Interdisziplinäres Zentrum
für Wissenschaftliches Rechnen (IWR)
Ruprecht–Karls–Universität Heidelberg

Dr. Peter Zaspel

Übungsblatt 4

17.05.2016 – 24.05.2016

Einführung in die Numerik SS 2016

Aufgabe 1. Trigonometrische Polynome (6 Punkte)
Man zeige, dass die Funktionen

$$\textbf{a. } \varphi_0(x) = \frac{1}{\sqrt{2\pi}}, \quad \textbf{b. } \varphi_k(x) = \frac{1}{\sqrt{\pi}}\cos(kx), \quad \textbf{c. } \psi_k(x) = \frac{1}{\sqrt{\pi}}\sin(kx), \quad k = 1, \dots, n$$

ein Orthonormalsystem des Vektorraumes $T_n \subset C[0, 2\pi]$ der trigonometrischen Polynome vom Grad $\leq n$ bzgl. des Skalarprodukts

$$(u, v) = \int_0^{2\pi} u(x)v(x)\mathrm{d}x$$

bilden.

Aufgabe 2. Interpolationskoeffizienten (4 Punkte)
Zeigen Sie, dass die trigonometrischen Ausdrücke

$$q_o(x) = \frac{a_0}{2} + \sum_{l=1}^{m} (a_l cos(lx) + b_l sin(lx)) \quad \text{für } n = 2m + 1$$

bzw.

$$q_e(x) = \frac{a_0}{2} + \sum_{l=1}^{m} (a_l cos(lx) + b_l sin(lx)) + \frac{a_m}{2} cos(mx) \quad \text{für } n = 2m$$

den Interpolationsbedingungen

$$q(x_k) = y_k \quad \forall k = 0, ..., n-1$$

genau dann gen[ge, wenn für die Koeffizienten gilt

$$a_l = \frac{2}{n} \sum_{m=0}^{n-1} y_m cos(mx_l) = \frac{2}{n} \sum_{m=0}^{n-1} y_m cos(\frac{2\pi km}{n})$$

$$b_l = \frac{2}{n} \sum_{m=0}^{n-1} y_m sin(mx_l) = \frac{2}{n} \sum_{m=0}^{n-1} y_m sin(\frac{2\pi km}{n}),$$

d.h. beweisen Sie Satz 2.3.3 aus der Vorlesung.

Aufgabe 3. Berechnungskomplexität der FFT (4 Punkte)

Zeigen Sie, dass sich der Aufwand zur Bewältigung der vollständigen Fourier-Synthese durch $\frac{1}{2}nlog(n)$ Multiplikationen abschätzen lässt, d.h. beweisen Sie Satz 2.3.5 aus der Vorlesung.

Aufgabe 4. Trigonometrische Interpolation (6 Punkte)
Die $2\pi-$periodische Funktion f sei durch

$$f(x) = \begin{cases} x: & 0 < x < 2\pi \\ \pi: & x = 0 \end{cases}$$

definiert. Bestimme das trigonometrische Interpoaltionspolynom $t_3(x)$ zu den Stützstellen $x_k = \frac{k\cdot\pi}{2},\ k = 0, ..., 3$. Man stelle das Ergebnis graphisch dar.

Aufgabe 5. Praktische Aufgabe zur Trigonometrischen Interpolation (1+2+2+2+2+1 Punkte)
Das Ziel ist es, ein Programm zu schreiben, mit dem die trigonometrische Interpolante q_e aus Aufgabe 2 im Fall von n vielen Stützstellen (n gerade, daher q_e) berechnet wird und anschließend ausgewertet wird. Zur Berechnung der Koeffizienten a_l, b_l in der Interpolante q_e sollen zunächst die Koeffizienten β_l mittels der Diskreten Fourier-Transformation aus den Funktionswerten y_l an den Stützstellen x_l bestimmt werden. Anschließend soll die Relation

$$a_0 = 2\beta_0, \quad a_l = \beta_l + \beta_{n-l}, \quad b_l = i(\beta_l - \beta_{n-l}), \quad a_{n/2} = 2\beta_{n/2} \qquad \forall l = 1, ..., n/2-1 \tag{1}$$

genutzt werden, um aus den berechneten Koeffizienten β_l die Interpolante q_e aufzubauen. Diese wird schließlich ausgewertet.

Als Basis für die Programmieraufgabe dient der Quelltext `dft.cpp` der im Moodle-Kurs heruntergeladen werden kann.

- **a.** Informieren Sie sich übers Internet, z.B. unter `http://en.cppreference.com/w/c/numeric/complex` über den Datentyp `double complex` zum Speichern komplexer Zahlen und über die dazugehörigen Funktionen wie `creal`, `cimag`, `conj`, `cexp`, ...
- **b.** Implementieren Sie die Funktion `adjoint_mat_vec_prod` im Quellcode, bei der die Operation $y = A^\star x$ mit $A \in \mathbb{R}^{n\times n}$ und $x, y \in \mathbb{R}^n$ ausgeführt wird.
- **c.** Ergänzen Sie die Funktion `fill_matrix_T` bei der die Matrix T_n aus der Vorlesung aufgestellt wird. Die Matrix wir als 2D-Array übergeben.
- **d.** Implementieren Sie die Funktion `convert_beta_to_a_b_even`, die die Transformation gemäß (1) durchführt.
- **e.** Schreiben Sie die Funktion `eval_sin_cos_expression_even`, die die Funktion q_e an der Stelle x (siehe Aufgabe 2) auswertet.
- **f.** Ergänzen Sie das Hauptprogramm unter Anwendungen der Routinen aus **b**, **c** und **d** so, dass für gegebene Funktionswerte y_l die Koeffizienten a_l, b_l berechnet werden. Der Rest des Hauptprogrammes übernimmt es für sie, die ensprechenden Funktionen zur Auswertung und zur Ausgabe der Interpolante auszuführen.

g. Nutzen Sie Ihr Programm mit $n = 16, 32, 64$ um die Funktion aus Aufgabe 4 zu interpolieren und plotten Sie die Ergebnisse mit Gnuplot.

Abgabe: 24.05.2016, 14:00-14:15 Uhr. (Mappen in der Vorlesung)