A Literature Review on UAVs in Low Pressure Atmospheres

Ansha Kottapalli

American High School, Polygence Student Research Project

Abstract

Over the years, there has been a rapid development of UAVs, but there is still limited research on their function in low pressure/high altitude systems. This is mostly because of limited access to low pressure testing environments, technological constraints, and high research and development costs. However, UAVs in low pressure environments can be extremely beneficial for space exploration and even search-and-rescue missions on high peaks like Mount Everest. In such environments, reduced air density makes it difficult for UAVs to maintain lift. [1] Specially designed UAVs, however, could provide aerial support as well as payload delivery. To facilitate the design of UAVs capable of flying in low pressure environments, this paper investigates previous related missions and lays the groundworks for dynamics and control of these unique vehicles.

1. Introduction

Unmanned aerial drones (UAVs), commonly known as drones, have evolved from first being used for military purposes and recreational uses to now being specialized for cutting-edge planetary exploration and other industries. Operating drones in low pressure atmospheres like Mars and high altitude locations on Earth presents its own set of challenges. As pressure, temperature, and air density decreases with increasing altitude, drones will need to adapt through their structural design, propulsion, and energy management systems to perform effectively. By exploring the FlyCart 30, a drone developed by DJI that carried two oxygen tanks from Base Camp to Camp 1, and the Ingenuity Mars Helicopter, an autonomous helicopter developed by NASA, insight can be gained into how UAVs can be engineered to perform in low-pressure environments, both on Earth and on other planets with similar conditions. [1, 2, 3]

This paper explores the control and dynamic model of the spring-mass-damper system and the 2D Quadrotor model. Through a SPYDER Python simulation, the behaviors of the systems changed as the dampening coefficient, initial conditions, and other control strategies were altered.

2. Problem Statement

Controlling system dynamics can be challenging when the system dynamics exhibit extreme changes under specific operating conditions, specifically under low-pressure control. The variation in dynamics makes it necessary to have adaptive control strategies, since the fixed system gains may not be optimal. The system gains must be adjusted to maintain desired performance and stability. For example, a quadcopter that's intended to hover at sea level on Earth may be unstable at very high altitude because of the significantly lower air density. The control gains would need to be adjusted to compensate for the reduced aerodynamic forces so that the quadcopter can maintain its position and orientation. The simulation results in this paper illustrate how the net thrust and torque need to be altered to get the desired results.

3. Drones in Extraterrestrial Low-Pressure Environments

NASA's Ingenuity Helicopter

Mars has a notably thin atmosphere, about 1% of Earth's, which makes generating lift extraordinarily difficult.[2] NASA's Ingenuity helicopter, however, overcame this by developing a lightweight, carbon-fiber design and fast-spinning coaxial rotors (about 2,500 rpm), allowing autonomous flight on Mars despite air density of about 0.017 km/m^3 and temperature of around -60 degrees Celsius. [2, 7] This proves that flight in an extremely low pressure atmosphere is feasible with innovative rotorcraft engineering and autonomous navigation.

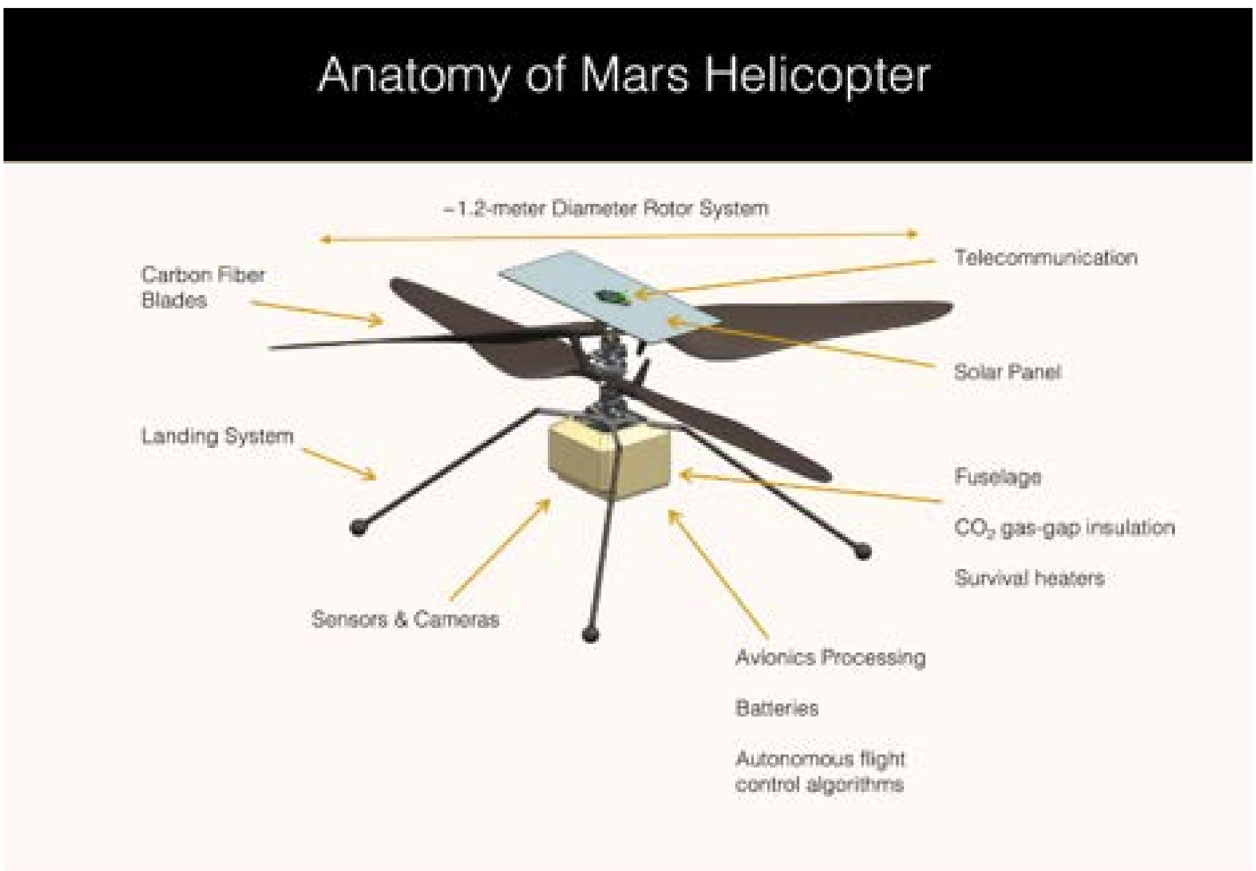

Figure 1: Mars Helicopter (NASA/JPL - Caltech) This figure highlights the structure of the Mars Helicopter. As noted previously, the Ingenuity Helicopter has carbon fiber blades, making this helicopter lightweight and easier to maneuver. The fuselage, which is about the size of a tissue box, is packed with computers, batteries, sensors, heaters, and telecommunications. [5]

Dragonfly

Conversely, one of Saturn's moon Titan has an atmosphere 4.4 times denser than Earth's, extremely low gravity of about 1.35 m/s^2, and a mean surface temperature of 94 K, approximately -179 degrees Celsius (Lorenz et al., 2018). Although these conditions favor generation of lift, thermal protection and a robust energy system is required to prevent failure due to freezing. The Dragonfly rotorcraft lander, designed by Johns Hopkins Applied Physics Laboratory, will be powered by a radioisotope thermoelectric generator and it will be equipped with sampling drills and onboard mass spectrometry to explore Titan's organic-rich surface (Ata, 2021; Lorenz et al., 2018).

Figure 2: Dragonfly (Thor Design) This figure has different parts of the Dragonfly labeled, describing each part's functionality. Unlike the Ingenuity Helicopter, Dragonfly doesn't utilize solar panels and instead utilized a radioisotope thermoelectric generator. [6]

4. High Altitude UAV Tests on Earth

Engineers and researchers have also turned to the Himalayas to test drones in low-pressure systems. In 2022, Da-Jiang Innovations (DJI) successfully conducted drone delivery trials on Mount Everest, the world's tallest mountain with an atmospheric pressure of about one-third of the atmospheric pressure at sea level. [1] These trials demonstrated the ability to carry oxygen tanks and other supplies between Base Camp and Camp 1, while operating in temperatures below freezing and air densities comparable to that of Mars. [3]

DJI engineers had to account for reduced thrust, shorter battery life, and the difficulty of maintaining GPS signal at extreme altitudes.

