



Numerische Lineare Algebra - Matlab-Blatt 4

(Besprechung in den MATLAB-Tutorien in KW 49/50)

Aufgabe 8 (Auslöschung)

(8 Punkte)

In dieser Aufgabe wollen wir den Ausdruck $f(x) = \sqrt{x^2 + 1} - x$ berechnen.

- (i) Laden Sie sich das MATLAB-Skript `cancellation.m` von der Homepage herunter.
- (ii) Vervollständigen Sie die Zeilen 16-20, indem Sie für $x \in \{10^0, 10^1, \dots, 10^{10}\}$ f auswerten und jeweils den relativen Fehler berechnen. Die "exakten" Werte sind im Skript schon vorgegeben. Stellen Sie außerdem den relativen Fehler über x in einem Schaubild mit doppelt-logarithmischer Skala dar. Was beobachten Sie? Wie nennt sich dieser Effekt?
- (iii) Geben Sie eine alternative Berechnungsmethode für f an, bei der das Phänomen aus Teil (ii) nicht auftritt. Erweitern Sie ihr Skript, sodass f mit Ihrer Methode ausgewertet wird. Berechnen Sie auch hier den relativen Fehler und zeichnen Sie diesen mit in das Schaubild ein. Erklären Sie das Ergebnis.

Aufgabe 9 (A simple recurrence formula)

(12 Punkte)

We consider the sequence

$$\omega_n(x) = n!(e^x - e_n(x)), \quad n \geq 0, \quad \text{where } e_n(x) = \sum_{k=0}^n \frac{x^k}{k!} \quad (x \in \mathbb{R}_{\geq 0}, \quad n \in \mathbb{N}_{\geq 0}).$$

- (i) Show that $\omega_n(x)$ satisfies $(n+1)\omega_n(x) - \omega_{n+1}(x) = x^{n+1}$ ($n \geq 0, x \geq 0$) with $\omega_0(x) = e^x - 1$.
- (ii) Show $0 \leq \omega_n(x) \leq \frac{x^{n+1}}{n+1}e^x$, $\forall x \geq 0, n \geq 0$.
- (iii) Write a Matlab function `val = omegan(x,N)` which returns a vector `val`, containing the values $\omega_n(x)$, $n = 0, \dots, N$ for given $x \geq 0$ and $N \geq 0$. The values should be calculated with the relation from part (i).
- (iv) Write a script `main1.m` in which you test the function `val = omegan(x,N)` for $x = 1$, and $N \in \{15, 18, 20\}$. For each value of N plot the vector `val` over the vector `0:N`. Compare your computational results with what you proved in part (ii). Explain your observations.
- (v) Write a Matlab function `val = rd2n(x,n)`, which rounds the number `x` to `n` digits of exactness. Example:
 $1260.0 = \text{rd2n}(1264.3, 3)$ und $0.2365 = \text{rd2n}(0.23649, 4)$
- (vi) Write a function `val = omeganrd(x,N)`, by modifying the function `val = omegan(x,N)` such that after each operation the result is rounded to four digits (Round with the function `val = rd2n(x,n)`). In the script `main1.m`, test the function `val = omeganrd(x,N)` for $x = 1$ and $N = 7$. What do you observe?