Numerik Prohl WS2023

Inhalt:	Live-Transkription
Datum:	WS 2023
Author:	Prohl

`[ Prohl 16.10.2023 ]`

0.1

0.2

0.3

0.4

0.4.1

0.4.2

0.4.3

0.4.4

0.4.5

`[ Prohl 18.10.2023 ]`

- Gleitpunktzahl $g \in G$.
  - Darstellung: $g = \pm d_1, \dots, d_2 \cdot \beta^e$, hierbei ist das 4-Tupel $(\beta, m, \underline{e}, \bar{e})$
  - speziell: $d_1 > 0$ ("Normaleinheit")
- $g_{\max} = \max_{g \in G} g, \; g_{\min} = \min_{g \in G, g > 0}$
- Runddungsabbildung $rd \colon G' \to G$ mit

$$G' = \{0\} \cup \{x \in \mathbb{R} \colon g_{\min} \leq |x| \leq g_{\max}$$

- eps, die Maschinengenauigkeit:

$$eps := \frac{1}{2}\beta^{1-m}$$

0.4.6 Lemma

Sei $0 \neq x \in G'$. Dann:

$$|\frac{rd(x) - x}{x}| \leq eps$$

Beweis

1. Ist $\beta^{e-1} \leq x \leq \beta^e$, so heißen die beiden Gleitkommazahlen, welche x einschließen, der Abstand β. Dann aber: $|x - rd(x)| \leq \frac{1}{2}\beta$
2. Aus 1. haben wir: $|x| \geq \beta^{e-1}$

3. $\frac{|x-rd(x)|}{|x|} \leq \frac{1}{2}\frac{\beta^{e-n}}{\beta^{e-1}} = eps.$

Beispiel

- $\beta = 1, n = 1$, betrachte $x = 0.33$ (ist das eine Zahl?)
- Damit $10^{-1} \leq x \leq 1$, mit $e = 0$
- Also $0.3 \leq x \leq 0.4$

Arithmetische Grundoperationen

Arithmetische Grundoperationen $* \in \{+, -, \cdot, :\}$ werden auf dem Rechner durch entsprechende *Maschinenoperationen* realisiert, kurz:

$$\circ \in \{\oplus, \ominus, \odot, \oslash\} \text{ (Maschinenoperationen).}$$

Ihre Eigenschaft ist, dass sie asu Maschinenzahlen wieder solche machen , gemäß

$$a \circ b = rd(a * b), \ \forall a, b \in G.$$

Dann werden die Operationen intern mit meist erhöhter Stellenzahl der Nachkommezahlen ausgeführt, dann in normale Form gebracht.

0.4.7 Bemerkung

1. - Überlauf: Falls $|x + y| > g_{\max}$
   - Unterlauf: Falls $0 < |x + y| < g_{\min}$
2. Distributiv- und Assoziativgesetze gelten nicht mehr. D.h.

$$(a \oplus b) \odot c \neq a \odot c \oplus b \odot c$$
$$(a \oplus b) \oplus c \neq a \oplus (b \oplus c)$$

3. *Auslöschung* ist unangenehmer Effekt in Gleitkommazahlenarithmetik.

Beispiel (weggelassen)

0.5 Konditionierung einer numerischen Aufgabe

Eine numerische Aufgabe wird als *gut konditioniert* bezeichnet, wenn eine kleine Störung der Eingangsdaten nur eine kleine Änderung der Ergebnisse zur Folge hat.

0.5.1 Beispiel

(weggelassen)

0.5.2 Bezeichnung (numerische Aufgabe)

Unter einer numerischen Aufgabe verstehen wir die Berechnung endlich vieler Größen y_i $(1 \leq i \leq n)$ und gewissen Größen x_i $(1 \leq j \leq m)$ mittels der funktionalen Vorschrift:

$$y_i = f_i(x_1, \dots, x_m) = f_i(\vec{x})$$

Beispiel: $f_1(x_1, x_2) = x_1 + x_2$.