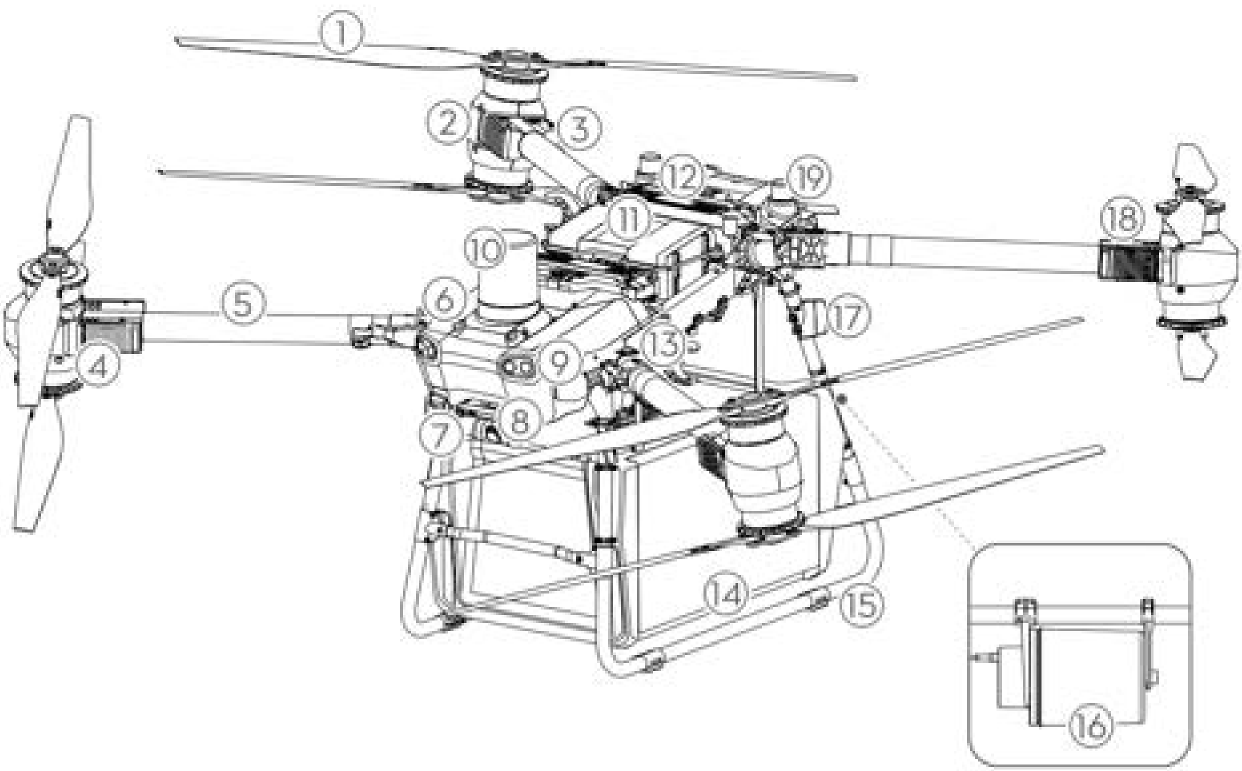

Figure 3 (DJI) This figure shows how the FlyCart 30 looks. For example, circle one is the propellers of the drone, and circle fourteen is where the cargo boxes go. [4]

5. Problem Statement

Controlling system dynamics can be challenging when the system dynamics exhibit extreme changes under specific operating conditions, specifically under low-pressure control. The variation in dynamics makes it necessary to have adaptive control strategies, since the fixed system gains may not be optimal. The system gains must be adjusted to maintain desired performance and stability. For example, a quadcopter that's intended to hover at sea level on Earth may be unstable at very high altitude because of the significantly lower air density. The control gains would need to be adjusted to compensate for the reduced aerodynamic forces so that the quadcopter can maintain its position and orientation. The simulation results in this paper illustrate how the net thrust and torque need to be altered to counteract gravity and maintain a desired trajectory. This was demonstrated in SPYDER using Python with the equations that will be derived in the following section.

6. Methodology

This study uses two models, the spring-mass-damper system and the 2D Quadcopter Model to simulate how control principles apply to dynamic systems from a foundational model to a more complex, nonlinear model. The spring-mass-damper system is a simple, linear system that is ideal for understanding basic control topics like dampening and stability. The 2D Quadcopter Model, on the other hand, is a nonlinear system that demonstrates the actual complexities of UAV flight. It would require advanced control in order to maintain stability. However, if linearized, PID would suffice.

Spring-Mass-Damper System:

The derivation of the equations of motion for the Spring-Mass-Damper system begins with Newton's Second Law.

m = mass

a = acceleration

F = force

$$\sum F = ma$$

In Figure 4 the constant c is the damping force, and the constant k is the spring constant. Both the damping force and the spring force are acting on this mass, making it the total net force on the mass. This means the equation can be rewritten as:

$F_s + F_d = ma$ (1)

The forces acting on the mass in the system are:

1. Spring force (Fs): According to Hooke's Law, where k is the spring constant and x is the displacement from the equilibrium. This force would be negative because the spring force acts in the opposite direction of the object's displacement.

   $F_s = -kx$ (2)

2. Damping force (Fd): The damping force is proportional to the velocity, where c is the damping coefficient and ẋ is the velocity. The damping force works to dissipate the object's motion and energy.

   $F_d = -c\dot{x}$ (3)

Substitute equations (2) and (3) into equation (1) to get:

$-kx - c\dot{x} = ma$ (4)

Since acceleration a is the second derivative of position with respect to time

$a = \ddot{x}$ (5)

Substitute Equation (5) into Equation (4) to get

$-kx - c\dot{x} = m\ddot{x}$ (6)

This is the **Equation of Motion**.

Formulating **Equation of Motion** in state-space form:

$$\text{State vector: } \vec{x} = \begin{bmatrix} x \\ \dot{x} \end{bmatrix}$$

$$\textbf{Derivative of state vector:}[\dot{\vec{x}} = \frac{d}{dt}\begin{bmatrix} \dot{x} \\ x \end{bmatrix} = \begin{bmatrix} \ddot{x} \\ \dot{x} \end{bmatrix}]$$

Solve for $\ddot{x}$ in equation (6) to get:

$$\ddot{x} = -\frac{k}{m}x - \frac{c}{m}\dot{x}$$

$$\dot{\vec{x}} = A\vec{x} = \begin{bmatrix} 0 & 1 \\ -\frac{k}{m} & -\frac{c}{m} \end{bmatrix} \begin{bmatrix} x \\ \dot{x} \end{bmatrix}$$

A is the system matrix

2D Quadcopter Model

The equations of motion for the 2D Quadcopter Model are derived from Newton's Second Law and for translational dynamics in the y-z plane. It is analyzed by its motion in the z-axis, y-axis, and rotational planes.

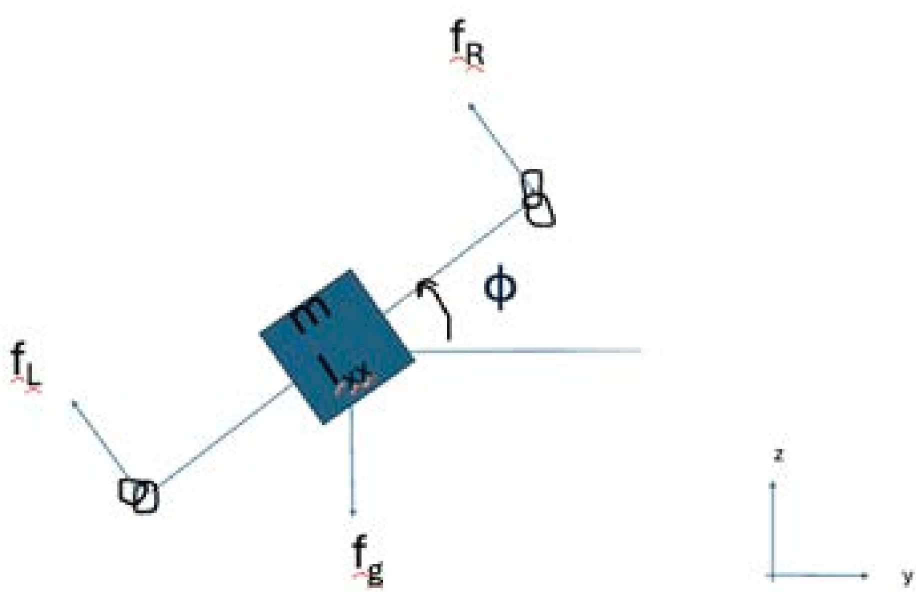

Figure 5: This figure represents the 2D quadrotor with fL and fR representing the forces, fg representing the quadrotor's weight, ϕ representing its orientation above the horizontal, m representing the quadrotor's mass, and finally Ixx representing its rotational inertia.

