Universität Tübingen
Mathematisches Institut
Prof. Dr. Christian Lubich

Tübingen, den 08.12.2020

5. Übungsblatt zur Numerik

Aufgabe 17: Das Polynom p sei gegeben in seiner Entwicklung nach Tschebyscheff-Polynomen,

$$p(x) = \frac{1}{2}c_0 + c_1 T_1(x) + \ldots + c_n T_n(x).$$

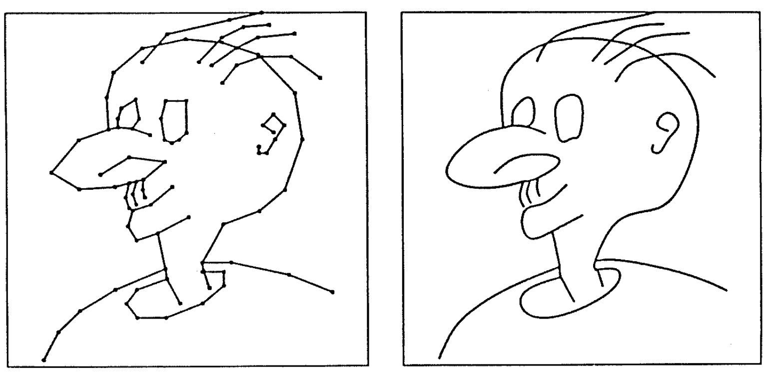
Falls

$$\begin{aligned} d_k &= c_k + 2x\, d_{k+1} - d_{k+2} \quad (k = n, n-1, \ldots, 0), \quad d_{n+1} = d_{n+2} = 0, \\ e_k &= d_k + 2x\, e_{k+1} - e_{k+2} \quad (k = n, n-1, \ldots, 1), \quad e_{n+1} = e_{n+2} = 0, \end{aligned}$$

dann ist bekanntlich $p(x) = \frac{1}{2}(d_0 - d_2)$ (Clenshaw). Zeigen Sie, dass außerdem gilt:

$$p'(x) = e_1 - e_3.$$

Aufgabe 18: Gegeben sei eine Menge von Punkten $(x_j, y_j)_{1\le j\le n} \in \mathbb{R}^2$, die beispielsweise aus einer Vektorgrafik oder einem gescannten Bild stammen. Erfinden Sie einen Algorithmus, der die durch lineare Interpolation dieser Punkte entstehenden Kanten glättet.



Aufgabe 19:

(a) Berechnen Sie mit dem Newton-Schema das Interpolationspolynom $p(x)$ zu folgenden Daten:

x_j	0	1	2	3	4
y_j	-1	-1	-7	-7	35

Berechnen Sie mit diesem Newton-Schema auch alle Ableitungen von p an der Stelle $x = -1$.

(b) Stellen Sie das Polynom $p(x) = x^3 + 2x^2 + x + 3$ mit Hilfe des Newtontableaus in der Form $p(x) = a(x-1)^3 + b(x-1)^2 + c(x-1) + d$ dar.

Aufgabe 20: Der kubische Spline minimiert $\int_a^b [s''(x)]^2 dx$. Minimiert man allgemeiner

$$\int_a^b [s''(x)]^2 dx + \lambda^2 \int_a^b [s'(x)]^2 dx,$$

erhält man einen Spline, bei dem zusätzlich die Länge minimiert wird. Dies entspricht physikalisch einem Balken unter Zug. Für einen derartigen Spline ergibt sich der Ansatz

$$s_i(x) = a_i + b_i x + c_i e^{\lambda x} + d_i e^{-\lambda x}.$$

Erklären Sie den Ansatz und definieren Sie natürliche, periodische und eingespannte Splines unter Zug.

Programmieraufgabe 3: Implementieren Sie die Interpolation mit eingespannten kubischen Splines zu gegebenen Wertepaaren (x_i, y_i), $0 \le i \le n$, und gegebenen Ableitungen $s'(x_0) = v_0$, $s'(x_n) = v_n$ für den Fall äquidistanter Stützstellen. Gehen Sie wie folgt vor:

(a) Lösen Sie das in der Vorlesung hergeleitete lineare Gleichungssystem $Av = b$ zur Bestimmung der fehlenden Ableitungen $v = (v_1, \ldots, v_{n-1})^{\mathrm{T}}$ mit Hilfe des Matlab-Befehls `v = A\b`. Schreiben Sie dazu Funktionen der Gestalt

```
function A = MatrixAufstellen(n)        function b = rechteSeite(xv,yv,v0,vn)
        ⋮                                        ⋮
end                                     end
```

Dabei stehen `xv` und `yv` für die gegebenen Stützstellen und die zugehörigen Werte.

(b) Schreiben Sie eine Funktion `SplineAuswerten(x,xv,yv,vv)`, welche den Spline an der Stelle $x \in [x_{i-1}, x_i]$ gemäß der in der Vorlesung hergeleiteten Darstellung von $s_i(x)$ auswertet. Berechnen Sie dazu den zu x gehörigen Index i in einer einzigen Programmzeile (z.B. mit dem Befehl `floor`).

Testen Sie Ihre Programme in einem Skript `mainSI.m` anhand des Beispiels aus Aufgabe 14.

Wenden Sie Ihr Programm außerdem auf die Funktion $f(x) = \sin(x)$ auf dem Intervall $[0, \pi]$ an. Erstellen Sie dafür einen Konvergenzplot für verschiedene Schrittweiten h und plotten Sie diesen im einem loglog Plot gegen den maximalen Fehler auf dem Intervall. Verwenden Sie $h = \frac{\pi}{2^j}, j = 1, ..., 7$. Plotten Sie weiterhin die Referenzlinien $(h_j, h_j^k)_{j=1,\ldots,7}$ für verschiedene k in dasselbe Schaubild. Welche Referenzlinie stimmt am besten mit dem beobachteten Fehler überein? Ist dies im Einklang mit der in der Vorlesung besprochenen Theorie?

Besprechung der Übungsaufgaben am 15. und 16. Dez. 2020.
Abgabe der Programmieraufgabe bis 22.12.2020 an progtutor@na.uni-tuebingen.de.
Abgabe in einem Zip-Ordner mit Name im Format: PA3_Nachname1_Nachname2_Nachname3.