Serie 4

Besprechung: in der Woche vom 27.10.08

4.1. Zur Erinnerung: die Notation $O(f(x))$, $x \to x_0$ bezeichnet eine Funktion $\widetilde{f}$ mit der Eigenschaft, daß es eine Umgebung $\mathcal{U}$ von x_0 und eine Konstante $c > 0$ gibt, so daß $|\widetilde{f}(x)| \le C|f(x)|$ für alle $x \in \mathcal{U}$. Analog verfährt man mit "einseitigen" Annäherungen, d.h. $x \to x_0+$ oder $x \to x_0-$.

a) Bestimmen Sie a, b so, daß $\arctan z = a + O(z^b)$, $\quad z \to +\infty$. (*Hinweis:* $\arctan z = \int_0^z (1+t^2)^{-1}\,dt$)

b) Sei $f(x) := \displaystyle\int_0^1 \frac{1}{x^2+t^2}\,dt$. Geben Sie eine (einfache) Funktion $h(x)$ an, so daß

$$f(x) = h(x) + O(1), \qquad x \to 0+\,.$$

c) Geben Sie p, $q \in \mathbb{R}$ und eine Funktion $h(x)$ an, so daß $\displaystyle\frac{x^3+27}{x^4-81} = p(x-3)^q + O(h(x))$, $x \to 3-$.

4.2. Betrachten Sie das Polynom

$$P(x) = x^4 - 4x^3 + 6x^2 - 4x + a, \qquad a = 1.$$

Wie ändern sich die Nullstellen von P, wenn der Koeffizient a zu $\tilde{a} = a - \varepsilon$, $0 < \varepsilon \ll 1$, verfälscht wird? Geben Sie die Kondition des Problems $a \mapsto$ "eine Nullstelle von P" an. Auf wieviele Stellen genau ist die Nullstelle bei einer (relativen) Maschinengenauigkeit von 10^{-16} berechenbar?

4.3. Betrachten Sie das Bestimmen einer Nullstelle des quadratischen Polynoms $x^2 + 2px - q$ mithilfe der Formel $x = -p + \sqrt{p^2+q}$.

a) Bestimmen Sie die absolute und relative Kondition der Abbildung $(p,q) \mapsto x$. Zeigen Sie: Für $q > 1$ ist das Bestimmen der Nullstelle x gut konditioniert. Für $q \approx -p^2$ ist es schlecht konditioniert.

b) Betrachten Sie folgende zwei Algorithmen zur Berechnung von x:

Algorithmus 1:	$s := p^2, \quad t := s + q, \quad u := \sqrt{t}, \quad x := -p + u;$
Algorithmus 2:	$s := p^2, \quad t := s + q, \quad u := \sqrt{t}, \quad v := p + u, \quad x := q/v.$

Bestimmen Sie mit einem MATLAB-Programm die Nullstelle mit beiden Algorithmen für $p = 10^5$ und $q = 0.018$. Geben Sie den relativen Fehler beider Auswertungsarten an. Das exakte Ergebnis ist $x = 0.899999999999595000\ldots \cdot 10^{-7}$.

c) Erklären Sie das Verhalten der beiden Algorithmen in Teilaufg. b) mithilfe der Vorwärtsfehleranalyse, d.h. simulieren Sie das Verhalten der Gleitkommaarithmetik für beide Algorithmen. Verwenden Sie hierzu das Modell, daß die Computerrealisierungen (immer mit * gekennzeichnet) der "Elementaroperationen" $+$, $-$, $*$, $/$, $\sqrt{\ }$ folgendes erfüllen: für Gleitkommazahlen x, y und $\mathtt{op} \in \{+,-,*,/\}$ gilt $x\,\mathtt{op}^*y = (x\,\mathtt{op}\,y)(1+\delta)$, wobei $|\delta| \le \mathtt{eps}$ mit $\mathtt{eps} = 10^{-16}$. Für $\sqrt{\ }$ gilt: $\sqrt{x}^* = \sqrt{x}(1+\delta)$, wobei $|\delta| \le \mathtt{eps}$. Sie dürfen zudem die Taylorapproximation $\sqrt{1+x} \approx 1 + x/2$ (für kleine x) einsetzen.

d) Erklären Sie das unterschiedliche Verhalten der beiden Algorithmen aus Teilaufg. b), indem Sie sich überlegen, welcher der beiden Algorithmen einen größeren Stabilitätsindikator hat.