System Parameters:

$$\begin{aligned} m &= \text{mass of quadcopter} \\ I_{xx} &= \text{moment of inertia about the} x\text{-axis (roll axis)} \\ L &= \text{distance from the center of mass to the propeller (lever arm for torque)} \\ f_L &= \text{thrust of the left propeller} \\ f_R &= \text{thrust of the right propeller} \\ F_g &= \text{force of gravity} \\ \phi &= \text{roll angle (angle of the quadcopter body with respect to the horizontal)} \end{aligned}$$

Forces & Torques

The control inputs for the 2D quadcopter are the net force and the net torque. The net force comes from the right propeller and the left propeller. The net torque comes from the difference of forces of the left propeller and right propeller multiplied by the lever arm.

$f_{tot} = f_L + f_R$ (net force)

$\tau_{tot} = (f_L - f_R)L$ (net torque)

Equations of Motion

$$1) \quad -f_{\text{tot}} \sin\phi = m\frac{d^2y}{dt^2}$$

- (Y-axis) The horizontal component of the total thrust is responsible for the motion along the y-axis. Additionally, $\sum F_y = md^2y/dt^2$ by Newton's Second Law because the double derivative is acceleration.

2) $f_{\text{tot}} \cos\phi - F_g = m\frac{d^2z}{dt^2}$

- (Z-axis) The vertical component of the total thrust acts upward, opposing the force of gravity. Additionally, $\sum F_z = md^2z/dt^2$ by Newton's Second Law because the double derivative is acceleration.

3) $\tau_{tot} = Ixx\,(d^2\phi/dt^2)$

$$\vec{x}(t) = \begin{bmatrix} x_1(t) \\ x_2(t) \\ x_3(t) \\ x_4(t) \\ x_5(t) \\ x_6(t) \end{bmatrix} = \begin{bmatrix} y(t) \\ z(t) \\ \phi(t) \\ \dot{y}(t) \\ \dot{z}(t) \\ \dot{\phi} \end{bmatrix}$$

$$\text{Input vector: } \mathbf{u} = \begin{bmatrix} u_1 \\ u_2 \end{bmatrix} = \begin{bmatrix} f \\ \tau \end{bmatrix}$$

Next, rewrite the Equations of Motion in terms of the state variables and inputs

$$\dot{x}_1 = x_4$$
$$\dot{x}_2 = x_5$$
$$\dot{x}_3 = x_6$$
$$\dot{x}_4 = -\frac{1}{m}u_1 \sin(x_3)$$
$$\dot{x}_5 = -\frac{1}{m}u_1 \cos(x_3) - g$$
$$\dot{x}_6 = \frac{1}{I_{xx}}u_2$$

PID Control Law

PID (Proportional Integral Derivative) control is a control scheme that works well for linear systems, like the spring mass damper, but not necessarily for the nonlinear dynamics of the quadcopter, but it doesn't always work well for nonlinear systems like quadcopters. However, it is possible to get around this for small deviations from hover state which is achieved through linearization, where the nonlinear system is approximated as linear in a specific region.

For example, $y = sinx$ is linearized as $y = x$ at $x = 0$.

$y = cosx$ is linearized as $y = 1$ at $x = 0$

Simulation

Euler's Method is used to simulate the system dynamics. The update step for the simulation is given by: $x_{k+1} = x_k + Ax_k\mathbf{\Delta}t$

Where x_k is the state vector at time step k, A is the system matrix, and **Δ**t is the time step. For simulation accuracy, a time step of **Δ**t < 0.01 seconds is recommended. The simulation begins with an initial condition x_0 and continues for the number of time steps.

The Python code that was used for the simulation results is linked here.

7. Results

Spring-mass-damper

Without control

No Dampening (c = 0)

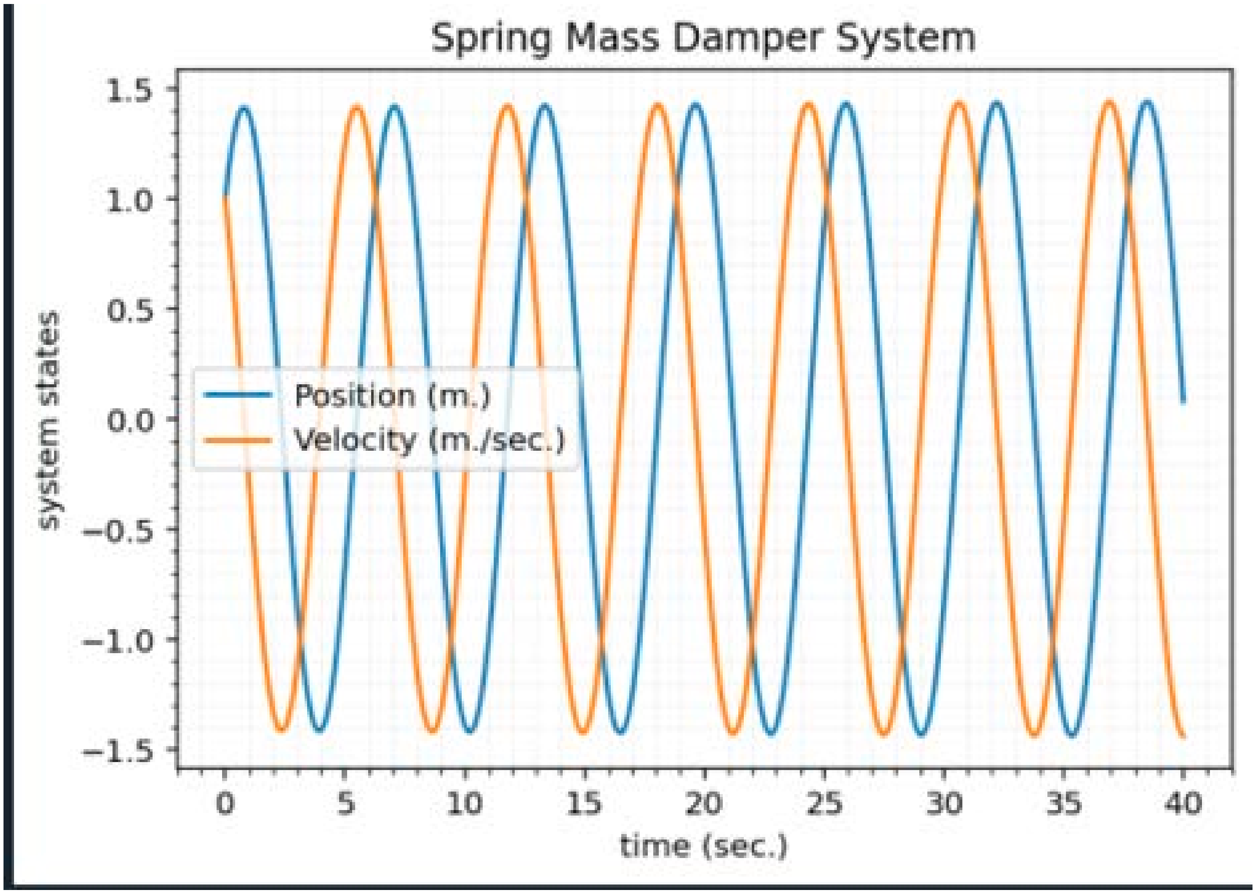

Figure 5: The graph shows the time on the x-axis and the system states on the y-axis. The system states just represents the value of the position or velocity. Velocity and position are color coded as shown.

Without any damping, the simulation showed persistent oscillations in the position and velocity graphs. The amplitude of these graphs remained constant over forty seconds and is implied it will stay constant forever which is expected since there is no energy dissipation in this system. This demonstrates a need for a control system so that the mass can stabilize and be brought to rest.

