Fachbereich Mathematik & Informatik
Freie Universität Berlin
Prof. Dr. Carsten Gräser, Lasse Hinrichsen

6. Übung zur Vorlesung

Computerorientierte Mathematik II

SoSe 2020

http://numerik.mi.fu-berlin.de/wiki/SS_2020/CoMaII.php

Abgabe: Fr., 19. Juni 2020, 12:00 Uhr

1. Aufgabe (2 TP)
Seien $\lambda_0, \cdots, \lambda_n$ die Gewichte der Newton-Côtes Formeln bezüglich des Referenzintervalls $[0, 1]$, d.h. es gelte

$$\lambda_i = \int_0^1 L_i(x)dx,$$

wobei L_i das i-te Lagrangepolynom bezüglich der Knotenpunkte $0 \leq x_0 < \cdots < x_n \leq 1$ bezeichne.
Angenommen, es gilt für alle $f \in P_m$

$$I_n(f) = \sum_{i=0}^{n} f(x_i)\lambda_i = \int_0^1 f(x)dx,$$

(d.h. die Quadraturformel ist exakt für Polynome vom Grad m auf dem Referenzinterval).
Zeigen Sie, dass auch die transformierte Quadraturformel

$$I_{[a,b]}(f) = (b-a)\sum_{i=0}^{n} f(a + (b-a)x_i)\lambda_i$$

exakt für alle Polynome vom Grad m auf einem beliebigen Interval $[a, b]$ ist, d.h.

$$I_{[a,b]}(f) = \int_a^b f(x)dx, \quad \forall f \in P_m.$$

2. Aufgabe (3 TP + 3 TP)
Sei $f \in C([a, b])$ gegeben. Wir möchten $I(f) := \int_a^b f(x)\,\mathrm{d}x$ mit Hilfe einer Quadraturformel $I_1(f)$ approximieren, die von einer Gitterweite h abhängt. Um den Fehler $e := |I(f) - I_1(f)|$ abschätzen zu können, verwenden wir eine zweite Quadraturformel $I_2(f)$ und den Fehlerschätzer $\widetilde{e} := |I_1(f) - I_2(f)|$.
Wir machen die folgenden Annahmen:

- Die Formel I_1 hat höchstens Ordnung $p_1 > 0$, es gilt also für ein $\alpha \in \mathbb{R}_{>0}$

$$\alpha h^{p_1} \leq |I(f) - I_1(f)|.$$

- Die Formel I_2 hat mindestens Ordnung $p_2 > p_1$, es gilt also für ein $\beta \in \mathbb{R}_{>0}$

$$\beta h^{p_2} \geq |I(f) - I_2(f)|.$$

Zeigen Sie:

a) Mit der Effizienzspanne $c := \frac{\beta}{\alpha}$ sowie mit $q := p_2 - p_1$ gilt

$$\frac{\widetilde{e}}{1 + ch^q} \leq e \leq \frac{\widetilde{e}}{1 - ch^q}.$$

b) Es gilt

$$\frac{|e - \widetilde{e}|}{e} \in \mathcal{O}(h^q) \text{ für } h \to 0.$$

3. Aufgabe (2 PP + 2 PP + 2 PP)
Wir möchten die summierten Newton–Côtes-Formeln implementieren und deren Fehlerverhalten untersuchen.

a) Schreiben Sie eine Funktion `summed_newton_cotes(a, b, f, k, n)`, die die Newton–Côtes-Formel $S_n^{(k)}$ implementiert. Dabei bestimmen `a` und `b` das Integrationsintervall, `f` sei die zu integrierende Funktion und `n` die Anzahl der Intervalle, auf denen die `k`-te Newton–Côtes-Formel (mit $k \in \{1, 2, 6\}$) angewendet wird. In der Variablen `S` werde hierbei der Wert der Quadraturformel zurückgegeben und in `A` die Anzahl der durchgeführten `f`-Auswertungen.

   Hinweis: Sie können die entsprechenden Quadraturgewichte dem Vorlesungsskript entnehmen.

b) Wir möchten die Approximation des Integrals $I := \int_0^\pi \sin(x)\,\mathrm{d}x$ untersuchen. Erstellen Sie hierzu für die Funktion $f(x) = \sin(x)$ und für das Intervall $[a, b] = [0, \pi]$

   - eine Abbildung, in der die Fehlerkurven $|I - S_n^{(k)}|$ für $k \in \{1, 2, 6\}$ doppelt logarithmisch gegen $n \in \{1, \dots, 1000\}$ geplottet werden,
   - und eine Abbildung, in der die Anzahl der `f`-Auswertungen über den Quadraturfehler $|I - S_n^{(k)}|$ für $k \in \{1, 2, 6\}$ doppelt logarithmisch als Punkt-Plot dargestellt wird.

   Kommentieren Sie Ihre Ergebnisse insbesondere vor dem Hintergrund der unterschiedlichen Ordnung der Verfahren und der Glattheit von f.

c) Verfahren sie analog für $f(x) = \sqrt{x} + \sin(21\pi x)$ auf dem Intervall $[0, 1]$.

Hinweis: Es gilt $\int_0^\pi \sin(x)\,\mathrm{d}x = 2$ und $\int_0^1 \sqrt{x} + \sin(21\pi x)\,\mathrm{d}x = \frac{2}{21}\left(7 + \frac{1}{\pi}\right)$.

Allgemeine Hinweise

Die Punkte unterteilen sich in Theoriepunkte (TP) und Programmierpunkte (PP). Bitte beachten Sie die auf der Vorlesungshomepage angegebenen Hinweise zur Bearbeitung und Abgabe der Übungszettel, insbesondere der Programmieraufgaben.