirc$

(1.4.4) Beispiel: Das Polynom $f(x) = x^3 - 3x + 1$ hat im Intervall $\bar{D} = [\sqrt{2}, 2]$ eine Nullstelle x^* . Wie groß darf ρ gewählt werden, damit für Startwerte $x^{(0)} \in B_\rho(x^*)$ das Newton-Verfahren konvergiert?

Abgeschätzt werden müssen $\sup_{\bar{D}} |f''|$ und $\inf_{\bar{D}} |f'|$. Offenbar ist $\sup_{\bar{D}} |f''| = \sup_{\bar{D}} 6x = 12$. Außerdem ist $f'(x) = 3x^2 - 3$ in \bar{D} monoton wachsend (Begründung?), also $\inf_{\bar{D}} |f'| \geq 3$. Das ergibt einen Wert $\omega = 4$. Der Satz erfordert damit eine Genauigkeit des Startwerts von $|x^{(0)} - x^*| < 2/\omega = 0.5$.

(1.4.5) Bemerkungen: (a) Wir fassen die Ergebnisse des Satzes damit zusammen, dass das Newton-Verfahren **lokal quadratisch konvergent** ist. Die quadratische Konvergenz ergibt sich aus Teil (c) des Satzes. Das Verfahren konvergiert allerdings nur bei hinreichend gutem Startwert, daher der Zusatz "lokal".

(b) Die Ergebnisse lassen sich übertragen auf Systeme. Man sehe hierzu das Vorlesungsskript.

2 Interpolation und Approximation

2.1 Polynominterpolation

In diesem Kapitel geht es um die Frage, wie Funktionen rekonstruiert werden können, von welchen nur gewisse Funktionswerte bekannt sind. Anwendungen sind z.B.:

- Aus Messungen im Labor sind nur diskrete Werte $f(t_1), f(t_2), \dots$ bekannt;
- Funktionswerte einer Funktion sind tabellarisch aufgelistet;
- eine kontinuierliche Funktion soll abgespeichert und später rekonstruiert werden.

2.1.1 Das einfache Interpolationsproblem

Gegeben sei ein Funktionenraum \mathcal{V} . Weiter seien Knoten $t_0 < \dots < t_n$ und Funktionswerte f_0, \dots, f_n gegeben. Das **Interpolationsproblem** in \mathcal{V} lautet: Finde eine Funktion $P(t) \in \mathcal{V}$, welches die **Interpolationseigenschaft** erfüllt:

$$P(t_i) = f_i, \quad i = 0, \dots, n.$$

(2.1.1) Definition: Ein System (b_0, \dots, b_n) von Funktionen in \mathcal{V} heißt **IP-System** zu den Knoten t_0, \dots, t_n , falls gilt

$$b_i(t_j) = \begin{cases} 1 & \text{falls } i = j \\ 0 & \text{sonst} \end{cases}.$$

Der von b_0, \dots, b_n aufgespannte Raum

$$\mathcal{V}_n = \text{span}(b_0, \dots, b_n)$$

heißt der **IP-Unterraum** von \mathcal{V} zur Basis b_0, \dots, b_n .

(2.1.2) Satz: In \mathcal{V}_n gibt es genau eine Lösung des IP-Problems. Diese ist gegeben durch

$$P(t) = \sum_{j=0}^n f_j b_j(t).$$

Beweis: Nach Definition der b_i ist $P(t)$ offenbar Lösung des IP-Problems. Ist $Q(t) = \sum_{j=0}^n \lambda_j b_j(t)$ eine weitere Lösung des IP-Problems, so folgt für alle $i = 0, \dots, n$ aus $Q(t_i) = f_i$ und den Eigenschaften der b_i die Beziehung $\lambda_i = f_i$. \circ

(2.1.3) Beispiele: Als Funktionenraum \mathcal{V} können wir z.B. die Menge der stetigen Funktionen auf $[t_0, t_n]$ wählen.

(a) Polynominterpolation: Ein IP-System wird gebildet durch die **Lagrange-Polynome**

$$L_i(t) := \prod_{\substack{j=0 \\ j \neq i}}^n \frac{t - t_j}{t_i - t_j}.$$

(b) Bandbegrenzte Interpolation: Die Funktionen des IP-Systems sind

$$b_i(t) = \frac{h}{\pi(t - ih)} \cdot \sin\left(\frac{\pi(t - ih)}{h}\right).$$

Diese Funktionen werden vor allem in der Elektrotechnik häufig benutzt zur Interpolation (Stichwort *Abtastreihe*).

2.2 Lagrange-Interpolation

Im Folgenden seien L_i die Lagrangepolynome zu den Knoten t_0, \dots, t_n , und

$$\mathcal{V}_n = \text{span}(L_0, \dots, L_n).$$

(2.2.4) Lemma: Es ist

$$\mathcal{V}_n = \mathcal{P}_n = \text{Menge der Polynome vom Grad } \leq n.$$

$\{L_i | i = 0, \dots, n\}$ ist eine Basis von \mathcal{P}_n .

Beweis: Offenbar liegen alle L_i in \mathcal{P}_n ; damit ist $\mathcal{V}_n \subseteq \mathcal{P}_n$. Die L_i sind (als Funktionen auf $[t_0, t_n]$) linear unabhängig, denn: Sei

$$\sum_{j=0}^n c_j L_j(t) = 0 \quad \text{für alle } t \in [t_0, t_n].$$

Wegen $L_i(t_j) = \delta_{ij}$ folgt dann aber

$$\sum_{j=0}^n c_j L_j(t_i) = c_i = 0 \quad \text{für alle } t \in [t_0, t_n]. \quad (2.1)$$

Damit ist $\{L_i | i = 0, \dots, n\}$ ist eine Basis von \mathcal{V}_n $\dim \mathcal{V}_n = n + 1$. Da außerdem gilt $\dim \mathcal{P}_n = n + 1$ (Beweis?), folgt $\mathcal{V}_n = \mathcal{P}_n$. \circ

Zusammen mit Satz (3.1.2) erhalten wir

(2.2.5) Folgerung: Es gibt genau ein Polynom $P(t) \in \mathcal{P}_n$, welches die Interpolationsbedingungen (1.1) erfüllt. (Dieses wird das **Interpolationspolynom** genannt.) Es ist gegeben durch

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

(2.2.6) Beispiele: (a) Die zu den Knoten $t_0 = 0$, $t_1 = 1$, $t_2 = 3$ und $t_3 = 4$ gehörigen Lagrange-Polynome sind

$$\begin{aligned} L_0(t) &= -\frac{1}{12}(t-1)(t-3)(t-4) = -\frac{1}{12}(t^3 - 8t^2 + 19t - 12), \\ L_1(t) &= \frac{1}{6}t(t-3)(t-4) = \frac{1}{6}(t^3 - 7t^2 + 12t), \\ L_2(t) &= -\frac{1}{6}t(t-1)(t-4) = -\frac{1}{6}(t^3 - 5t^2 + 4t), \\ L_3(t) &= \frac{1}{12}t(t-1)(t-3) = \frac{1}{12}(t^3 - 4t^2 + 3t). \end{aligned}$$

Das Interpolationspolynom zu den Werten $f_0 = 1$, $f_1 = 0$, $f_2 = -1$ und $f_3 = 2$ ist

$$P(t) = L_0(t) - L_2(t) + 2L_3(t) = \frac{1}{12}(3t^3 - 10t^2 - 5t + 12).$$

(b) Der Wert $\sin(66.3^\circ)$ soll bestimmt werden. In einem Tafelwerk finden wir die tabellierten Werte

$$\begin{aligned} \sin(60^\circ) &= 0.866025 \\ \sin(65^\circ) &= 0.906308 \\ \sin(70^\circ) &= 0.939693 \\ \sin(75^\circ) &= 0.965926. \end{aligned}$$

(i) Das *lineare* Interpolationspolynom zu den Knoten $t_1 = 65$ und $t_2 = 70$ ist

$$P_1(t) = 0.0066770000 \cdot t + 0.47230300$$

und liefert

$$\sin(66.3^\circ) \approx P_1(66.3) = 0.914988.$$

(ii) Das Interpolationspolynom $P_3(t)$ dritten Grades zu den Knoten $t_0 = 60$, $t_1 = 65$, $t_2 = 70$ und $t_3 = 75$ ist

$$P_3(t) = -0.33866667 \cdot 10^{-6}t^3 - 0.71920000 \cdot 10^{-4}t^2 + 0.021017467 \cdot t - 0.062959000.$$

Dies führt auf die Approximation

$$\sin(66.3^\circ) \approx P_3(66.3) = 0.915662.$$

Zum Vergleich: Der exakte Wert ist $\sin(66.3^\circ) = 0.91566259\dots$.

Wir untersuchen zunächst die **Kondition** des Interpolationsproblems. Hierbei geht es nicht um die Frage, *wie genau* ein Funktionsverlauf durch Interpolation rekonstruiert werden kann (dies geschieht in Abschnitt 1.3), sondern darum, wie stark sich *Rundungsfehler* auf das Ergebnis auswirken. Zunächst fassen wir die Knoten t_i und die Funktionswerte f_i zu Vektoren zusammen:

$$\mathbf{t} := (t_0, \dots, t_n)^T, \quad \mathbf{f} := (f_0, \dots, f_n)^T.$$

Im Folgenden werde \mathbf{t} als Folge paarweise unterschiedlicher Punkte festgehalten. Es bezeichne $P[\mathbf{f}](t)$ das nach Satz [1.3] eindeutige Interpolationspolynom zu den Werten \mathbf{f} . Aus der Darstellung (1.2) für das Interpolationspolynom folgt, dass die Abbildung $P[\cdot] : \mathbb{R}^{n+1} \rightarrow \mathcal{P}_n$ linear ist. Hieraus folgt: Wird der Vektor \mathbf{f} durch Rundungsfehler $\mathbf{h} = (h_0, \dots, h_n)^T$ gestört, so hat dies den Fehler

$$P[\mathbf{f} + \mathbf{h}] - P[\mathbf{f}] = P[\mathbf{h}] \tag{2.2}$$

zur Folge. Zur Untersuchung des Rundungsfehlereinflusses muss also nur $P[\mathbf{h}]$ betrachtet werden. Wir führen die beiden folgenden Maximumnormen ein:

$$\|\mathbf{h}\|_\infty := \max_{i=0, \dots, n} |h_i|, \tag{2.3}$$

$$\|P[\mathbf{h}]\|_\infty := \sup_{t \in [a, b]} |P[\mathbf{h}](t)|. \tag{2.4}$$

Die **absolute Kondition** bezüglich der Maximumnorm ist definiert durch

$$\kappa_{\text{abs}} = \sum_{\mathbf{h} \in \mathbb{R}^{n+1}} \frac{\|P[\mathbf{h}]\|_\infty}{\|\mathbf{h}\|_\infty}. \tag{2.5}$$

n	äquidistante Knoten	Tschebyscheff-Knoten
5	3.106	2.104
10	29.890	2.489
15	512.052	2.728
20	10986.533	2.901

Tabelle 2: Lebesgue-Konstante Λ_n

Diese kann mit Hilfe der Lagrange-Polynome wie folgt beschrieben werden.

(2.2.7) Satz: Es seien $a \leq t_0 < \dots < t_n \leq b$ paarweise verschiedene Knoten und L_{in} , $i = 0, \dots, n$ die zugehörigen Lagrange-Polynome. Dann gilt

$$\kappa_{\text{abs}} = \max_{t \in [a,b]} \sum_{i=0}^n |L_{in}(t)|. \quad (2.6)$$

Beweis: Zur Abkürzung definieren wir

$$\Lambda_n := \max_{t \in [a,b]} \sum_{i=0}^n |L_{in}(t)|.$$

(Λ_n heißt auch “Lebesgue-Konstante”.) Wir haben zunächst zu zeigen, dass $\kappa_{\text{abs}} \leq \Lambda_n$. Dies folgt aus

$$|P[\mathbf{h}](t)| = \left| \sum_{i=0}^n h_i L_{in}(t) \right| \leq \sum_{i=0}^n |h_i| \cdot |L_{in}(t)| \leq \|\mathbf{h}\|_{\infty} \cdot \Lambda_n. \quad (2.7)$$

Es sei nun $t^* \in [a, b]$ der Wert, an dem $\sum_{i=0}^n |L_{in}(t)|$ sein Maximum annimmt. Definieren wir den speziellen Vektor \mathbf{h}^* durch

$$h_i^* := \text{sign}(L_{in}(t^*)),$$

so ist $\|\mathbf{h}^*\|_{\infty} = 1$ und

$$\|P[\mathbf{h}]\|_{\infty} = |P[\mathbf{h}](t^*)| = \sum_{i=0}^n |L_{in}(t^*)| = \Lambda_n. \quad \circ$$

(2.2.8) Beispiel: Die Kondition κ_{abs} hängt wesentlich von der Wahl der Knoten ab.

Tabelle 1 gibt für das Intervall $[-1, 1]$ für verschiedene n die Lebesgue-Konstante an. Als Knoten wurden hierbei einmal *äquidistante Knoten* $t_i = (2i+1)/(n+1) - 1$ gewählt, und andererseits zum Vergleich die sog. *Tschebyscheff-Knoten* $t_i = \cos(\pi \cdot (2i+1)/(2n+2))$. (Als Begründung für die Wahl der Tschebyscheff-Knoten vgl. Abschnitt 1.4.)

Sollen von einem Interpolationspolynom $P(\cdot)$ nur einzelne Werte $P(t)$ berechnet werden, so ist es nicht nötig, die Koeffizienten des Polynoms zu berechnen. Numerisch günstiger ist die Anwendung des Schemas von Aitken-Neville. Grundlage hierfür ist das folgende Ergebnis. Es seien die Knoten \mathbf{t} und die Funktionswerte \mathbf{f} fest vorgegeben; $P[\mathbf{f}]$ bezeichne wie vorher das zugehörige Interpolationspolynom n -ten Grades. Des weiteren bezeichne $P[\mathbf{f}|\hat{t}_i]$ das Interpolationspolynom $n-1$ -ten Grades zu den n Paaren $(t_0, f_0), \dots, (t_{i-1}, f_{i-1}), (t_{i+1}, f_{i+1}), \dots, (t_n, f_n)$. Das folgende Lemma zeigt, dass sich $P[\mathbf{f}]$ rekursiv aus Interpolationspolynomen niedrigeren Grades aufbauen lässt.

(2.2.9) Lemma: Für $P[\mathbf{f}]$ gilt die Rekursionsformel

$$P[\mathbf{f}](t) = \frac{(t_0 - t)P[\mathbf{f}|\hat{t}_0](t) - (t_n - t)P[\mathbf{f}|\hat{t}_n](t)}{t_0 - t_n} .$$

Beweis: Zur Abkürzung definieren wir

$$\phi(t) := \frac{(t_0 - t)P[\mathbf{f}|\hat{t}_0](t) - (t_n - t)P[\mathbf{f}|\hat{t}_n](t)}{t_0 - t_n} .$$

Offenbar ist $\phi \in \mathcal{P}_n$. Außerdem ist

$$\begin{aligned} \phi(t_0) &= P[\mathbf{f}|\hat{t}_n](t_0) = f_0, \\ \phi(t_n) &= P[\mathbf{f}|\hat{t}_0](t_n) = f_n, \end{aligned}$$

und für $i = 1, \dots, n-1$ gilt

$$\phi(t_i) = \frac{(t_0 - t_i)f_i - (t_n - t_i)f_i}{t_0 - t_n} = f_i.$$

Damit ist $\phi = P[\mathbf{f}]$. $\quad \circ$

Berücksichtigt man nun, dass auch die Polynome $P[\mathbf{f}|\hat{t}_i]$ auf Polynome niedrigeren Grades zurückgeführt werden können, so ergibt sich schließlich das folgende Rekursionsverfahrens.

(2.2.10) Schema von Aitken-Neville: Berechne die Werte π_{ik} , $k = 0 \dots n$, $i = k \dots n$ nach dem Schema

$$\begin{array}{ccccccc}
 f_0 & = & \pi_{00} & & & & \\
 & & & \searrow & & & \\
 f_1 & = & \pi_{10} & \rightarrow & \pi_{11} & & \\
 & & \vdots & & \ddots & & \\
 & & \vdots & & & \ddots & \\
 & & \vdots & & & & \ddots \\
 f_{n-1} & = & \pi_{n-1,0} & \rightarrow & \pi_{n-1,1} & \rightarrow & \dots \rightarrow \pi_{n-1,n-1} \\
 & & & \searrow & & \searrow & & \searrow & & \searrow \\
 f_n & = & \pi_{n0} & \rightarrow & \pi_{n1} & \rightarrow & \dots \rightarrow \pi_{n,n-1} & \rightarrow & \pi_{nn}
 \end{array}$$

wobei die π_{ik} für $k \geq 1$ aus den Werten der vorhergehenden Spalte bestimmt werden nach der Rekursionsformel

$$\begin{aligned}
 \pi_{ik} &= \frac{(t - t_{i-k})\pi_{i,k-1} + (t_i - t)\pi_{i-1,k-1}}{t_i - t_{i-k}} \\
 &= \pi_{i,k-1} + \frac{t - t_i}{t_i - t_{i-k}} \cdot (\pi_{i,k-1} - \pi_{i-1,k-1}).
 \end{aligned}$$

Dann ist $P(t) = \pi_{nn}$.⁵

(2.2.11) Beispiel: Der Wert $P_3(66.3)$ des Beispiels [1.4](b) folgt somit aus dem Schema

$$\begin{aligned}
 \sin(60^\circ) &= 0.866025 \\
 \sin(65^\circ) &= 0.906308 \quad 0.9167816 \\
 \sin(70^\circ) &= 0.939693 \quad 0.9149881 \quad 0.9156517 \\
 \sin(75^\circ) &= 0.965926 \quad 0.9202806 \quad 0.9156761 \quad 0.9156619.
 \end{aligned}$$

Aus dem Schema von Aitken-Neville kann man sich leicht den folgenden Algorithmus herleiten.

(2.2.12) Algorithmus zum Schema von Aitken-Neville:

- (S1) Für $k = 0(1)n$: Setze $p_k := f_k$.
- (S2) Für $k = 1(1)n$
 Für $i = n(-1)k$:
 Setze $p_i := p_i + (t - t_i) \cdot (p_i - p_{i-1}) / (t_i - t_{i-k})$.

⁵Die Werte π_{kk} sind die Werte des Interpolationspolynoms zu den Knoten t_0, \dots, t_k an der Stelle t .

2.3 Das erweiterte Interpolationsproblem

Wir wollen nun das Interpolationsproblem verallgemeinern und außer Funktionswerten f_i auch Ableitungen $f_i^{(l)}$ an den Knoten t_i vorschreiben. Diese Art der Polynominterpolation heißt **Hermite-Interpolation**. Anstelle einer strikt wachsenden Folge von Knoten $t_0 < t_1 < \dots$ lassen wir nun gleiche Knoten zu: $t_0 \leq t_1 \leq \dots$. Werden an einem Knoten außer dem Funktionswert f_i auch die Ableitungen $f_i', \dots, f_i^{(k)}$ vorgeschrieben, so muss dieser Knoten $k + 1$ -mal in der Knotenfolge auftreten. Wir definieren

$$d_i := \max\{j : t_i = t_{j-i}\}.$$

Beispiel: Zu den Knoten $\tau_0 < \tau_1 < \tau_2 < \tau_3$ seien die Werte $f(\tau_0)$, $f'(\tau_0)$, $f(\tau_1)$, $f'(\tau_1)$, $f''(\tau_1)$, $f(\tau_2)$, $f(\tau_3)$ und $f'(\tau_3)$ vorgegeben. Hieraus ergibt sich mit $t_0 = \tau_0$, $t_2 = \tau_1$, $t_5 = \tau_2$, $t_6 = \tau_3$ die folgende Knotenfolge für die Hermite-Interpolation.