Some Dampening

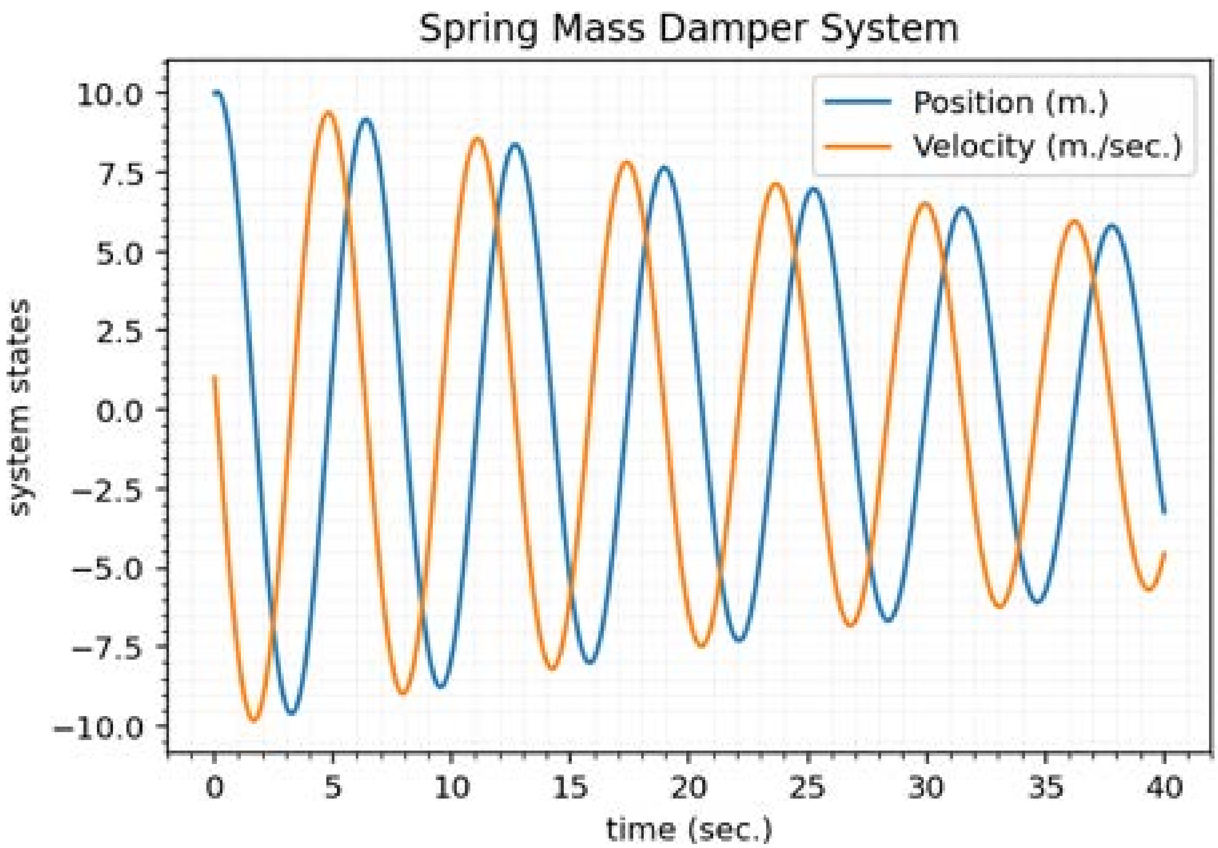

Figure 6: The graph shows the time on the x-axis and the system states on the y-axis. The system states just represents the value of the position or velocity. Velocity and position are color coded as shown.

In the situation when the damping coefficient is 0.03, both the position and velocity start decreasing slowly. The graph shows a decreasing amplitude as time goes to infinity which is expected because there is some energy dissipation in this system but not quite enough for the mass to completely stabilize fast.

Initial position further from equilibrium

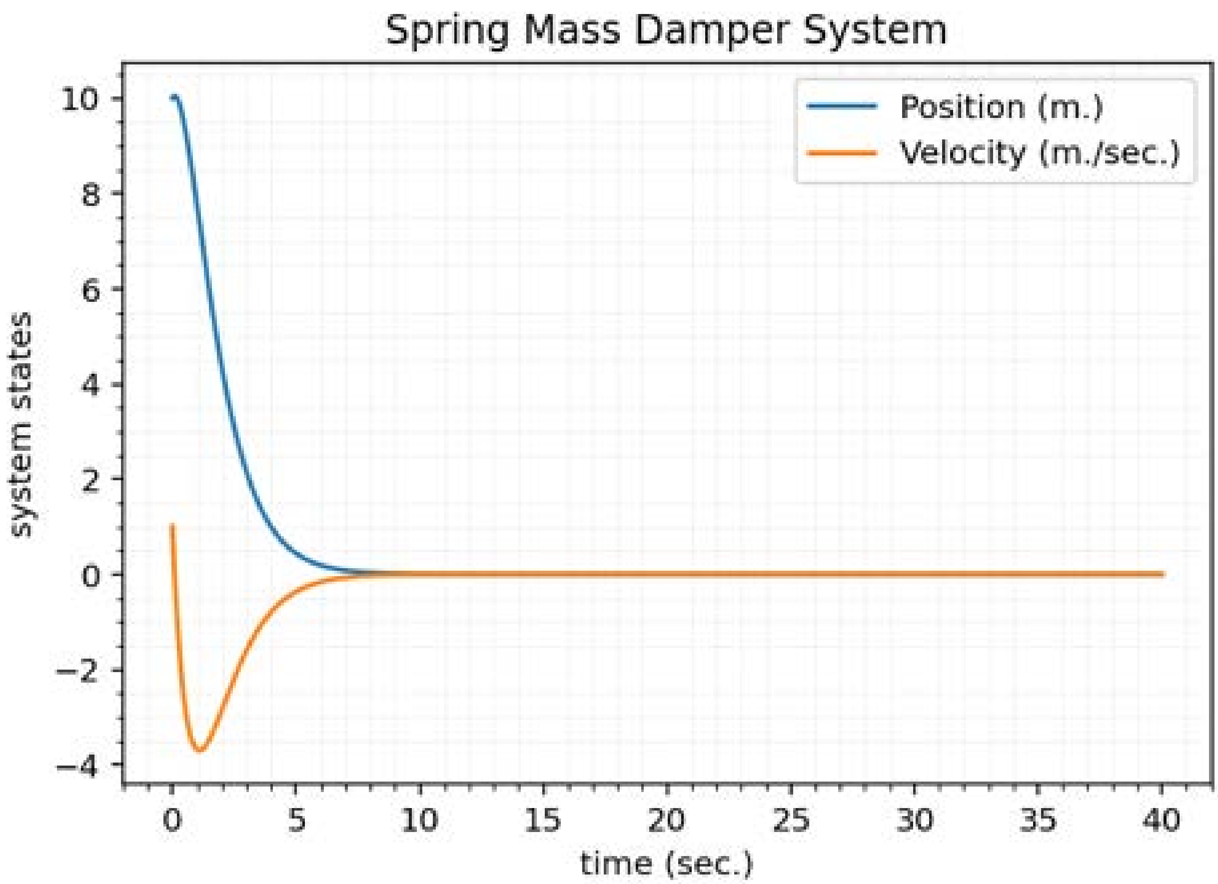

Figure 7: The graph shows the time on the x-axis and the system states on the y-axis. The system states just represents the value of the position or velocity. Velocity and position are color coded as shown.

This motion is different from the other two that were observed. Instead of the continuous oscillations, the position and velocity decayed smoothly back to the equilibrium point. The motion looks like a smooth gradual decrease until the mass comes to a complete stop.

