Prof. Dr. Karsten Urban
Dipl.-Math. oec. Klaus Stolle
Institut für Numerische Mathematik
Universität Ulm

SoSe 2017
Blatt 12
10.07.2017

Angewandte Numerik 1

Besprechung in den Tutorien in der Woche vom 17.07.2017 bis 21.07.2017

Für dieses Übungsblatt gibt es 12 Theorie- und 23 Matlab-Punkte, sowie 44 Theorie- und 21 Matlab-Zusatzpunkte. Punkte, die mit einem * gekennzeichnet sind, sind Zusatzpunkte.
Somit sind für das Bestehen der Vorleistung insgesamt **121,8 Theorie- und 119,4 Matlabpunkte** nötig.

Bitte melden Sie sich bis **spätestens Montag, 17.07.2017** zur Vorleistung an. Die Anmeldung ist bereits jetzt möglich.

Die beiden **Klausuren** finden voraussichtlich am **02.08.2017** und am **04.10.2017** statt. Die genaue Uhrzeit und die Raumeinteilung werden über die Homepage bekannt gegeben. Beachten Sie die Homepage bitte auch wegen möglicherweise kurzfristig veröffentlichter Hinweise zu den Klausuren.

Als **Hilfsmittel zu den Klausuren** ist nur ein eigenhändig beidseitig beschriebenes DIN-A4-Blatt (keine Kopien oder Ausdrucke) zugelassen. Insbesondere sind Taschenrechner, Handys, Smartphones, Smartwatches und Ähnliches nicht erlaubt.

Aufgabe 46 *(Fortsetzung Aufgabe 42: Dividierte Differenzen)* (4T+2T Punkte)

In Aufgabe 42 vom letzten Übungsblatt haben Sie $\sqrt{3}$ mit Hilfe des Interpolationspolynoms $P = P(f|x_0, \dots, x_3)$ zu $f(x) = 3^x$ und den Stützstellen

x_0	x_1	x_2	x_3
-2	-1	0	1

näherungsweise berechnet. Sie haben dazu die Lagrange-Darstellung des Interpolationspolynoms und das Schema von Aitken-Neville verwendet.

a) Stellen Sie jetzt P als Newton'sches Interpolationspolynom mit Hilfe der Methode der dividierten Differenzen explizit dar und werten Sie anschließend das Polynom an der richtigen Stelle x aus.

b) Erweitern Sie wie in Aufgabe 42 auch diese Berechnung mit den dividierten Differenzen um die zusätzliche Stützstelle $x_4 = 2$.

Hinweis: Rechnen Sie in der ganzen Aufgabe mit Brüchen.

Aufgabe 47 *(Programmieraufgabe: Horner-Schema)* (3M+2M Punkte)

Polynome $p(x) = \sum_{k=0}^{n} a_k x^k = a_0 + a_1 x + \cdots + a_n x^n$ (mit $x^0 := 1$) können mit Hilfe des Horner-Schemas effizient ausgewertet werden. Dazu wird das Polynom $p(x) = \sum_{k=0}^{n} a_k x^k$ umgeformt zu

$$p(x) = \sum_{k=0}^{n} a_k x^k = (\dots((x a_n + a_{n-1})\, x + a_{n-2})\, x + \dots a_1)\, x + a_0.$$

a) Schreiben Sie eine Matlab-Funktion `pt = hornerSchema(a, t)` zur effizienten Auswertung des Polynoms $p(x) = \sum_{k=0}^{n} a_k x^k$ mit den Koeffizienten $\mathtt{a} = (a_0, a_1, a_2, \ldots, a_n)^T \in \mathbb{R}^{n+1}$. Dabei enthält der Vektor $\mathtt{t} = (t_1, t_2, \ldots, t_m)^T$ mit $t_k \in \mathbb{R}$, $(k = 1, \ldots, m)$ die Stellen, an denen das Polynom p ausgewertet werden soll. Der Ergebnisvektor $\mathtt{pt} = (pt_1, pt_2, \ldots, pt_m)^T$ soll die Werte des Polynoms an den Stellen `t` enthalten, also $pt_k = p(t_k)$, $(k = 1, \ldots, m)$.

b) Vergleichen Sie Ihre Funktion mit der Matlab-Funktion `polyval`. Diese Funktion verwendet auch das Hornerschema. Schreiben Sie hierzu ein Matlab-Skript `testHornerSchema`, das Ihre Funktion `hornerSchema(a,x)` und die Matlab-Funktion `polyval` für verschiedene Polynome aufruft und die Ergebnisse vergleicht.
Tipp: Die Matlab-Funktion `flipud` könnte Ihnen beim Aufruf der beiden Funktionen helfen.