τ_i	τ_0	τ_0	τ_1	τ_1	τ_1	τ_2	τ_3	τ_3
t_i	t_0	$= t_1$	$< t_2$	$= t_3$	$= t_4$	$< t_5$	$< t_6$	$= t_7$
d_i	0	1	0	1	2	0	0	1
ξ_i	$f(\tau_0)$	$f'(\tau_0)$	$f(\tau_1)$	$f'(\tau_1)$	$f''(\tau_1)$	$f(\tau_2)$	$f(\tau_3)$	$f'(\tau_3)$

Definieren wir nun $\xi = (\xi_0, \dots, \xi_n)^T$ mit $\xi_i := f_i^{(d_i)}$ und die Abbildungen

$$\mu_i : \mathcal{P}_n \longrightarrow \mathbb{R}, \quad \mu_i(P) := P^{(d_i)}(t_i), \quad i = 0, \dots, n$$

so lautet das neue Interpolationsproblem

(2.3.13) Aufgabe der Hermite-Interpolation: Finde $P \in \mathcal{P}_n$ mit

$$\mu_i(P) = \xi_i. \tag{2.8}$$

Die Lösung $P =: P(\xi|t_0, \dots, t_n)$ heißt **Hermite-Interpolierende** zu den Daten ξ an den Knoten t_i .

Ähnlich wie in Satz [1.1] kann gezeigt werden

(2.3.14) Satz: Zu jedem Vektor $\xi \in \mathbb{R}^{n+1}$ und zu jeder monotonen Knotenfolge

$$a = t_0 \leq t_1 \leq \dots \leq t_n = b$$

gibt es genau eine Hermite-Interpolierende $P(\xi|t_0, \dots, t_n)$.

(2.3.15) Spezialfälle: (a) Im Falle paarweise verschiedener Knoten t_i entspricht die Hermite-Interpolation der Interpolation des Abschnitts 1.1.

(b) Der Fall $t_0 = t_1 = \dots = t_n$ führt auf die *Taylor-Interpolation*. Die zugehörige Hermite-Interpolierende ist die abgeschnittene Taylorreihe

$$P(\xi|t_0, \dots, t_n)(t) = \sum_{j=0}^n \frac{(t - t_0)^j}{j!} \xi_j \quad .$$

(c) *Kubische Hermite-Interpolation:* Hierbei ist $n = 3$; an den Knoten $\tau_0 < \tau_1$ seien die Funktionswerte f_i sowie die ersten Ableitungen f'_i vorgegeben. Zur Konstruktion der Hermite-Interpolierenden wird zunächst die *kubische Hermite-Basis* $\{H_0, \dots, H_3\}$ konstruiert. Diese ist eindeutig bestimmt durch die folgenden Vorgaben.

	$H_i(\tau_0)$	$H'_i(\tau_0)$	$H_i(\tau_1)$	$H'_i(\tau_1)$
i=0	1	0	0	0
i=1	0	1	0	0
i=2	0	0	1	0
i=3	0	0	0	1

Dann ist die Hermite-Interpolierende gegeben durch

$$P(\xi|t_0, \dots, t_3) = \sum_{i=0}^3 \xi_i \cdot H_i \quad . \quad (2.9)$$

(2.3.16) Beispiel: Gesucht ist die Hermite-Interpolierende, welche an den Knoten $\tau_0 = 0$ und $\tau_1 = \pi$ die selben Funktionswerte und die selben ersten Ableitungen hat wie $\sin(t)$. Offenbar sind die Knotenfolge \mathbf{t} und die Folge ξ gegeben durch

$$\mathbf{t} = (0, 0, \pi, \pi)^T, \quad \xi = (0, 1, 0, -1)^T,$$

und für die Interpolierende gilt nach (1.11) die Darstellung

$$P(\xi|\mathbf{t})(t) = H_1(t) - H_3(t).$$

H_1 und H_3 sind Polynome dritten Grades. H_1 ergibt sich aus den Bedingungen $H_1(0) = H_1(\pi) = H'_1(\pi) = 0$ und $H'_1(0) = 1$ als

$$H_1(t) = \frac{1}{\pi^2} \cdot t \cdot (t - \pi)^2 \quad .$$

Entsprechend folgt H_3 aus $H_3(0) = H_3(\pi) = H_3'(0) = 0$ und $H_3'(\pi) = 1$,

$$H_3(t) = \frac{1}{\pi^2} \cdot t^2 \cdot (t - \pi) \quad .$$

Damit ist die Hermite-Interpolierende

$$P(\xi|\mathbf{t})(t) = \frac{1}{\pi^2} \left[t \cdot (t - \pi)^2 - t^2 \cdot (t - \pi) \right] = -\frac{1}{\pi} \cdot t \cdot (t - \pi) \quad .$$

Zur Lösung des Interpolationsproblems führen wir eine weitere Basis von \mathcal{P}_n ein. Die Menge $\{\omega_0, \dots, \omega_n\}$, welche definiert ist durch

$$\omega_0(t) := 1, \quad \omega_i(t) := \prod_{j=0}^{i-1} (t - t_j) \quad (\omega_i \in \mathcal{P}_i)$$

heißt **Newton-Basis**. (Wieso ist dies eine Basis des \mathcal{P}_n ?)

Der führende Koeffizient a_n des Interpolationspolynoms

$$P(\xi|t_0, \dots, t_n) = a_n t^n + a_{n-1} t^{n-1} + \dots + a_0$$

zu den Knoten $t_0 \leq t_1 \leq \dots \leq t_n$ heißt *n-te dividierte Differenz* und wird mit

$$\xi[t_0, \dots, t_n] := a_n$$

bezeichnet.

(2.3.17) Satz: Bezüglich der Newton-Basis besitzt $P(\xi|t_0, \dots, t_n)$ die Darstellung

$$P_n := P(\xi|t_0, \dots, t_n) = \sum_{i=0}^n \xi[t_0, \dots, t_i] \cdot \omega_i \quad . \quad (2.10)$$

Beweis durch Induktion: Für $n = 0$ ist die Aussage richtig.

Es gelte

$$P_{n-1} := P(\xi_0, \dots, \xi_{n-1}|t_0, \dots, t_{n-1}) = \sum_{i=0}^{n-1} \xi[t_0, \dots, t_i] \cdot \omega_i \quad .$$

Wir fügen den zusätzlichen Knoten t_n hinzu und erhalten

$$\begin{aligned} P_n(t) &= \xi[t_0, \dots, t_n] \cdot t^n + a_{n-1} t^{n-1} + \dots + a_0 \\ &= \xi[t_0, \dots, t_n] \cdot \omega_n(t) + Q_{n-1}(t) \end{aligned}$$

mit einem Polynom $Q_{n-1} \in \mathcal{P}_{n-1}$. Da $\omega_n(t)$ an den Knoten $t_i, i = 0, \dots, n-1$ verschwindet, ist

$$Q_{n-1} = P_n - \xi[t_0, \dots, t_n] \cdot \omega_n(t)$$

das Interpolationspolynom zu den Knoten t_0, \dots, t_{n-1} , also $Q_{n-1} = P_{n-1}$. \circ

Ähnlich wie Funktionswerte des Interpolationspolynoms mit Hilfe des Aitken-Neville-Schemas können die dividierten Differenzen rekursiv durch das folgende Schema berechnet werden.

(2.3.18) Berechnung der dividierten Differenzen: Die Berechnung erfolgt durch das Schema

$$\begin{array}{cccccccc}
 \xi_0 & = & \xi[t_0] & & & & & \\
 & & & \searrow & & & & \\
 \xi_{1-d_1} & = & \xi[t_1] & \rightarrow & \xi[t_0, t_1] & & & \\
 & & \vdots & & \ddots & & & \\
 & & \vdots & & & \ddots & & \\
 & & \vdots & & & & \ddots & \\
 \xi_{n-1-d_{n-1}} & = & \xi[t_{n-1}] & \rightarrow & \xi[t_{n-2}, t_{n-1}] & \rightarrow & \cdots & \rightarrow & \xi[t_0, \dots, t_{n-1}] \\
 & & & \searrow & & \searrow & & \searrow & \\
 \xi_{n-d_n} & = & \xi[t_n] & \rightarrow & \xi[t_{n-1}, t_n] & \rightarrow & \cdots & \rightarrow & \xi[t_1, \dots, t_n] & \rightarrow & \xi[t_0, \dots, t_n]
 \end{array}$$

mit den folgenden "Rechenregeln".

(a) Für zusammenfallende Knoten $t_k = \dots = t_{k+l}$ ist

$$\xi[t_k, \dots, t_{k+l}] = \frac{\xi_{k+l}}{l!} .$$

(b) Ist $t_i \neq t_j$, so gilt die Rekursionsformel

$$\xi[t_0, \dots, t_n] = \frac{\xi[t_0, \dots, \hat{t}_i, \dots, t_n] - \xi[t_0, \dots, \hat{t}_j, \dots, t_n]}{t_j - t_i} ,$$

wobei " \hat{t}_k " bedeutet, dass der Knoten t_k und in der Liste $\xi = (\xi_0, \dots, \xi_n)^T$ die Komponente ξ_k gestrichen werden.

Beweis: **(a)** Ist $t_k = \dots = t_{k+l}$, so ist nach [1.14](b)

$$P(\xi_k, \dots, \xi_{k+l} | t_k, \dots, t_{k+l})(t) = \sum_{j=0}^l \frac{(t - t_k)^j}{j!} \xi_{k+j} \quad ;$$

insbesondere ist der führende Koeffizient gleich $\xi_{k+l}/l!$

(b) Ist $t_i \neq t_j$, so gilt

$$P(\xi|t_0, \dots, t_n) = \frac{(t_i - t)P(\xi|t_1, \dots, \hat{t}_j, \dots, t_n) - (t_j - t)P(\xi|t_1, \dots, \hat{t}_i, \dots, t_n)}{t_i - t_j} .$$

(Begründung?) Hieraus folgt die Darstellung für den führenden Koeffizienten. ○

(2.3.19) Beispiele: (a) Das Schema für das Beispiel [1.15] lautet

$$\begin{array}{cccc} [0] & & & \\ [0] & [1] & & \\ [0] & 0 & -1/\pi & \\ [0] & [-1] & -1/\pi & 0 \end{array}$$

wobei die in eckige Klammern gesetzten Werte durch die Regel (a), die anderen durch die Regel (b) berechnet wurden. Damit ist die Hermite-Interpolierende gegeben durch

$$P(\xi|t) = 0 \cdot 1 + 1 \cdot t - 1/\pi \cdot t^2 + 0 \cdot t^2 \cdot (t - \pi) = -\frac{1}{\pi} \cdot t(t - \pi) .$$

(b) Zu den Vorgaben des Interpolationsproblems [1.15] soll hinzugefügt werden: $P(\pi/6) = P(5 \cdot \pi/6) = 0.5$. Hieraus ergibt sich das folgende Schema.

$$\begin{array}{l|cccccc} t_0 = 0 & [0] & & & & & \\ t_1 = 0 & [0] & [1] & & & & \\ t_2 = \pi/6 & [0.5] & 3/\pi & -6/\pi + 18/\pi^2 & & & \\ t_3 = 5\pi/6 & [0.5] & 0 & -18/5\pi^2 & 36/5\pi^2 - 648/25\pi^3 & & \\ t_4 = \pi & [0] & -3/\pi & -18/5\pi^2 & 0 & -36/5\pi^3 + 648/25\pi^4 & \\ t_5 = \pi & [0] & [-1] & -6/\pi + 18/\pi^2 & -36/5\pi^2 + 648/25\pi^3 & -36/5\pi^3 + 648/25\pi^4 & 0 \end{array}$$

Damit ist

$$\begin{aligned} P(\xi|t)(t) &= 1 \cdot \omega_1(t) + \frac{6(3 - \pi)}{\pi^2} \cdot \omega_2(t) + \frac{36(5\pi - 18)}{25\pi^3} \omega_3(t) - \frac{36(5\pi - 18)}{25\pi^4} \omega_4(t) \\ &= t + 0.016104 \cdot t^2 - 0.212895 \cdot t^3 + 0.033883 \cdot t^4 . \end{aligned}$$

(c) Die Exponentialfunktion soll in der Nähe von $t = 0$ durch ein Polynom $P(t)$ approximiert werden. Dieses Polynom soll möglichst niedrigen Grades sein und definiert durch die Bedingungen $P(-1) = 1/e$, $P(0) = P'(0) = P''(0) = 1$ sowie $P(1) = e$. Aus

dem Schema

$$\begin{array}{l|l}
 t_0 = -1 & [1/e] \\
 t_1 = 0 & [1] \quad (e-1)/e \\
 t_2 = 0 & [1] \quad [1] \quad 1/e \\
 t_3 = 0 & [1] \quad [1] \quad [1/2] \quad (e-2)/2e \\
 t_4 = 0 & [1] \quad [1] \quad [1/2] \quad [1/6] \quad (3-e)/3e \\
 t_5 = 1 & [e] \quad e-1 \quad e-2 \quad e-5/2 \quad e-8/3 \quad 0.5 \cdot (e-1/e-7/3)
 \end{array}$$

folgt

$$\begin{aligned}
 P(t) &= \frac{1}{e} + \left(1 - \frac{1}{e}\right)(t+1) + \frac{1}{e}(t+1)t + \left(\frac{1}{2} - \frac{1}{e}\right)(t+1)t^2 + \left(\frac{1}{e} - \frac{1}{3}\right)(t+1)t^3 \\
 &\quad + 0.5 \cdot \left(e - \frac{1}{e} - \frac{7}{3}\right)(t-1)t^4 \\
 &= 1 + t + 0.5t^2 + 0.166667t^3 + 0.043081t^4 + 0.008535t^5 \quad .
 \end{aligned}$$

(2.3.20) Übung: Überprüfen Sie: Ist es im Fall paarweise verschiedener Knoten t_i wichtig, die Knoten in wachsender Reihenfolge anzuordnen? Was ergibt sich hieraus, wenn zum bereits berechneten Interpolationspolynom ein weiterer Knoten hinzugefügt werden soll?

2.4 Approximationsfehler

Bisher haben wir die Werte $\xi = (\xi_0, \dots, \xi_n)^T$ immer als isoliert vorgegeben gedacht. Jetzt wollen wir dies leicht modifizieren. Wir stellen uns eine Knotenfolge $a = t_0 \leq \dots \leq t_n = b$ und eine hinreichend glatte Funktion $f : [a, b] \rightarrow \mathbb{R}$ vor und fragen uns, wie gut diese Funktion durch das zugehörige Interpolationspolynom im gesamten Intervall $[a, b]$ approximiert wird. Letzteres ist durch das eindeutig bestimmte Polynom $P \in \mathcal{P}_n$ definiert, welches die Aufgabe [1.12] löst, wobei $\xi_i = f^{(d_i)}(t_i)$. Dieses Polynom werde im Folgenden auch mit $P[\mathbf{f}|\mathbf{t}]$ bezeichnet, wobei $\mathbf{f} = (f^{(d_0)}(t_0), \dots, f^{(d_n)}(t_n))^T$.

(2.4.21) Spezialfall: Für den Spezialfall $t_0 = \dots = t_n$ und für mindestens $n+1$ -mal stetig differenzierbare Funktionen f kann der Approximationsfehler $f(t) - P[\mathbf{f}|\mathbf{t}](t)$ leicht aus den bisher bekannten Ergebnissen der Analysis hergeleitet werden. Laut [1.14](b)

ist nämlich das Interpolationspolynom gegeben als abgeschnittene Taylorreihe

$$P[\mathbf{f}|\mathbf{t}](t) = \sum_{j=0}^n \frac{f^{(j)}(t_0)}{j!} \cdot (t - t_0)^j.$$

Andererseits besagt die bekannte Restgliedformel für die Taylorreihe, dass

$$f(t) = \sum_{j=0}^n \frac{f^{(j)}(t_0)}{j!} \cdot (t - t_0)^j + \frac{f^{(n+1)}(\tau)}{(n+1)!} \cdot (t - t_0)^{n+1}$$

mit einer geeigneten Stelle τ zwischen t_0 und t . Berücksichtigen wir nun noch, dass in unserem Spezialfall gilt $(t - t_0)^{n+1} = \omega_{n+1}(t)$, so lässt sich der Approximationsfehler darstellen durch

$$f(t) - P[\mathbf{f}|\mathbf{t}](t) = \frac{f^{(n+1)}(\tau)}{(n+1)!} \cdot \omega_{n+1}(t) \quad (2.11)$$

Wir werden im Folgenden zumindest skizzieren, dass sich die Formel (1.13) auf den allgemeinen Fall übertragen lässt.

(2.4.22) Lemma: Für $t \notin \{t_0, \dots, t_n\}$ und für die erweiterten Vektoren $\bar{\mathbf{t}} = (t_0, \dots, t_n, t)^T$ sowie $\bar{\xi} = (\xi_0, \dots, \xi_n, f(t))^T$ gilt

$$f(t) = P[\mathbf{f}|\mathbf{t}](t) + \xi[t_0, \dots, t_n, t] \cdot \omega_{n+1}(t).$$

Beweis: Nach Satz [1.16] ist

$$P[\mathbf{f}|\mathbf{t}](t) = \sum_{i=0}^n \xi[t_0, \dots, t_i] \cdot \omega_i(t) \quad ,$$

und das Polynom

$$P_{n+1}(s) := P[\mathbf{f}|\mathbf{t}](s) + \xi[t_0, \dots, t_n, t] \cdot \omega_{n+1}(s)$$

ist das Interpolationspolynom von f zu den Knoten $\bar{\mathbf{t}}$. Insbesondere ist

$$P_{n+1}(t) = f(t). \quad \bigcirc$$

Eine Art Mittelwertsatz liegt der folgenden Darstellungsformel für dividierte Differenzen

zugrunde; wir wollen sie hier nicht beweisen, da hierzu die Berechnung höherdimensionaler Integrale benötigt wird.⁶

(2.4.23) Lemma: Für beliebige $n \in \mathbb{N}$ und Knotenfolgen $a = t_0 \leq \dots \leq t_n = b$ existieren Zwischenwerte $\tau \in [a, b]$ mit

$$\xi[t_0, \dots, t_n] = \frac{f^{(n)}(\tau)}{n!} \quad .$$

Als zentrales Ergebnis folgt aus den Lemmata [1.21] und [1.22]

(2.4.24) Satz: Ist $f : [a, b] \rightarrow \mathbb{R}$ $n + 1$ -mal stetig differenzierbar, so gilt für den Approximationsfehler der Hermite-Interpolierenden die Darstellung

$$f(t) - P[\mathbf{f}|\mathbf{t}](t) = \frac{f^{(n+1)}(\tau)}{(n+1)!} \cdot \omega_{n+1}(t)$$

mit einer geeigneten Zwischenstelle $\tau \in [a, b]$.

(2.4.25) Übung: (a) Plotten Sie $\omega_n(t)$ für $n = 5, 10, 20$ für

(i) äquidistante, (ii) Tschebyscheff-Knoten.

(b) Die Funktion $\sin(t)$ sei im Intervall $[60^\circ, 75^\circ]$ an $n + 1$ äquidistanten Stützstellen gegeben. Plotten Sie mit Hilfe eines Grafikprogramms die Interpolationspolynome, die Interpolationsfehler sowie die Polynome $\omega_{n+1}(t)$ für $n = 4, 6, 8, 10$.