If The Spring Would to Push Instead of Pull

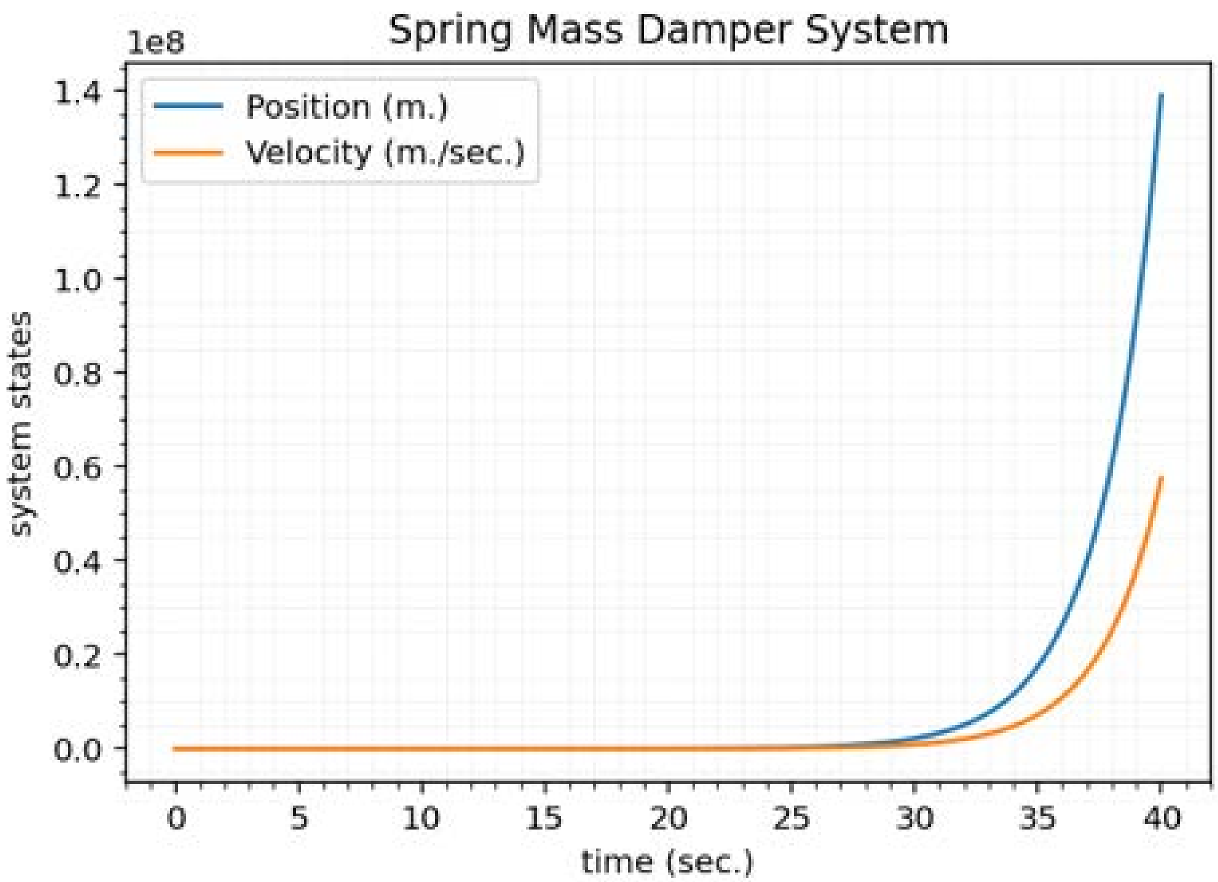

Figure 8: The graph shows the time on the x-axis and the system states on the y-axis. The system states just represents the value of the position or velocity. Velocity and position are color coded as shown.

The spring's position and velocity stayed constant for a long period of time until about t=30 where they both started increasing exponentially. This is because the spring constant is negative which indicates an unstable system because a spring is supposed to pull.

Feedback Control

The PID controller is a type of feedback control system that aims to regulate a system's output to a desired setpoint. It's the most common control system as PID comprises about 70% of all control systems. Feedback control is extremely important because it allows a system to be dynamic and adapt to disturbances. However, selecting the wrong gains can make the control system unstable which can make tuning gains a challenge. An unstable control system can lead to extreme failures like satellites going out of orbit or planes crashing. That's why observing the system's behavior with different gains is a very important step in control system design.

Deriving the PID Control Law

The PID control law is based on proportional, integral, and derivative gains. The control input force u is a function of the position, its derivative, and its integral:

$$u = k_p x + k_d \dot{x} + k_I \int x\, dt$$

For the closed-loop spring-mass-damper system, the system dynamics can be derived by including the control input force u in the net force equation. The total force is the sum of the input force, the spring force, and the damping force:

$$F_{\text{total}} = F_{\text{input}} + F_s + F_d = m\ddot{x} \quad (7)$$

Substituting Equations (2) and (3) into Equation (7) yields:

$$F_{\text{input}} - kx - c\dot{x} = m\ddot{x}$$

This equation can be rearranged to show the dynamics of the closed-loop system:

$$\ddot{x} = -\frac{k}{m}x - \frac{c}{m}\dot{x} + \frac{F_{\text{input}}}{m}$$

With Control
Lightly Damped

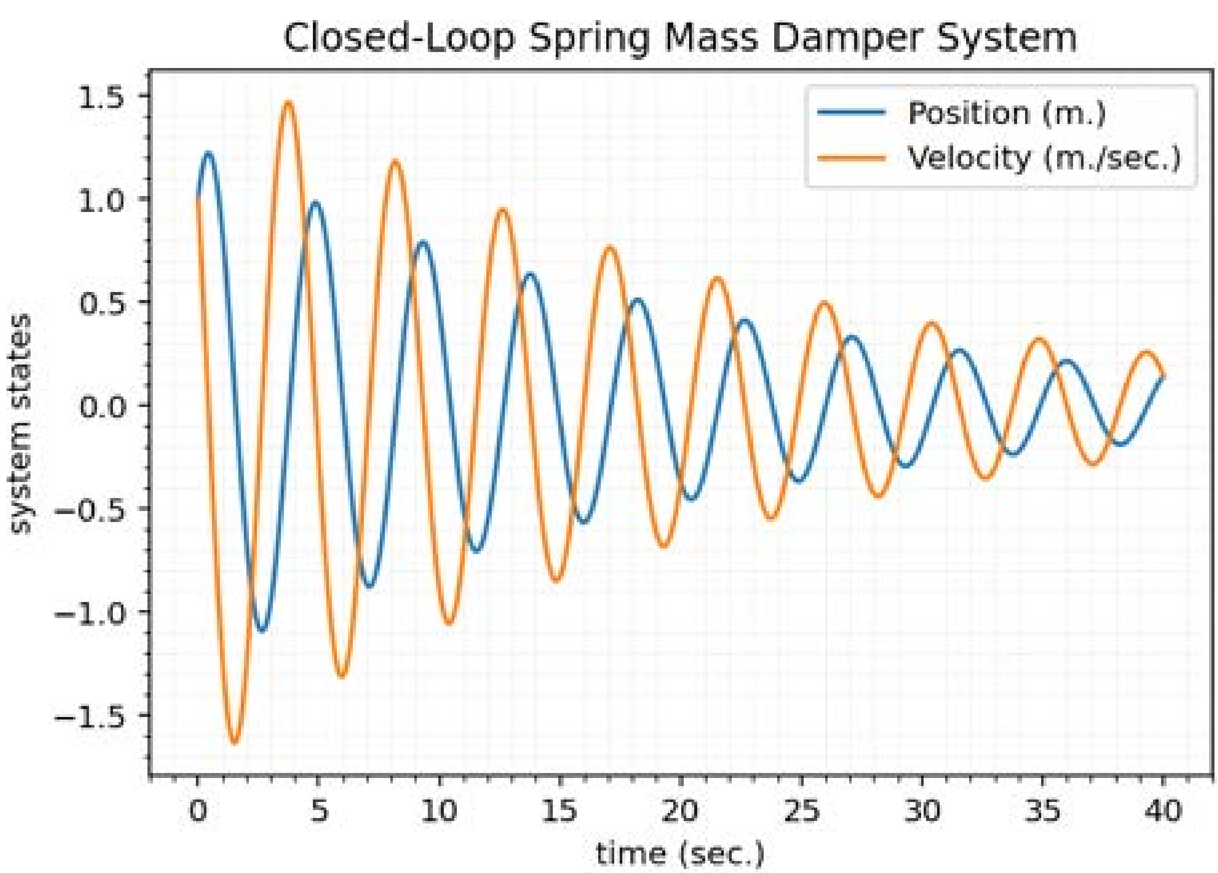

Figure 9: The graph shows the time on the x-axis and the system states on the y-axis. The system states just represents the value of the position or velocity. Velocity and position are color coded as shown.

When the proportional gain Kp was set to 1 and the derivative gain Kd to 0.1, the simulation produced a dissipating sinusoidal wave. This demonstrates that the control system was trying to bring the mass back to the equilibrium position.

