Projekt 16: summierte Trapezregel und adaptive Quadratur

Ziel sind: besseres Verständnis der Eigenschaften der summierten Trapezregel, Grundzüge von adaptiven Quadraturstrategien.

1. **a)** Geben Sie eine Funktion $f \in C^\infty(\mathbb{R})$ und eine Konstante $c > 0$ an, so daß für die summierte Trapezregel $T(h)$ gilt: $\left|\int_0^1 f(x)\,dx - T(h)\right| \geq ch^2$.

b) Zeigen Sie: Die summierte Simpsonregel entsteht in der Spalte $n = 1$ bei Romberg-Extrapolation der summierten Trapezregel für $h_i = \frac{b-a}{2^i}$, $i = 0, \ldots$.

2. Wie in Aufg. 1 gezeigt, kann man bei der summierten Trapezregel keine Konvergenz erwarten, die besser als $O(h^2)$ ist. Eine Ausnahme bildet die Integration von *periodischen* Funktionen, wie wir jetzt zeigen.

Betrachten Sie die summierte Trapezregel auf dem Intervall $[0, 2\pi]$ angewandt auf trigonometrische Polynome.

a) Sei $f \in C^{2m+2}(\mathbb{R})$ und 2π-periodisch. Zeigen Sie: die Trapezregel $T(h)$ mit Schrittweite $h = \frac{2\pi}{N}$ $(N \in \mathbb{N})$ erfüllt

$$\left|\int_0^{2\pi} f(x)\,dx - T(h)\right| \leq C_m h^{2m+2} \|f^{(2m+2)}\|_{C([0,2\pi])}$$

für eine Konstante $C_m > 0$, die nur von $m \in \mathbb{N}_0$ abhängt.

b) Werten Sie die Integrale $\int_0^{2\pi} f(x)\,dx$ mittels der summierten Trapez- und Simpsonregel (dies entspricht einem Extrapolationsschritt) für $f_1(x) = e^x$ und $f_2(x) = \sin^2 x$ aus. Was beobachten Sie? Wann ist Extrapolation sinnvoll?

c) Zeigen Sie: Die summierte Trapezregel mit Schrittweite $h = \frac{2\pi}{N}$ ist exakt für trigonometrische Polynome vom Grad $n < N$, d.h.

$$\int_0^{2\pi} f(x)\,dx - T(h) = 0 \qquad \forall f(x) = \sum_{k=0}^{n} a_n \sin nx + b_n \cos nx$$

Hinweis: Betrachten Sie die summierte Trapezregel angewandt auf $x \mapsto e^{\mathbf{i}kx}$ für $-n \leq k \leq n$.

d) Zeigen Sie folgendes Approximationsresultat: Für $h = \frac{2\pi}{N}$ und jedes $f \in C([0, 2\pi])$ gilt

$$\left|\int_0^{2\pi} f(x)\,dx - T(f)\right| \leq 4\pi \inf_{p \in T_{N-1}} \|f - p\|_{C([0,2\pi])},$$

wobei T_{N-1} der Raum der trigonometrischen Polynome from Grad $N - 1$ ist.

e) Zeigen Sie, daß für $f(x) = e^{\sin x}$ die summierte Trapezregel exponentiell (in N) konvergiert, d.h. es existieren $C > 0$, $q \in (0, 1)$ so daß

$$\left|\int_0^{2\pi} f(x) - T(h)\right| \leq Cq^N, \qquad h = \frac{2\pi}{N}.$$

Plotten Sie den Quadraturfehler der summierten Trapezregel und der summierte Simpsonregel für $N = 2, \ldots, 11$. Es gilt: $\int_0^{2\pi} e^{\sin x}\,dx \approx 7.95492652101284527 45$. Beschreiben Sie Ihr Ergebnis.

3. Sei $\Delta = \{x_0, \ldots, x_n\}$ ein Gitter auf dem Intervall $[x_0, x_n]$. Sei $h_i := x_{i+1} - x_i$ die Länge des i-ten Teilintervalls $I_i = (x_i, x_{i+1})$. Mit $I_\Delta : C([x_0, x_n]) \to S^1(\Delta)$ bezeichnen wir den nodalen Interpolationsoperator, d.h. $I_\Delta f \in S^1(\Delta)$ ist gekennzeichnet durch die Bedingungen:

$$(I_\Delta f)|_{I_i} \in \mathcal{P}_1, \qquad (I_\Delta f)(x_i) = f(x_i) \qquad \forall i.$$

Für ein Intervall $[a, b]$ definieren wir weiter $\|u\|_{L^1([a,b])} := \int_a^b |u(x)|\, dx$. Im folgenden bezeichnet $C > 0$ immer eine Konstante, die nicht von Δ, f und i abhängt.

