Ann and H.J. Smead
Aerospace Engineering Sciences
UNIVERSITY OF COLORADO **BOULDER**

FLAME Project Final Report

Fire Limiting Aerial Management Ensemble:

Christian Bowman	Ian McCarty
Maximillian Brown	Braden Nelson
Ethan Davis	Jack Pearse
Donovan Gavito	Alex Putnam
Joshua Geeting	Brady Sivey
Drew Kane	Jared Steffen

Advisor: **Professor Glusman**
Industry Mentor: **Jack Elston**

May 5, 2025

Contents

List of Tables

List of Figures

List of Acronyms

CAD - Computer Aided Design
CDR - Critical Design Review
DR - Design Requirement
EKF - Extended Kalman Filter
ELRS - Express Long Range System
ESC - Electronic Speed Controller
FAA - Federal Aviation Administration
FLAME - Fire Limiting Aerial Management Ensemble
FPV - First Person View
FR - Functional Requirement
GPS - Global Positioning System
I2C - Inter-Integrated Circuit
IDR - Intenal Design Review
LIDAR - Light Detection and Ranging
Li-Ion - Lithium Ion
LiPo - Lithium Polymer
PFR - Project Final Report
PID - Proportional, Integral, and Derivative
PLA - Polylactic Acid
RC - Remote Control
RTK - Real-Time Kinematics
TRR - Test Readiness Review
UAS - Unmanned Aerial System

Nomenclature

NOTE: A dot above a variable indicates a time rate of change (e.g., $\dot{x}$), and a superscript "E" indicates inertial frame (e.g., x^E).

β: Cone Slant Angle
ρ: Density
g: Gravitational Acceleration Constant
ϕ: Roll Angle
θ: Pitch Angle
ψ: Yaw Angle
X: x Drag Force Component
Y: y Drag Force Component
Z: z Drag Force Component
x: x-Axis Position
y: y-Axis Position
z: z-Axis Position
u: Body Frame x Velocity Component
v: Body Frame y Velocity Component
w: Body Frame z Velocity Component
p: Roll Rate
q: Pitch Rate
r: Yaw Rate
L: Roll Moment
M: Pitch Moment
N: Yaw Moment
L_c: Roll Control Moment
M_c: Pitch Control Moment
N_c: Yaw Control Moment
Z_c: Motor Control Forces
I_x: x-Axis Moment of Inertia
I_y: y-Axis Moment of Inertia
I_z: z-Axis Moment of Inertia
A: Exit Area
V: Exit Velocity

1 Project Purpose and Design

Authors: Ian McCarty, Donovan Gavito, Jack Pearse, Alex Putnam, Jared Steffen

1.1 Project Purpose

The Fire Limiting Aerial Management Ensemble (FLAME) is a proof-of-concept for a firefighting unmanned aerial system (UAS). With the growing need for firefighting capabilities in areas with dangerous terrain, the FLAME UAS will be able to reduce response time. With the use of a UAS, the time it will take to reach that fire will be cut drastically, as well as minimizing risks to the firefighting personnel. The general goal of FLAME UAS is to help combat fires in the front-range area. Once the hexacopter proves the validity of an autonomous firefighting drone, the goal is to have a fleet of hexacopters, or similar vehicles, that would be working on the same fire together. This would help prevent fires from spreading in areas that are difficult to reach, as well as protect more people from the physical harm caused by wildfires. A successful project would result in a valid concept that would ultimately lead to fewer fires in the Colorado Valley area and a higher overall level of safety within fire departments.

High-Level Requirements

The high-level functional requirements for the system are summarized in Table 1. These requirements were developed based on the original request for proposal from August 2024 and outline the essential capabilities the drone must meet to successfully complete its mission. They cover key aspects such as autonomy, payload capacity, range, safety, and regulatory compliance. These requirements guided the overall design, implementation, and testing of the system to ensure that it meets its intended operational objectives.