High Dampening

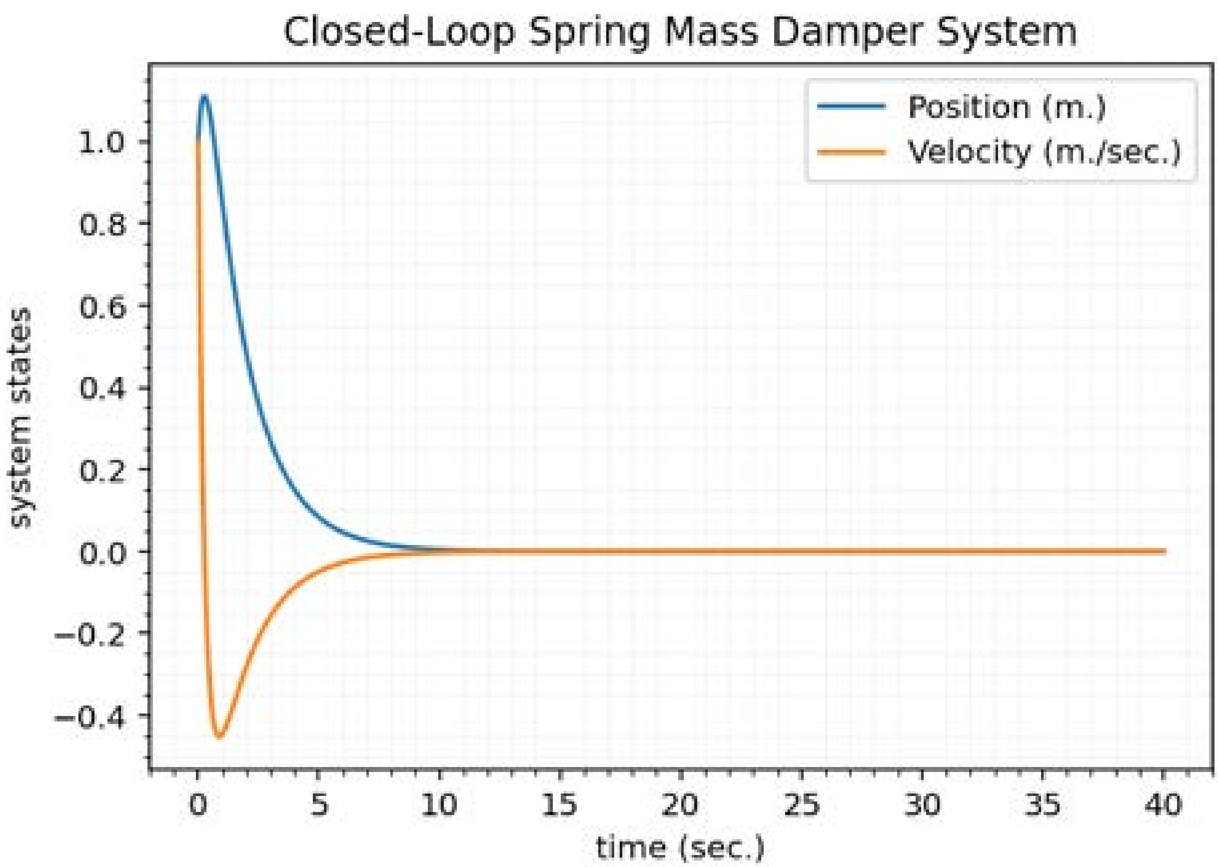

Figure 10: The graph shows the time on the x-axis and the system states on the y-axis. The system states just represents the value of the position or velocity. Velocity and position are color coded as shown.

Increasing the derivative gain Kd to 4, while keeping Kp at 1, caused the system to be highly damped. The position and velocity shows a fast decay to its equilibrium position without any oscillations. This shows that a high derivative gain can bring a system to its setpoint.

Negative Gain

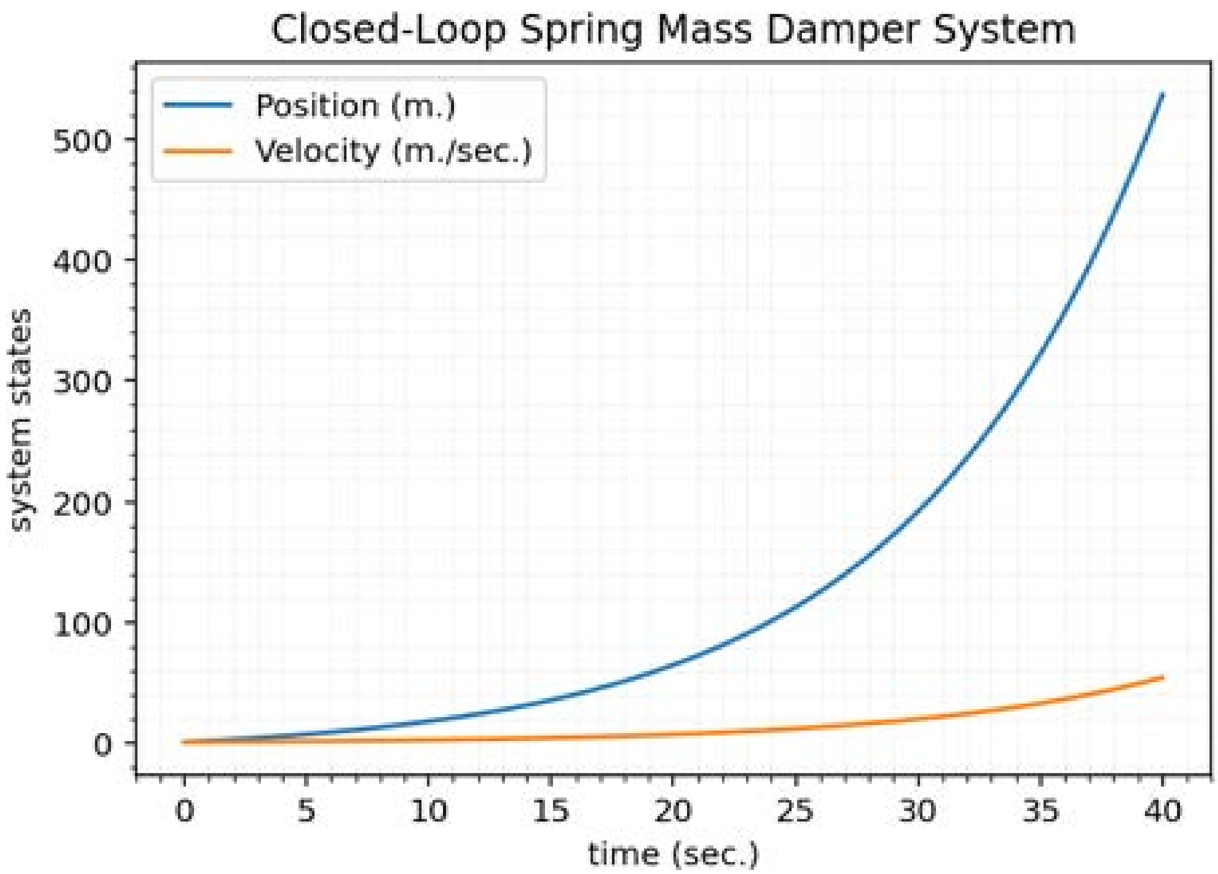

Figure 11: The graph shows the time on the x-axis and the system states on the y-axis. The system states just represents the value of the position or velocity. Velocity and position are color coded as shown.

A negative proportional gain Kp = -1 resulted in an unstable system. The simulation showed both position and velocity going to infinity. This result demonstrates that incorrect gain tuning can lead to failure, causing the system to spiral out of control.

2D Quadrotor

To explore open-loop dynamics of the quadrotor, four test cases were simulated with varying thrust u1 and torque u2.

Case 1 (u1 = 0.99*m*g, u2 = 0)