4.4. (**schriftlich**) Machen Sie folgende "Standardannahme" an die Gleitkommaarithmetik: Für zwei Gleitkommazahlen x, y und $\mathtt{op} \in \{+,-,*,/\}$ gibt es ein $\delta = \delta(x,y,\mathtt{op})$ mit $|\delta| \le \mathtt{eps}$, so daß für die Computerrealisierung $\mathtt{op}^*$ von $\mathtt{op}$ gilt:

$$x\mathtt{op}^*y = (x\,\mathtt{op}\,y)(1+\delta).$$

(Dies besagt, daß die Grundrechenarten rückwärtsstabil realisiert werden.) Für die Maschinengenauigkeit $\mathtt{eps}$ und $n \in \mathbb{N}_0$ definieren wir

$$\gamma_n := \frac{n\mathtt{eps}}{1 - n\mathtt{eps}}$$

Im Folgenden wird immer die implizite Annahme getroffen, daß für alle auftretenden n die Bedingung $n\mathtt{eps} < 1$ erfüllt ist.

a) Zeigen Sie: für θ_n und δ mit $|\theta_n| \leq \gamma_n$ und $|\delta| \leq \texttt{eps}$ gilt

$$|(1+\theta_n)(1+\delta) - 1| \leq \gamma_{n+1}.$$

b) Seien $(x_i)_{i=1}^n$ und $(y_i)_{i=1}^n$ zwei Folgen von Gleitkommazahlen. Wir bezeichnen mit $\texttt{skalar}(x, y)$ die Funktion, die das Skalarprodukt $x^\top y$ durch $\texttt{skalar}(x, y) := s_n$ mittels folgender Rekursion definiert:

$$s_1 := x_1 *^* y_1, \quad s_{i+1} := s_i +^* (x_{i+1} *^* y_{i+1}), \qquad i = 1, \ldots, n-1$$

Zeigen Sie folgendes Rückwärtsstabilitätsresultat: Das erhaltene Ergebnis $s_n := \texttt{skalar}(x, y)$ erfüllt für geeignet gewählte θ_n, θ'_n, $\theta_{n-1}, \ldots, \theta_2$:

$$s_n = x_1 y_1 (1+\theta_n) + x_2 y_2 (1+\theta'_n) + x_3 y_3 (1+\theta_{n-1}) + \cdots + x_n y_n (1+\theta_2);$$

hier erfüllen alle θ_i, θ'_i die Abschätzungen $|\theta_i| \leq \gamma_i$ und $|\theta'_i| \leq \gamma_i$. Begründen Sie, warum dies ein Rückwärtsstabilitätsresultat ist.

c) Schließen Sie auf folgendes Vorwärtsstabilitätsresultat ($|x|$ und $|y|$ sind Vektoren, bei denen die Betragsfunktion komponentenweise angewendet wurde):

$$|x^\top y - \texttt{skalar}(x, y)| \leq \gamma_n |x|^\top |y|$$

4.5. (**Programmieraufgabe**) In der VO wurde folgendes Rückwärtsstabilitätsresultat für die LU-Zerlegung diskutiert: Bestimmt man eine LU-Zerlegung ohne Pivotsuche, so erhält man Faktoren $\widehat{L}$ und $\widehat{U}$, für die es ein ΔA gibt mit $A + \Delta A = \widehat{L}\widehat{U}$ und $\|\Delta A\|_2 \leq \gamma_n \| \, |\widehat{L}| \, |\widehat{U}| \, \|_2$.

Ziel der Aufgabe ist zu sehen, inwieweit diese Abschätzung qualitativ richtig ist. Hierzu wird in Matlab die LU-Zerlegung (*Hinweis:* `help lu`) in *einfacher Genauigkeit* durchgeführt (*Hinweis:* `help single`, `help eps`, `help cast`). Die Differenz $A - LU$ wird in doppelter Genauigkeit bestimmt, so daß man davon ausgehen kann, den Fehler $A - LU$ "exakt" bestimmt zu haben.

Hierzu seien die Matrizen $A \in \mathbb{R}^{n \times n}$ gegeben durch

$$A = W + 0.1\,\texttt{eye}(n), \qquad W = \begin{pmatrix} 1 & 0 & \ldots & 0 & 1 \\ -1 & \ddots & \ddots & \vdots & \vdots \\ \vdots & \ddots & \ddots & 0 & \vdots \\ \vdots & & \ddots & 1 & 1 \\ -1 & \ldots & \ldots & -1 & 1 \end{pmatrix} \in \mathbb{R}^{n \times n}.$$

Schreiben Sie ein Matlabprogramm $[\texttt{schranke}, \texttt{fehler}] = \texttt{wilkinson\_example(n)}$, welches eine Fehlerschranke (`schranke`) für die LU-Zerlegung und den tatsächlichen Fehler (`fehler`) zurückgibt. Das Programm bestimme A, konvertiere A anschließend mittels `single` auf einfache Genauigkeit, bestimme dann die LU-Zerlegung (ebenfalls in einfacher Genauigkeit—dies geschieht in Matlab automatisch, wenn die Eingabe vom Typ `single` ist) von A. Berechnen Sie die Differenz $A - LU$ in doppelter Genauigkeit, indem Sie A und die erhaltenen Faktoren L und U in "doubles" konvertieren. Die Ausgaben $\texttt{fehler} = \texttt{norm(A - L * U)}$ und $\texttt{schranke} = \gamma_n \texttt{norm}(|L||U|)$ sind nun einfach zu bestimmen.

Schreiben Sie ein zweites Matlabprogramm `serie4_5`, welches für $n = 1, \ldots, 50$ die Zahlen `schranke` und `fehler` bestimmt und anschließend semilogarithmisch (`help semilogy`) gegen n plottet. Diskutieren Sie Ihre Ergebnisse.