

Quantenmechanik von 1D-Potentialen II:

Zeitentwicklung

Aufgabe 7.1:

(16 Punkte)

Betrachten Sie für ein Gauß'sches Wellenpaket

$$\varphi(x, t=0) = \frac{1}{(2\pi(\Delta x)^2)^{1/4}} \exp\left(-\frac{(x-x_0)^2}{4(\Delta x)^2}\right) \exp\left(\frac{i}{\hbar_{\text{eff}}} p_0 x\right)$$

mit mittlerem Ort x_0 und mittlerem Impuls p_0 die Zeitentwicklung in einem asymmetrischen Doppelmuldenpotential, $V(x) = x^4 - x^2 + Ax$, für Startpunkte x_0 , die mit der Maus gewählt werden. Verwenden Sie für die anderen Parameter $p_0 = 0$, $A = 0,055$, $\hbar_{\text{eff}} = 0,06$ und $\Delta x = 0,1$.

- Geben Sie ein Programm ab, das die Funktionen des Moduls `quantenmechanik.py` der Musterlösung zu Blatt 6 nutzt. Hierzu soll dieses Modul mittels

```
import quantenmechanik as qm
```

in Ihr Programm importiert werden, so dass z.B. mit `qm.diskretisierung` auf die Funktionen im Modul zugegriffen werden kann.

- Berechnen Sie die Entwicklungskoeffizienten c_n der Entwicklung des Wellenpaketes φ in die Eigenfunktionen. Konstruieren Sie aus diesen c_n das Wellenpaket $\tilde{\varphi}$ und überprüfen Sie, ob dieses das ursprüngliche Wellenpaket rekonstruiert, indem Sie die Norm der Differenz $\varphi - \tilde{\varphi}$ berechnen und für jedes Anfangswellenpaket auf der Konsole ausgeben.
- Berechnen Sie den Energieerwartungswert $\langle \varphi | H | \varphi \rangle$ unter Verwendung der Entwicklungskoeffizienten c_n .
- Zur grafischen Darstellung zeichnen Sie das Betragsquadrat des zeitentwickelten Wellenpakets dynamisch als Funktion der Zeit in Höhe des Energieerwartungswertes. Zeichnen Sie zusätzlich das Betragsquadrat der Eigenfunktionen (mit $E_n < 0,15$) auf Höhe ihrer Eigenenergie in das Potential ein.
- Geben Sie Ihr Programm für obige Parameter ab.
- Diskutieren Sie im Kommentar am Ende des Programms folgende Punkte:
 - a) Welche Bewegung beobachten Sie beim Start des Wellenpaketes im Minimum bzw. im Maximum?
 - b) Was beobachten Sie für $p_0 = 0.3$ beim Start des Wellenpaketes im Minimum bzw. im Maximum?
 - c) Untersuchen Sie den Fall $A = 0$ für sehr große Zeiten ($t_{\text{max}} = 10000$), wenn das Wellenpaket (mit $p_0 = 0$) in einem der Minima gestartet wird.

Bonus (4 Punkte)

Zu c): Berechnen Sie analytisch die auftretende Zeitskala für die Superposition (mit gleichen Anteilen) der beiden niedrigsten Eigenfunktionen. Schätzen Sie diese Zeitskala unter Verwendung der Eigenwerte ab und vergleichen Sie mit der numerisch für das Gauß'sche Wellenpaket erhaltenen Zeit.

Vorgaben und Hinweise:

- ❶ Entwickeln und testen Sie Ihr Programm am harmonischen Oszillator (mit $\Delta x = 1$). Vergleichen Sie die numerischen Entwicklungskoeffizienten mit den analytisch bekannten.

Anmerkung: für den harmonischen Oszillator behält das Wellenpaket immer seine Gauß'sche Form, allerdings mit periodisch oszillierender Breite.

- ❷ Beispiel zur dynamischen Darstellung des Wellenpakets:

```
1 import numpy as np
2 import matplotlib.pyplot as plt
3 import functools
4
5 def start_sinus(event, ax, phi_t):
6     """Plotte Sinus-Kurve."""
7     x = np.linspace(0.0, 2.0*np.pi, 100)
8     sinus = ax.plot(x, np.sin(x-phi_t[0])) # Anfangsplot
9     for phi in phi_t[1:]:
10        sinus_t = np.sin(x-phi)           # Neue Daten
11        sinus[0].set_ydata(sinus_t)      # Plotdaten aktualisieren,
12        event.canvas.flush_events()      # und dynamisch
13        event.canvas.draw()              # darstellen.
14
15 def main():
16     """Hauptprogramm."""
17     phi_t = np.linspace(0.0, 6.0, 100)   # Phasenverschiebung
18
19     fig = plt.figure()
20     ax = fig.add_subplot(1, 1, 1)
21     ax.set_xlabel("x")
22     ax.set_ylabel("y")
23     ax.set_title("Maustaste zum Starten klicken")
24     klick_funktion = functools.partial(start_sinus, ax=ax,
25                                       phi_t=phi_t)
26     fig.canvas.mpl_connect("button_press_event", klick_funktion)
27     plt.show()
28
29 if __name__ == "__main__":
30     main()
```

- ❸ Zu Arrays mit komplexen Zahlen als Einträgen siehe Einführung II, Abschnitt 3.