2.5 Tschebyscheff-Interpolation

Interpolationspolynome hohen Grades zu äquidistanten Stützstellen sind praktisch unbrauchbar, da diese starke Oszillationen aufweisen. Die Oszillationen stehen gemäß Satz [1.23] im Zusammenhang mit dem Verhalten der Polynome $\omega_n(t)$ in der Nähe der Ränder (vgl. Übung [1.24](a)). Die Untersuchung der Frage, wie die Knoten t_0, \dots, t_n anzuordnen sind, damit

$$\max_{t \in [t_0, t_n]} |\omega_{n+1}(t)|$$

möglichst klein wird, führt auf die *Tschebyscheff-Polynome*.

⁶Ein Beweis findet sich in P. Deuffhard/ A. Hohmann: Numerische Mathematik I, Satz 7.12 sowie Folgerung 7.13.

(2.5.26) Definition: Die Funktion $T_n : [-1, 1] \rightarrow \mathbb{R}$,

$$T_n(t) = \cos(n \cdot \arccos(t))$$

heißt das **Tschebyscheff-Polynom n -ten Grades**.

Zunächst einmal ist nicht klar, wieso T_n ein Polynom sein soll. Dies und weitere Eigenschaften werden in Satz [1.27] gezeigt. Zunächst wird eine wichtige Rekursionseigenschaft bewiesen.

(2.5.27) Hilfssatz: Es ist

$$T_0(t) = 1 \quad \text{und} \quad T_1(t) = t.$$

Für $n \geq 1$ gilt die **Drei-Term-Rekursion**

$$T_{n+1}(t) = 2tT_n(t) - T_{n-1}(t). \tag{2.12}$$

Beweis: Die Aussagen für $n = 0$ und $n = 1$ folgen aus den wohlbekanntenen Beziehungen

$$\cos(0) = 1, \quad \cos(\arccos(t)) = t.$$

Zur Herleitung der Drei-Term-Rekursion benötigen wir das Additionstheorem für den Kosinus,

$$\cos(a \pm b) = \cos(a) \cos(b) \mp \sin(a) \sin(b).$$

Mit $a = n \cdot \arccos(t)$ und $b = \arccos(t)$ folgt

$$\begin{aligned} \cos((n+1) \arccos(t)) &= \cos(n \cdot \arccos(t)) \cos(\arccos(t)) - \sin(n \cdot \arccos(t)) \sin(\arccos(t)), \\ \cos((n-1) \arccos(t)) &= \cos(n \cdot \arccos(t)) \cos(\arccos(t)) + \sin(n \cdot \arccos(t)) \sin(\arccos(t)). \end{aligned}$$

Aufsummieren der beiden Gleichungen ergibt

$$\cos((n+1) \cdot \arccos(t)) + \cos((n-1) \arccos(t)) = 2t \cos(n \cdot \arccos(t)).$$

Hieraus folgt die Drei-Term-Rekursion. \square

(2.5.28) Satz: Die Tschebyscheff-Polynome erfüllen für $t \in [-1, 1]$ die folgenden Eigenschaften.

(a) T_n ist ein Polynom n -ten Grades und ist damit darstellbar in der Form

$$T_n(t) = a_n^{(n)}t^n + a_{n-1}^{(n)}t^{n-1} + \dots + a_1^{(n)}t + a_0^{(n)}.$$

(b) Für die führenden Koeffizienten gilt

$$a_n^{(n)} = 2^{n-1} \quad \text{für } n \geq 1.$$

(c) T_n ist beschränkt durch 1:

$$\max_{t \in [-1, 1]} |T_n(t)| = 1.$$

(d) T_n hat die n paarweise verschiedenen Nullstellen

$$t_k^{(n)} = \cos\left(\frac{2k-1}{2n} \cdot \pi\right), \quad k = 1, \dots, n.$$

Beweis: Zu (a), (b): Für $n = 0$ und $n = 1$ wurden die Aussagen im Hilfssatz [1.26] gezeigt. Der Beweis für $n > 1$ folgt durch vollständige Induktion aus der Drei-Term-Rekursion.

Zu (c): Die Funktionswerte von T_n sind Werte der Kosinus-Funktion und daher durch 1 beschränkt. An den Stellen $\cos(k\pi/n)$ nimmt $|T_n|$ den Wert 1 an.

Zu (d):

$$T_n(t_k^{(n)}) = \cos\left(n \cdot \frac{(2k-1)\pi}{2n}\right) = \cos\left(\left(k - \frac{1}{2}\right)\pi\right) = 0. \quad \square$$

(2.5.29) Bemerkung: Mit Hilfe des Hilfssatzes [1.26] können die Tschebyscheff-Polynome leicht berechnet werden. Es ist

$$\begin{aligned} T_0(t) &= 1 \\ T_1(t) &= t \\ T_2(t) &= 2t^2 - 1 \\ T_3(t) &= 4t^3 - 3t \\ T_4(t) &= 8t^4 - 8t^2 + 1 \\ &\vdots \end{aligned}$$

Werden die Nullstellen $t_k^{(n)}$ des n -ten Tschebyscheff-Polynoms als Knoten zur Interpolation gewählt, so können die Interpolationsfehler nicht groß werden; aus dem Satz [1.27] folgt nämlich

(2.5.30) Folgerung: $t_k^{(n)}$ seien die Nullstellen von T_n aus Satz [1.27](d). Dann gilt für

$$\omega(t) = \prod_{k=1}^n (t - t_k^{(n)})$$

die Abschätzung

$$|\omega(t)| \leq \frac{1}{2^{n-1}}.$$

Dass Tschebyscheff-Knoten optimal sind, ergibt sich aus dem folgenden Resultat.

(2.5.31) Satz: Jedes Polynom $P \in \mathcal{P}_n$ mit führendem Koeffizienten $a_n \neq 0$ nimmt im Intervall $[-1, 1]$ einen Wert vom Betrag $\geq |a_n|/2^{n-1}$ an. Insbesondere sind die Tschebyscheff-Polynome minimal bezüglich der Maximumnorm $\|f\|_\infty = \max_{t \in [-1, 1]} |f(t)|$ unter den Polynomen in \mathcal{P}_n mit führendem Koeffizienten 2^{n-1} .

Beweis: $P \in \mathcal{P}_n$ sei ein Polynom mit führendem Koeffizienten $a_n = 2^{n-1}$.

Annahme: $|P(t)| < 1$ für alle $t \in [-1, 1]$.

Es ist $T_n - P \in \mathcal{P}_{n-1}$; weiterhin gilt für $\bar{x}_k = \cos(k\pi/n)$

$$\begin{aligned} T_n(\bar{x}_{2k}) &= 1, & P(\bar{x}_{2k}) &< 1, \\ T_n(\bar{x}_{2k+1}) &= -1, & P(\bar{x}_{2k+1}) &> -1. \end{aligned}$$

Hieraus folgt

$$P(\bar{x}_{2k}) - T_n(\bar{x}_{2k}) < 0 \quad \text{und} \quad P(\bar{x}_{2k+1}) - T_n(\bar{x}_{2k+1}) > 0 \quad .$$

Daher ist $T_n - P$ an $n + 1$ Tschebyscheff-Knoten abwechselnd positiv und negativ und hat damit mindestens n Nullstellen in $[-1, 1]$. Dies ist aber im Widerspruch zu

$$0 \neq T_n - P \in \mathcal{P}_{n-1}.$$

Ist nun P ein beliebiges Polynom in \mathcal{P}_n , so lässt sich die Aussage des Satzes durch $\tilde{P} := (2^{n-1}/a_n) \cdot P$ auf den oben ausgeführten Fall zurückführen. \circ

3 Spline-Interpolation

3.1 Polynom-Splines

Die Idee der Spline-Interpolation ist es, das Interpolationsproblem nicht durch ein einziges Polynom vom Grad n zu lösen (da – wie wir gesehen haben – Polynome hohen Grades zu starken Oszillationen neigen können), sondern durch eine möglichst glatte Funktion, welche sich stückweise aus Polynomen niedrigeren Grades zusammensetzt. Die Vorgehensweise soll zunächst an einem Beispiel demonstriert werden.

(3.1.1) Beispiel: Gegeben seien die Punktepaare (t_i, f_i) , $i = 0, \dots, 3$, mit den Werten $(t_0, f_0) = (0, 1)$, $(t_1, f_1) = (2, 2)$, $(t_2, f_2) = (3, 0)$ und $(t_3, f_3) = (5, 0)$. Diese sollen durch eine stetige zusammengesetzte Funktion P_k interpoliert werden, welche in jedem Intervall $[t_i, t_{i+1}]$ ein Polynom $p_i^{(k)}$ k -ten Grades ist.

(a) $k = 1$: Das Polynom $p_i^{(1)}$ 1. Grades durch die Punkte (t_i, f_i) und (t_{i+1}, f_{i+1}) ist gegeben durch die Gleichung

$$\frac{p_i^{(1)}(t) - f_i}{t - t_i} = \frac{f_{i+1} - f_i}{t_{i+1} - t_i}.$$

Hieraus folgt als interpolierende Funktion

$$P_1(t) = \begin{cases} 0.5t + 1 & \text{für } t \in [0, 2] \\ -2t + 6 & \text{für } t \in [2, 3] \\ 0 & \text{für } t \in [3, 5] \end{cases}$$

Diese Funktion ist stetig, in den Interpolationspunkten aber nicht differenzierbar.

(b) $k = 2$: Für das Polynom zweiten Grades in $[t_i, t_{i+1}]$ machen wir den Ansatz

$$p_i^{(2)}(t) = a_i t^2 + b_i t + c_i.$$

Einsetzen der Interpolationsbedingungen

$$p_i^{(2)}(t_i) = f_i, \quad p_i^{(2)}(t_{i+1}) = f_{i+1}$$

führt auf die 6 linearen Gleichungen für die 9 unbekanntenen Koeffizienten a_i , b_i und c_i :

$$c_0 = 1, \quad 4a_0 + 2b_0 + c_0 = 2, \tag{3.1}$$

$$4a_1 + 2b_1 + c_1 = 2, \quad 9a_1 + 3b_1 + c_1 = 0, \tag{3.2}$$

$$9a_2 + 3b_2 + c_2 = 0, \quad 25a_2 + 5b_2 + c_2 = 0. \tag{3.3}$$

Für eine eindeutige Lösung benötigen wir 3 weitere Bedingungen. Hiervon verwenden wir zwei Bedingungen, um zu erreichen, dass die Funktion an den Interpolationspunkten stetig differenzierbar ist. Dies wird erreicht durch die Bedingungen

$$\frac{dp_i^{(2)}}{dt}(t_{i+1}) = \frac{dp_{i+1}^{(2)}}{dt}(t_{i+1}) \quad \text{für } i = 1, 0,$$

also durch

$$4a_0 + b_0 = 4a_1 + b_1, \quad 6a_1 + b_1 = 6a_2 + b_2. \quad (3.4)$$

Das immer noch unterbestimmte Gleichungssystem (4.6) \cdots (4.9) kann nach einem freien Parameter – z.B. a_0 – aufgelöst werden:

$$\begin{aligned} b_0 &= 0.5 - 2a_0, & c_0 &= 1, \\ a_1 &= -2.5 - 2a_0, & b_1 &= 10.5 + 10a_0, & c_1 &= -9 - 12a_0, \\ a_2 &= 2.25 + a_0, & b_2 &= -18 - 8a_0, & c_2 &= 33.75 + 15a_0. \end{aligned}$$

Beispielsweise erhalten wir für $a_0 = 0$ die Lösung

$$P_2(t) = \begin{cases} 0.5t + 1 & \text{für } t \in [0, 2] \\ -2.5t^2 + 10.5t - 9 & \text{für } t \in [2, 3] \\ 2.25t^2 - 18t + 33.75 & \text{für } t \in [3, 5] \end{cases}$$

während die Wahl $a_0 = 0.25$ auf die Lösung

$$P_2(t) = \begin{cases} 0.25t^2 + 1 & \text{für } t \in [0, 2] \\ -13t^2 + 13t - 12 & \text{für } t \in [2, 3] \\ 2.5t^2 - 20t + 37.5 & \text{für } t \in [3, 5] \end{cases}$$

führt.

(3.1.2) Definition: $a = t_0 < t_1 < \cdots < t_n = b$ sei eine aufsteigende Folge von Knoten. Ein **Spline vom Grad k** ist eine $(k - 1)$ -mal stetig differenzierbare Funktion, welche auf jedem der Intervalle $[t_i, t_{i+1}]$ ein Polynom (höchstens) k -ten Grades ist.

3.2 Kubische Splines

In der Praxis spielen kubische Splines P_3 eine wichtige Rolle. Zu ihrer Konstruktion wählen wir für die Intervalle $[t_i, t_{i+1}]$ den Ansatz

$$p_i^{(3)}(t) = \alpha_i(t - t_i)^3 + \beta_i(t - t_i)^2 + \gamma_i(t - t_i) + \delta_i, \quad i = 0, \dots, n - 1.$$

Die Koeffizienten δ_i folgen unmittelbar aus den *Interpolationsbedingungen*:

$$\delta_i = f_i.$$

Beziehungen zwischen den übrigen Koeffizienten erhalten wir aus den *Stetigkeitsbedingungen* für $P_3(t)$, $P_3'(t)$ und $P_3''(t)$ an den Knotenpunkten t_{i+1} :

$$\begin{aligned} I & \quad \alpha_i h_i^3 + \beta_i h_i^2 + \gamma_i h_i + f_i = f_{i+1} & (i = 0, \dots, n-1), \\ II & \quad 3\alpha_i h_i^2 + 2\beta_i h_i + \gamma_i = \gamma_{i+1} & (i = 0, \dots, n-2), \\ III & \quad 6\alpha_i h_i + 2\beta_i = 2\beta_{i+1} & (i = 0, \dots, n-2), \end{aligned} \quad (3.5)$$

wobei wir zur Abkürzung definiert haben:

$$h_i = t_{i+1} - t_i.$$

Mit Hilfe der Gleichungen *I* und *III* lassen sich die Größen α_i und γ_i durch β_i und β_{i+1} ausdrücken gemäß

$$(i) \quad \alpha_i = \frac{\beta_{i+1} - \beta_i}{3h_i}, \quad (ii) \quad \gamma_i = -\frac{h_i}{3}(\beta_{i+1} + 2\beta_i) + \frac{f_{i+1} - f_i}{h_i} \quad (i = 0, \dots, n-2). \quad (3.6)$$

Einsetzen in *II* führt schließlich auf die $n-2$ Gleichungen

$$r_i \beta_i + 2\beta_{i+1} + (1 - r_i)\beta_{i+2} = q_i \quad (i = 0, \dots, n-3) \quad (3.7)$$

für die n Unbekannten β_i , wobei zur Abkürzung

$$r_i = \frac{h_i}{h_{i+1} + h_i}, \quad q_i = \frac{3}{h_i + h_{i+1}} \cdot \left(\frac{f_{i+2} - f_{i+1}}{h_{i+1}} - \frac{f_{i+1} - f_i}{h_i} \right)$$

verwendet wurde. Beachten Sie, dass auch γ_{n-1} mit Hilfe der Gleichung *II* ($i = n-2$) und der Beziehungen (4.11) durch die Koeffizienten β_i beschrieben werden kann:

$$\gamma_{n-1} = \frac{h_{n-2}}{3}(2\beta_{n-1} + \beta_{n-2}) + \frac{f_{n-1} - f_{n-2}}{h_{n-2}}. \quad (3.8)$$

Durch Einsetzen in *I* ($i = n-1$) folgt außerdem

$$\alpha_{n-1} h_{n-1}^2 = -\beta_{n-1} \left(h_{n-1} + \frac{2}{3} h_{n-2} \right) - \frac{1}{3} \beta_{n-2} h_{n-2} + \frac{h_{n-1} + h_{n-2}}{3} q_{n-2}. \quad (3.9)$$

Damit können alle Koeffizienten berechnet werden, sobald die β_i bestimmt sind.

Für ein vollständiges Gleichungssystem fehlen zwei weitere Bedingungen, welche häufig in Form zusätzlicher Randbedingungen ergänzt werden.

(ii) Die Sinusfunktion lässt sich “antiperiodisch” über das Intervall $[0, \pi]$ hinaus fortsetzen. (D.h. $\sin(\pi + t) = -\sin(t)$.) Entwickeln Sie geeignete “antiperiodische” Randbedingungen, stellen Sie das zugehörige Gleichungssystem auf und lösen Sie dieses. Vergleichen Sie das Ergebnis mit denen aus (i).

3.3 B-Splines

Im Folgenden seien die Knoten $a = t_0 < \dots < t_{\ell+1} = b$ fest; wir definieren das Gitter $\Delta := \{t_0, \dots, t_{\ell+1}\}$. Mit $\mathcal{S}_{k,\Delta}$ bezeichnen wir den linearen Vektorraum aller Splines vom Grad $k - 1$ bezüglich des Gitters Δ . Die Dimension dieses Raums kann man durch Abzählen der Freiheitsgrade leicht ermitteln.

(3.3.6) Bemerkung: Es ist $\dim(\mathcal{S}_{k,\Delta}) = k + \ell$.

Beweis: Übung.

Die Konstruktion einer geeigneten Basis von $\mathcal{S}_{k,\Delta}$ führt auf den Begriff der B-Splines.

(3.3.7) Definition: Sei $\tau_1 \leq \dots \leq \tau_n$ eine beliebige Folge von Knoten. Dann sind die B-Splines $N_{ik}(t)$ der Ordnung k für $k = 1, \dots, n$ und $i = 1, \dots, n - k$ rekursiv definiert durch

$$N_{i1}(t) := \chi_{[\tau_i, \tau_{i+1})}(t) = \begin{cases} 1 & \text{falls } \tau_i \leq t < \tau_{i+1} \\ 0 & \text{sonst} \end{cases} \quad (3.17)$$

$$N_{ik}(t) = \frac{t - \tau_i}{\tau_{i+k-1} - \tau_i} \cdot N_{i,k-1}(t) + \frac{\tau_{i+k} - t}{\tau_{i+k} - \tau_{i+1}} \cdot N_{i+1,k-1}(t) \quad . \quad (3.18)$$

Hierbei wird die Konvention $0/0 = 0$ verwendet.

Wir wollen zunächst an einem Beispiel zeigen, wie sich hierdurch eine Basis von $\mathcal{S}_{k,\Delta}$ konstruiert werden kann.

(3.3.8) Beispiel: Für $k = 1, 2, 3, 4$ und das Gitter $\Delta = \{0, 1, 2, 3\}$ sollen die B-Splines bestimmt werden. Aus Gründen, welche später klar werden, wählen wir die erweiterte Knotenfolge $\tau_1 = \tau_2 = \tau_3 = \tau_4 = 0$, $\tau_5 = 1$, $\tau_6 = 2$, $\tau_7 = \tau_8 = \tau_9 = \tau_{10} = 3$.

