

Numerische Mathematik I für Ing. – Übung 4 –

Theoretische Aufgaben: (Abgabe im jeweiligen Tutorium, 16.-18. Mai)

1. Aufgabe: (3+1 P.)

Wir untersuchen die Kondition der Berechnungsvorschrift $x^* = f(a, b) = a^2 - b^2$.

- Berechne die Faktoren, um die die relativen Fehler in den Daten a, b im relativen Fehler des Ergebnisses verstärkt werden.
- Begründe an Hand der Verstärkungsfaktoren, für welchen Fall die Berechnung schlecht konditioniert ist.

2. Aufgabe: (1+2+2 Punkte)

Betrachte Bsp. 29 der VL.

- Zeige: Die Lösung

$$x^* = a_1 - \sqrt{a_1^2 - a_2} \quad (1)$$

der quadratischen Gleichung $x^2 - 2a_1x + a_2 = 0$ kann auch durch

$$x^* = \frac{a_2}{a_1 + \sqrt{a_1^2 - a_2}} \quad (2)$$

beschrieben werden.

- Schreibe (??) als Algorithmus analog zu Seite 36 unten links im Skript.
- Rechne das Zahlenbeispiel Seite 36 unten rechts für diesen Algorithmus durch. Berechne den relativen Fehler und vergleiche ihn mit dem für Formel (??) im Skript.

3. Aufgabe: (1+1+3+2 P.)

Wir betrachten folgendes Differentialgleichungssystem, das trotz seiner Einfachheit eine außerordentlich reiche und komplizierte Dynamik besitzt. Die Gleichungen wurden 1963 von dem Meteorologen und Mathematiker E.N. Lorenz als grobes Modell einer konvektiven (vorwiegend vertikalen) Strömung aufgestellt. Sie sind ein klassisches Beispiel aus der sogenannten Chaos-Theorie.

$$\begin{aligned}\dot{x} &= \sigma(y - x) \\ \dot{y} &= rx - y - xz \\ \dot{z} &= xy - bz\end{aligned}$$

für $t \geq 0$. Dabei sind $\sigma, r, b > 0$ konstant.

- Zeige: Wenn $(x(t), y(t), z(t))$ eine Lösung ist, so ist auch $(-x(t), -y(t), z(t))$ eine Lösung.
- Zeige, dass die z -Achse eine *invariante Menge* ist, d.h. dass Lösungen, die sich zu einer bestimmten Zeit auf ihr befinden, von da an auf ihr bleiben.
- Berechne alle *stationären Punkte*, d.h. die Punkte mit $(\dot{x}, \dot{y}, \dot{z}) = (0, 0, 0)$ für alle t .
- Schreibe das System in Vektorschreibweise und berechne die Funktionalmatrix der rechten Seite.

Programmieraufgabe: (Abgabe per email bis zum 25. Mai) (16 P.)

Das Programm vom 2. Übungsblatt soll so abgeändert werden, dass nun das Eulerverfahren selbst als Funktion in MATLAB realisiert wird. Es soll für beliebige Systeme von gewöhnlichen Differentialgleichungen anwendbar sein.

- Eingabe: Vektor der Startwerte, Start- und Endwert für die unabhängige Variable t_0, t_e , konstante Schrittweite h , Name der Funktion, die die rechte Seite der Differentialgleichung beschreibt (als String).
- Ausgabe: Vektor der Stützstellen t_i und Matrix der approximierten Funktionswerte u_i .
- Teste das Programm an dem System aus Aufgabe 3 mit den Anfangsbedingungen $x(0) = 18.7, y(0) = 29.9, z(0) = 102.5$ und $\sigma = 10, r = 100.5, b = 8/3, t \in [0, 1.1]$.
- Plotte das Ergebnis als dreidimensionale Kurve $\{(x(t), y(t), z(t))\}$.

- Hinweis (nicht Teil der Aufgabe): Interessant ist auch der Fall $r = 24, x(0) = 1, y(0) = 1, z(0) = 0$ (Lorenz-Attraktor).

Die Dateien müssen **als Anhänge/Attachments per email** an den **Betreuer des jeweiligen Tutoriums** geschickt werden. Betreuer sind

Thomas Slawig, email: nm1-001@pool.math.tu-berlin.de
Uwe Prüfert, email: nm1-011@pool.math.tu-berlin.de