Aufgabe 48 *(Programmieraufgabe: Schema der Dividierten Differenzen)* (4M+4M+2M Punkte)

a) Schreiben Sie eine Matlab-Funktion `c = divDiff(x, f)`, die mittels dividierter Differenzen den Vektor $\mathtt{c} = (c_0, \ldots, c_n)^T$ der Koeffizienten des Newton-Interpolationspolynoms zu den Daten (x_j, f_j), $j = 0, \ldots, n$, berechnet. $\mathtt{x} = (x_0, \ldots, x_n)^T$ sind dabei die Stützstellen, $\mathtt{f} = (f_0, \ldots, f_n)^T$ die Funktionswerte an diesen Stützstellen.

b) Schreiben Sie eine Matlab-Funktion `v = evalNewtonpolynom(x, c, t)`, die das Newton-Polynom zu den Stützstellen $\mathtt{x} = (x_0, \ldots, x_n)^T$ und den Koeffizienten $\mathtt{c} = (c_0, \ldots, c_n)^T$ mittels eines modifizierten Horner-Schemas in den Punkten $\mathtt{t} = (t_1, \ldots, t_m)^T$, $m \in \mathbb{N}$ auswertet.

c) Testen Sie Ihre Funktionen am Beispiel aus Aufgabe 46 mit 4 und mit 5 Stützstellen.

Aufgabe 49 *(Programmieraufgabe: Fortsetzung Aufgabe 45: Beispiel von Runge mit Dividierten Differenzen)* (3M+2T+(3M+2T)+(2M+2T)+(3M*+2T*) Punkte)

In Aufgabe 45 vom letzten Übungsblatt haben wir die Entwicklung des Interpolationsfehlers bei zunehmender Anzahl n der Stützstellen anhand des sogenannten Gegenbeispiels von Runge genauer untersucht. Wir haben dabei das Schema von Aitken-Neville verwendet, obwohl das Interpolationspolynom an „vielen“ Stellen ausgewertet werden musste.
In dieser Aufgabe wollen wir Aufgabe 45 um das Verfahren der Dividierten Differenzen erweitern und auch die jeweils benötigten Laufzeiten untersuchen.

a) Erweitern Sie Ihr Matlab-Skript `Aufgabe45` um das Verfahren der Dividierten Differenzen. Sie dürfen dazu Ihre Matlab-Funktionen `c = divDiff(x, f)` und `v = evalNewtonpolynom(x, c, t)` aus Aufgabe 48 verwenden. Berechnen Sie also für $n = 1, \ldots, 200$ äquidistante Stützstellen $x = (x_0, \ldots, x_n)^T$ jeweils das Interpolationspolynom der Funktion $f : [-5, 5] \to \mathbb{R}$, $f(x) = \frac{1}{1+x^2}$ mit dem Verfahren der Dividierten Differenzen. Werten Sie dieses Interpolationspolynom mit Ihrem modifizierten Horner-Schema an mindestens 500 Stellen aus. Berechnen Sie analog zu Aufgabe 45 jeweils den maximalen Interpolationsfehler und plotten Sie diesen in Abhängigkeit von der Anzahl der Stützstellen. Verwenden Sie dabei eine logarithmische Darstellung der y-Achse.

b) Vergleichen Sie Ihren Plot der maximalen Interpolationsfehler bei äquidistanten Stützstellen mit dem entsprechenden Plot für das Schema von Aitken-Neville aus Aufgabe 45. Was fällt Ihnen auf? Interpretieren Sie Ihr Schaubild. Können Sie Ihre Beobachtung erklären?

c) Erweitern Sie Ihr Matlab-Skript `Aufgabe45` auch um das mit dem Verfahren der Dividierten Differenzen berechnete Interpolationspolynom zu den auf das Intervall $[-5, 5]$ transformierten Tschebyscheff-Knoten. Plotten Sie wiederum den Interpolationsfehler in Abhängigkeit von der Anzahl der Stützstellen und vergleichen Sie diesen mit dem entsprechenden durch das Schema von Aitken-Neville berechneten Interpolationsfehler. Was fällt Ihnen auf? Haben Sie dieses Ergebnis erwartet? Interpretieren Sie das Ergebnis und erklären Sie Ihre Beobachtungen!

d) Mit Hilfe der Leja-Sortierung der Stützstellen kann der bei Verwendung des Verfahrens der Dividierten Differenzen aufgetretene Interpolationsfehler reduziert werden.

