Double Pendulum Simulation

A Chaotic Classical Mechanics Benchmark

Joseph Barton

February 2026

Abstract

This document derives the equations of motion for the planar double pendulum using Lagrangian mechanics and describes their numerical integration using `scipy.integrate.odeint`. The simulation demonstrates the characteristic chaotic behavior of the system for certain initial conditions. Source code is available in the repository. Link at end of paper.

1 Introduction

The double pendulum is a canonical example in nonlinear dynamics and computational physics. Despite its simple construction—two point masses connected by massless rods—it exhibits extreme sensitivity to initial conditions (chaos) for moderate-to-large amplitudes.

This work accompanies a Python implementation that numerically solves the system and visualizes the motion (phase space, time series, animations).

2 Theoretical Background

2.1 Lagrangian Formulation

We consider two point masses m_1 and m_2 attached to massless rods of lengths l_1 and l_2. The generalized coordinates are the angles θ_1 and θ_2 measured from the downward vertical.

The kinetic energy is

$$T = \frac{1}{2}(m_1 + m_2)l_1^2\dot{\theta}_1^2 + \frac{1}{2}m_2 l_2^2\dot{\theta}_2^2 + m_2 l_1 l_2 \dot{\theta}_1\dot{\theta}_2\cos(\theta_1 - \theta_2), \tag{1}$$

and the potential energy (zero at lowest point) is

$$V = -(m_1 + m_2)g l_1 \cos\theta_1 - m_2 g l_2 \cos\theta_2. \tag{2}$$

The Lagrangian is $L = T - V$. Applying the Euler–Lagrange equations

$$\frac{\mathrm{d}}{\mathrm{d}t}\left(\frac{\partial L}{\partial \dot{q}_i}\right) - \frac{\partial L}{\partial q_i} = 0, \qquad q_i = \theta_1, \theta_2$$

yields the coupled second-order system.

2.2 Equations of Motion

After algebraic manipulation, the accelerations are

$$\ddot{\theta}_1 = \frac{-g(2m_1 + m_2)\sin\theta_1 - m_2 g \sin(\theta_1 - 2\theta_2) - 2\sin(\theta_1 - \theta_2)m_2\big(\dot{\theta}_2^2 l_2 + \dot{\theta}_1^2 l_1 \cos(\theta_1 - \theta_2)\big)}{l_1\big(2m_1 + m_2 - m_2\cos\big(2(\theta_1 - \theta_2)\big)\big)}, \tag{3}$$

$$\ddot{\theta}_2 = \frac{2\sin(\theta_1 - \theta_2)\big(\dot{\theta}_1^2 l_1(m_1 + m_2) + g(m_1 + m_2)\cos\theta_1 + \dot{\theta}_2^2 l_2 m_2 \cos(\theta_1 - \theta_2)\big)}{l_2\big(2m_1 + m_2 - m_2\cos\big(2(\theta_1 - \theta_2)\big)\big)}. \tag{4}$$

(These are standard forms found in many references; small variations exist depending on sign conventions for potential.)

3 Numerical Solution

The second-order system is rewritten as four first-order ODEs and integrated with `scipy.integrate.odeint` (Runge–Kutta based). Typical parameters used are $m_1 = m_2 = 1\,\mathrm{kg}$, $l_1 = l_2 = 1\,\mathrm{m}$, $g = 9.81\,\mathrm{m/s^2}$.

Key results include: - Regular (periodic) motion at small angles, - Aperiodic, chaotic motion for larger initial displacements (strong dependence on initial conditions).

Trajectories, Poincaré sections, and animations are generated in the accompanying code.

4 Conclusion

The double pendulum serves as an excellent test case for understanding numerical stability, sensitivity to initial conditions, and visualization of chaotic dynamics in Python/NumPy/SciPy/Matplotlib.

Repository: `https://github.com/jbarton727/Portfolio`