Table 1: High-Level Functional Requirements

FR 1.1	The system shall have a minimum range of 100 meters
FR 1.2	The system shall be able to take off autonomously
FR 1.3	The system shall be able to land autonomously
FR 1.4	The system shall be capable of making 3 drops in a 20-minute span
FR 1.5	The system shall reach target location above the 30-foot hard deck
FR 1.6	The system shall only vertically translate below the 30-foot hard deck, at the location of interest and take-off/landing position
FR 1.7	The system shall be operable in mild weather conditions
FR 1.8	The system shall return to a known safe waypoint altitude if loss of communication and control
FR 2.1	The system shall carry and deploy 5 lbs of fire retardant per target location
FR 2.2	The system shall be able to take off, fly, and land with a minimum fire retardant weight of 5 lbs
FR 2.3	The payload shall consist of dyed water solution to mimic the viscosity and density of a PHOS-CHEK mixture
FR 2.4	The system components shall be less than 55 lbs with payload
FR 3.1	The system shall utilize human-in-the-loop to navigate from 10 meters RMS accuracy to 1 meter from location of interest
FR 3.2	The system shall be able to fly in US airspace
FR 3.3	The system components shall maintain visual line of sight from pilot
FR 3.4	The system components meet the certification requirements set by CU flight operations
FR 3.5	The system shall deploy the payload and have it land within a $\pm$ 1-meter accuracy or target location.

1.2 Design

The figures below showcase the FLAME hexacopter developed for this project. Figs. 1a and 1b presents images of the fully assembled drone during field testing, highlighting the physical implementation of the design. Fig. 2a and Fig. 2b display the CAD model used during the design phase, providing an isometric and front view respectively. These visualizations reflect the transition from digital design to a functional, test-ready platform. When fully assembled, the UAS weighed in at 18.2 lbs. Since we are not deploying the payload in one go and it was susceptible to winds, the team decided to carry an additional 0.5 lbs of payload, leading to a 5.5 lbs payload and a fully loaded weight of 23.7 lbs. This total weight verifies FR 2.4 and all child requirements of FR 2.4.

(a) Side View **(b) Isometric View**

Figure 1: FLAME Hexacopter

(a) Isometric View **(b) Front View**

Figure 2: CAD Model

1.2.1 Concept of Operations

In the interest of showing the actions the system will need to perform and in what order, a ConOps is shown for the mission. The low-level ConOps, as seen in Fig. 3, is used to break down every signal, action, and movement the system performs on one leg of the mission. Stage 1 begins with a fully loaded hexacopter, and an autonomous take-off is performed. Stage 2 is autonomous navigation to the target location. Stage 3 is the human-in-the-loop movement and payload delivery, where the pilot will manually adjust the position of the system to be accurately above the target and then deploy the payload onto the target. Stage 4 is autonomous navigation to the home station, and stage 5 is autonomous landing. Stages 1 and 5 are vertical motion only, with stages 2 and 4 consisting of horizontal motion. Stage 3 will consist of vertical and horizontal motion, although any movement below the 30-foot hard deck must be vertical. The signals present in the low-level ConOps detail the constant ground station communication and feedback, the constant GPS communication with satellites, and the human-in-the-loop communication during stage 3. The whole mission will be contained in a 100-meter diameter zone along with the home station. This mission zone will contain 3 targets with given GPS coordinates. The above low-level ConOps is repeated for each target location. In between the mission legs, once the hexacopter has landed at the home station, the hexacopter is disarmed, and the ground team will reload the hexacopter with a new 5.5 lb payload. 5.5 lbs is used as a buffer in the event that some water does not hit the target. During this time, other members of the ground team will be transitioning to the next mission segment within Mission Planner to reduce the delay between mission legs. Once this process is completed 3 times within a 20-minute window, the mission is considered completed and successful.

