Abschlussprojekte

Übungen zu „Numerische Mathematik“

Sommersemester 2012

In Gruppen zu max. 4 Leuten ist **eines** der folgenden Beispiele zu bearbeiten, wobei jedes Beispiel an mehrere Gruppen vergeben werden kann (die Lösungen sollten aber nicht vollständig identisch sein!), aber alle fünf Beispiele vergeben werden müssen. Meldung darüber, welche Gruppen welches Beispiel machen wollen, bitte per E-Mail oder vor/nach einer der letzten Stunden. Mit „Buch“ ist Prof. Neumaiers Buch *Introduction to Numerical Analysis* gemeint.

Jede Gruppe hat **ein** Exemplar der dazugehörigen Programme, Resultate (`diary`) und Kommentare der Resultate (im Fall von Bsp. **5** ist auch eine Herleitung der Formeln nötig, die auch handgeschrieben abgegeben werden kann) abzugeben; Namen der Gruppenmitglieder nicht vergessen! (Programme sind am besten in elektronischer Form abzugeben, damit ich sie auch ausprobieren kann.)

Beurteilung: Für ein ganz korrekt von einer Vierergruppe bearbeitetes Beispiel gibt es für jedes Gruppenmitglied 6 Punkte. Zusatzpunkte gibt es, wenn ein Beispiel nur von 1-2 Personen bearbeitet wurde oder „Extras“ enthalten sind (z.B. Bsp. 1a mit Pivotisierung), Punkteabzüge (von den 6 Punkten) für Fehler. Jedes Mitglied einer Gruppe bekommt die gleiche Punkteanzahl fürs Projekt gutgeschrieben.

Ab 10 Punkten gibt es auf jeden Fall ein „Sehr gut“; wer bereits mit den Punkten für Tafelmeldungen und Mitarbeit und den Zusatzpunkten für einen hohen Kreuzerlprozentsatz 10 Punkte erreicht, braucht also kein Projekt mehr zu machen!

Abgabetermin: Ab sofort bis spätestens **15.10.2012**, bzw. (mind.) 3 Wochen, bevor Sie die eingetragene Note brauchen!

Beispiel 1: Das Zweipunkt-Randwertproblem

$$-y''(x) + \frac{k(k-1)}{(1-x)^2}y(x) = 0, \quad y(0) = 1, \quad y(1) = 0,$$

kann durch das lineare Tridiagonal-Gleichungssystem $Ay = b$ mit

$$A = \begin{pmatrix} 2+q_1 & -1 & & & 0 \\ -1 & 2+q_2 & -1 & & \\ & \ddots & \ddots & \ddots & \\ & & \ddots & \ddots & -1 \\ 0 & & & -1 & 2+q_n \end{pmatrix}, \quad b = \begin{pmatrix} 1 \\ 0 \\ \vdots \\ 0 \end{pmatrix},$$

$q_i = \dfrac{k(k-1)}{(n+1-i)^2}$ approximiert werden; es ist dann $y(\frac{i}{n+1}) \approx y_i$, $i = 1, \ldots, n$.

a) Schreiben Sie ein Programm für die Gauß-Elimination (ohne Pivotisierung) für Tridiagonalmatrizen (d.h. Dreieckszerlegung, Vorwärts- und Rückwärtssubstitution; Spezialfall einer Bandmatrix, siehe Buch S. 74), wobei der Rechenaufwand und Speicherplatz von der Größenordnung $O(n)$ sein sollen. (D.h. es soll nicht die gesamte Matrix A abgespeichert werden, sondern nur die drei relevanten Diagonalen, entweder in der Form von drei Vektoren oder in Form einer $(n \times 3)$- oder $(3 \times n)$-Matrix mit zwei „überflüssigen“

Eintragungen, weil die Nebendiagonalen ja um 1 kürzer sind.) Das Programm soll die Diagonalen von A mit denen von L und R überschreiben und ausgeben und einen zusätzlichen Eingabeparameter enthalten, der angibt, ob die LR-Zerlegung gemacht werden soll oder die eingegebenen Diagonalen bereits die einer LR-Zerlegung sind und nur mehr die Vorwärts- und Rückwärtssubstitution gemacht werden sollen. (Diese Option wird in b) und c) für die Nachiteration gebraucht.)

b) Benutzen Sie das Programm, um das obige System für $k = 0.5$ und $n = 2^s - 1$, $s = 9$ und 15, zu lösen, und verbessern Sie den erhaltenen Lösungsvektor y durch (mindestens) einen Schritt Nachiteration zu $\tilde{y}$. (Für die Nachiteration sollen nur mehr die Vorwärts- und Rückwärtssubstitution mit der schon berechneten Faktorisierung gemacht werden.) Plotten Sie die Approximationen, d.h. $(\frac{1}{n+1}, y_1), \ldots, (\frac{n}{n+1}, y_n)$ und $(\frac{1}{n+1}, \tilde{y}_1), \ldots, (\frac{n}{n+1}, \tilde{y}_n)$, und die exakte Lösung $y(x) = (1-x)^k$ des Randwertproblems auf $[0,1]$. Drucken Sie ferner die Näherungswerte mit und ohne Nachiteration und die exakte Lösung an den Stellen $x = 0.25, 0.5$ und 0.75 aus.

c) Wie b) für $k = 4$ und (nur) für $n = 2^9 - 1 = 511$.