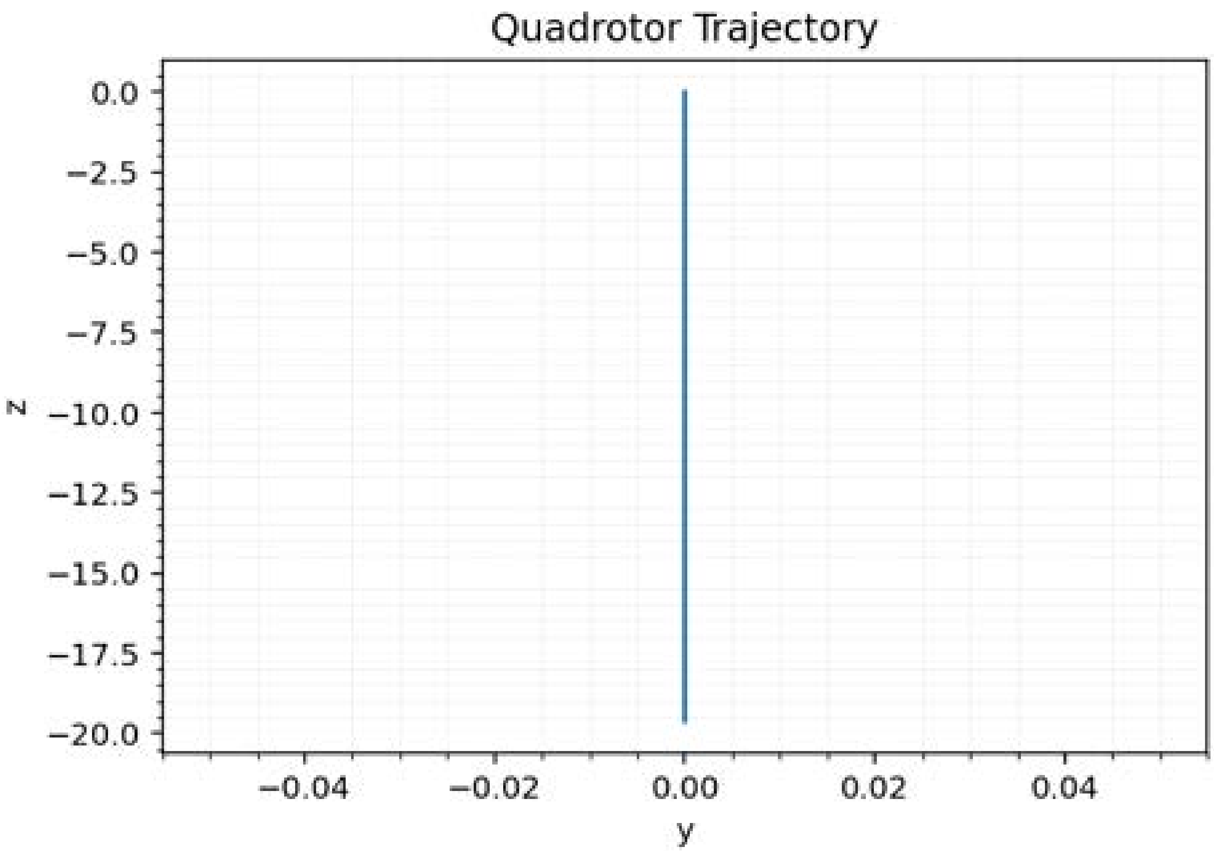

Figure 12: The x-axis would be the horizontal position of the quadrotor while the x-axis is the vertical position of the quadrotor. The graph demonstrates the quadrotor's position based on the conditions.

In this case the thrust is one percent less than the weight, so there is no tilt. The vertical acceleration is small and negative causing the quadrotor to sink slowly in a straight line downward.

Case 2 (u1 = m*g, u2 = 1)

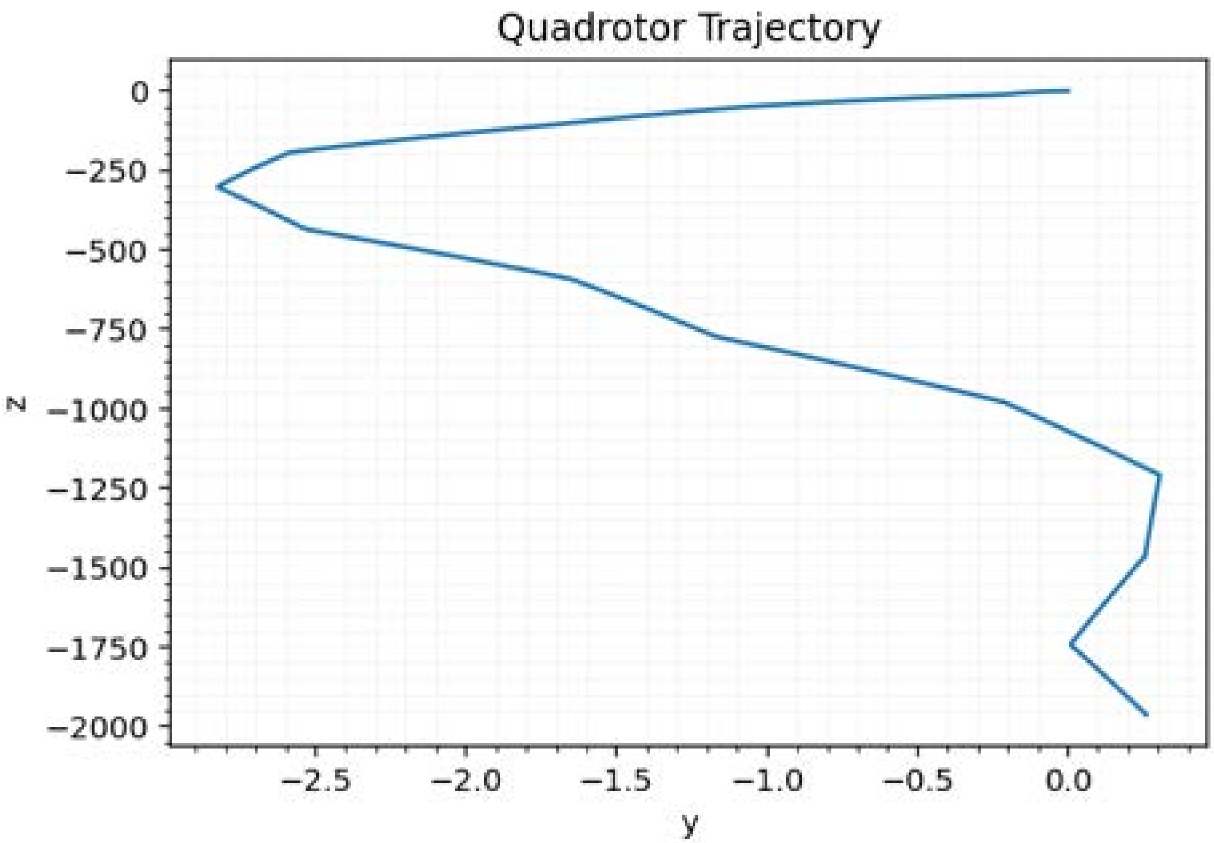

Figure 13: the x-axis would be the horizontal position of the quadrotor while the x-axis is the vertical position of the quadrotor. The graph demonstrates the quadrotor's position based on the conditions.

The thrust matches the weight, but torque is positive. Positive torque means positive phi which means the quadrotor will rotate continuously. As phi increases, the thrust gets a horizontal component, so it will accelerate sideways while slowly losing altitude because the vertical component of thrust decreases.

Case 3 (u1 = m*g, u2 = -1)

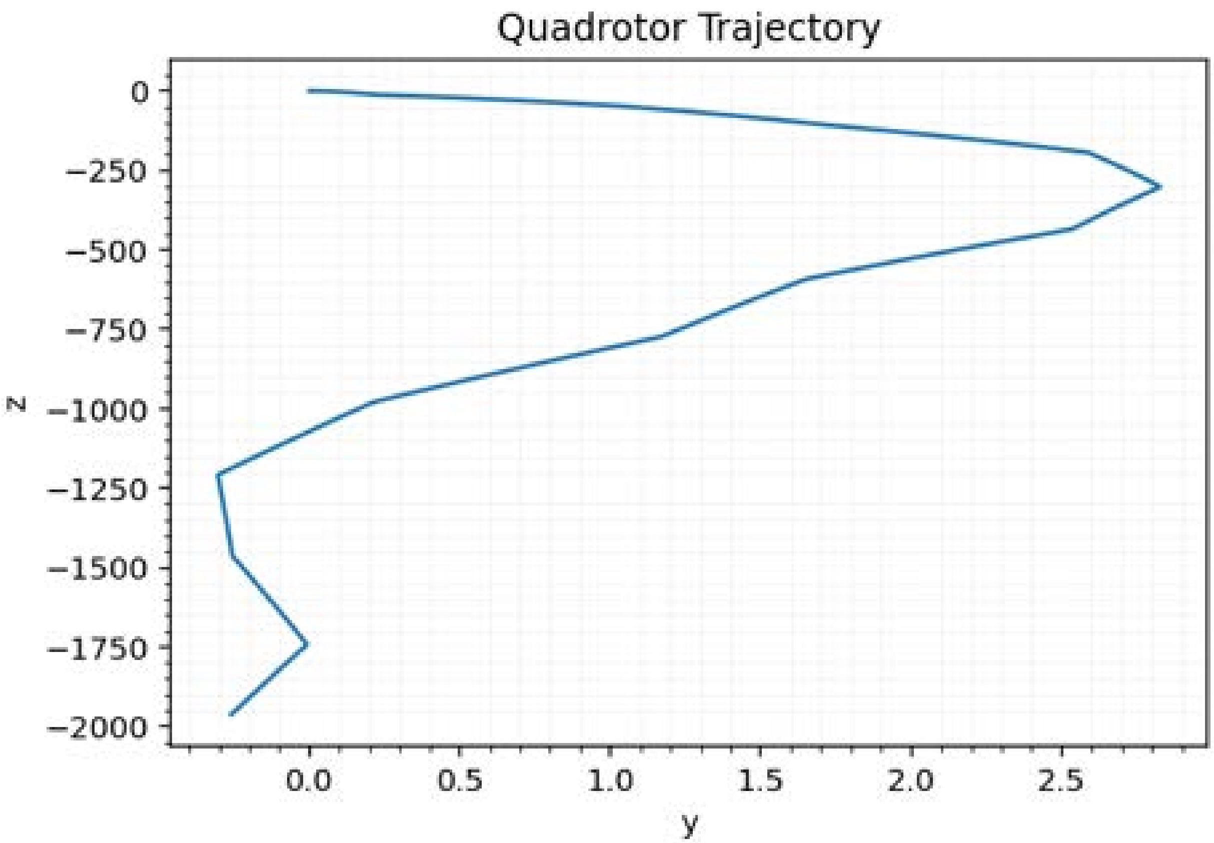

Figure 14: the x-axis would be the horizontal position of the quadrotor while the x-axis is the vertical position of the quadrotor. The graph demonstrates the quadrotor's position based on the conditions.

