

double_pendulum_exercise_0

May 13, 2025

1 Doppelpendel

1.0.1 Dein Vor- und Nachname: XXX XXX

1.1 Ergänze wo ein XXX steht

1.1.1 Wichtige Module

```
In [1]: import numpy as np
import matplotlib.pyplot as plt
from scipy.integrate import solve_ivp
```

Matplotlib is building the font cache; this may take a moment.

1.1.2 Parameter

```
In [2]: # Parameters
g = 9.81          # gravity acceleration
l = 1.0           # pendulum arm length
m = 1.0           # mass of each pendulum
```

1.1.3 Bewegungsgleichung in Matrixform

```
In [3]: def eq_of_motion(t, y):
    th1, th2, w1, w2 = y
    delta = th1 - th2
    sin_d = np.sin(delta)
    cos_d = np.cos(delta)

    # Equation of motion matrix
    A = np.array([[XXX, XXX],
                  [XXX, XXX]])

    # Right-hand side vector (students should derive this)
    b = np.array([XXX, XXX])

    acc = np.linalg.solve(A, b)
    return [w1, w2, acc[0], acc[1]]
```

1.1.4 Numerische Lösung der Differentialgleichung

```
In [4]: y0 = [XXX, XXX, XXX, XXX] # Initial angles and velocities
        t_span = (0, 10)
        t_eval = np.linspace(*t_span, 1000)

        sol = solve_ivp(eq_of_motion, t_span, y0, t_eval=t_eval)
```

NameError

Traceback (most recent call last)

```
Cell In[4], line 1
----> 1 y0 = [XXX, XXX, XXX, XXX] # Initial angles and velocities
      2 t_span = (0, 10)
      3 t_eval = np.linspace(*t_span, 1000)
```

NameError: name 'XXX' is not defined

1.1.5 Zeitlicher Verlauf der Winkel

```
In [5]: th1, th2, w1, w2 = sol.y
        fig, axs = plt.subplots(nrows=1, ncols=2, figsize=(9,3))

        axs[0].plot(sol.t, th1)
        axs[0].set_xlabel(r'$t$')
        axs[0].set_ylabel(r'$\varphi_1$')

        axs[1].plot(XXX, XXX)
        axs[1].set_xlabel(r'$t$')
        axs[1].set_ylabel(r'$\varphi_2$')
        plt.show()
```

NameError

Traceback (most recent call last)

```
Cell In[5], line 1
----> 1 th1, th2, w1, w2 = sol.y
      2 fig, axs = plt.subplots(nrows=1, ncols=2, figsize=(9,3))
      4 axs[0].plot(sol.t, th1)
```

NameError: name 'sol' is not defined

1.1.6 Energien als Funktion der Zeit

```
In [6]: th1, th2, w1, w2 = sol.y
        K = XXX
        V = XXX
        E = XXX
```

```
plt.figure()
plt.plot(XXX, XXX)
plt.plot(XXX, XXX)
plt.plot(XXX, XXX)
plt.xlabel(r'$t$')
plt.ylabel('Energy')
plt.grid(True)
plt.show()
```

NameError

Traceback (most recent call last)

```
Cell In[6], line 1
----> 1 th1, th2, w1, w2 = sol.y
      2 K = XXX
      3 V = XXX
```

NameError: name 'sol' is not defined

1.1.7 Phasenraumplot

```
In [7]: th1, th2, w1, w2 = sol.y
        fig, axs = plt.subplots(nrows=1, ncols=2, figsize=(9,3))
```

```
axs[0].plot(th1, w1)
axs[0].set_xlabel(r'$\varphi_1$')
axs[0].set_ylabel(r'$\dot{\varphi}_1$')

axs[1].plot(th2, w2)
axs[1].set_xlabel(r'$\varphi_2$')
axs[1].set_ylabel(r'$\dot{\varphi}_2$')
plt.show()
```

NameError

Traceback (most recent call last)

```

Cell In[7], line 1
----> 1 th1, th2, w1, w2 = sol.y
      2 fig, axs = plt.subplots(nrows=1, ncols=2, figsize=(9,3))
      4 axs[0].plot(th1, w1)