Hinweis: Die Approximationen für $k = 0.5$ sind nicht besonders gut, die für $k = 4$ jedoch schon. Daher ist b) auch nur für ein n zu berechnen.

Beispiel 2: Schreiben Sie ein Programm, das die Ableitung $f'(x)$ einer gegebenen differenzierbaren Funktion f am Punkt $x \in \mathbb{R}$ berechnet. Benutzen Sie dazu zu vorgegebenem $h_0 > 0$ die zentralen Differenzenquotienten

$$f[x - h_i, x + h_i] := \frac{f(x + h_i) - f(x - h_i)}{2h_i}$$

für $h_i := 2^{-i} h_0$, $i = 0, 1, \ldots$ p_n sei das Interpolationspolynom vom Grad $\leq n$ mit

$$p_n(h_i^2) = f[x - h_i, x + h_i], \quad i = 0, \ldots, n.$$

Wählen Sie den Grad n so groß, dass erstmals

$$|p_{n+1}(0) - p_n(0)| \geq |p_n(0) - p_{n-1}(0)|$$

gilt. Dann ist $\frac{1}{2}(p_n(0) + p_{n-1}(0))$ ein Näherungswert für $f'(x)$ und $\frac{1}{2}|p_n(0) - p_{n-1}(0)|$ ein Schätzwert für den Fehler.

Bestimmen Sie für $f(x) = e^x$ auf diese Weise eine Näherung und einen Schätzwert für den Fehler für $f'(1)$, wobei nacheinander $h_0 = 1, 0.1, 0.01, 0.001$ gewählt werden soll, und vergleichen Sie mit dem tatsächlichen Fehler.

Hinweis: Verwenden Sie das *Neville-Verfahren* zur Extrapolation am Punkt $x = 0$ (siehe Buch S. 146–148): Sei $p_{ik}(x)$ das (eindeutig bestimmte) Polynom vom Grad $\leq k$, das die Funktion $f(x)$ an $x_{i-k}, \ldots, x_i$ interpoliert, und sei $p_{ik} := p_{ik}(0) - f(x_i)$. Dann gilt für $i = 0, 1, \ldots$:

$$p_{i0} = 0,$$
$$p_{ik} = p_{i,k-1} + x_i \frac{p_{i,k-1} - p_{i-1,k-1} + f(x_i) - f(x_{i-1})}{x_{i-k} - x_i}, \quad k = 1, \ldots, i.$$

Wenn man p_{ik} in q_{i+1-k} speichert und nicht benötigte Zahlen überschreibt, erhält man für $n = 1, \ldots$:

$$q_n = 0,$$
$$q_j = q_{j+1} + (q_{j+1} - q_j + f(x_n) - f(x_{n-1}))x_n/(x_j - x_n), \quad j = n-1, n-2, \ldots, 1,$$
$$p_n(0) = q_1 + f(x_n).$$

Hier wird der Spezialfall $x_i = 4^{-i}h_0^2$ betrachtet, und der Ausdruck $x_n/(x_j - x_n)$ vereinfacht sich (ausrechnen!).

Beispiel 3: Sei ein erweitertes Gitter $x_{-2} < x_{-1} < x_0 < \cdots < x_n < x_{n+1} < x_{n+2} < x_{n+3}$ gegeben. Für $k = 0, \ldots, n+1$ definieren wir

$$B_k(x) := \begin{cases} \alpha_{k,k-2}(x - x_{k-2})^3 & \text{für } x_{k-2} < x \leq x_{k-1}, \\ \alpha_{k,k-2}(x - x_{k-2})^3 + \alpha_{k,k-1}(x - x_{k-1})^3 & \text{für } x_{k-1} < x \leq x_k, \\ \alpha_{k,k+2}(x_{k+2} - x)^3 + \alpha_{k,k+1}(x_{k+1} - x)^3 & \text{für } x_k < x \leq x_{k+1}, \\ \alpha_{k,k+2}(x_{k+2} - x)^3 & \text{für } x_{k+1} < x < x_{k+2}, \\ 0 & \text{sonst,} \end{cases}$$

wobei

$$\alpha_{k,j} = \frac{x_{k+2} - x_{k-2}}{\prod\limits_{\substack{|i-k|\leq 2 \\ i \neq j}} (x_i - x_j)}$$