Nachfolgend sei

- $\vec{x} = (x_1, \dots, x_m)^T$
- und $\vec{y} = (y_1, \dots, y_n)$
- und $\vec{f} = (f_1, \dots, f_n)$.

Nachfolgend nehmen wir die Differenzierbarkeit an von $\vec{f}\colon \mathbb{R}^n \to \mathbb{R}^n$. Mittels der *differentiellen Fehlanalyse* wollen wir den Einfluss kleiner Datenfehler $(\Delta_{x_1}, \dots, \Delta_{x_m})$ auf die Resultate y_i untersuchen.

Das geschieht mit dem Taylor'schen Satz:

$$\Delta_{y_i} := f_i(\vec{x} + \vec{\Delta x}) - \underbrace{f_i(\vec{x})}_{=\, y_i} = \sum_{j=1}^{m} \frac{\partial f_i}{\partial x_j}(\vec{x})\underbrace{\Delta x_j} + R_f(\vec{x}, \vec{\Delta x})$$

mit dem Restglied $R_f(\vec{x}, \vec{\Delta x}) = o(|\vec{\Delta x}|)$, falls $f \in C^2$, d.h. zweimal stetig differenzierbar.

Wir schreiben nun:

$$\Delta_{y^i} = \sum_{j=1}^{m} \frac{\partial f_i}{\partial x_j}(\vec{x})\Delta x_j + o(|\Delta x|).$$

Nun dividieren wir durch $y_i \neq 0$:

$$\begin{aligned} |\frac{\Delta y_i}{y_i}| &\leq |\sum_{j=1}^{m} \frac{\partial f_i}{\partial x_j}(\vec{x})\frac{\Delta x_j}{y_i}| + \frac{o(|\Delta x|)}{|y|} \\ &= \sum_{j=1}^{m} \underbrace{|\frac{\partial f_i}{\partial x_j}(\vec{x})\frac{x_i}{f_i(\vec{x})}|}_{:=\, k_{ij}(\vec{x})} |\frac{\Delta x_i}{x_j}| + \frac{o(|\Delta x|)}{|y_i|}. \end{aligned}$$

Die Größen $k_{ij}(\vec{x})$ heißen (relative) *Konditionszahlen* der Funktion $\vec{f}$ im Punkt $\vec{x}$. Sie sind ein Maß dafür, wie sich relative Eingabefehler verstehen/auswirken.

0.5.3 Definition (Landau'sche Symbole)

Es seien $g, h\colon \mathbb{R} \to \mathbb{R}$. Die Schreibweise

a. $g(t) = \mathcal{O}(h(t)\ (t \to 0)$ bedeutet, dass für kleine $t \in (0, t_0]$ mit einer Konstanten $c \geq 0$ gilt:

$$|g(t)| \leq c|h(t)|$$

b. $g(t) = o(h(t))\ (t \to 0)$ bedeutet, dass für kleine $t \in (0, t_9]$ mit einer Funktion $c(t) \downarrow 0$, gilt: $|g(t)| \leq c(t)|h(t)|$. Analoge Sichtweisen gelten für $t \uparrow \infty$.

0.5.4 Sprechweise: Doe numerische Aufgabe

$\vec{y} = f(\vec{x})$ heißt

a. *schlecht konditioniert*, wenn ein $|k_y(x)| >> 1$ ist, ansonsten

b. gut konditioniert.

c. Falls $|k_{ij}(\vec{x})$ spricht man von *Fehlerauslöschung* ansonsten von Verstärkung.

0.5.5 Beispiele

1. Die Addition $y = f(x_1, x_2) = x_1 + x_2$ ist für Zahlen $x_1 \approx -x_2$ *schlecht konditioniert* ($\Rightarrow$ Auslöschung), während Multiplikation und Division *gut konditioniert* sind.