$k = 1$: Nach (2.17) ist

$$N_{11} \equiv N_{21} \equiv N_{31} \equiv N_{71} \equiv N_{81} \equiv N_{91} \equiv 0$$

und nach (2.18)

$$N_{41}(t) = \begin{cases} 1, & t \in [0, 1) \\ 0 & \text{sonst} \end{cases}, \quad N_{51}(t) = \begin{cases} 1, & t \in [1, 2) \\ 0 & \text{sonst} \end{cases}, \quad N_{61}(t) = \begin{cases} 1, & t \in [2, 3) \\ 0 & \text{sonst} \end{cases}.$$

$k = 2$: Es ist

$$N_{i2}(t) = \frac{t - \tau_i}{\tau_{i+1} - \tau_i} \cdot N_{i1}(t) + \frac{\tau_{i+2} - t}{\tau_{i+2} - \tau_{i+1}} \cdot N_{i+1,1}(t) \quad .$$

Bestätigen Sie: Ist $\tau_{i+1} = \tau_i$, so ist auch $N_{i1} \equiv 0$; nach der Konvention der Definition [2.8] verschwinden die zugehörigen Summanden. Es folgt

$$N_{12} \equiv N_{22} \equiv N_{72} \equiv N_{82} \equiv 0 \quad .$$

Für die restlichen Funktionen ergibt sich

$$N_{32}(t) = \begin{cases} 1 - t, & t \in [0, 1) \\ 0 & \text{sonst} \end{cases}, \quad N_{42}(t) = \begin{cases} t, & t \in [0, 1) \\ 2 - t, & t \in [1, 2) \\ 0 & \text{sonst} \end{cases},$$

$$N_{52}(t) = \begin{cases} t - 1, & t \in [1, 2) \\ 3 - t, & t \in [2, 3) \\ 0 & \text{sonst} \end{cases}, \quad N_{62}(t) = \begin{cases} t - 2, & t \in [2, 3) \\ 0 & \text{sonst} \end{cases}.$$

$k = 2$: Hier ergibt sich $N_{13} \equiv N_{73} \equiv 0$ und

$$N_{23}(t) = \frac{\tau_5 - t}{\tau_5 - \tau_3} \cdot N_{32}(t) = \begin{cases} (1 - t)^2, & t \in [0, 1) \\ 0 & \text{sonst} \end{cases}$$

sowie

$$N_{33}(t) = \begin{cases} -\frac{3}{2} \left(t - \frac{2}{3}\right)^2 + \frac{2}{3}, & t \in [0, 1) \\ \frac{(2-t)^2}{2}, & t \in [1, 2) \\ 0 & \text{sonst} \end{cases}, \quad N_{43}(t) = \begin{cases} \frac{t^2}{2}, & t \in [0, 1) \\ -\left(t - \frac{3}{2}\right)^2 + \frac{3}{4}, & t \in [1, 2) \\ \frac{(3-t)^2}{2}, & t \in [2, 3) \\ 0 & \text{sonst} \end{cases},$$

$$N_{53}(t) = \begin{cases} \frac{(t-1)^2}{2}, & t \in [1, 2) \\ -\frac{3}{2} \left(t - \frac{7}{3}\right)^2 + \frac{2}{3}, & t \in [2, 3) \end{cases}, \quad N_{63}(t) = \begin{cases} (t - 2)^2, & t \in [2, 3) \\ 0 & \text{sonst} \end{cases}.$$

$k = 4$: Die Elemente N_{i4} sind "nichttrivial" für $i = 1, \dots, 6$ und ergeben sich zu

$$N_{14}(t) = \begin{cases} (1 - t)^3, & t \in [0, 1) \\ 0 & \text{sonst} \end{cases}, \quad N_{24}(t) = \begin{cases} \frac{7}{4}t^3 - \frac{9}{2}t^2 + 3t, & t \in [0, 1) \\ \frac{(2-t)^3}{4}, & t \in [1, 2) \\ 0 & \text{sonst} \end{cases},$$

$$N_{34}(t) = \begin{cases} -\frac{11}{12}t^3 + \frac{3}{2}t^2, & t \in [0, 1) \\ \frac{7}{12}t^3 - 3t^2 + \frac{9}{2}t - \frac{3}{2}, & t \in [1, 2) \\ \frac{(3-t)^3}{6}, & t \in [2, 3) \\ 0 & \text{sonst} \end{cases}, \quad N_{44}(t) = \begin{cases} \frac{t^3}{6}, & t \in [0, 1) \\ -\frac{7}{12}t^3 + \frac{9}{4}t^2 - \frac{9}{4}t + \frac{3}{4}, & t \in [1, 2) \\ \frac{11}{12}t^3 - \frac{27}{4}t^2 + \frac{63}{4}t - \frac{45}{4}, & t \in [2, 3) \\ 0 & \text{sonst} \end{cases},$$

$$N_{54}(t) = \begin{cases} \frac{(t-1)^3}{4}, & t \in [1, 2) \\ -\frac{7}{4}t^3 + \frac{45}{4}t^2 - \frac{93}{4}t + \frac{63}{4}, & t \in [2, 3) \\ 0 & \text{sonst} \end{cases}, \quad N_{64}(t) = \begin{cases} (t-2)^3, & t \in [2, 3) \\ 0 & \text{sonst} \end{cases}.$$

Durch Induktion weist man leicht die folgenden Eigenschaften nach.

(3.3.9) Lemma: Für die B-Splines N_{ik} gilt

- (a) $\text{supp } N_{ik} \subseteq [\tau_i, \tau_{i+k}]$ (lokaler Träger)
- (b) $N_{ik}(t) \geq 0 \quad \forall t \in \mathbb{R}$ (Nichtnegativität)
- (c) N_{ik} ist stückweise ein Polynom vom Grad $\leq k-1$.

Wir wollen nun eine geeignete Basis von $\mathcal{S}_{k,\Delta}$ zum Gitter $\Delta = \{t_0, \dots, t_{\ell+1}\}$ konstruieren, wobei $a = t_0 < \dots < t_{\ell+1} = b$. Hierbei definieren wir zunächst $n = \ell + k$ sowie die erweiterte Knotenfolge $\tau_1 \leq \dots \leq \tau_{n+k}$ durch

$$\begin{aligned} \tau_1 = \dots = \tau_k &= t_0, \\ \tau_{k+r} &= t_r, \quad r = 1, \dots, \ell, \\ \tau_{n+1} = \dots = \tau_{n+k} &= t_{\ell+1}. \end{aligned}$$

Ohne Beweis stellen wir fest⁷

(3.3.10) Lemma: Für die bezüglich der erweiterten Knotenfolge definierten B-Splines gilt $N_{ik} \in \mathcal{S}_{k,\Delta}$, $i = 1, \dots, n$.

Wir haben die lineare Unabhängigkeit der N_{ik} , $i = 1, \dots, n$, zu zeigen. Nützlich ist die folgende *Marsden-Identität*.

(3.3.11) Lemma: Für beliebige $t \in [a, b]$ und $s \in \mathbb{R}$ ist

$$(t-s)^{k-1} = \sum_{i=1}^n \phi_{ik}(s) N_{ik}(t) \quad \text{mit} \quad \phi_{ik}(s) = \prod_{j=1}^{k-1} (\tau_{i+j} - s).$$

⁷vgl. Folgerung 7.52 in Deuffhard/Hohmann, Numerische Mathematik I

Beweis durch Induktion: Für $k = 1$ gilt die Aussage wegen $\sum_{i=1}^n N_{ik}(t) = 1$.

Die Aussage gelte für alle $\ell < k - 1$. Dann gilt nach der Rekursionsformel für N_{ik}

$$\begin{aligned} \sum_{i=1}^n \phi_{ik}(s) N_{ik}(t) &= \sum_{i=2}^n \left(\frac{t - \tau_i}{\tau_{i+k-1} - \tau_i} \phi_{ik}(s) + \frac{\tau_{i+k-1} - t}{\tau_{i+k-1} - \tau_i} \phi_{i-1,k}(s) \right) N_{ik}(t) \\ &= \sum_{i=2}^n \prod_{j=1}^{k-2} (\tau_{i+j} - s) \cdot \underbrace{\left(\frac{t - \tau_i}{\tau_{i+k-1} - \tau_i} (\tau_{i+k-1} - s) + \frac{\tau_{i+k-1} - t}{\tau_{i+k-1} - \tau_i} (\tau_i - s) \right)}_{(*)} N_{i,k-1}(t) \\ &= (t - s) \sum_{i=2}^n \phi_{i,k-1}(s) N_{i,k-1}(t) = (t - s)(t - s)^{k-2} = (t - s)^{k-1}. \end{aligned}$$

Hierbei ist – wie man leicht nachprüft – der Ausdruck $(*)$ das lineare Interpolationspolynom der Abbildung $t \rightarrow t - s$ zu den Knoten τ_i und τ_{i+k-1} , also gleich $t - s$. \circ

Hieraus ergibt sich

(3.3.12) Folgerung: Es sei $\mathcal{P}_{k-1}[a, b]$ der Raum der Polynome auf dem Intervall $[a, b]$ vom Grad $\leq k - 1$. Dann ist

$$\mathcal{P}_{k-1}[a, b] \subset \text{span}(N_{1k}, \dots, N_{nk}).$$

Außerdem bilden die N_{ik} eine *Zerlegung der Eins* in dem Sinne, dass

$$\sum_{i=1}^n N_{ik}(t) = 1 \quad \text{für alle } t \in [a, b].$$

Beweis: Definiere $f(s) := (t - s)^{k-1}$. Dann ist für $\ell = 1, \dots, k - 1$

$$f^{(\ell)}(0) = (k - 1) \cdots (k - \ell) (-1)^\ell t^{k-1-\ell} = \frac{(k - 1)!}{(k - \ell - 1)!} (-1)^\ell t^{k-1-\ell}.$$

Aus der Marsden-Identität folgt

$$t^{k-1-\ell} = \frac{(-1)^\ell (k - \ell - 1)!}{(k - 1)!} \sum_{i=1}^n \phi_{ik}^{(\ell)}(0) N_{ik}(t).$$

Damit sind alle t^m , $m = k - \ell - 1$, Linearkombinationen der N_{ik} , und daher $\mathcal{P}_{k-1}[a, b] \subset \text{span}(N_{1k}, \dots, N_{nk})$.

Der Beweis der Zerlegung der Eins folgt aus $\ell = k-1$ mit $\phi^\ell(0) = (-1)^\ell$. Begründung? \circ

Als Beitrag zum Beweis der linearen Unabhängigkeit der N_{ik} beweisen wir zunächst die *lokale lineare Unabhängigkeit*.

(3.3.13) Lemma: Es sei $(c, d) \subset [a, b]$. Ist $\sum_{i=1}^n c_i N_{ik}(t) = 0$ für alle $t \in (c, d)$, so ist $c_j = 0$ für alle j mit $(c, d) \cap (\tau_j, \tau_{j+k}) \neq \emptyset$.

Beweis: Es sei $(c, d) \cap (\tau_j, \tau_{j+1}) \neq \emptyset$. Es gibt genau die k B-Splines

$$N_{j,j+k}, N_{j,j+k+1}, \dots, N_{j,j+2k-1}, \quad (3.19)$$

welche auf (τ_j, τ_{j+1}) *nicht* identisch verschwinden. Andererseits lassen sich nach Folgerung [2.11] die k linear unabhängigen Polynome $1, t, \dots, t^{k-1}$ auf $[c, d]$ als Linearkombinationen der N_{ik} darstellen. Damit sind die k Funktionen aus (2.19) als Funktionen auf $[c, d]$ linear unabhängig. \circ

Als zentrales Ergebnis folgt hieraus

(3.3.14) Satz: Die N_{ik} , $i = 1, \dots, \ell + 1$, bilden eine Basis von $\mathcal{S}_{k,\Delta}$.

Damit lässt sich jede Spline-Funktion durch eine Linearkombination der Form

$$S(t) = \sum_{i=1}^{\ell+1} d_i N_{ik}(t) \quad (3.20)$$

beschreiben. Da die N_{ik} eine Zerlegung der Eins bilden, lassen sich diese Funktionen abschätzen durch

$$\|S(t)\|_\infty \leq \max_i |d_i|.$$

Es zeigt sich damit, dass das Interpolationsproblem bezüglich dieser Basis gut konditioniert ist. Werden die Koeffizienten d_i durch 2- oder 3-dimensionale Vektoren ersetzt, so lassen sich auch ebene oder Raumkurven erzeugen. Die d_i heißen auch die **de Boor-Punkte** von S .

(3.3.15) Beispiele: (a) Das Gitter Δ und die N_{ik} seien gegeben wie in Beispiel [2.8]. Gesucht ist der interpolierende kubische Spline zu natürlichen Randbedingungen. Die

	$N_{14}(t_i)$	$N_{24}(t_i)$	$N_{34}(t_i)$	$N_{44}(t_i)$	$N_{54}(t_i)$	$N_{64}(t_i)$
$t_i = 0$	1	0	0	0	0	0
$t_i = 1$	0	1/4	7/12	1/6	0	0
$t_i = 2$	0	0	1/6	7/12	1/4	0
$t_i = 3$	0	0	0	0	0	1

Tabelle 3: Knotenwerte $N_{j4}(t_i)$ zu Beispiel [2.15](a)

	$N''_{14}(t_i)$	$N''_{24}(t_i)$	$N''_{34}(t_i)$	$N''_{44}(t_i)$	$N''_{54}(t_i)$	$N''_{64}(t_i)$
$t_i = 0$	6	-9	3	0	0	0
$t_i = 3$	0	0	0	3	-9	6

Tabelle 4: Zweite Ableitungen $N''_{j4}(t_i)$ zu Beispiel [2.15](a)

Funktionswerte $N_{j4}(t_i)$ sind in Tabelle 2, die zweiten Ableitungen in den Endpunkten in Tabelle 3 angegeben. Als Gleichungssystem für die de Boor-Punkte $\mathbf{d} = (d_1, \dots, 6)^T$ folgt

$$\begin{pmatrix} 1 & 0 & 0 & 0 & 0 & 0 \\ 6 & -9 & 3 & 0 & 0 & 0 \\ 0 & 1/4 & 7/12 & 1/6 & 0 & 0 \\ 0 & 0 & 1/6 & 7/12 & 1/4 & 0 \\ 0 & 0 & 0 & 3 & -9 & 6 \\ 0 & 0 & 0 & 0 & 0 & 1 \end{pmatrix} \cdot \mathbf{d} = \begin{pmatrix} f_0 \\ 0 \\ f_1 \\ f_2 \\ 0 \\ f_3 \end{pmatrix}$$

(3.3.16) Übung: Benutzen Sie die Rekursionsformel (2.18) zur Konstruktion eines Schemas, mit Hilfe dessen einzelne Funktionswerte $S(t)$ aus der Darstellung (2.20) berechnet werden können. Zeigen Sie hierzu zunächst, dass

$$S(t) = \sum_{i=r+1}^n d_i^r(t) N_{i,k-1}(t),$$

wobei die d_i^r rekursiv definiert sind durch $d_i^0(t) := d_i$ und

$$d_k^r(t) := \begin{cases} \frac{t-\tau_i}{\tau_{i+k-r}-\tau_i} d_i^{r-1}(t) + \frac{\tau_{i+k-r}-t}{\tau_{i+k-r}-\tau_i} d_{i-1}^{r-1}(t) & \text{falls } \tau_{i+k-r} \neq \tau_i \\ 0 & \text{sonst} \end{cases}$$

für $r > 0$. Zeigen Sie dann, dass $S(t) = d_i^{k-1}(t)$ für $t \in [\tau_i, \tau_{i+1})$.

4 Approximation durch Orthogonalsysteme

Im folgenden soll eine Funktion $f : [a, b] \rightarrow \mathbb{R}$ nicht an vorgegebenen Knoten interpoliert werden, sondern durch ein Polynom approximiert werden, welche den *ganzen* Verlauf von f im Intervall $[a, b]$ gut beschreibt. Ein brauchbares Hilfsmittel hierfür liefern Orthogonalpolynome als Basis des Polynomraums. Wir benötigen einige Vorbereitungen.

Mit $\mathcal{C}([a, b], \mathbb{R})$ bezeichnen wir die Menge der stetigen beschränkten reellwertigen Funktionen auf $[a, b]$. Auf $\mathcal{C}([a, b], \mathbb{R})$ sei ein **Skalarprodukt**⁸ $\langle \cdot, \cdot \rangle : \mathcal{C}([a, b], \mathbb{R}) \times \mathcal{C}([a, b], \mathbb{R}) \rightarrow \mathbb{R}$ definiert durch

$$\langle f, g \rangle = \int_a^b \omega(t) f(t) g(t) dt,$$

wobei die *Gewichtsfunktion* $\omega(t)$ positiv und integrierbar sei. Die durch

$$\|f\| := \sqrt{\langle f, f \rangle}$$

definierte Norm auf $\mathcal{C}([a, b], \mathbb{R})$ heißt die **durch $\langle \cdot, \cdot \rangle$ induzierte Norm**.

(4.0.1) Definition: Eine Menge $\{P_n | n = 0, 1, 2, \dots\}$ von Polynomen (wobei $0 \neq P_n$ ein Polynom n -ten Grades sei) heißt **Orthogonalsystem** bzgl. des Skalarprodukts $\langle \cdot, \cdot \rangle$, wenn gilt

$$\langle P_n, P_m \rangle = 0 \quad \text{falls} \quad m \neq n.$$

Es bezeichne

$$\pi_n := \|P_n\|$$

die Norm des n -ten Orthogonalpolynoms.

(4.0.2) Beispiele: (a) Wir wollen ein Orthogonalsystem auf dem Intervall $[0, 1]$ zur Gewichtsfunktion $\omega(t) \equiv 1$ konstruieren und wählen hierfür den Ansatz

$$P_n(t) = t^n + a_{n-1}^{(n)} t^{n-1} + \dots + a_0^{(n)}.$$

⁸Ein Skalarprodukt auf einem Vektorraum V ist eine Abbildung $\langle \cdot, \cdot \rangle : V \times V \rightarrow \mathbb{R}$ mit den Eigenschaften (i) $\langle f, f \rangle > 0$ falls $f \neq 0$, (ii) $\langle f, \alpha g + \beta h \rangle = \alpha \langle f, g \rangle + \beta \langle f, h \rangle$, (iii) $\langle f, g \rangle = \langle g, f \rangle$.

Intervall $[a, b]$	Gewichtsfunktion $\omega(t)$	Orthogonalsystem
$[-1, 1]$	1	Legendre-Polynome P_n
$[-1, 1]$	$\frac{1}{\sqrt{1-t^2}}$	Tschebyscheff-Polynome T_n
$[0, \infty]$	$\exp(t)$	Laguerre-Polynome L_n
$[-\infty, \infty]$	$\exp(-t^2)$	Hermite-Polynome H_n

Tabelle 5: Häufig benutzte Orthogonalpolynome

Dann ist $P_0(t) \equiv 1$; wurden $P_i(t)$, $i = 0, \dots, n-1$ bereits bestimmt, so folgen die Koeffizienten für P_n aus den n Gleichungen

$$\langle P_n, P_i \rangle = 0, \quad i = 0, \dots, n-1.$$

Aus elementaren Rechnungen folgt

$$\begin{aligned} P_0(t) &= 1, & \pi_0 &= 1, \\ P_1(t) &= t - \frac{1}{2}, & \pi_1 &= \frac{1}{2\sqrt{3}}, \\ P_2(t) &= t^2 - t + \frac{1}{6}, & \pi_2 &= \frac{1}{6\sqrt{5}} \\ & \vdots \end{aligned}$$

(b) Einige Orthogonalsysteme zu häufig verwendeten Grundintervallen $[a, b]$ und Gewichtsfunktionen $\omega(t)$ sind in Tabelle 3 angegeben.

Bis auf eine Normierungsbedingung sind Orthogonalpolynome eindeutig bestimmt. Neben der in Beispiel [3.2](a) angedeuteten Möglichkeit zur Berechnung von Orthogonalsystemen gibt es auch ein rekursives Verfahren. Die Ergebnisse sind im folgenden Satz zusammengefasst.