Dann sind B_k, $k = 0, \ldots, n+1$, kubische Splines.
Plotten Sie für das Gitter $(-5, -2, -1, 0, 1, 2, 5)$ und das äquidistante Gitter $(-3, -2, -1, 0, 1, 2, 3)$ (jeweils um je drei beliebige Punkte auf beiden Seiten zu einem erweiterten Gitter ergänzt) die oben definierten Funktionen $B_k(x)$, $k = 0, \ldots, 8$, und ihre Summe $\sum_{k=0}^{8} B_k(x)$ auf $[x_1, x_n]$. Beschriften Sie die Achsen und geben Sie den Plots einen Titel.
Hinweis: Es gilt $\sum_{k=0}^{n+1} B_k(x) = 1$ auf $[x_1, x_n]$.

Beispiel 4: Das *Romberg-Verfahren* für die näherungsweise Berechnung von $\int_a^b f(x)\,dx$ ist rekursiv gegeben durch

$$\begin{aligned} T_{0,0} &:= T_N(f), \\ T_{i,0} &:= T_{2^i N}(f) = \frac{1}{2}(T_{i-1,0} + R_{2^{i-1}N}(f)), \quad i = 1, 2, \ldots, \\ T_{i,k} &:= T_{i,k-1} + \frac{T_{i,k-1} - T_{i-1,k-1}}{4^k - 1}, \quad k = 1, \ldots, i, \end{aligned}$$

wobei

$$T_N(f) := \frac{h_N}{2}\Big(f(a) + f(b) + 2\sum_{i=1}^{N-1} f(a + ih_N)\Big) \quad \text{(zusammengesetzte Trapezregel)},$$

$$R_N(f) := h_N \sum_{i=1}^{N} f\Big(a + (i - \frac{1}{2})h_N\Big)$$

und $h_N := (b-a)/N$. Schreiben Sie ein Programm für die näherungsweise Berechnung des Integrals $\int_a^b f(x)\,dx$ mit dem Romberg-Verfahren. Die Anfangsschrittweite sei $h := \frac{b-a}{2}$ (d.h. $N = 2$). Anschließend ist die Schrittweite so lange zu halbieren, bis der geschätzte absolute Fehler $|T_{k,k} - T_{k,k-1}|$ kleiner ist als 10^{-6}, höchstens aber zehnmal. Berechnen und drucken Sie die Werte des „Romberg-Dreiecks“ sowie den geschätzten Fehler des besten Wertes. Testen Sie das Programm an den drei Funktionen

a) $f(x) = \sin x, \quad a = 0, \quad b = \pi,$
b) $f(x) = x^{13/2}, \quad a = 0, \quad b = 1,$

c) $f(x) = (1 + 15\cos^2 x)^{-1/2}, \quad a = 0, \quad b = \pi/2.$

Vergleichen Sie für a) und b) die erhaltenen Werte auch mit den analytisch berechneten Werten und berechnen Sie die wirklichen Fehler.

Beispiel 5: Das Anfangswertproblem

$$\dot{y} = \frac{2t}{y^2}, \quad y(0) = 1,$$

besitzt die Lösung $y(t) = \sqrt[3]{3t^2 + 1}$. Berechnen Sie im Intervall $[0, 3]$ die Näherungslösungen (y^i Approximation für $y(t_i)$, $t_i = ih$)

a) mit dem Euler-Verfahren

$$y^0 = 1, \quad y^{i+1} = y^i + hF(t_i, y^i),\ i = 1, 2, \ldots, \quad F(t, y) := \frac{2t}{y^2},$$

mit den Schrittweiten $h = 0.1, 0.01, 0.001$;

b) mit dem Verfahren der Taylorreihe, für das die Formeln für die Koeffizienten bis zur Ordnung 4 in h hergeleitet werden sollen. Wählen Sie als Schrittweiten $h = 0.2, 0.1, 0.05$.

Vergleichen Sie die wahre Lösung und die Approximationen an den Stellen $s_i = 0.2i$ ($i = 1, 2, \ldots, 15$). (Damit lassen sich die Fehlerordnungen der Verfahren verifizieren.)
Hinweis: Taylorpolynom 4. Ordnung:
$y(t + h) \approx y(t) + h\dot{y}(t) + \dfrac{h^2}{2}\ddot{y}(t) + \dfrac{h^3}{6}y^{(3)}(t) + \dfrac{h^4}{24}y^{(4)}(t).$
Aus der Differentialgleichung folgt $\dot{y} = \dfrac{2t}{y^2}$; durch sukzessives Differenzieren erhält man daraus auch Ausdrücke für $\ddot{y}$, $y^{(3)}$ und $y^{(4)}$, die nur von y und t abhängen. Diese sind in das Taylorpolynom 4. Grades einzusetzen, um eine Formel $y^{i+1} = G(h, t_i, y^i)$ zu erhalten.