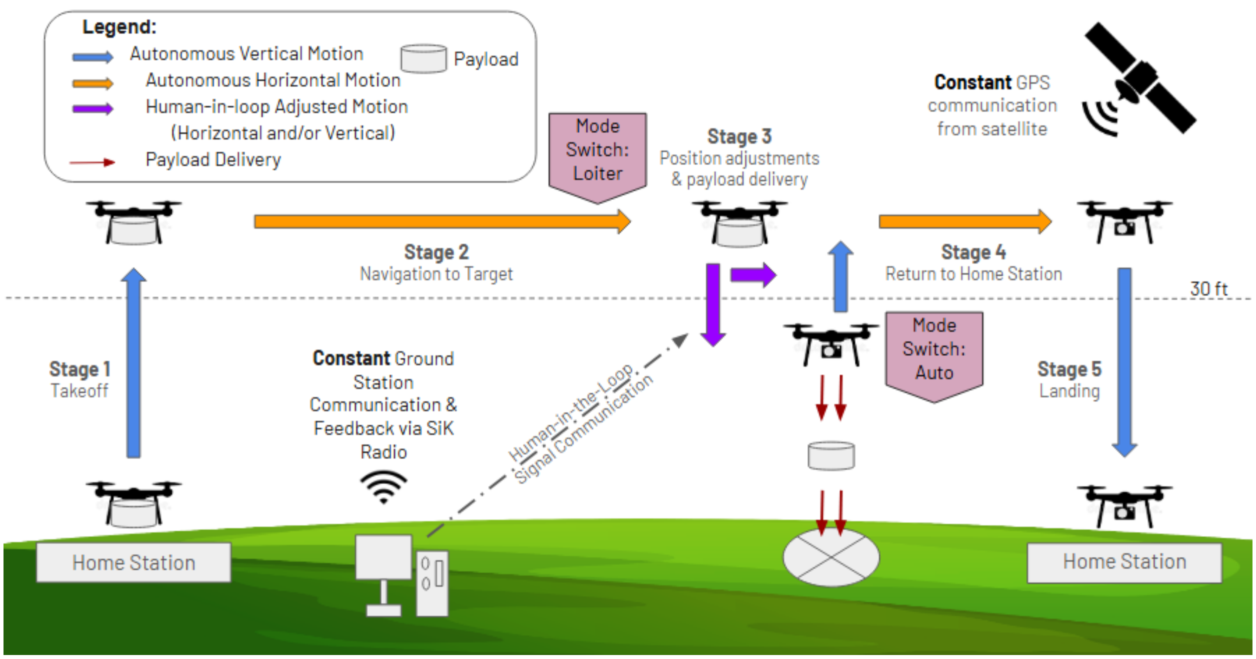


Figure 3: Concept of Operations

1.2.2 Functional Block Diagram

The functional block diagram shown in Fig. 4 illustrates the power, data, and control architecture of the UAS hexacopter integrated with a ground control station. At its core is the Pixhawk 6X flight controller running ArduPilot firmware, which interoperates with multiple sensors, including barometers, a magnetometer, accelerometers, gyros, and a GPS module for navigation. The system is powered by a 16,000 mAh 6S LiPo battery supplying 24 V (nominally), which feeds into a PM03D power module that regulates voltage for onboard electronics and powers six 50 A OPTO ESCs, each connected to a 5015 380KV motor. Peripheral hardware includes a 1080p FPV camera with video transmitter, a zero differential solenoid valve, and a cooling fan, each with dedicated voltage regulation. A remote switch, wirelessly controlled from the ground control station via a WOODGUILLIN remote controller, governs the activation of the solenoid valve. When the switch is triggered, it allows current to flow from a 3S (12 V) LiPo battery to the valve, generating a magnetic field that retracts the internal plunger and enables the release of the payload. Communication is handled via multiple radio links: SiK radios for telemetry (915 MHz), Ranger ELRS RC receiver for manual control (2.4 GHz), and FPV video (5.8 GHz). There is also an isolated Remote ID module onboard that broadcasts flight information over Bluetooth. The ground control station includes a TX16S transmitter powered by a 2S (8.4 V) LiPo battery with an ELRS transmitter and a SiK V3 telemetry module, which interfaces with Mission Planner software via USB.