(4.0.3) Satz: Zu jedem gewichteten Skalarprodukt gibt es eindeutig bestimmte Orthogonalpolynome $P_k \in \mathcal{P}_k$ ($k = 0, 1, 2, \dots$) mit führendem Koeffizienten 1. Sie genügen der *Drei-Term-Rekursion*

$$\begin{aligned} P_0 &\equiv 1, & P_1(t) &= (t + a_1)P_0(t), \\ P_k(t) &= (t + a_k)P_{k-1}(t) + b_k P_{k-2}(t), & k &= 1, 2, \dots, \end{aligned}$$

wobei

$$a_k = -\frac{\langle tP_{k-1}, P_{k-1} \rangle}{\langle P_{k-1}, P_{k-1} \rangle}, \quad b_k = -\frac{\langle P_{k-1}, P_{k-1} \rangle}{\langle P_{k-2}, P_{k-2} \rangle}.$$

Beweis: Die Gültigkeit der Formeln für $k = 0, 1$ ist leicht einzusehen. Zu $k > 1$ seien die Orthogonalpolynome P_j , $j = 0, \dots, k-1$ bereits kontruiert. Gesucht ist das Orthogonalpolynom $P_k \in \mathcal{P}_k$. Da die Leitkoeffizienten von P_{k-1} und P_k gleich 1 sind, ist $P_k - t \cdot P_{k-1} \in \mathcal{P}_{k-1}$. Da die P_0, \dots, P_{k-1} eine Basis von \mathcal{P}_{k-1} bilden, ist

$$P_k - t \cdot P_{k-1} = \sum_{i=0}^{k-1} c_i P_i \quad ,$$

und wegen der Orthogonalität der P_i ist

$$c_i = \frac{\langle P_k - t \cdot P_{k-1}, P_i \rangle}{\langle P_i, P_i \rangle} \quad .$$

Aus der Forderung $\langle P_k, P_i \rangle = 0$ ($i = 0, \dots, k-1$) folgt

$$c_i = -\frac{\langle t \cdot P_{k-1}, P_i \rangle}{\langle P_i, P_i \rangle} = -\frac{\langle P_{k-1}, t \cdot P_i \rangle}{\langle P_i, P_i \rangle} \quad .$$

$i = 0, \dots, k-3$: Es ist $t \cdot P_i \in \mathcal{P}_{k-2}$ und $P_{k-1} \perp \mathcal{P}_{k-2}$. Damit ist $c_i = 0$.

$i = k-2$: Wegen $\mathcal{P}_{k-2} \ni t \cdot P_{k-2} - P_{k-1} = \sum_{j=0}^{k-2} \tilde{c}_j P_j$ ist

$$c_{k-2} = -\frac{\langle P_{k-1}, t \cdot P_{k-2} \rangle}{\langle P_{k-2}, P_{k-2} \rangle} = -\frac{\langle P_{k-1}, P_{k-1} - \sum_{j=0}^{k-2} \tilde{c}_j P_j \rangle}{\langle P_{k-2}, P_{k-2} \rangle} = -\frac{\langle P_{k-1}, P_{k-1} \rangle}{\langle P_{k-2}, P_{k-2} \rangle} \quad .$$

$i = k-1$:

$$c_{k-1} = -\frac{\langle P_{k-1}, t \cdot P_{k-1} \rangle}{\langle P_{k-1}, P_{k-1} \rangle} \quad . \quad \circ$$

(4.0.4) Übung: (a) Bestimmen Sie die ersten Orthogonalpolynome P_i zum Intervall $[0,1]$ und zur Gewichtsfunktion $\omega(t) = t \cdot (1-t)$.

(b) Für das Intervall $[-1,1]$ und die Gewichtsfunktion $\omega(t) = 1$ ist die Orthogonalbasis gegeben durch die Legendre-Polynome. (Vgl. Tabelle 4.)

(i) Zeigen Sie:

$$P_m(t) = \frac{1}{2^m \cdot m!} \cdot \frac{d^m}{dx^m} [(t^2 - 1)^m], \quad m = 0, 1, 2, \dots$$

(ii) Zeigen Sie: $\langle P_m, P_m \rangle = 2/(2m+1)$.

(iii) Leiten Sie eine Rekursionsformel her zur punktweisen Berechnung von

$$h(t) = \sum_{k=0}^n c_k P_k(t).$$

Die ersten $n + 1$ Orthogonalpolynome P_0, \dots, P_n bilden eine Basis von \mathcal{P}_n . Ist daher $P \in \mathcal{P}_n$ ein beliebiges Polynom, so existiert eine Darstellung der Form

$$P = \sum_{i=0}^n \alpha_i P_i.$$

Die Koeffizienten α_j können leicht bestimmt werden. Aus der Orthogonalitätseigenschaft der P_i folgt nämlich

$$\langle P, P_j \rangle = \alpha_j \pi_j^2.$$

Damit besitzt P die Darstellung

$$P = \sum_{i=0}^n \frac{\langle P, P_i \rangle}{\pi_i^2} \cdot P_i.$$

Diese Darstellung lässt sich leicht auf andere Funktionen übertragen.

(4.0.5) Definition: Für $f \in \mathcal{C}([a, b], \mathbb{R})$ heißt das Polynom

$$\sum_{i=0}^n \frac{\langle f, P_i \rangle}{\pi_i^2} \cdot P_i \tag{4.1}$$

Entwicklung von f in die Orthogonalpolynome P_0, \dots, P_n .

(4.0.6) Beispiel: Die Funktion $f : [-1, 1] \rightarrow \mathbb{R}$,

$$f(t) = 1 - |t|$$

soll bezüglich der Gewichtsfunktion $\omega \equiv 1$ in die *Legendre-Polynome* P_0, \dots, P_4 entwickelt werden (vgl. Tabelle 4). Die Legendre-Polynome können aus geeigneten Tafelwerken abgelesen werden⁹:

$$\begin{aligned} P_0(t) &= 1, \\ P_1(t) &= t, \\ P_2(t) &= \frac{1}{2}(3t^2 - 1), \\ P_3(t) &= \frac{1}{2}(5t^3 - 3t), \\ P_4(t) &= \frac{1}{8}(35t^4 - 30t^2 + 3). \end{aligned}$$

⁹vgl. z.B. I. N. Bronstein et al.: Taschenbuch der Mathematik, Harri Deutsch, 1999

Ihre Norm ist

$$\|P_n\| = (n + 0.5)^{-1/2} \quad (\text{vgl. Übung[3.4](b)(ii)}).$$

Durch Auswerten der Skalarprodukte $\langle f, P_i \rangle$ erhält man die folgende Entwicklung für f :

$$\sum_{i=0}^4 \frac{\langle f, P_i \rangle}{\pi_i^2} \cdot P_i = \frac{1}{2} \cdot P_0 - \frac{5}{8} \cdot P_2 + \frac{3}{16} \cdot P_4 = \frac{113}{128} - \frac{105}{64}t^2 + \frac{105}{128}t^4.$$

Im folgenden Sinne optimieren diese Entwicklungen die Approximation stetiger Funktionen durch Polynome.

(4.0.7) Satz: Es sei $f \in \mathcal{C}([a, b], \mathbb{R})$. Unter allen Polynomen P n -ten Grades minimiert die Entwicklung (3.1) in die Orthogonalpolynome P_0, \dots, P_n den Abstand $\|f - P\|$. Es ist

$$\left\| f - \sum_{i=0}^n \frac{\langle f, P_i \rangle}{\pi_i^2} \cdot P_i \right\|^2 = \sum_{i=n+1}^{\infty} \left(\frac{\langle f, P_i \rangle}{\pi_i} \right)^2.$$

5 Numerische Integration

In diesem Kapitel seien a und b fest gewählte reelle Koeffizienten ($a < b$). Das Ziel ist die Herleitung numerischer Verfahren für das Integral

$$I(f) := \int_a^b f(t) dt$$

für hinreichend glatte Funktionen $f : [a, b] \rightarrow \mathbb{R}$.

5.1 Quadraturformeln

Gesucht sind Näherungsformeln für $I(f)$ der Form

$$\hat{I}(f) := (b - a) \cdot \sum_{i=0}^n \lambda_i f(t_i) \quad (5.1)$$

mit geeigneten Knoten

$$a \leq t_0 < t_1 < \dots < t_n \leq b$$

und Gewichten λ_i . Beginnen wir mit einer ‘Denksportaufgabe’, und versuchen wir die Gewichte λ_i so zu bestimmen, dass wenigstens die Polynome aus \mathcal{P}_n exakt integriert werden. Dann sind die Gewichte eindeutig bestimmt.

(5.1.1) Lemma: Zu $n + 1$ paarweise verschiedenen Knoten t_i gibt es eine eindeutige Folge $\lambda_0, \dots, \lambda_n$, für die alle Polynome $P \in \mathcal{P}_n$ durch die Formel (4.1) exakt integriert werden.

Beweis: Formel (4.1) ist genau dann für alle $P \in \mathcal{P}_n$ exakt, wenn sie exakt ist für alle Lagrange-Polynome L_{in} . Dies ist äquivalent zu

$$I(L_{in}) = \hat{I}(L_{in}) = (b - a) \cdot \sum_{j=0}^n \lambda_j L_{in}(t_j) = (b - a) \sum_{j=0}^n \lambda_j \delta_{ij} = (b - a) \lambda_i. \quad \circlearrowright$$

Wir schwächen die Forderung der exakten Integration von Polynomen ab und wählen praxisnähere Mindestanforderungen.

(5.1.2) Mindestanforderungen: (a) Konstante Funktionen sollen exakt integriert werden. Das heißt: Für $f(\cdot) \equiv c$ ist $I(f) = \hat{I}(f)$.

(b) Ist $f(t) \geq 0$ für alle $t \in [a, b]$, so ist auch $\hat{I}(f) \geq 0$.

Notwendige und hinreichende Bedingungen, welche sich aus den Mindestanforderungen ergeben, lassen sich leicht herleiten. Ist $f(t) = c \neq 0$ für alle $t \in [a, b]$, so ist $I(f) = c \cdot (b - a)$ und

$$\hat{I}(f) = (b - a) \cdot c \cdot \sum_{i=0}^n \lambda_i.$$

Die Forderung (a) ist daher äquivalent zur Forderung $\sum_{i=0}^n \lambda_i = 1$.

Die Forderung (b) führt zur Bedingung $\lambda_i \geq 0$, $i = 0, \dots, n$. Um dies zu sehen, nehmen wir an, dass für ein $j \in \{0, \dots, n\}$ gilt $\lambda_j < 0$. Wir konstruieren die lineare Spline-Funktion f , welche eindeutig bestimmt ist durch

$$f(t_i) = \begin{cases} 0 & \text{falls } i \neq j \\ 1 & \text{für } i = j. \end{cases}$$

Diese Funktion ist sicherlich nichtnegativ. Für die Integralnäherung gilt

$$\hat{I}(f) = (b - a)\lambda_j < 0.$$

Damit ist die Forderung (b) nicht erfüllt.

Im folgenden betrachten wir nur Näherungsformeln, für welche die Mindestanforderungen erfüllt sind und welche durch die folgende Definition charakterisiert sind.

(5.1.3) Definition: Eine Formel der Form (5.1) heißt **Quadraturformel für f** , wenn gilt

$$\begin{aligned} \text{(a)} \quad & \sum_{i=0}^n \lambda_i = 1, \\ \text{(b)} \quad & \lambda_i \geq 0, \quad i = 0, \dots, n. \end{aligned}$$

Quadraturformeln können leicht auf heuristischem Weg hergeleitet werden, wie die folgenden Beispiele zeigen.

(5.1.4) Beispiele: (a) Mittelpunktsregel. Durch die Wahl der Knoten

$$t_i := a + i \cdot \frac{b - a}{N}, \quad i = 0, \dots, N,$$

zerlegen wir das Intervall $[a, b]$ in N gleich große Teilintervalle der Länge $h = (b - a)/N$; das Integral über dem i -ten Teilintervall approximieren wir durch die Fläche des Rechtecks mit Höhe $f(\tau_i)$, wobei

$$\tau_i = a + (i - 0.5)h, \quad i = 1, \dots, N$$

der Mittelpunkt des Teilintervalls ist. Dies führt auf die Näherungsformel

$$\hat{I}(f) = h \cdot \sum_{i=1}^N f(\tau_i) = (b - a) \cdot \sum_{i=1}^N \frac{1}{N} f(\tau_i).$$

Dies ist eine Quadraturformel mit den Gewichten $\lambda_i = 1/N$ ($i = 1, \dots, N$).

(b) Trapezregel. Das Integral $[a, b]$ wird zerlegt wie in (a). Über jedem Teilintervall $[t_i, t_{i+1}]$ wird die Funktion f approximiert durch das lineare Interpolationspolynom zu den Knoten t_i und t_{i+1} . Die Fläche über dem Teilintervall wird approximiert durch das hierdurch entstehende Trapez mit der Fläche $0.5 \cdot h \cdot (f(t_i) + f(t_{i+1}))$. Damit erhalten wir als Approximation des Integrals $I(f)$

$$\begin{aligned} \hat{I}(f) &= 0.5 \cdot h \cdot \sum_{i=0}^N (f(t_i) + f(t_{i+1})) \\ &= 0.5 \cdot h \cdot (f(a) + f(b)) + h \cdot \sum_{i=1}^{N-1} f(t_i). \end{aligned}$$

Dies ist eine Quadraturformel mit den Knoten t_i , $i = 0, \dots, N$, und den Gewichten $\lambda_0 = \lambda_N = 1/(2N)$ und $\lambda_1 = \dots = \lambda_{N-1} = 1/N$.

(c) Eine weitere Möglichkeit der Herleitung von Quadraturformeln ist es, die Funktion f durch das Interpolationspolynom zu geeigneten Knoten t_0, \dots, t_n zu approximieren und als Näherungsformel für $I(f)$ das exakte Integral des Interpolationspolynoms zu berechnen. Ist L_{in} das i -te Lagrange-Polynom zu den Knoten t_0, \dots, t_n (vgl. Abschnitt (5.1)), so ist das Interpolationspolynom gegeben durch

$$P(t) = \sum_{i=0}^n f(t_i) \cdot L_{in}(t);$$

das Integral von $P(t)$ erfüllt die Gleichung

$$\int_a^b P(t) dt = (b - a) \cdot \sum_{i=0}^n \lambda_i f(t_i)$$

Dies ist eine Approximationsformel der Form (5.1) mit den Gewichten

$$\lambda_i = \frac{1}{b - a} \int_a^b L_{in}(t) dt. \quad (5.2)$$

5.2 Newton-Cotes-Formeln

Die in Beispiel ... hergeleiteten Integrationsformeln heißen im Fall äquidistanter Knoten

$$t_i = a + i \cdot h, \quad i = 0, \dots, n \quad \text{mit} \quad h = \frac{b-a}{n} \quad (5.3)$$

Newton-Cotes-Formeln.

(5.2.5) Beispiele: (a) $n = 1$: Die Lagrange-Polynome zu den Knoten $t_0 = a$ und $t_1 = b$ sind

$$L_{01}(t) = \frac{b-t}{b-a} \quad \text{und} \quad L_{11}(t) = \frac{t-a}{b-a}.$$

Durch Integration folgt

$$\lambda_0 = \lambda_1 = \frac{1}{2}.$$

In Analogie zu Beispiel [5.3](b) wird dieses Verfahren als **Trapezregel** bezeichnet.

(b) $n = 2$: Die Lagrange-Polynome zu den Knoten $t_0 = a$, $t_1 = 0.5(a+b)$ und $t_2 = b$ sind

$$\begin{aligned} L_{02}(t) &= \frac{1}{(b-a)^2} \cdot (2t^2 - (a+3b)t + (a+b)b), \\ L_{12}(t) &= \frac{-4}{(b-a)^2} \cdot (t^2 - (a+b)t + ab), \\ L_{22}(t) &= \frac{1}{(b-a)^2} \cdot (2t^2 - (3a+b)t + a(a+b)). \end{aligned}$$

Für die Gewichte folgt

$$\lambda_0 = \lambda_2 = \frac{1}{6}, \quad \lambda_1 = \frac{4}{6}.$$

Das entsprechende Verfahren heißt **Simpson-Regel**.

Die Gewichte für die Newton-Cotes-Formeln sind für $n = 1, \dots, 4$ in Tabelle 4 zusammengestellt. Man überzeugt sich leicht, dass dies Quadraturformeln im Sinne der Definition [5.2] sind. Dies ändert sich ab $n = 8$, da hier auch negative Gewichte auftreten. Wie bei der Interpolation gilt auch für die Newton-Cotes-Formeln, dass Näherungen für große n nicht sinnvoll sind.

B – Fehlerrechnung

n	Gewichte					Verfahren			
1		$\frac{1}{2}$		$\frac{1}{2}$		Trapezregel			
2		$\frac{1}{6}$		$\frac{4}{6}$		$\frac{1}{6}$	Simpson-Regel		
3		$\frac{1}{8}$		$\frac{3}{8}$		$\frac{3}{8}$	$\frac{1}{8}$	Newtonsche 3/8-Regel	
4	$\frac{7}{90}$		$\frac{32}{90}$		$\frac{12}{90}$		$\frac{32}{90}$	$\frac{7}{90}$	Milne-Regel

Tabelle 6: Newton-Cotes-Formeln

Um den Fehler bei der Integration mit Hilfe der Newton-Cotes-Formeln abzuschätzen, erinnern wir uns an die Fehlerabschätzungen bei der Polynom-Interpolation. Die Formel (4.4) der Bemerkung [4.8] gibt eine Abschätzung des Interpolationsfehlers. Diese kann wie folgt in Art eines Zwischenwertsatzes präzisiert werden. Ist $P(t)$ das Interpolationspolynom zu den Knoten $t_0 < \dots < t_n$ und ist f mindestens $(n+1)$ -mal differenzierbar, so gibt es zu jedem $t \in [a, b]$ einen Zwischenwert $\tau = \tau(t) \in [a, b]$ derart, dass

$$P(t) - f(t) = \frac{f^{(n+1)}(\tau)}{(n+1)!} \cdot \omega(t). \quad (5.4)$$

Die Funktion $t \rightarrow \tau(t)$ ist stetig. Um hieraus eine Fehlerabschätzung für das Integral herzuleiten, benötigen wir folgendes Hilfsergebnis.

(5.2.6) Hilfssatz: g und h seien stetige Funktionen auf $[a, b]$. Für g gelte **entweder** $g(t) \geq 0$ für alle $t \in [a, b]$ **oder** $g(t) \leq 0$ für alle $t \in [a, b]$ (d.h. g hat keinen Vorzeichenwechsel in $[a, b]$). Dann gibt es ein $\tau \in [a, b]$ derart, dass

$$\int_a^b h(t)g(t)dt = h(\tau) \cdot \int_a^b g(t)dt.$$

Beweis. Wir setzen voraus, dass $g \geq 0$. (Der Beweis für $g \leq 0$ erfolgt analog.) Dann ist

$$\min_{t \in [a, b]} h(t) \cdot \int_a^b g(s)ds \leq \int_a^b h(s)g(s)ds \leq \max_{t \in [a, b]} h(t) \cdot \int_a^b g(s)ds.$$

Die stetige Funktion

$$F(t) := \int_a^b h(s)g(s)ds - h(t) \cdot \int_a^b g(s)ds$$

hat daher Werte t_0 und t_1 in $[a, b]$ mit $F(t_0) \geq 0$ und $F(t_1) \leq 0$. Nach dem Zwischenwertsatz für stetige Funktionen gibt es daher einen Wert $\tau \in [a, b]$ mit $F(\tau) = 0$. \square

Dieses Ergebnis erlaubt die folgenden Fehlerabschätzungen für die Newton-Cotes-Formeln.

