Computergrundlagen 2025

Blatt 10: Python

- Abgabetermin für die Lösungen: **04.01.2026, 20 Uhr**
- Bei Fragen wendet euch bitte an eure/n Tutor/in:
  - Mo 11:30: Stephan Haag: `st170833@stud.uni-stuttgart.de`
  - Di 09:45: Julian Hoßbach: `julian.hossbach@icp.uni-stuttgart.de`
  - Mi 14:00: Julian Peters: `julian.peters@icp.uni-stuttgart.de`
  - Do 09:45: Rebecca Stephan: `rebecca.stephan@icp.uni-stuttgart.de`
  - Fr 09:45: Jonas Höpker: `st182335@stud.uni-stuttgart.de`
- Die Übungsaufgaben sollen in der Regel in **Zweiergruppen** bearbeitet werden. Nur in **begründeten Ausnahmefällen** sind Dreiergruppen möglich.
- Die Abgabe der Übungsblätter erfolgt über Ilias.
- Mit Abgabe der Lösungen erklärt Ihr, dass Ihr die Lösung euren Mitstudierenden im Rahmen der Übungsbesprechung vorstellen könnt. Um dies zu überprüfen, muss mindestens zweimal von jedem Teilnehmenden vorgetragen werden. Wenn Ihr das nicht könnt, werden euch die Punkte für die entsprechenden Aufgaben wieder abgezogen.
- **Befehle, die nicht in der Vorlesung besprochen wurden, müssen gegebenenfalls recherchiert werden.**
- **Alle erstellen Skripte (`.py` and `.ipynb`) sowie ein mit markown oder Latex erstellter Report (`.pdf`) sind Teil der Abgabe**

  Tipp: Für Abgaben in Python kann der Editor https://jupyter.org/ hilfreich sein. Hier könnt ihr euren Code *interactive* ausführen sowie mit Markdown Zellen ausführlich kommentieren. Die `*.ipynb*` Dateien können ebenfalls als Lösungen abgegeben werden und sind leicht zu validieren.

Problem des Handlungsreisenden (5 Punkte)

Schreibe ein Programm, welches die kürzeste Strecke durch n zufällig ausgewählte Punkte mit den Koordinaten (x, y) findet. Berechne hierzu alle möglichen Permutationen mit Hilfe von `itertools` und wähle die insgesamt kürzeste Strecke aus. Erstelle abschließend eine Grafik, welche die Punkte darstellt. Außerdem soll die kürzeste Strecke zwischen den Punkten durch einen Polygonzug dargestellt werden. Demonstriere das Programm für $n = 10$. (Hinweis: Beachte, dass die Richtung in der ein Pfad durchlaufen wird (vorwärts oder rückwärts) keinen Einfluss auf die Länge hat.)

Fehleranalyse einer gedämpften Schwingung (5 Punkte)

In dieser Aufgabe lernst du, wie man die symbolische Mathematik von `sympy` nutzt, um Fehlerfortpflanzungsformeln aufzustellen, und diese anschließend mit `matplotlib` zu visualisieren.

Die Auslenkung $x(t)$ einer schwach gedämpften Schwingung ist gegeben durch:

$$x(t) = A \cdot e^{-\lambda t} \cdot \cos(\omega t)$$

Angenommen, wir messen die Amplitude A, den Dämpfungskoeffizienten γ und die Kreisfrequenz ω. Jede dieser Größen hat eine kleine Messunsicherheit (ΔA ,$\Delta\gamma$,$\Delta\omega$). Wir möchten wissen, wie groß der resultierende Fehler Δx zum Zeitpunkt t ist.

Nach linearer Fehlerfortpflanzung ist dieser Fehler gegeben durch:

$$\Delta x = \Delta A|\frac{\partial x}{\partial A}| + \Delta\gamma|\frac{\partial x}{\partial \gamma}| + \Delta\omega|\frac{\partial x}{\partial \omega}|$$

Teil A: Symbolische Berechnung (sympy)

Schreibe ein Python Skript, das die folgenden Punkte ausführt:

1. Definiere die Symbole für alle Variablen (t, A, γ, ω) und deren Fehler $(\Delta A, \Delta\gamma, \Delta\omega)$.
2. Erstelle den symbolischen Ausdruck für $x(t)$.
3. Berechne die partiellen Ableitungen nach A, γ und ω mit `sympy`.
4. Setze diese zur Formel für den Gesamtfehler Δx zusammen.
5. Nutze `sp.lambdify`, um die symbolische Fehlerformel in eine effiziente numerische Funktion für `numpy` umzuwandeln.

Teil B: Numerische Simulation & Plot (matplotlib)

Ergänge nun das Python Skript aus Teil A um die folgenden Punkte:

1. Erstelle einen Zeitvektor t von 0 bis 20 mit 400 Punkten.
2. Verwende folgende Messwerte:
   - $A = 10 \pm 1.0$
   - $\gamma = 0.1 \pm 0.0$
   - $\omega = 2.0 \pm 0.0$
3. Berechne die Werte für $x(t)$ und den dazugehörigen Fehler $\Delta x(t)$.
4. Erstelle einen Plot:
   - Zeichne die Hauptkurve $x(t)$ als Linie.
   - Stelle den Unsicherheitsbereich $[x - \Delta x, x + \Delta x]$ als schattierte Fläche dar (`plt.fill_between`).
   - Beschrifte die Achsen und füge eine Legende hinzu.
5. Erkläre den Verlauf des Plots.
6. Verwende nun die Werte
   - $A = 10 \pm 0.0$
   - $\gamma = 0.1 \pm 0.0$
   - $\omega = 2.0 \pm 0.2$

   und erkläre den Verlauf des resultierenden Plots.