```

NameError: name 'sol' is not defined

1.1.8 Erzeugung einer Animation

```
In [8]: from matplotlib.animation import FuncAnimation
```

```
In [9]: # Cartesian coordinates
        th1, th2 = sol.y[0], sol.y[1]
        x1 = XXX
        y1 = XXX
        x2 = XXX
        y2 = XXX

```

NameError

Traceback (most recent call last)

```

Cell In[9], line 2
      1 # Cartesian coordinates
----> 2 th1, th2 = sol.y[0], sol.y[1]
      3 x1 = XXX
      4 y1 = XXX

```

NameError: name 'sol' is not defined

```
In [10]: # Animation
         fig, ax = plt.subplots(figsize=(5, 5))
         ax.set_xlim(-2.2, 2.2)
         ax.set_ylim(-2.2, 2.2)
         ax.set_aspect('equal')
         line1, = ax.plot([], [], 'r-', lw=2)
         line2, = ax.plot([], [], 'b-', lw=2)
         bob1, = ax.plot([], [], 'ro', markersize=8)
         bob2, = ax.plot([], [], 'bo', markersize=8)

         def init():
             return line1, line2, bob1, bob2

         def update(i):

```

```

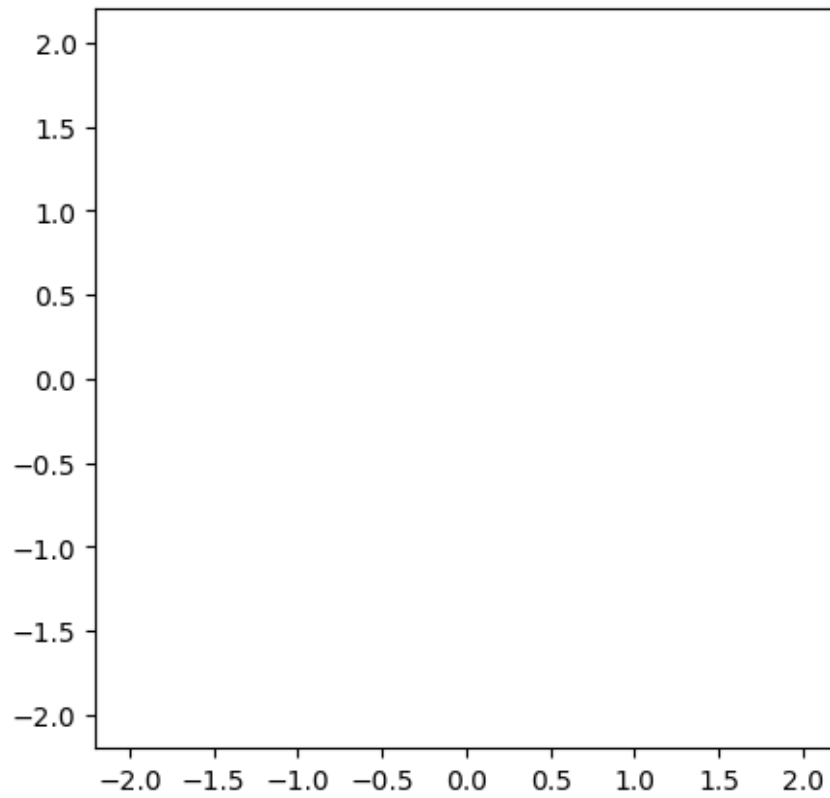
line1.set_data([0, x1[i]], [0, y1[i]])
line2.set_data([x1[i], x2[i]], [y1[i], y2[i]])
bob1.set_data([x1[i]], [y1[i]])
bob2.set_data([x2[i]], [y2[i]])
return line1, line2, bob1, bob2

```

```

ani = FuncAnimation(fig, update, frames=len(t_eval), init_func=init, blit=
ani.save('double_pendulum.gif', writer='pillow', fps=30)

```



NameError

Traceback (most recent call last)

```

Cell In[10], line 21
    18     bob2.set_data([x2[i]], [y2[i]])
    19     return line1, line2, bob1, bob2
--> 21 ani = FuncAnimation(fig, update, frames=len(t_eval), init_func=init, bl
    22 ani.save('double_pendulum.gif', writer='pillow', fps=30)

```

```
NameError: name 't_eval' is not defined
```

```
In [ ]:
```