(5.2.7) Fehlerabschätzungen: (a) Trapezregel. Es sei

$$T(f) := \frac{b-a}{2} (f(a) + f(b))$$

die Approximation von $I(f)$ nach der Trapezregel. Ist f zweimal stetig differenzierbar, so gilt mit einem geeigneten $\tau \in [a, b]$

$$T(f) - \int_a^b f(t) dt = \frac{(b-a)^3}{12} \cdot f''(\tau).$$

Beweis: $P(t)$ sei das lineare Interpolationspolynom von f zu den Knoten a und b . Nach Formel (5.4) ist

$$P(t) = f(t) + \frac{f''(\tau)}{2} \cdot (t-a) \cdot (t-b).$$

Die Funktion $(t-a) \cdot (t-b)$ ist ≤ 0 für $t \in [a, b]$. Damit ist die Voraussetzung des Hilfssatzes [5.6] erfüllt und es gilt mit einem geeigneten Zwischenwert $\bar{\tau}$

$$I(f) - T(f) = \frac{f''(\bar{\tau})}{2} \cdot \underbrace{\int_a^b (t-a)(t-b) dt}_{-\frac{(b-a)^3}{6}} = -\frac{f''(\bar{\tau})}{12} \cdot (b-a)^3. \quad \square$$

(b) Simpsonregel. Es sei

$$S(f) := \frac{b-a}{6} \left(f(a) + 4f\left(\frac{a+b}{2}\right) + f(b) \right)$$

die Approximation von $I(f)$ nach der Simpsonregel.

(i) Ist $f \in \mathcal{P}_3$, so ist $I(f) = S(f)$.

(ii) Ist f viermal stetig differenzierbar, so gilt mit einem geeigneten $\tau \in [a, b]$

$$S(f) - \int_a^b f(t) dt = \frac{(b-a)^5}{90} \cdot f^{(4)}(\tau).$$

Beweis: (i) Es sei $Q \in \mathcal{P}_3$ und $P \in \mathcal{P}_2$ das quadratische IP-Polynom zu den Knoten a , $(a+b)/2$ und b . Nach der Formel (1.13) zum Approximationsfehler für IP-Polynome gilt mit der Konstante $\gamma = Q'''(\tau)/6$

$$Q(t) = P(t) + \gamma \underbrace{(t-a)(t-(a+b)/2)(t-b)}_{\omega_3(t)}.$$

Da ω_3 ungerade bezüglich des Punktes $(a+b)/2$ ist, ist $\int_a^b \omega_3(t) dt = 0$ und

$$\int_a^b Q(t) dt = \int_a^b P(t) dt + \gamma \int_a^b \omega_3(t) dt = \int_a^b P(t) dt \quad .$$

(ii) Sei $f \in \mathcal{C}^4[a, b]$ und $Q \in \mathcal{P}_3$ das zugehörige Hermite-IP-Polynom zu den Knoten a , $(a+b)/2$, $(a+b)/2$ und b . Das Newton-Polynom hat in $[a, b]$ gleichbleibendes Vorzeichen:

$$\omega_4(t) = (t-a)(t-(a+b)/2)^2(t-b) \leq 0.$$

Nach der Fehlerformel für Hermite-Interpolierende folgt

$$f(t) = Q(t) + \frac{f^{(4)}(\tau)}{4!} \cdot \omega_4(t)$$

und damit (wie im Beweis für die Trapezregel)

$$\int_a^b f(t) dt = \underbrace{\int_a^b Q(t) dt}_{=S} + \frac{f^{(4)}(\tau)}{4!} \cdot \underbrace{\int_a^b \omega_4(t) dt}_{=-4h^5/15} = S - \frac{f^{(4)}(\tau)}{90} \cdot h^5 \quad . \quad \circ$$

Entsprechend können die folgenden Fehlerformeln bewiesen werden.

(c) Newtons 3/8-Regel. Bei mindestens viermal stetig differenzierbaren Funktionen f erfüllt der Fehler der Integrationsformel

$$N(f) := \frac{b-a}{8} \left(f(a) + 3f\left(a + \frac{b-a}{3}\right) + 3f\left(a + 2\frac{b-a}{3}\right) + f(b) \right)$$

eine Gleichung der Form

$$N(f) - \int_a^b f(t) dt = \frac{3(b-a)^5}{80} \cdot f^{(4)}(\tau)$$

mit einem geeigneten $\tau \in [a, b]$.

(d) Milne-Regel. Die Integrationsformel

$$M(f) := \frac{b-a}{90} \left(7f(a) + 32f\left(a + \frac{b-a}{4}\right) + 12f\left(a + \frac{b-a}{2}\right) + 32f\left(a + 3\frac{b-a}{4}\right) + 7f(b) \right)$$

hat für mindestens sechsmal differenzierbare Funktionen f einen Fehler der Form

$$M(f) - \int_a^b f(t) dt = \frac{8(b-a)^7}{945} \cdot f^{(6)}(\tau).$$

mit einem geeigneten $\tau \in [a, b]$.

$\frac{1}{24}$	$\frac{1}{12}$	$\frac{1}{12}$	$\frac{1}{12}$	$\frac{1}{12}$	$\frac{1}{12}$	$\frac{1}{12}$	$\frac{1}{12}$	$\frac{1}{12}$	$\frac{1}{12}$	$\frac{1}{12}$	$\frac{1}{12}$	$\frac{1}{24}$	Trapezregel
$\frac{1}{36}$	$\frac{4}{36}$	$\frac{2}{36}$	$\frac{4}{36}$	$\frac{2}{36}$	$\frac{4}{36}$	$\frac{2}{36}$	$\frac{4}{36}$	$\frac{2}{36}$	$\frac{4}{36}$	$\frac{2}{36}$	$\frac{4}{36}$	$\frac{1}{36}$	Simpson-Regel
$\frac{1}{32}$	$\frac{3}{32}$	$\frac{3}{32}$	$\frac{2}{32}$	$\frac{3}{32}$	$\frac{3}{32}$	$\frac{2}{32}$	$\frac{3}{32}$	$\frac{3}{32}$	$\frac{2}{32}$	$\frac{3}{32}$	$\frac{3}{32}$	$\frac{1}{32}$	Newtonsche 3/8-Regel
$\frac{7}{270}$	$\frac{32}{270}$	$\frac{12}{270}$	$\frac{32}{270}$	$\frac{14}{270}$	$\frac{32}{270}$	$\frac{12}{270}$	$\frac{32}{270}$	$\frac{14}{270}$	$\frac{32}{270}$	$\frac{12}{270}$	$\frac{32}{270}$	$\frac{7}{270}$	Milne-Regel

Tabelle 7: Gewichte für summierte Newton-Cotes-Formeln ($N = 12$)

C – Summierte Newton-Cotes-Formeln

Anstelle der Anwendung der Newton-Cotes-Formeln auf das ganze Intervall $[a, b]$ ist es in der Regel sinnvoller, das Intervall in M gleich große Teilintervalle zu teilen und die Quadraturformeln auf jedes Teilintervall anzuwenden.

(a) Summierte Trapexregel. Über jedem der $M = N$ Teilintervalle $[t_i, t_{i+1}]$ wenden wir die Trapezregel an und erhalten als Flächeninhalt der Teilintervalle

$$T_i = \frac{h}{2}(f(t_i) + f(t_{i+1})).$$

Durch Summation der Teilflächen erhalten wir die summierte Trapezregel

$$\hat{I}_T(f) = \sum_{i=0}^{N-1} T_i = \frac{b-a}{N} \left(\frac{1}{2} (f(a) + f(b)) + \sum_{i=1}^{N-1} f(t_i) \right).$$

Dies entspricht der Trapexregel des Beispiels [5.3](b). Der Fehler für jedes Teilintervall ist mit $h = (b-a)/N$ nach [5.7](a) gleich

$$T_i(f) - \int_{t_i}^{t_{i+1}} f(t) dt = \frac{h^3}{12} f''(\tau_i)$$

mit geeigneten $\tau_i \in [t_i, t_{i+1}]$. Da nach dem Mittelwertsatz gilt

$$\sum_{i=0}^{N-1} f''(\tau_i) = N \cdot f''(\tau)$$

für ein $\tau \in [a, b]$, kann der Gesamtfehler geschrieben werden in der Form

$$\hat{I}_T(f) - I(f) = h^2 \cdot \frac{b-a}{12} f''(\tau) = \mathcal{O}(h^2).$$

(b) Summierte Simpson-Regel. Wir teilen das Gesamtintervall in $M = N/2$ Teilin-

tervalle der Länge $2h$ und wenden auf jedes der Teilintervalle $[t_{2i}, t_{2i+2}]$ die Simpsonregel zu den Knoten t_{2i} , t_{2i+1} und t_{2i+2} an:

$$S_i(f) = \frac{2h}{M}(f(t_{2i}) + 4f(t_{2i+1}) + f(t_{2i+2})) = \frac{b-a}{36}(f(t_{2i}) + 4f(t_{2i+1}) + f(t_{2i+2})).$$

Durch Aufsummieren der M Teilflächen entsteht eine Summationsformel $\hat{I}_S(f)$. Für die Einzelfehler gilt nach [5.7](b)

$$S_i(f) - \int_{t_{2i}}^{t_{2i+2}} f(t)dt = \frac{(2h)^5}{90} \cdot f^{(4)}(\tau_i).$$

Den Gesamtfehler erhält man mit Hilfe des Mittelwertsatzes gemäß

$$\hat{I}_S(f) - I(f) = h^4 \cdot \frac{8(b-a)}{45} f^{(4)}(\tau) = \mathcal{O}(h^4).$$

(c) Summierte Newtonsche 3/8-Regel. Durch Einteilung von $[a, b]$ in $M = N/3$ Teilintervalle und Anwendung der 3/8-Regel erhält man eine Quadraturformel $\hat{I}_N(f)$ mit den in Tabelle 4 angegebenen Gewichten. Der Fehler ist

$$\hat{I}_N(f) - I(f) = h^4 \cdot \frac{243(b-a)}{80} f^{(4)}(\tau) = \mathcal{O}(h^4).$$

(d) Summierte Milne-Regel. Die summierte Milne-Regel $\hat{I}_M(f)$ zu $M = N/4$ Teilintervallen hat die in Tabelle 4 angegebenen Gewichte und den Fehler

$$\hat{I}_M(f) - I(f) = h^6 \cdot \frac{32768(b-a)}{945} f^{(6)}(\tau) = \mathcal{O}(h^6).$$

Für den Spezialfall $N = 12$ sind die Gewichte in Tabelle 6 zusammengestellt.

Wir fassen zusammen.

(5.2.8) Fehlerordnungen: Für summierte Newton-Cotes-Formeln zur Integration hinreichend glatter Funktionen mit der Schrittweite h gelten die Fehlerordnungen

$$\begin{aligned} \mathcal{O}(h^2) & \text{ für die Trapezregel,} \\ \mathcal{O}(h^4) & \text{ für die Simpsonregel,} \\ \mathcal{O}(h^4) & \text{ für Newtons 3/8-Regel,} \\ \mathcal{O}(h^6) & \text{ für die Milneregel.} \end{aligned}$$

(5.2.9) Übung: Wenden Sie die zusammengesetzten Newton-Cotes-Formeln an auf die Funktionen $f_i : [-1, 1] \rightarrow \mathbb{R}$,

$$f_0(t) = \cos(\pi t/2), \quad f_1(t) = 1 - |t|, \quad f_2(t) = \begin{cases} 1 & \text{in } [-0.5, 0.5] \\ 0 & \text{sonst} \end{cases}.$$

und beobachten Sie das Fehlerverhalten. Wie interpretieren Sie die Ergebnisse? Sind die oben beschriebenen Fehlerformeln anwendbar? Wie können die Verfahren modifiziert werden zur Erhöhung der Fehlerordnung?

5.3 Gauß-Christoffel-Quadratur

Bei den Newton-Cotes-Formeln sind die Knoten als äquidistant vorgegeben. Im Gegensatz hierzu sollen im Folgenden $n + 1$ “optimale” Knoten und zugehörige “optimale” Gewichte konstruiert werden.

In Verallgemeinerung zur bisherigen Fragestellung wollen wir nun das *gewichtete* Integral

$$I(f) := \int_a^b \omega(t)f(t)dt \quad \text{mit} \quad \omega(t) > 0 \quad \forall t \in (a, b) \quad (5.5)$$

numerisch lösen. Als **Voraussetzung** an die Gewichtsfunktion $\omega(\cdot)$ fordern wir, dass

$$\mu_k := \int_a^b t^k \omega(t)dt < \infty \quad , \quad k = 0, 1, 2, \dots$$

(Typische Gewichte sind in Tabelle 4 des Abschnitts 3.1 dargestellt.) Das **Ziel** ist es nun, Quadraturformeln der Form

$$\hat{I}_n(f) := \sum_{i=0}^n \lambda_{in} f(\tau_{in}) \quad (5.6)$$

zu entwerfen, wobei die Knoten τ_{in} und die Gewichte λ_{in} so zu wählen sind, dass Polynome möglichst hohen Grades *exakt* gelöst werden. Da die Anzahl der Freiheitsgrade gleich $2n + 2$ ist, sollten wir erwarten, dass Polynome bis zum Grad $2n + 1$ exakt integriert werden können.

Wie in Abschnitt 3.1 definieren wir das Skalarprodukt

$$\langle f, g \rangle := \int_a^b \omega(t)f(t)g(t)dt$$

und die zugehörige induzierte Norm

$$\|f\| := \langle f, f \rangle^{1/2} \quad .$$

(5.3.10) Lemma: Ist \hat{I}_n für alle Polynome $P \in \mathcal{P}_{2n+1}$ exakt (für $n = 0, 1, 2, \dots$), so sind die Polynome $P_n \in \mathcal{P}_{n+1}$, definiert durch

$$P_{n+1}(t) := (t - \tau_{0n}) \cdots (t - \tau_{nn})$$

paarweise orthogonal.

Beweis: Für $j < n + 1$ ist $P_{n+1} \cdot P_j \in \mathcal{P}_{2n+1}$, und daher

$$\langle P_j, P_{n+1} \rangle = \int_a^b \omega(t) P_j(t) P_{n+1}(t) dt = \hat{I}_n(P_j P_{n+1}) = \sum_{i=0}^n \lambda_{in} P_j(\tau_{in}) \underbrace{P_{n+1}(\tau_{in})}_{=0} = 0. \quad \circ$$

Da die im Lemma definierten Polynome P_0, \dots, P_n eine Basis von \mathcal{P}_n bilden, folgt $P_{n+1} \perp \mathcal{P}_n$. Umgekehrt gilt

(5.3.11) Lemma: Ist $P \in \mathcal{P}_{n+1}$ und $P \perp \mathcal{P}_n$, so hat P in (a, b) genau $n + 1$ einfache Nullstellen.

Beweis: Es seien $t_1, \dots, t_m \in (a, b)$ diejenigen Nullstellen von P , an denen P das Vorzeichen ändert. Wir definieren

$$Q(t) := (t - t_1) \cdots (t - t_m) \in \mathcal{P}_m \subseteq \mathcal{P}_{n+1}.$$

Damit hat die Funktion

$$\omega(t) \cdot P(t) \cdot Q(t)$$

in (a, b) keinen Vorzeichenwechsel und es folgt $\langle P, Q \rangle \neq 0$. Damit ist $Q \in \mathcal{P}_{n+1} \setminus \mathcal{P}_n$, also $m = n + 1$. \circ

Damit ergibt sich die folgende Situation: Konstruieren wir eine Folge P_0, P_1, P_2, \dots zueinander orthogonaler Polynome, $P_n \in \mathcal{P}_n$, so hat P_{n+1} genau $n + 1$ Nullstellen $\tau_{0n}, \dots, \tau_{nn}$. Da die Konstruktion des Orthogonalsystems bis auf die Wahl des führenden Koeffizienten eindeutig ist, sind diese Knoten eindeutig bestimmt.

Wir kommen zur Bestimmung der Gewichte λ_{in} . Mit Hilfe der Lagrange-Polynome L_{in} lässt sich $P \in \mathcal{P}_n$ darstellen durch

$$P(t) = \sum_{i=0}^n P(\tau_{in}) L_{in}(t).$$

Damit $P \in \mathcal{P}_n$ durch \hat{I}_n exakt integriert wird, muss demnach gelten

$$\int_a^b \omega(t)P(t)dt = \sum_{i=0}^n P(\tau_{in}) \int_a^b \omega(t)L_{in}(t)dt = \sum_{i=0}^n \lambda_{in}P(\tau_{in}) \quad ,$$

also

$$\lambda_{in} = \int_a^b \omega(t)L_{in}(t)dt. \quad (5.7)$$

(5.3.12) Lemma: $\tau_{0n}, \dots, \tau_{nn}$ seien wie oben definiert. Dann gilt für jede Quadraturformel der Form

$$\tilde{I}(f) := \sum_{i=0}^n \lambda_i f(\tau_{in}) \quad :$$

Ist \tilde{I} exakt auf \mathcal{P}_n , so auch auf \mathcal{P}_{2n+1} .

Beweis: \tilde{I} sei exakt auf \mathcal{P}_n , und es sei $P \in \mathcal{P}_{2n+1}$. Ferner sei $P_{n+1} \in \mathcal{P}_{n+1}$ das $(n+1)$ -te Orthogonalpolynom. Durch Polynomdivision finden wir zwei Polynome $Q, R \in \mathcal{P}_n$ mit $P = Q \cdot P_{n+1} + R$. Wegen $P_{n+1} \perp \mathcal{P}_n$ folgt

$$I(P) = \underbrace{\int_a^b \omega(t)Q(t)P_{n+1}(t)dt}_{=0} + I(R) = I(R) = \tilde{I}_n(R).$$

Andererseits ist

$$\tilde{I}_n(R) = \sum_{i=0}^n \lambda_{in}R(\tau_{in}) = \sum_{i=0}^n \lambda_{in} \left(Q(\tau_{in}) \underbrace{P_{n+1}(\tau_{in})}_{=0} + R(\tau_{in}) \right) = \tilde{I}(P) \quad ,$$

also $I(P) = \tilde{I}(P)$. $\quad \circ$

Wir fassen die bisherigen Ergebnisse zusammen.

(5.3.13) Satz: Es existieren eindeutig bestimmte Knoten $\tau_{0n}, \dots, \tau_{nn} \in [a, b]$ und Gewichte $\lambda_{0n}, \dots, \lambda_{nn}$ derart, dass die Quadraturformel \hat{I} (vgl. (4.6)) das gewichtete Integral I (vgl. (4.5)) für beliebige $P \in \mathcal{P}_{2n+1}$ exakt integriert. Die Knoten τ_{in} sind die Nullstellen des $(n+1)$ -ten Orthogonalpolynoms P_{n+1} . Die Gewichte λ_{in} sind gegeben durch (4.7).

Die Quadraturformel (4.6) mit den obigen Knoten und Gewichten heißt **Gauß-Christoffel-Formel**. Ohne Beweis bemerken wir, dass die Gewichte darstellbar sind als

$$\lambda_{in} = \frac{\langle P_n, P_n \rangle}{P'_{n+1}(\tau_{in}) \cdot P_n(\tau_{in})} \quad (5.8)$$

und stets positiv sind.