   Erweitern Sie Ihr Matlab-Skript `Aufgabe45` erneut und verwenden Sie zusätzlich sowohl die mit Hilfe der Leja-Sortierung umsortierten äquidistanten Stützstellen als auch die mit Hilfe der Leja-Sortierung umsortierten Tschebyscheff-Stützstellen (Aufruf jeweils `x = leja(x);`). Was stellen Sie nun fest? Woran liegt das?

e) Vergleichen Sie die Laufzeiten des in Aufgabe 45 verwendeten Schemas von Aitken-Neville mit den Laufzeiten des in dieser Aufgabe verwendeten Schemas der Dividierten Differenzen einschließlich der anschließenden Auswertung mit dem modifizierten Hornerschema aus Aufgabe 48. Plotten Sie die Laufzeiten der Verfahren jeweils in Abhängigkeit von der Anzahl der Stützstellen und achten Sie dabei auf eine geeignete Skalierung der Achsen.

   Berücksichtigen Sie bei Ihrem Laufzeitenvergleich sowohl die nicht vektorisierte als auch Ihre vektorisierten Varianten des Schemas von Aitken-Neville.

   Was fällt Ihnen auf? Haben Sie dieses Ergebnis so erwartet? Welches Verfahren ist das für diese Problemstellung geeignetere? Warum?

Achtung: Die Berechnungen dieser Aufgabe können durchaus etwas länger dauern.

Aufgabe 50 *(Programmieraufgabe: Zusammengesetzte Quadraturformeln)*

(10M*+6M*+(2M*+2T*) Punkte)

Sei $m \in \mathbb{N}$ und $-\infty < a < b < \infty$.
Wir wollen in dieser Aufgabe zusammengesetzte Quadraturformeln zur Approximation von

$$I(f) := \int_a^b f(x)\,dx$$

betrachten, und zwar die zusammengesetzte (summierte) linke Rechteckregel, die zusammengesetzte (summierte) Mittelpunktregel, die zusammengesetzte Trapezregel (Trapezsumme) und die zusammengesetzte Simpsonregel. Die Simpsonregel finden Sie in Beispiel 7.2.5.

a) Schreiben Sie vier Matlabfunktionen `intApprox = linkeRechteckSumme(f, a, b, m)`, `intApprox = mittelpunktSumme(f, a, b, m)`, `intApprox = trapezSumme(f, a, b, m)` und `intApprox = simpsonSumme(f, a, b, m)` zur näherungsweisen Berechnung des Integrals $I(f)$ mit den oben genannten Methoden. Die Funktionen erhalten jeweils den Integrand `f` als function handle, die Intervallgrenzen `a` und `b`, sowie die Anzahl `m` der Teilintervalle. Zurückgegeben wird jeweils der Näherungswert `intApprox` für das Integral .

b) Schreiben Sie ein Matlab-Skript, welches Ihre Funktionen für das Integral

$$I(f) \;:=\; \int_0^1 \frac{4}{x^2+1}\,dx \;=\; \pi$$

   testet. Approximieren Sie dazu für $m = 2^1, ..., 2^{12}$ mit allen vier Regeln das Integral und berechnen Sie jeweils den absoluten Fehler. Stellen Sie anschließend die jeweiligen Fehler in doppelt logarithmischer Skala in Abhängigkeit von m dar.

c) Erklären Sie das Ergebnis. Zeichnen Sie dazu geeignete Steigungsgeraden in Ihr Schaubild ein. War das Ergebnis so zu erwarten? Begründen Sie Ihre Aussage.

Aufgabe 51 *(Gewichte interpolatorischer Quadraturformeln)* (6T* Punkte)

Bestimmen Sie die Gewichte λ_i der interpolatorischen Quadraturformel

$$\hat{I}_2(f) \;=\; (b-a)\sum_{i=0}^{2} \lambda_i f(x_i) \;\approx\; \int_a^b f(x)\,dx$$

mit den Knoten

$$x_0 = \frac{3a+b}{4}, \quad x_1 = \frac{2a+2b}{4}, \quad x_2 = \frac{a+3b}{4}.$$

Hinweis: Approximieren Sie f durch das Interpolationspolynom $P(f|x_0, x_1, x_2)$ in der Lagrange-Darstellung und integrieren Sie anschließend dieses Interpolationspolynom.