a) Zeigen Sie: Ist $f|_{I_i} \in C^1(I_i) \cap C(\overline{I}_i)$ und gilt $\|f'\|_{L^1(I_i)} < \infty$, so ist:

$$\begin{aligned} \|(I_\Delta f)'\|_{L^1(I_i)} &\le \|f'\|_{L^1(I_i)} \\ \|f - I_\Delta f\|_{L^1(I_i)} &\le C h_i \|f'\|_{L^1(I_i)}. \end{aligned}$$

b) Zeigen Sie: Ist $f|_{I_i} \in C^2(\overline{I_i})$, so gilt

$$\|f - I_\Delta f\|_{L^1(I_i)} \le C h_i^2 \|f''\|_{L^1(I_i)}.$$

4. (graduierte Gitter) Sei $f(x) = x^\alpha$, $\alpha \in (0, 1)$, $I = (0, 1)$. $I_\Delta : C([0, 1]) \to S^1(\Delta)$ bezeichnet wie in Aufg. 3 den nodalen Interpolationsoperator.

a) Sei $\Delta_{unif} = \{i/n, i = 0, \ldots, n\}$ ein uniformes Gitter. Zeigen Sie:
$\|f - I_{\Delta_{unif}} f\|_{L^1(I)} \le C n^{-(\alpha+1)}$ für ein $C > 0$, das nicht von n abhängt.

b) Sei Δ_{unif} das uniforme Gitter aus Teilaufg. a). Zeigen Sie:
$\|f - I_{\Delta_{unif}} f\|_{L^1(I)} \ge C n^{-(\alpha+1)}$ für ein $C > 0$, das nicht von n abhängt.

c) Sei $\Delta_{grad} = \{(i/n)^\beta, i = 0, \ldots, n\}$ mit $\beta = 2/\alpha$.
Zeigen Sie: $\|f - I_{\Delta_{grad}} f\|_{L^1(I)} \le C n^{-2}$ für ein $C > 0$.

d) Progammieren Sie die summierte Trapezregel für allgemeine Gitter Δ. Für $f(x) = x^\alpha$, $\alpha = 0.1$, und $n = 2^i$, $i = 1, \ldots, 12$ plotten Sie doppelt logarithmisch den Quadraturfehler der summierten Trapezregel gegen die Anzahl n der Teilintervalle im Fall des uniformen Gitters Δ_{unif} und des graduierten Gitters Δ_{grad}. Welche Konvergenz (Fehler gegen n) beobachten Sie? Erklären Sie.

Hinweis für c): Zeigen Sie $h_i \le C n^{-1} x_i^{1-1/\beta}$ ($i \ge 1$, $C > 0$ geeignet) und verwenden Sie

$$\|f - I_\Delta f\|_{L^1(I_0)} \le C h_0 \|f'\|_{L^1(I_0)}, \qquad \|f - I_\Delta f\|_{L^1(I_i)} \le C h_i^2 \|f''\|_{L^1(I_i)}, \quad i \ge 1.$$

5. Aufg. 4 zeigt, daß die summierte Trapezregel bei richtiger Wahl des Gitters das optimale Konvergenzverhalten $O(N^{-2})$ (Fehler gegen Funktionsauswertungen) erreichen kann. In der Praxis ist ein *adaptiver* Algorithmus, der automatisch das Gitter erzeugt, besser. Eine Rohfassung eines adaptiver Algorithmus ist in Fig. 1 dargestellt (hier bezeichnet $T([a, b])$ die Trapezregel für das Intervall $[a, b]$ und $S([a, b])$ die Simpsonregel). Ein sinnvolles Abbruchkriterium ist

$$|I_1 - I_2| \le \tau \text{ oder } |I_1| \le \tau I, \tag{1}$$

wobei I eine positive Zahl von der Größenordnung des gesuchten Integralwertes ist.

a) Programmieren Sie einen adaptiven Quadraturalgorithmus zur Auwertung von $\int_a^b f(x)\, dx$, der auf den obigen Ideen beruht. Eingabeparameter für die Quadraturroutine sollen τ, a, b, I. Ausgabeparameter sollen den Integralwert, der geschätze Fehler und die Anzahl benötigter Teilintervalle sein.
Hinweis: in MATLAB kann es notwendig sein, mit `set(0,'RecursionLimit',10000)` die Rekursionstiefe zu erhöhen.

b) Verwenden Sie Ihren Quadraturalgorithmus zur Bestimmung von $\int_0^1 x^{0.1}\,dx$. Verwenden Sie als Toleranzen $\tau = 10^{-n}$, $n = 1, \ldots, 8$ und $I = 1$. Plotten Sie doppelt logarithmisch den geschätzen und den tatsächlichen Quadraturfehler über den Anzahl benötigter Teilintervalle. Plotten Sie im gleichen Bild auch die Kurve $N \mapsto N^{-2}$.

c) Ihr adaptiver Algorithmus "akzeptiert" den Wert der Trapezregel. Sinnvoller ist es, den Wert der Simsponregel zu akzeptieren. Modifizieren Sie Ihr Programm entsprechend und wiederholen Sie die Rechnung aus Teilaufg. b).

