Übungen zu Physik auf dem Computer SS 2012

Übungsblatt 11: Schrödingergleichung

4. Juli 2012

Allgemeine Hinweise

- Abgabetermin ist **Montag, 9.7.2012, 13:00**.
- Zur Abgabe schickst Du die Lösungsdatei(en) im Anhang einer Email an Deinen Tutor:
  - Florian (`floh@icp.uni-stuttgart.de`; Dienstag, 15:45–17:15)
  - Dominic (`dominic@icp.uni-stuttgart.de`; Dienstag, 15:45–17:15)
  - Olaf (`olenz@icp.uni-stuttgart.de`; Mittwoch, 15:45–17:15)
- Die Übungen werden in Gruppen von jeweils zwei oder drei Leuten bearbeitet. Diese dürfen sich gerne von Blatt zu Blatt unterscheiden. Aus formalen Gründen muss allerdings jeder von Euch eine eigene Lösung abgeben. Schreibt bitte auf die Lösungen, mit wem Ihr zusammengearbeitet habt, um uns das Korrigieren zu erleichtern.
- Die Übungen finden statt im CIP-Pool des Instituts für Computerphysik (ICP) im Pfaffenwaldring 27.

Auf diesem Blatt geht es darum, die (stationäre, eindimensionale) Schrödingergleichung numerisch zu lösen. Diese lautet[1]

$$-\frac{\mathrm{d}^2}{\mathrm{d}x^2}\psi(x) + V(x)\psi(x) = E\psi(x).$$

Im Gegensatz zur Poissongleichung handelt es sich dabei um ein Eigenwertproblem, das zur Lösung andere Verfahren benötigt.

Aufgabe 11.1 (5 Punkte): Unendliche quadratische Senke

Die unendliche quadratische Senke ist durch das folgende Potential definiert:

$$V(x) = \begin{cases} 0 & \text{, wenn } 0 < x < L \\ \infty & \text{, sonst} \end{cases}$$

Die Energieeigenwerte und die dazugehörigen Wellenfunktionen können für die unendliche quadratische Senke analytisch berechnet werden

- 11.1.1 (1 Punkt) Diskretisiere die eindimensionale, stationäre Schrödingergleichung mit Hilfe von finiten Differenzen in linearer Ordnung (wie im letzten Übungsblatt und im Skript). Implementiere eine Pythonfunktion `schroedinger_matrix(V, h)`, die die zum Lösen des Eigenwertproblems benötigte $N \times N$-Matrix berechnet und zurückgibt. Dabei sei `V` ein eindimensionales NumPy-Array, das N Werte des Potentials V enthält; `h` ist der Abstand zwischen den einzelnen Werten von $x \in [0, Nh]$, an denen `V` berechnet wurde. Gib ein Pythonskript als Lösung ab, das die Pythonfunktion enthält.

[1] O. B. d. A. arbeiten wir im reduzierten Einheitensystem und vernachlässigen den Vorfaktor, setzen also $\frac{\hbar^2}{2m} = 1$.

- 11.1.2 (1 Punkt) Implementiere eine Pythonfunktion `solve_schroedinger_scipy(V,h,n)`, die mit Hilfe der Pythonfunktionen `scipy.linalg.eigh` und `schroedinger_matrix` die `n` kleinsten Energieeigenwerte und die dazugehörigen numerischen Lösungen der Schrödungergleichung berechnet. Gib ein Pythonskript als Lösung ab, das die Pythonfunktion enthält.
- 11.1.3 (1 Punkt) Wie musst Du `V` wählen, damit die Gleichung für die tatsächlich unendliche quadratische Senke gelöst wird? Schreibe die Lösung in die Lösungsemail.

  Hinweis Wenn Du nicht auf die Lösung kommst, setze einfach den ersten und den letzten Wert des Potentials auf einen großen Wert (z.B. 1000).

- 11.1.4 (2 Punkte) Löse die Schrödingergleichung für ein Teilchen in der unendlichen quadratischen Senke ($L = 10$). Verwende dafür $N = 50$ Punkte. Erzeuge einen Plot, der die verschiedenen Lösungen ψ zeigt und die einzelnen Kurven mit den Energieeigenwerten E beschriftet. Dabei kannst Du die Ergebnisse mit den analytischen Lösungen vergleichen: $E = \frac{n\pi}{L}, n = 1, 2, 3, \dots$. Gib ein Pythonskript als Lösung ab, das den Plot erzeugt.

  Hinweis Damit die Kurven im Plot besser zu unterscheiden sind, bietet es sich an, die einzelnen Kurven ψ um ihren Energieeigenwert E entlang der y-Achse zu verschieben.

Aufgabe 11.2 (2 Punkte): QR-Verfahren

- 11.2.1 (1 Punkt) Implementiere eine Pythonfunktion `solve_schroedinger_qr(V,h,n,tolerance)`, die die Energieeigenwerte und Eigenvektoren mit Hilfe des QR-Verfahrens aus dem Skript berechnet und zurückgibt. Gib ein Pythonskript als Lösung ab, das die Pythonfunktion enthält.
- 11.2.2 (1 Punkt) Löse mit Hilfe dieser Pythonfunktion die unendliche quadratische Senke und erzeuge entsprechende Plots wie in Aufgabe 11.1.4. Verwende dabei eine Genauigkeit von 10^{-1}. Gib ein Pythonskript als Lösung ab, das den Plot erzeugt.

  Hinweis Dieses Verfahren konvergiert schlecht, wenn man ein wirklich unendliches Potential am Rand verwendet. Daher solltest Du hierbei den ersten und den letzten Wert auf 1000 setzen.

Aufgabe 11.3 (3 Punkte): Andere Potentiale

Löse numerisch die Schrödingergleichung für die folgenden Potentiale und plotte die Ergebnisse wie in Aufgabe 11.1.4. Gib ein Pythonskript als Lösung ab, das die Plots erzeugt:

- 11.3.1 (1 Punkt) Endliche quadratische Senke ($N = 50$, $L = 10$):
$$V(x) = \begin{cases} \infty & \text{, wenn } x < 0 \lor x > L \\ 2 & \text{, wenn } 0 < x < \frac{1}{5}L \lor \frac{3}{5}L < x < L \\ 0 & \text{, sonst} \end{cases}$$
- 11.3.2 (1 Punkt) Doppelte quadratische Senke ($N = 50$, $L = 10$):
$$V(x) = \begin{cases} \infty & \text{, wenn } x < 0 \lor x > L \\ 2 & \text{, wenn } \frac{2}{5}L < x < \frac{3}{5}L \\ 0 & \text{, sonst} \end{cases}$$
- 11.3.3 (1 Punkt) Harmonisches Potential ($N = 50$, $L = 10$):
$$V(x) = \begin{cases} \infty & \text{, wenn } x < 0 \lor x > L \\ \frac{1}{10}(x - \frac{L}{2})^2 & \text{, sonst} \end{cases}$$