

Modeling the Skydiving Problem with Quadratic Air Resistance: A Computational Approach Using Python

Mihai-Alexandru Radu

August 31, 2024

Abstract

This paper addresses the classical skydiving problem, focusing on the effects of air resistance by modeling it with a quadratic air friction force. Using Python, we derive and solve the relevant differential equations. The study highlights how air resistance influences the skydiver's motion, offering insights into key concepts such as terminal velocity and the trajectory of the fall.

Introduction

The skydiving problem is a well-known example in physics, used to demonstrate fundamental principles of Newtonian mechanics. In its simplest form, the problem is often solved without considering air resistance, leading to an idealized model of free fall. However, in realistic scenarios, air resistance plays a significant role in the motion of a falling object. This study aims to extend the classical approach by incorporating a quadratic air friction force into the model. By solving the resulting system of ordinary differential equations (ODEs) using Python, we explore the impact of air resistance on the skydiver's velocity and position over time. This approach not only provides a more accurate representation of the skydiving process but also serves as an educational tool for illustrating the application of computational methods in physics.

Methodology

Formulation of the Problem

The motion of a skydiver is governed by Newton's Second Law of Motion, which states that the net force acting on an object is equal to the mass of the object multiplied by its acceleration:

$$\vec{F} = m\vec{a} = m \cdot \begin{bmatrix} \frac{dv_x}{dt} \\ \frac{dv_y}{dt} \end{bmatrix}$$

Forces Acting on the Skydiver

In this scenario, two primary forces are considered:

1. **Gravitational Force:** The force due to gravity is given by:

$$\vec{F}_g = -m\vec{g} = -m \cdot \begin{bmatrix} 0 \\ g \end{bmatrix}$$

2. **Air Resistance (Friction):** The force due to air friction is assumed to be quadratic in nature and is expressed as:

$$\vec{F}_f = -b \cdot |\vec{v}| \cdot \vec{v} = -b \cdot \sqrt{v_x^2 + v_y^2} \cdot \begin{bmatrix} v_x \\ v_y \end{bmatrix}$$

Derivation of the Differential Equations

Applying Newton's First Law of Motion, the sum of the forces acting on the skydiver is given by:

$$\vec{F}_{net} = \sum_{i=1}^n \vec{F}_i = m\vec{a}$$

This leads to the following system of differential equations describing the motion of the skydiver:

$$m \cdot \begin{bmatrix} \frac{dv_x}{dt} \\ \frac{dv_y}{dt} \end{bmatrix} = -m \cdot \begin{bmatrix} 0 \\ g \end{bmatrix} - b \cdot \sqrt{v_x^2 + v_y^2} \cdot \begin{bmatrix} v_x \\ v_y \end{bmatrix}$$

Final Form of the ODEs

The final ordinary differential equations (ODEs) governing the motion are expressed as:

$$\begin{aligned} \frac{dv_x}{dt} &= -\frac{b}{m} \cdot \sqrt{v_x^2 + v_y^2} \cdot v_x \\ \frac{dv_y}{dt} &= -g - \frac{b}{m} \cdot \sqrt{v_x^2 + v_y^2} \cdot v_y \end{aligned}$$

Results and Discussion

The derived equations were implemented in Python, using numerical methods to simulate the skydiver's descent. The results demonstrate the impact of air resistance on the velocity and trajectory of the skydiver. A comparison with the idealized case (no air resistance) highlights the significance of frictional forces in real-world scenarios. The analysis provides insights into terminal velocity and the time required for the skydiver to reach a steady state.

Conclusion

This study successfully models the skydiving problem with air resistance, offering a detailed computational approach to understanding the dynamics involved. The use of Python for solving the ODEs proves effective, making complex physical scenarios more accessible for analysis. Future work could extend this model to account for additional factors such as variable air density or wind effects.

References

- [1] Luke Polson, *Python Metaphysics Tutorials*, Unpublished.