(5.3.14) Satz: Der Approximationsfehler der Gauß-Christoffel-Quadratur lässt sich für $f \in \mathcal{C}^{2n+2}$ darstellen durch

$$I(f) - \hat{I}_n(f) = \frac{f^{(2n+2)}(\tau)}{(2n+2)!} \cdot \langle P_{n+1}, P_{n+1} \rangle \quad \text{für ein } \tau \in (a, b) \quad .$$

Beweis: $P \in \mathcal{P}_{2n+1}$ sei die Hermite-Interpolierende von f zu den Knoten $\tau_{0n}, \tau_{0n}, \dots, \tau_{nn}, \tau_{nn}$. Nach Lemma [1.21] (Darstellung des Interpolationsfehlers) gilt

$$f(t) = P(t) + \xi[\tau_{0n}, \tau_{0n}, \dots, \tau_{nn}, \tau_{nn}, t] \cdot \underbrace{(t - \tau_{0n})^2 \cdot (t - \tau_{nn})^2}_{=: P_{n+1}^2} \quad . \quad (5.9)$$

Nach Lemma [1.22] ist

$$\xi[\tau_{0n}, \tau_{0n}, \dots, \tau_{nn}, \tau_{nn}, t] = \frac{f^{(2n+2)}(\tau(t))}{(2n+2)!}$$

mit einer stetigen Funktion $\tau(t)$. Wegen $P_{n+1}^2(t) \geq 0$ ist bei der Integration der Gleichung (4.9) das Lemma [4.7] anwendbar, welches auf das gewünschte Ergebnis führt. \circ

(5.3.15) Beispiel (Gauß-Tschebyscheff-Quadratur: $[a, b] = [-1, 1]$, $\omega(t) = 1/\sqrt{1-t^2}$. Die Orthogonalpolynome P_n mit führendem Koeffizienten 1 sind bis auf Normierung die Tschebyscheff-Polynome T_n (vgl. Abschnitte 1.4, 3.1). Aus ihren Nullstellen (vgl. Satz [1.27](d)) ergeben sich die Knoten zu

$$\tau_{in} = \cos\left(\frac{(2i+1)\pi}{2n+2}\right) \quad .$$

Die Gewichte sind aus Formel (4.8) berechenbar zu $\lambda_{in} = \pi/(n+1)$. Hieraus ergibt sich die Gauß-Christoffel-Quadraturformel

$$\hat{I}_n(f) = \frac{\pi}{n+1} \cdot \sum_{i=0}^n \cdot f(\tau_{in})$$

mit einem Quadraturfehler

$$I(f) - \hat{I}_n(f) = \frac{\pi}{2^{2n+1}(2n+2)!} \cdot f^{(2n+2)}(\tau) \quad .$$

Im Folgenden spielen *Bernoulli-Zahlen* und *Bernoulli-Polynome* eine Rolle; wir müssen sie deshalb kurz einführen, werden aber auf Details nicht eingehen.

(5.4.16) Definition: (a) Die *Bernoulli-Zahlen* B_n sind definiert als die Koeffizienten der Taylorreihe von $f(z) = z/(\exp(z) - 1)$; es gilt

$$\frac{z}{\exp(z) - 1} = \sum_{n=0}^{\infty} \frac{B_n}{n!} z^n. \quad (5.10)$$

(Der Konvergenzradius dieser Reihe ist 2π .)

(b) Mit Hilfe der Bernoullizahlen sind die *Bernoulli-Polynome* $P_n : [0, 1] \rightarrow \mathbb{R}$ definiert durch

$$P_n(x) = \frac{1}{n!} \sum_{k=0}^n \binom{n}{k} B_k x^{n-k} \quad (5.11)$$

(5.4.17) Bemerkungen: (a) Ab $n = 3$ verschwinden die Bernoulli-Zahlen mit ungeradem Index, d.h. $B_{2n+1} = 0$ für $n \geq 1$.

(b) Die Bernoulli-Zahlen mit geradem Index wachsen sehr schnell. Es gilt

$$2 \cdot \frac{2 \cdot (2n)!}{(2\pi)^{2n}} > (-1)^{n-1} B_{2n} > \frac{2 \cdot (2n)!}{(2\pi)^{2n}}. \quad (5.12)$$

Insbesondere folgt, dass $|B_{2n+2}/B_{2n}| \rightarrow \infty$. (Vgl. Knopp, S. 246.)

(c) Es ist

$$P_1(x) = x - \frac{1}{2}, \quad P_2(x) = \frac{x^2}{2} - \frac{x}{2} + \frac{1}{12}. \quad (5.13)$$

Für $k \geq 1$ ist

$$P'_{k+1}(x) = P_k(x). \quad (5.14)$$

Außerdem gilt für $k \geq 2$

$$P_k(0) = P_k(1) = \frac{B_k}{k!}. \quad (5.15)$$

(d) Für $k \geq 2$ gilt die Abschätzung

$$|P_k(x)| \leq \frac{4}{(2\pi)^k}. \quad (5.16)$$

(5.4.18) Folgerung: Ist $f : [0, 1] \rightarrow \mathbb{R}$ hinreichend glatt, so folgt aus Bemerkung (8.2)(c) durch fortgesetzte partielle Integration

$$\begin{aligned}
 \int_0^1 f(x)dx &= \int_0^1 P_1'(x)f(x)dx = P_1(x)f(x)|_0^1 - \int_0^1 P_1(x)f'(x)dx \\
 &= \frac{1}{2}[f(1) + f(0)] - \int_0^1 P_2'(x)f'(x)dx \\
 &= \frac{1}{2}[f(1) + f(0)] - [P_2(x)f'(x)]_0^1 + \int_0^1 P_2(x)f''(x)dx \\
 &= \frac{1}{2}[f(1) + f(0)] - \frac{B_2}{2!}[f'(1) - f'(0)] + \frac{B_3}{3!}[f''(1) - f''(0)] - \int_0^1 P_3(x)f'''(x)dx \\
 &= \dots
 \end{aligned}$$

Durch periodische Fortsetzung erweitern wir nun den Definitionsbereich für die Bernoulli-Polynome auf ganz \mathbb{R} , indem wir definieren

$$P_k(x) := P_k(x - [x]). \quad (5.17)$$

Hierbei ist $[x] = \max\{z \in \mathbb{Z} : z \leq x\}$. Es ist klar, dass die Formeln der Folgerung (8.3) sich auf beliebige Integrale $\int_n^{n+1} f(x)dx$ übertragen lassen. Hieraus folgt die für uns zentrale Formel, die

(5.4.19) Eulersche Summenformel (auch *Euler-MacLaurinsche Summenformel*): Ist f in $[0, n]$ $(2k + 1)$ -mal stetig differenzierbar, so gilt

$$\sum_{\nu=0}^n f(\nu) = \int_0^n f(x)dx + \frac{1}{2}(f_0 + f_n) + \sum_{\mu=1}^k c_\mu(f) + R_k(f) \quad (5.18)$$

mit den Koeffizienten

$$c_\mu(f) = \frac{B_{2\mu}}{(2\mu)!} \cdot [f^{(2\mu-1)}(n) - f^{(2\mu-1)}(0)] \quad (5.19)$$

und dem Restglied

$$R_k(f) = \int_0^n P_{2k+1}(x)f^{(2k+1)}(x)dx. \quad (5.20)$$

Die Eulersche Summenformel ist ein wichtiges Hilfsmittel z.B. zur Berechnung von Fol-gengrenzwerten und von unendlichen Reihen.

(5.4.20) Beispiel: *Berechnung der Eulerschen Konstante:* Die Funktion $f : \mathbb{R}_+ \rightarrow \mathbb{R}$, $f(x) = 1/(1+x)$ ist monoton fallend, und es ist

$$\int_0^{n-1} f(x) dx = \ln(n). \quad (5.21)$$

Durch Bildung von Ober- und Untersummen findet man leicht, dass

$$\sum_{k=2}^n \frac{1}{k} \leq \ln n \leq \sum_{k=1}^{n-1} \frac{1}{k} \quad (5.22)$$

und damit

$$\frac{1}{n} \leq \sum_{k=1}^n \frac{1}{k} - \ln n \leq 1. \quad (5.23)$$

Die Eulersche Konstante γ ist definiert durch

$$\gamma = \lim_{n \rightarrow \infty} \underbrace{\left(\sum_{\nu=1}^n \frac{1}{\nu} - \ln n \right)}_{=: \gamma_n} \quad (5.24)$$

Diese Zahl spielt eine wichtige Rolle z.B. im Zusammenhang mit der Eulerschen Γ -Funktion und der Riemannschen ζ -Funktion. Zu ihrer Berechnung setzen wir die Funktion $f(x) = 1/(1+x)$ in die Eulersche Summenformel ein und erhalten

$$\sum_{\nu=1}^n \frac{1}{\nu} = \underbrace{\int_1^n f(x) dx}_{=\ln n} + \frac{1}{2} + \frac{1}{2n} + \sum_{\mu=1}^k c_{\mu}(f) + R_k(f), \quad (5.25)$$

also für $n \rightarrow \infty$

$$\begin{aligned} \gamma &= \frac{1}{2} - \sum_{\mu=1}^k B_{\mu} f^{(2\mu-1)}(0) + \int_0^{\infty} P_{2k+1}(x) f^{(2k+1)}(x) dx \\ &= \frac{1}{2} + \sum_{\nu=1}^k \frac{B_{2\nu}}{2\nu} - (2k+1)! \int_1^{\infty} \frac{P_{2k+1}(x)}{x^{2k+2}} dx \quad \text{für } k = 1, 2, \dots \end{aligned} \quad (5.26)$$

Diese Darstellung ist für $k \rightarrow \infty$ unbrauchbar, da das Restglied divergiert. (Die Reihe $\sum_{\nu=1}^{\infty} \frac{B_{2\nu} x^{2\nu}}{2\nu}$ hat den Konvergenzradius 0!) Setzen wir $k = 3$, so folgt [Knopp S. 545]

$$\gamma = \frac{1}{2} + \sum_{\nu=1}^3 \frac{B_{2\nu}}{2\nu} - \underbrace{7! \int_1^{\infty} \frac{P_7(x)}{x^8} dx}_{R_3(f)} - E \quad (5.27)$$

mit dem Fehler

$$|E| = 7! \left| \int_4^{\infty} \frac{P_7(x)}{x^8} dx \right| \leq \frac{4 \cdot 7!}{(2\pi)^7} \int_4^{\infty} \frac{1}{x^8} dx < 10^{-6}. \quad (5.28)$$

$R_3(f)$ lässt sich aus (8.16) berechnen, und es folgt bis auf einen Fehler der Größenordnung 10^{-6} die Approximation $\gamma = 0.577215$.

Im Folgenden sei $\tau_n \searrow 0$ eine monoton fallende Nullfolge und $g : \{\tau_n, n \in \mathbb{N}\} \rightarrow \mathbb{R}$ eine Abbildung.

(5.4.21) Definition: Eine (formale) Reihe der Form

$$a_0 + a_1\tau + a_2\tau^2 + \dots \quad (5.29)$$

heißt *asymptotische Darstellung von g bzgl. der Nullfolge τ_n* , falls für jedes feste $k \in \mathbb{N}$ gilt

$$\left[g(\tau_n) - (a_0 + a_1\tau_n + \dots + a_k\tau_n^k) \right] \tau_n^{-k} \rightarrow 0 \quad (5.30)$$

für $n \rightarrow \infty$. Wir schreiben hierfür

$$g(\tau) \sim a_0 + a_1\tau + a_2\tau^2 + \dots \quad (5.31)$$

Der Unterschied zwischen Potenzreihenentwicklung und asymptotischer Darstellung soll im folgenden Beispiel illustriert werden.

(5.4.22) Beispiel: Für $c > 0$ definieren wir die Funktion $f_c : \mathbb{R}_+ \rightarrow \mathbb{R}$ durch

$$f_c(x) := \frac{1}{c+x} = \frac{1}{c} \cdot \frac{1}{1+x/c}. \quad (5.32)$$

Für $|x/c| < 1$ hat die *Potenzreihenentwicklung*

$$f_c(x) = \frac{1}{c} \sum_{\nu=0}^{\infty} (-1)^\nu \left(\frac{x}{c} \right)^\nu. \quad (5.33)$$

Dagegen hat $f_c(x)$ für beliebige $c > 0$ und beliebige Nullfolgen τ_n die *asymptotische Darstellung*

$$\frac{1}{c} - \frac{\tau}{c^2} + \frac{\tau^2}{c^3} - + \dots \quad (5.34)$$

Ist $a_0 + a_1\tau + a_2\tau^2 + \dots$ eine asymptotische Darstellung von g , so bedeutet dies nicht, dass dies eine Potenzreihe von g ist. In den für uns interessanten Fällen wird der Konvergenzradius der Reihe häufig gleich 0 sein. Dennoch können wir für unsere Zwecke wichtige Ergebnisse ableiten.

5.4.2 Trapezregel: Asymptotische Fehlerentwicklung

Gegeben sei eine hinreichend glatte Funktion $f : [a, b] \rightarrow \mathbb{R}$. Numerisch berechnet werden soll das Integral

$$I(f) = \int_a^b f(t) dt. \quad (5.35)$$

Wir führen eine äquidistante Zerlegung von $[a, b]$ durch:

$$t_i := a + ih, \quad i = 0, \dots, n, \quad h = (b - a)/n. \quad (5.36)$$

Die summierte Trapezregel für diese Zerlegung lautet

$$T_h(f) = \frac{b-a}{n} (0.5f(a) + (f(t_1) + \dots + f(t_{n-1})) + 0.5f(b)). \quad (5.37)$$

Aus Abschnitt 5.2 wissen wir, dass der Fehler wie folgt dargestellt werden kann:

$$T_h(f) - I(f) = h^2 \cdot \frac{b-a}{12} f''(\tau) = \mathcal{O}(h^2). \quad (5.38)$$

Die Eulersche Summenformel erlaubt eine genauere Darstellung des Fehlers. Hierzu definieren wir $F : [0, n] \rightarrow \mathbb{R}$ durch

$$F(t) = f\left(a + \frac{b-a}{n} \cdot t\right) = f(a + ht). \quad (5.39)$$

Offenbar gelten neben $F(k) = f(t_k)$ die Beziehungen

$$F^{(\ell)}(t) = h^\ell \cdot f^{(\ell)}(a + ht) \quad (5.40)$$

sowie

$$I(f) = h \cdot \int_0^n F(t) dt. \quad (5.41)$$

Die Anwendung der Eulerschen Summenformel auf F ergibt

$$h \cdot \int_0^n F(t) dt = h \left[\frac{1}{2} F(0) + F(1) + \dots + F(n-1) + \frac{1}{2} F(n) \right] \quad (5.42)$$

$$\begin{aligned} &+ h \sum_{\mu=1}^k \frac{B_{2\mu}}{(2\mu)!} [F^{(2\mu-1)}(n) - F^{(2\mu-1)}(0)] + h \int_0^n P_{2k+1}(t) F^{(2k+1)}(t) dt \\ &= h \underbrace{\left[\frac{1}{2} f(t_0) + f(t_1) + \dots + f(t_{n-1}) + \frac{1}{2} f(t_n) \right]}_{=T_h(f)} \quad (5.43) \end{aligned}$$

$$+ \sum_{\mu=1}^k h^{2\mu} \frac{B_{2\mu}}{(2\mu)!} [f^{(2\mu-1)}(b) - f^{(2\mu-1)}(a)] + h^{2k+1} \cdot R_k(h)$$

mit dem Restglied

$$R_k(h) = \int_a^b P_{2k+1} \left(\frac{t-a}{h} \right) f^{(2k+1)}(t) dt. \quad (5.44)$$

Mit Bemerkung (8.2)(b) folgt

$$|R_k(h)| \leq \frac{4}{(2\pi)^k} \cdot \int_a^b |f^{(2k+1)}(t)| dt, \quad (5.45)$$

ist also gleichmäßig (bzgl. h) beschränkt. Hieraus folgt für beliebig oft differenzierbare Funktionen

(5.4.23) Satz: Bezüglich der Nullfolge $h_n = (b-a)/n$, $n = 1, 2, \dots$ hat die Trapezregel die asymptotische Darstellung

$$T_h(f) = I(f) + \tau_1 h^2 + \tau_2 h^4 + \dots \quad (5.46)$$

mit den Koeffizienten

$$\tau_k = \frac{B_{2k}}{(2k)!} [f^{(2k-1)}(b) - f^{(2k-1)}(a)] \quad (5.47)$$

Wie dieses Ergebnis verwendet werden kann, um die Trapezregel zu verbessern, zeigt das folgende Beispiel. Wichtig ist, dass die Koeffizienten τ_ν von h unabhängig sind.

(5.4.24) Beispiel: Der Fehler bei der Trapezregel ist

$$T_h(f) - I(f) = \tau_1 h^2 + \tau_2 h^4 + h^5 \cdot R_k(h) = \mathcal{O}(h^2). \quad (5.48)$$

Berechnen wir die Trapezregel zu den Schrittweiten $h = (b-a)/n$ und $h/2 = (b-a)/(2n)$, so gilt

$$T_h(f) = I(f) + \tau_1 h^2 + \tau_2 h^4 + h^5 \cdot R_k(h) \quad (5.49)$$

$$T_{h/2}(f) = I(f) + \tau_1 \frac{h^2}{4} + \tau_2 h^4 + \frac{h^5}{32} \cdot R_k(h/2) \quad (5.50)$$

Hieraus folgt

$$4T_{h/2}(f) - T_h(f) = 3I(f) + 3\tau_2 h^4 + \mathcal{O}(h^5), \quad (5.51)$$

also

$$\frac{4T_{h/2}(f) - T_h(f)}{3} - I(f) = \mathcal{O}(h^4). \quad (5.52)$$

Wir können also durch Kombination von Ergebnissen der Trapezregel die Genauigkeit von $\mathcal{O}(h^2)$ auf $\mathcal{O}(h^4)$ erhöhen. Es ist aufschlussreich zu sehen, was mit der Formel $(4T_{h/2}(f) - T_h(f))/3$ berechnet wird. Bezeichnen wir die Knoten

$$t_i = a + i \cdot \frac{h}{2}, \quad i = 0, \dots, 2n, \quad (5.53)$$

so ist

$$T_h = \frac{b-a}{n} \left[\frac{1}{2}f(t_0) + f(t_2) + f(t_4) + \dots + f(t_{2n-2}) + \frac{1}{2}f(t_{2n}) \right] \quad (5.54)$$

$$T_{2h} = \frac{b-a}{2n} \left[\frac{1}{2}f(t_0) + f(t_1) + f(t_2) + \dots + f(t_{2n-1}) + \frac{1}{2}f(t_{2n}) \right] \quad (5.55)$$

und damit

$$\frac{4T_{h/2}(f) - T_h(f)}{3} = (b-a) \cdot \frac{1}{6n} \left[f(t_0) + \sum_{i=1}^{n-1} [4f(t_{2i-1}) + 2f(t_{2i})] + 4f(t_{2n-1}) + f(t_{2n}) \right].$$

Dies ist gerade die Formel für die summierte Simpsonregel.

5.4.3 Das Romberg-Verfahren

Im folgenden definieren wir $\eta := h^2$ und $\mathcal{T}(\eta) := T_h(f)$. Nach Satz (8.8) hat $\mathcal{T}(\eta)$ die asymptotische Darstellung (bzgl. der Nullfolge $\eta_n = [(b-a)/n]^2$)

$$\mathcal{T}(\eta) \sim I(f) + \tau_1\eta + \tau_2\eta^2 + \dots \quad (5.56)$$

(hinreichende Glattheit von f vorausgesetzt). Die Idee des klassischen Romberg-Verfahrens ist, durch Anwenden der Trapezregel zu verschiedenen Schrittweiten $h_0 > h_1 > \dots$ ein Interpolationspolynom zu $\mathcal{T}(\eta)$ zu berechnen und dieses an der Stelle $\eta = 0$ auszuwerten. (Da 0 außerhalb der Knoten η_i ist, heißt dieses Verfahren *Extrapolation*.)