6. 1D Quadraturregeln lassen sich in natürlicher Weise zu Quadraturregeln für Rechtecken (in 2D) und Quadern (in 3D) erweitern. Sei hierzu $Q_n^{1D}(f) = \sum_{i=0}^n w_i f(\xi_i) \approx \int_0^1 f(x)\,dx$ eine 1D Quadraturegel. Dann definiert

$$Q_n^{2D}(f) := \sum_{i=0}^n \sum_{j=0}^n w_i w_j f(\xi_i, \xi_j)$$

eine Quadraturregel zur Approximation von $\int_0^1 \int_0^1 f(x,y)\,dx\,dy$.

a) Sei Q_n^{1D} exakt für Polyome vom Grad m. Zeigen Sie: Q^{2D} ist exakt für alle $f \in \mathcal{Q}_m := \operatorname{span}\{x^i y^j \mid 0 \le i, j \le m\}$.

b) Erzeugen Sie mittels Q_n^{2D} eine "summierte" Quadraturregel für die Quadratur auf $\Omega := [a,b] \times [a,b]$. Sei hierzu $\mathcal{T} = \{[x_i, x_{i+1}] \times [y_i, y_{i+1}] \mid i = 1, \ldots, N\}$ ein "Gitter" auf $[a,b] \times [a,b]$, d.h. die Vereinigung der Rechtecke $[x_i, x_{i+1}] \times [y_i, y_{i+1}]$ überdeckt Ω und die offenen Rechtecke $(x_i, x_{i+1}) \times (y_i, y_{i+1})$ sind paarweise disjunkt. Sei $h_i := \operatorname{diam}[x_i, x_{i+1}] \times [y_i, y_{i+1}]$. Zeigen Sie: Ihre summierte Quadraturregel $Q_{\mathcal{T}}$ erfüllt für eine geeignete Konstante C, die nicht von f abhängt

$$\left| \int_\Omega f - Q_{\mathcal{T}}(f) \right| \le C h^{m+1} \sum_{i=1}^N \operatorname{meas}([x_i, x_{i+1}] \times [y_i, y_{i+1}]) \sum_{\substack{(\alpha_1,\alpha_2) \in \mathbb{N}_0^2 \\ \alpha_1 + \alpha_2 = m+1}} \|\partial_x^{\alpha_1} \partial_y^{\alpha_2} f\|_{C([x_i, x_{i+1}] \times [y_i, y_{i+1}])}$$

c) Programmieren Sie Ihre summierte Quadraturregel für $\Omega = [0,1]^2$ und uniforme Zerlegungen, d.h. zu Eingabeparameter $M \in \mathbb{N}$ ist $N = M^2$ und jedes Rechteck von $\mathcal{T}$ ist ein Quadrat der Kantenlänge $h = 1/M$. Verwenden Sie als 1D Quadraturroutine die 1D Gaußquadratur mit n Punkten. *Hinweis:* die Routine $[x, w] =$ `gauleg(n)`, die Sie auf `http://www.math.tuwien.ac.at/~melenk/teach/numerik_WS0708/projekte` finden, liefert die n Gaußpunkte und Gaußgewichte für das Intervall $[-1, 1]$.

Testen Sie Ihr Programm für $n = 1, 2, 3$ und $M = 2^i$, $i = 1, \ldots, 5$ zur Bestimmung der Integrale

$$\int_0^1 \int_0^1 e^{x+y}\,dx\,dy = (e-1)^2, \qquad \int_0^1 \int_0^1 \sqrt{x+y}\,dx\,dy = \frac{8}{15}(-1 + 2\sqrt{2});$$

d) Formulieren Sie einen adaptiven Algorithmus für die Auswertung von Integralen der Form $\int_0^1 \int_0^1 f(x,y)\,dx\,dy$. Gehen Sie vor wie bei der 1D adaptiven Quadratur. Verwenden Sie zwei verschiedenen Gaußregeln zum Schätzen des Fehlers.

```
adapt(f, a, b, τ)
I_1 := T([a,b])       %Approximation an ∫_a^b f(x) dx
I_2 := S([a,b])    %‘‘bessere’’ Approximation an ∫_a^b f(x) dx
if Abbruchkriterium erfüllt return(I_1)
else { % gewünschte Genauigkeit nicht erreicht → unterteile [a,b] in [a,m], [m,b]
      m := (a+b)/2
      return( adapt(f, a, m, τ/2) + adapt(f, m, b, τ/2))
}
```

Abbildung 1: Grundform eines adaptive Algorithmus