Aufgabe 52 *(Fehlerordnung von Quadraturformeln)* (4T*+3T* Punkte)

Der Fehler einer Quadraturformel $\hat{I}(f)$ zur Approximation des Integrals $I(f)$ ist definiert als

$$E(f) := I(f) - \hat{I}(f)$$

a) Zeigen Sie, dass der Fehler linear ist, d. h. dass

$$E(\lambda f + \mu g) = \lambda E(f) + \mu E(g).$$

Hinweis: Verwenden Sie hierbei, dass für die Addition von Funktionen

$$(f+g)(x) := f(x) + g(x)$$

und für die Multiplikation von Funktionen mit Skalaren

$$(\lambda \cdot f)(x) := \lambda \cdot f(x)$$

gilt.

Bemerkung: Aufgrund der Linearität des Quadraturfehlers kann die Fehlerordnung einer Quadraturformel leicht über die Monome $m_i(x) = x^i \;\; (i = 0, 1, \ldots)$ getestet werden.

b) Welche Fehlerordnung hat die folgende Quadraturformel?

$$\hat{I}(f) \;=\; \frac{1}{2} f\left(\frac{1}{4}\right) + \frac{1}{2} f\left(\frac{3}{4}\right) \;\approx\; \int_0^1 f(x)\,dx$$

Aufgabe 53 *(Gauß-Quadratur)* (3T*+1T*+4T*+4T*+1T*+4T* Punkte)

a) Transformieren Sie das Integral $\int_a^b f(t)dt$ auf das Intervall $[-1, 1]$, d. h. finden Sie Konstanten c_1, c_2 und c_3 so, dass

$$\int_a^b f(t)\,dt \;=\; c_1 \int_{-1}^{1} f(c_2 + c_3 \cdot s)\,ds.$$

b) Berechnen Sie die Nullstellen x_i des dritten Legendre-Polynoms $L_3(x) = \frac{1}{2}\left(5x^3 - 3x\right)$.

c) Bestimmen Sie zu diesen Nullstellen x_i Gewichte $\lambda_i \in \mathbb{R}$ so, dass die Quadraturformel

$$\hat{I}(f) \;=\; \sum_{i=0}^{2} \lambda_i \cdot f(x_i) \quad \approx \quad \int_{-1}^{1} f(x)\,dx$$

für alle Polynome vom Grad kleiner oder gleich 2 exakt ist.

Hinweis: Setzen Sie für f die Monome $m_i(x) = x^i \quad (i = 0, 1, 2)$ ein und lösen Sie das daraus resultierende Gleichungssystem. Mit der Linearität des Fehlers (vergleiche Aufgabe 52) ist dann die exakte Integration auch für Linearkombinationen der Monome gewährleistet.

d) Welchen Exaktheitsgrad hat die resultierende Quadraturformel?

e) Welchen Exaktheitsgrad kann eine Quadraturformel mit dieser Anzahl Stützstellen maximal haben?

f) Wie können Sie alternativ zum Hinweis in Aufgabenteil c) die Gewichte berechnen? Erhalten Sie mit dieser alternativen Berechnung die gleichen Werte für die Gewichte wie in Aufgabenteil c)?

Hinweis: Denken Sie daran, dass auch die obige Quadraturformel interpolatorisch ist. Einen Hinweis (bis auf den Faktor $\frac{1}{b-a}$) gibt Ihnen auch Bemerkung 7.3.22 und Tabelle 7.1.

Aufgabe 54 *(Vergleich von Quadraturformeln)* (3T*+3T*+4T* Punkte)

In dieser Aufgabe wollen wir drei verschiedene Quadraturformeln vergleichen. Die zu vergleichenden Quadraturformeln haben jeweils drei Stützstellen und damit drei Funktionsauswertungen.
Es gilt

$$\log(x^2 + 1) = \int_0^x \frac{2t}{t^2 + 1}\,dt.$$

Berechnen Sie hiermit $\log(5)$ näherungsweise

a) unter Verwendung der Trapezsumme mit 2 Teilintervallen,

b) mit Hilfe der Simpson-Regel und

c) mit der in Aufgabe 53 hergeleiteten Quadraturformel.

Berechnen Sie jeweils den relativen Fehler und vergleichen Sie diese Fehler. War das Ergebnis zu erwarten?

Hinweise:
Die Programmieraufgaben sind in Matlab zu erstellen. Der Source Code muss strukturiert und dokumentiert sein. Senden Sie **spätestens 24 Stunden vor Ihrem Tutorium** alle Matlab-Files und alle Ergebnisse in einer E-mail mit dem Betreff **Loesung-Blatt12** an **angewandte.numerik@uni-ulm.de**.