Aim

To simulate and visualize the dynamics of a double pendulum system with a spring.

Objective

- To model the motion of a double pendulum system using Lagrangian mechanics.
- To solve the resulting differential equations numerically.
- To visualize the motion of the pendulum and spring system through animation.

Summary

This project involves the simulation of a double pendulum system where the second pendulum is connected to a spring. Using Lagrangian mechanics, we derive the equations of motion for the system, which are then solved numerically. The results are visualized through an animated plot showing the movement of the pendulum and the spring over time.

Tools and Libraries Used

- **NumPy**: For numerical operations.
- **Matplotlib**: For plotting and animation.
- **SciPy**: For solving ordinary differential equations (ODEs).
- **SymPy**: For symbolic mathematics and deriving Lagrange's equations.

Procedure

1. **Define Symbols and Coordinates**:
    - Use SymPy to define symbols for time, gravitational constant, lengths, masses, spring constant, and initial conditions.
    - Express the positions of the masses in terms of the angles of the pendulums and use trigonometric functions for coordinates.
2. **Compute Kinetic and Potential Energy**:
    - Calculate the kinetic and potential energies of the system using the mass positions and velocities.
    - Define the Lagrangian as the difference between kinetic and potential energy.
3. **Derive Lagrange's Equations**:
    - Derive the equations of motion (Lagrange’s equations) from the Lagrangian.
4. **Solve Differential Equations**:
    - Convert the second-order differential equations to a system of first-order equations.
    - Use SymPy to convert symbolic expressions to numerical functions.

- Implement the system of ODEs in a Python function.

5. **Numerical Integration**:
    - Solve the system of ODEs using SciPy's odeint method to obtain the angles and angular velocities over time.
6. **Visualization**:
    - Plot the angles over time using Matplotlib.
    - Define a function to convert angles to mass coordinates.
    - Create an animation of the pendulum system showing the motion of the masses.

Highlights

- **Lagrangian Mechanics**: The use of Lagrangian mechanics to derive the equations of motion provides a powerful framework for dealing with complex mechanical systems.
- **Numerical Solutions**: Solving the differential equations numerically using odeint enables the handling of non-linear systems.
- **Animation**: Visualization of the dynamic behavior of the pendulum and spring system provides an intuitive understanding of the system's behavior.

Conclusion

The project successfully simulates and visualizes the motion of a double pendulum system with a spring. By employing Lagrangian mechanics and numerical integration, the system's dynamics were accurately modeled and visualized. The animation provides clear insights into the behavior of the pendulum and spring system, demonstrating the complex interactions and movements.