The thrust matches the weight, but the torque is negative. Negative torque means negative phi, so this quadrotor will rotate backward. This will cause horizontal acceleration in the opposite direction while losing altitude.

Case 4 (u1 = 2*m*g, u2 = -0.01)

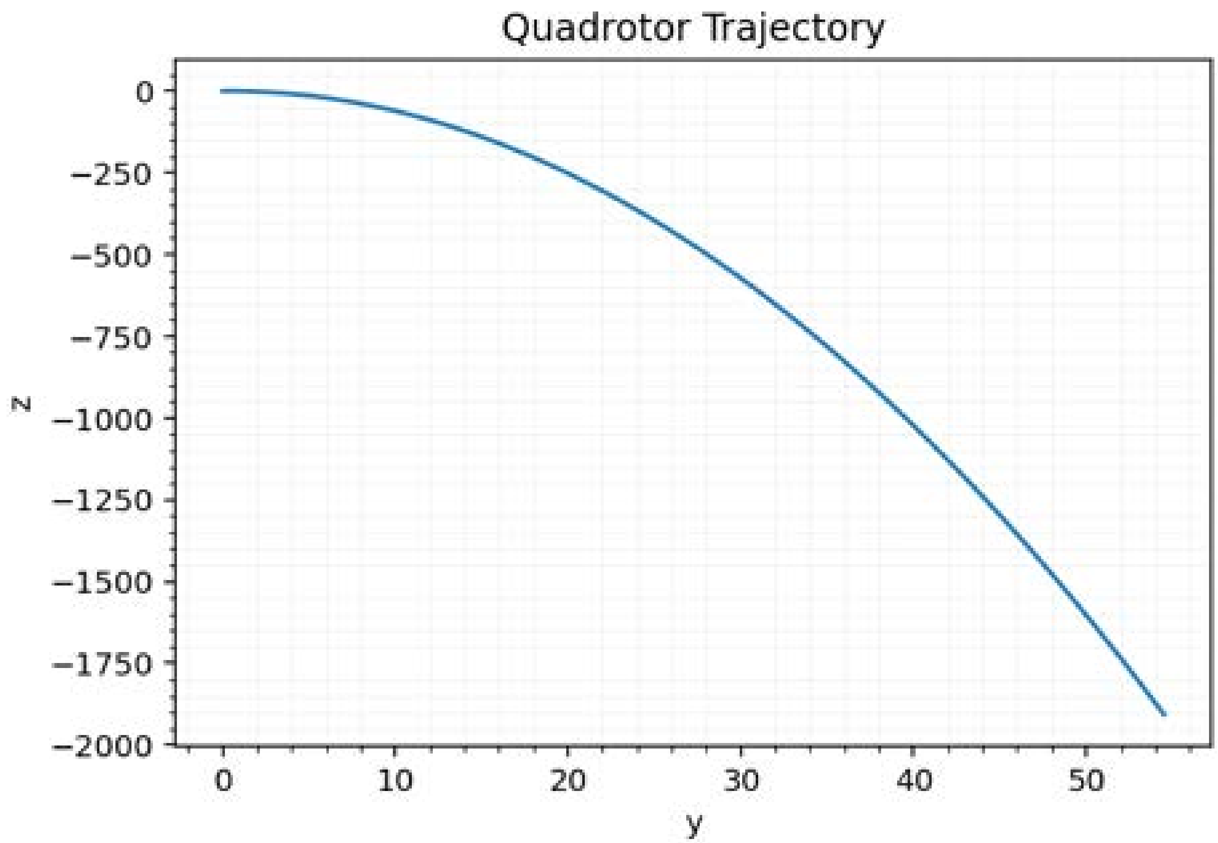

Figure 15: the x-axis would be the horizontal position of the quadrotor while the x-axis is the vertical position of the quadrotor. The graph demonstrates the quadrotor's position based on the conditions.

The small negative torque causes a backward tilt, which reduces the vertical component of thrust. The quadrotor slows its ascent then falls, creating a downward-opening parabolic trajectory in the y-z plane.

8. Conclusion

This paper explores the complexity of UAV operation in low-pressure environments through real world case studies and dynamic simulations. Exploring NASA's Ingenuity Rover, DJI's FlyCart 30, and the Dragonfly highlights how UAVs can be engineered to operate effectively in extreme climate and low pressure environments. The spring-mass-damper system showed control principles, while the 2D-Quadcopter Model presents complexities in real life flight dynamics. In the situation that was investigated, a positive gain would make the most sense because the quadrotor was supposed to produce a positive thrust, so instead of pushing upward, the quadrotor would be pushed downward.

The principles of control and the careful consideration of system dynamics are crucial for designing future aerial vehicles for both terrestrial and extraterrestrial missions.

The simulation provided a demonstration of the core control principles that were discussed in this paper. The spring-mass-damper system, a foundational linear model, was crucial for understanding proportional and derivative gains. Tuning these gains provided a visualization of an oscillating, undamped system to a system that was highly damped and stable. This simple model provided the important insights before moving to a more complex, nonlinear 2D Quadrotor. The Quadrotor simulation highlighted the challenges of controlling a vehicle in this space. Even a slight change in thrust or torque could significantly alter its position and orientation.

These simulations emphasize the vital connection between theoretical control dynamics and the engineering of UAVs in extreme environments. They demonstrate that control theory is applicable to solving real-world challenges like maintaining lift and stability in low-density atmospheres.

Works Cited

[1] “DJI Completes World's First Drone Delivery Tests on Mount Everest.” *DJI,* https://www.dji.com/media-center/announcements/dji-completes-world-first-drone-delivery-tests-on-mount-everest-en. Accessed 31 May 2025.

[2] Ata, Osama W. "Aerodynamic Performance Study of Ingenuity and Dragonfly Drones for Future Space Missions to Mars and Titan." *2021 4th International Symposium on Advanced Electrical and Communication Technologies (ISAECT)*, 2021, pp. 1-6. *IEEE Xplore*, Accessed 30 April 2025.

[3] Dreier, Frederick. “This Drone Video Shows What It’s Like to Climb to Mount Everest’s Summit.” *Outside*, https://www.outsideonline.com/outdoor-adventure/everest/mount-everest-drone-footage/. Accessed 31 May 2025.

[4] “FlyCart 30 Quick Start Guide.” *DJI,* https://manuals.plus/dji/flycart-30-drone-manual. Accessed 24 July 2025.

[5] "Ingenuity Mars Helicopter Landing Press Kit." *JPL,* https://www.jpl.nasa.gov/news/press_kits/ingenuity/landing/mission/spacecraft/. Accessed 25 April 2025.

[6] Marks, Paul. "Designing Dragonfly, NASA's Titan Explorer." *Aerospace America,* https://aerospaceamerica.aiaa.org/features/designing-dragonfly-nasas-titan-explorer/. Accessed 12 June 2025.

[7] Lorenz, Ralph D., et al. "Dragonfly: A Rotorcraft Lander Concept for Scientific Exploration at Titan." *Johns Hopkins APL Technical Digest*, vol. 34, no. 3, 2018, pp. 374–387. Johns Hopkins University Applied Physics Laboratory, Accessed 5 May 2020.

[8] "2D Quadrotor." *Cookie Robotics,* https://cookierobotics.com/052/. Accessed July 23 2025.

[9] anshakot-sys, *quadrotor-dynamics.* GitHub, 2025, https://github.com/anshakot-sys/quadrotor-dynamics. Accessed 22 August 2025.