Hierzu stellen wir fest, dass sich nach Satz (8.8) das Ergebnis $\mathcal{T}(h^2)$ der summierten Trapezregel zur Schrittweite h gut durch ein Polynom m -ten Grades in $\eta = h^2$ beschreiben lässt, falls die zu integrierende Funktion f genügend glatt ist. Beispielsweise kann die Trapezregel zu $m+1$ verschiedenen Knoten h_i berechnet werden und als approximierendes Polynom das Interpolationspolynom $\mathcal{P}(\eta)$ zu den Knoten $\eta_i = h_i^2$, $i = 0, \dots, m$ bestimmt werden. Da nur eine Approximation des gesuchten Integralwerts $I(f) = \mathcal{T}(0)$ gesucht ist, ist eine explizite Bestimmung der Koeffizienten von \mathcal{P} nicht nötig. Stattdessen kann der Wert $\mathcal{P}(0)$ mit Hilfe des Neville-Schemas [4.5] bestimmt werden. Dies

N	$T_h f_1$	Fehler	$T_h f_2$	Fehler	$T_h f_3$	Fehler
2	0.94805945	0.05194055	0.70050081	0.29949919	1.50000000	0.50000000
4	0.98711580	0.01288426	0.90357622	0.09642378	1.25000000	0.25000000
8	0.98711580	0.01288426	0.90357622	0.09642378	1.12500000	0.12500000
16	0.99678517	0.00321483	0.96801730	0.03198270	1.06250000	0.06250000
32	0.99919668	0.00080332	0.98920035	0.01079965	1.03125000	0.03125000
64	0.99979919	0.00020081	0.99630723	0.00369277	1.01562500	0.01562500
128	0.99994980	0.00005020	0.99872566	0.00127434	1.00781250	0.00781250
256	0.99998745	0.00001255	0.99955726	0.00044274	1.00390625	0.00390625
512	0.99999686	0.00000314	0.99984542	0.00015458	1.00195313	0.00195313
1024	0.99999922	0.00000078	0.99994583	0.00005417	1.00097656	0.00097656
2048	0.99999980	0.00000020	0.99998097	0.00001903	1.00048828	0.00048828
4096	0.99999995	0.00000005	0.99999330	0.00000670	1.00024414	0.00024414
8192	0.99999998	0.00000002	0.99999764	0.00000236	1.00012207	0.00012207
16384	1.00000000	0.00000000	0.99999917	0.00000083	1.00006104	0.00006104

Tabelle 8: Trapezregel $T_h(f)$ und Fehler $|T_h(f) - I(f)|$

(5.4.26) Beispiele: (a) Die Funktion $f_1 : [0, \pi/2] \rightarrow \mathbb{R}$ sei definiert durch $f_1(t) = \cos(t)$. Das Integral $I(f_1) = \int_0^{\pi/2} f_1(t) dt$ soll mit Hilfe der Trapezregel $T_h(f_1)$ mit den Schrittweiten $h = \pi \cdot 2^{-N}$, $N = 1, 2, 3, \dots$ gelöst werden. Da wir wissen, dass die Trapezregel die Fehlerordnung $\mathcal{O}(h^2)$ hat, nehmen wir als Faustregel an, dass der Fehler mit jedem Schritt etwa um den Faktor 4 abnehmen sollte. Die Ergebnisse und die Fehler $|T_h(f_1) - I(f_1)|$ sind in den ersten Spalten von Tabelle 8.1 angegeben. Die Faustregel bestätigt sich. (Das exakte Ergebnis ist $I(f_1) = 1$.) Einen Integrationsfehler kleiner als 10^{-8} erhalten wir für $N = 16384$.

Wenden wir nun das Romberg-Verfahren an, so finden wir, dass wir einen ähnlichen Fehler bereits für $N = 16$ erhalten (vgl. Tabelle 8.2).

(b) Wenden wir nun diese Verfahren auf die Funktionen $f_2, f_3 : [0, \pi/2] \rightarrow \mathbb{R}$, definiert durch

$$f_2(t) = \sqrt{\sin(t)} \cos(t), \quad f_3(t) = \begin{cases} 2/\pi & \text{für } t < \pi/2 \\ 0 & \text{sonst} \end{cases}$$

N	$T_h f_1$			
2	0.94805945			
4	0.98711580	1.00013459		
8	0.99678517	1.00000830	0.99999988	
16	0.99919668	1.00000051	1.00000000	1.00000000

Tabelle 9: Romberg-Verfahren für f_1

an, so finden wir wesentlich schlechtere Ergebnisse. Die Trapezregel für $N = 16384$ liefert die Fehler $8.3 \cdot 10^{-7}$ bzw. $6.104 \cdot 10^{-5}$. Das Romberg-Verfahren für $N = 16$ ergibt die Fehler 0.0032 und 0.038.

Der Grund für das wesentlich schlechtere Abschneiden von f_2 und f_3 (verglichen mit f_1) liegt in der mangelnden Regularität. Die Voraussetzung für die Gültigkeit der Fehlerformeln für die Trapezregel und das Rombergverfahren ist, dass der Integrand hinreichend oft stetig differenzierbar ist (mindestens zweimal für die Trapezregel). Dagegen hat f_2 eine unbeschränkte erste Ableitung, und die Funktion f_3 ist sogar unstetig.

(c) Als letztes Beispiel betrachten wir das Integral $\int_0^{\pi/2} n(t) dt$ für die Funktion $n(t) = c/(10^{-4} + (t-1)^2)$ ("Nadelimpuls"). Die Ergebnisse der Trapezregel sind in Tabelle ... dargestellt. Hier sehen wir ein auf den ersten Blick seltsames Verhalten. Für $N < 256$ sehen wir ein sehr irreguläres Verhalten mit großen Fehlern. Erst dann stabilisieren sich die Ergebnisse und konvergieren recht schnell gegen den korrekten Wert. Wie ist dieses Verhalten zu erklären?

Bei der Abschätzung des Fehlers hilft uns der folgende Satz.

(5.4.27) Satz: $\mathcal{T}(h^2)$ sei gegeben zu den Werten h_1^2, \dots, h_m^2 . Für hinreichend glatte f liefert das Extrapolationstableau einen Approximationsfehler der Form

$$\epsilon_{ik} = |\tau_k| \cdot h_{i-k+1}^2 \cdots h_i^2 + \sum_{j=i-k+1}^i \mathcal{O}(h_j^{2k+1}) \quad .$$

Zum Beweis benötigen wir folgendes Hilfsergebnis.

(5.4.28) Lemma: Für die Lagrange-Polynome zu den Stützstellen t_0, \dots, t_n gilt

$$\sum_{j=0}^n L_j(0)t_j^m = \begin{cases} 1 & \text{für } m = 0 \\ 0 & \text{für } 1 \leq m \leq n \\ (-1)^n t_0 \cdots t_n & \text{für } m = n + 1 \end{cases}$$

Beweis: $0 \leq m \leq n$: $P(t) := \sum_{j=0}^n L_j(t)t_j^m$ ist das Interpolationspolynom zu den Punkten (t_j, t_j^m) , $j = 0, \dots, n$. Damit interpoliert P die Funktion T^m . Aus $m \leq n$ folgt $P(t) = t^m$.

$m = n + 1$: Definiere $Q(t) := t^{n+1} - \sum_{j=0}^n L_j(t)t_j^{n+1}$. Dann ist $Q \in \mathcal{P}_{n+1}$ und hat die Nullstellen t_j , $j = 0, \dots, n$. Damit ist $Q(t) = (t - t_0) \cdots (t - t_n)$. \circ

Beweis von Satz [4.20]: O.B.d.A. sei $i = k$. Es ist

$$\mathcal{T}_{j1} = \mathcal{T}(h_j^2) = \tau_0 + \tau_1 h_j^2 + \cdots + \tau_k h_j^{2k} + \mathcal{O}(h_j^{2k+1}) \quad .$$

Für das Interpolationspolynom

$$P_{kk}(h^2) := \sum_{j=1}^k L_j(h^2) \mathcal{T}_{j1}$$

folgt mit Hilfe des Lemmas [4.21]

$$\begin{aligned} \mathcal{T}_{kk} &= P_{kk}(0) = \sum_{j=1}^k L_j(0) \mathcal{T}_{j1} \\ &= \sum_{j=1}^k L_j(0) \left(\tau_0 + \tau_1 h_j^2 + \cdots + \tau_k h_j^{2k} + \mathcal{O}(h_j^{2k+1}) \right) \\ &= \tau_0 + \tau_k \cdot (-1)^{k-1} h_1^2 \cdots h_k^2 + \mathcal{O}(h_j^{2k+2}) \quad . \quad \circ \end{aligned}$$

(5.4.29) Bemerkungen: (a) Geeignete Abbruchkriterien ergeben sich durch Beobachtung der Entwicklung der Näherungswerte \mathcal{T}_{ii} , $i = 1, 2, \dots$

(b) Bei fortschreitender Iteration ist zu beachten, dass das Romberg-Verfahren nur gerechtfertigt ist für hinreichend glatte Funktionen f . Ist diese Voraussetzung nicht erfüllt, so wird sich die Approximation mit fortschreitendem Index i i.a. verschlechtern.

(c) Ist f hinreichend glatt, so ergibt eine Fehleranalyse für den Wert \mathcal{T}_{ii} einen Fehler der Ordnung $\mathcal{O}(H^{-2i})$.

5.4.4 Adaptive Verfahren

5.5 Die Idee; Fehlerschätzer

Kommen wir kurz auf das Beispiel des Nadelimpulses aus (.,.) (b). Es ist offensichtlich, dass außerhalb eines kleinen Intervalls $[1 - \epsilon, 1 + \epsilon]$ die Schrittweite groß gewählt werden kann, während in der Nähe des Peaks die Funktion sehr genau aufgelöst werden muss. Das Ziel dieses Abschnitts ist es, numerische Verfahren zu skizzieren, welche im Bedarfsfall die numerische Approximation eines Teilintervalls ein Ergebnis zurückweisen, wenn die Genauigkeit zu klein ist, und mit feinerer Diskretisierung eine bessere Approximation erzeugen. Nötig für ein solches *adaptives Verfahren* ist die Existenz eines *Fehlerschätzers*, welcher Anhaltspunkte liefern kann, ob die gewählte Diskretisierung zu grob gewählt war.

Die Grundlage für einen Fehlerschätzer liefert die folgende Überlegung. Betrachten wir ein Teilintegral

$$I_t^{t+H} := \int_t^{t+H} f(\tau) d\tau. \quad (5.57)$$

Wir konstruieren das Extrapolationstableau zu den Schrittweiten $h_i = H/n_i$, $i = 1, \dots, m$. Ist \mathcal{T}_{ik} der (i, k) -te Eintrag im Tableau und $\epsilon_{ik} = |\mathcal{T}_{ik} - I_t^{t+H}|$ der zugehörige Integrationsfehler, so gilt nach Satz ... für den führenden Fehlerterm

$$\epsilon_{ik} \doteq |\tau_k| h_{i-k+1}^2 \cdots h_i^2 \quad (5.58)$$

Vergleichen wir diesen Fehler mit Nachbarwerten des Tableaus, so finden wir

$$\epsilon_{i,k+1} \doteq |\tau_{k+1}| h_{i-k}^2 \cdots h_i^2 = \frac{|\tau_{k+1}|}{|\tau_k|} h_{i-k}^2 \cdot \epsilon_{ik} \ll \epsilon_{ik} \quad (5.59)$$

für h_{i-k} hinreichend klein, sowie – mit $h_{i+1} = h_{i-k+1} \cdot 2^{-k}$ –

$$\epsilon_{i+1,k} \doteq |\tau_k| h_{i-k+2}^2 \cdots h_{i+1}^2 = \frac{h_{i+1}^2}{h_{i-k+1}^2} \cdot \epsilon_{ik} = 2^{-2k} \cdot \epsilon_{ik} \ll \epsilon_{ik} \quad (5.60)$$

falls k hinreichend groß ist. Dies rechtfertigt die folgende Annahme.

Im folgenden Schema bezeichne die Schreibweise $\epsilon \rightarrow \delta$ so viel wie $|\epsilon| \ll |\delta|$ bzw. $|\epsilon/\delta| \ll 1$. Gestützt auf die Aussage des Satzes ... ist folgende Annahme sinnvoll.

(5.5.30) **Annahme:** Im Fehlertableau des Extrapolationsverfahrens gelte

$$\begin{array}{ccccccc}
 & & \epsilon_{11} & & & & \\
 & & \uparrow & & & & \\
 & & \epsilon_{21} & \leftarrow & \epsilon_{22} & & \\
 & & \uparrow & & \uparrow & & \\
 & & \epsilon_{31} & \leftarrow & \epsilon_{32} & \leftarrow & \epsilon_{33} \\
 & & \vdots & & \vdots & & \vdots
 \end{array} \tag{5.61}$$

In diesem Fall können wir Schätzer für die einzelnen Einträge im Tableau aus den Nachbarwerten konstruieren. Beispielsweise gilt

$$\begin{aligned}
 |\mathcal{T}_{k,k-1} - I_t^{t+H}| &= |(\mathcal{T}_{k,k-1} - \mathcal{T}_{k,k}) + (\mathcal{T}_{k,k} - I_t^{t+H})| \\
 &\leq |\mathcal{T}_{k,k-1} - \mathcal{T}_{k,k}| + \underbrace{|\mathcal{T}_{k,k} - I_t^{t+H}|}_{\ll |\mathcal{T}_{k,k-1} - I_t^{t+H}|} \approx |\mathcal{T}_{k,k-1} - \mathcal{T}_{k,k}| \tag{5.62}
 \end{aligned}$$

Damit können wir als einen Schätzer für den Fehler $\epsilon_{k,k-1}$ definieren

$$\bar{\epsilon}_{k,k-1} := |\mathcal{T}_{k,k-1} - \mathcal{T}_{k,k}| \tag{5.63}$$

Hierauf aufbauend können wir nun adaptive numerische Verfahren konstruieren.

5.6 Die Idee der Schrittweitensteuerung

Die Berechnung des Teilintegrals

$$I_a^t = \int_a^t f(s) ds \tag{5.64}$$

sei bereits abgeschlossen, und ein folgendes Teilintegral \int_t^{t+H} soll numerisch approximiert werden. Dies geschieht durch die Berechnung des Romberg-Schemas. Zur Berechnung der k -ten Zeile sind folgende Schritte durchzuführen.

- (i) Als Eingabeparameter müssen eine Grundschriftweite H und eine Toleranzgrenze tol , welche bei der numerische Berechnung nicht überschritten werden darf, festgelegt werden.
- (ii) Als nächstes muss eine Formel hergeleitet werden für die maximal erlaubte Schrittweite bei gegebener Toleranz. Hierzu wissen wir aus Satz ...: Wird das Romberg-Tableau mit Schrittweiten $h_i \leq H$ berechnet, so haben die Einträge der k -ten Zeile einen Fehler von etwa

$$\epsilon(t, H) \leq \gamma \cdot H^{2k+1} \quad (5.65)$$

(mit unbekanntem γ); hierbei wurde berücksichtigt, dass

$$\tau_k = \frac{B_{2k}}{(2k)!} [f^{(2k-1)}(t+H) - f^{(2k-1)}(t)] = \frac{B_{2k}}{(2k)!} f^{(2k)}(\tau) \cdot H \quad (5.66)$$

Die maximale Schrittweite \tilde{H} wird nun erreicht, wenn $\text{tol} = \gamma \cdot \tilde{H}^{2k+1}$, also – durch Einsetzen von γ

$$\tilde{H} = \left(\frac{\text{tol}}{\epsilon} \right)^{1/(2k+1)} \cdot H \quad (5.67)$$

- (iii) Da ϵ nicht bekannt ist, wird die k -te Spalte des Romberg-Tableaus mit der Schrittweite H berechnet und ϵ durch den Schätzer $\bar{\epsilon}_{k,k+1}$ ersetzt (oder besser: durch $\bar{\epsilon}_{k,k+1}/\rho$ mit einem ‘‘Sicherheitsfaktor’’ $\rho < 1$). Dies führt auf den Schrittweitevorschlag für die k -te Spalte

$$\tilde{H}_k = \left(\frac{\rho \cdot \text{tol}}{\bar{\epsilon}_{k,k+1}} \right)^{1/(2k+1)} \cdot H \quad (5.68)$$

- (iv) Die optimale Ordnung erhält man durch Minimierung des Aufwands pro Schrittweite: $W_k := A_k/\tilde{H}_k$, wobei der Aufwand A_k z.B. die Anzahl der benötigten Funktionsaufrufe sein kann.

Ein **erster Entwurf** zur Steuerung eines Romberg-Schrittes könnte wie folgt aussehen.

- *Eingabeparameter*: linker Intervallrand t , aktuelle Schrittweite H , vorgeschlagene Tiefe k des Tableaus
- *Ausgabeparameter*: neu berechnete Werte \tilde{H} , $\tilde{t} = t + \tilde{H}$ und \tilde{k} , sowie numerischer Integralwert $\mathcal{T}_{\tilde{k}\tilde{k}}$ für $I_t^{\tilde{t}}$.
- *Programmsegment*:
 Berechne Extrapolationsschema $\mathcal{T}_{11}, \dots, \mathcal{T}_{kk}$
 Solange $k < kmax$ und $\bar{\epsilon}_{k,k+1} > tol$:
 setze $k := k + 1$
 berechne k -te Zeile des Extrapolationsschemas
 Berechne $\tilde{H}_1, \dots, \tilde{H}_k$ und W_1, \dots, W_k
 Wähle \tilde{k} für minimalen Aufwand
 Berechne $\tilde{H} := H_{\tilde{k}}$, $\tilde{t} := t + \tilde{H}$ und $\tilde{I}_t^{\tilde{t}} := \mathcal{T}_{\tilde{k}\tilde{k}}$.

Ein von Deuffhard-Hohmann (Beispiel (9.31) in Numerische Mathematik I) berichtetes Testbeispiel zur Berechnung der Nadelfunktion (vgl. Beispiel ... in diesem Skript) ergibt für eine geforderte Genauigkeit von 10^{-9} für das klassische Rombergverfahren 4097 Funktionsaufrufe, während das adaptive Verfahren mit 321 Aufrufen auskommt.

5.7 Die Idee der Mehrgitterverfahren

Eine verwandte Idee liegt dem Mehrgitterverfahren zugrunde, welches in vielen Integrationsproblemen verwendet wird. Eine Variante wollen wir kurz besprechen. Hier teilt man zunächst das Gesamtintervall in N gleich große Teilintervalle $[t_i, t_{i+1}]$ ein und berechnet in jedem Intervall die einfache Trapezregel T_i . Liegt der Fehler unter der Toleranzschwelle, so wird das Teilergebnis T_i akzeptiert. Wenn nicht, dann wird das Intervall $[t_i, t_{i+1}]$ halbiert und in jedem der kleineren Intervalle die Trapezregel berechnet. Die Intervallhalbierung wird so lange fortgeführt, bis der Fehler akzeptiert werden kann.

Als Fehlerschätzer wird häufig die einfache Simpsonregel benutzt, welche ja um zwei Ordnungen genauer als die Trapezregel ist.

Ein zugehöriges Programm kann leicht rekursiv definiert werden (vgl. Hanke-Bourgeois, Abschnitt 42).