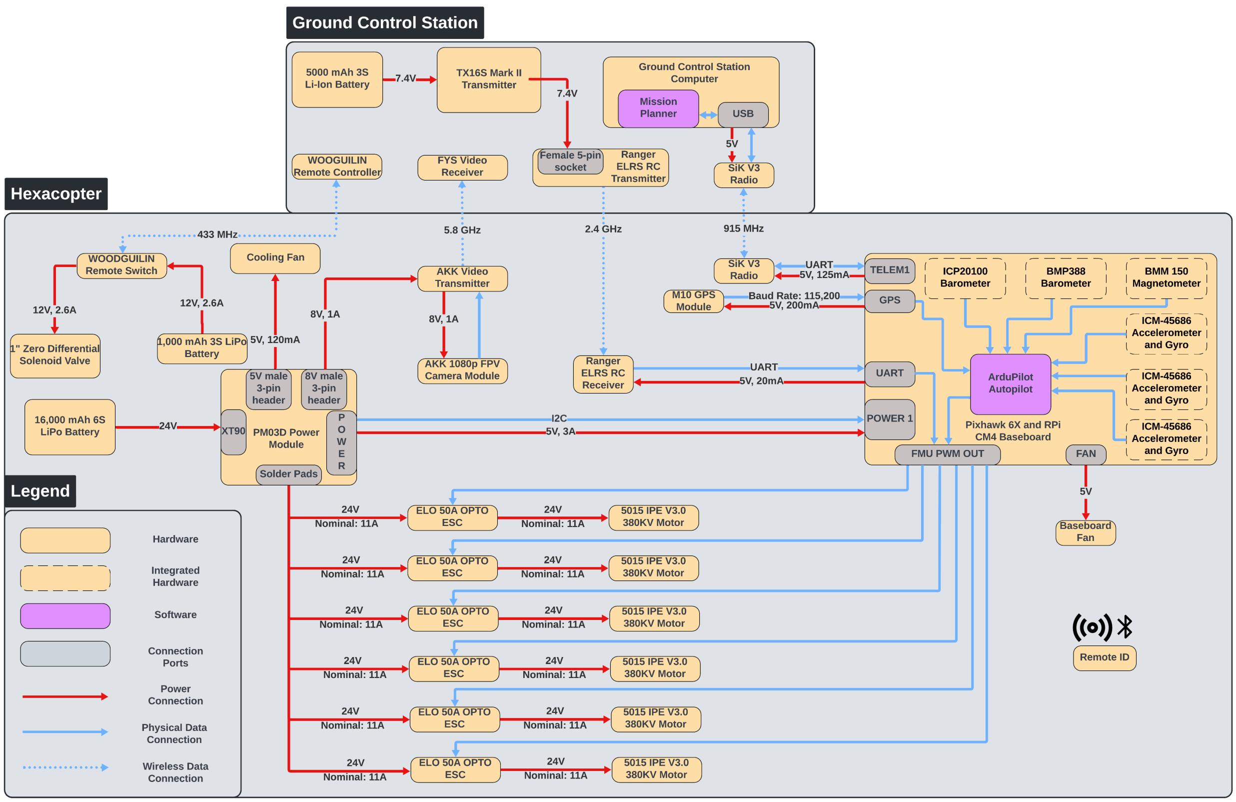
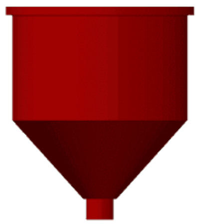


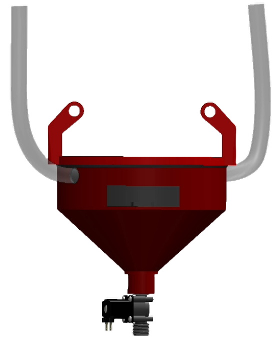
Figure 4: Functional Block Diagram

1.2.3 CAD Modeling

In order to determine an acceptable geometry for the payload container, a code was written that simulated all possible combinations of dimensions for a cylindrical tank with a funnel on the bottom, limited by the size of the hexacopter airframe. The dimensions were optimized to find the greatest viable ratio of contained volume to container volume, in order to minimize unnecessary weight while still allowing for the required payload volume. A mounting structure was added to the top of the container to attach it to the airframe, which consists of two arms that the rails of the frame's battery tray slide into horizontally.



(a) Original Payload Design



(b) Final Payload Design

Figure 5: Payload Container CAD Models

The addition of the mounting structure to the top required that the container undergo vertical compression and compensatory horizontal expansion to fit under the frame without touching the ground, as can be seen in Figs. 5a and 5b, while still maintaining the same total contained volume. The two tubes on the sides of Fig. 5b are used for two purposes. The first reason is to allow us to fill the water in the payload container, without it interfering with the electrical components onboard. The reason for having the second tube is to allow for the pressure to escape the container, thus preventing the 'glugging' phenomenon that the water payload would otherwise produce while deploying.

To reduce vibration reaching Pixhawk accelerometers, a CAD model of the vibration-damping mounting table was obtained from Thingiverse and can be seen in Fig. 6. The mount utilizes eight rubber grommets, similar to seismic springs, to isolate the Pixhawk from the airframe and the vibration produced by the motors. This had the effect of improving overall stability during flight and reducing the EKF's state estimation error. The vibrational readings before and after the implementation of this mount are shown in Fig. 7.

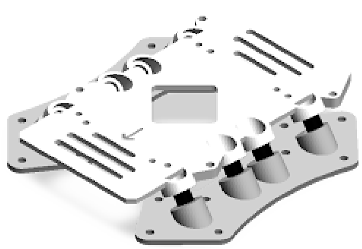


Figure 6: Flight Controller Vibrational Damping Mount

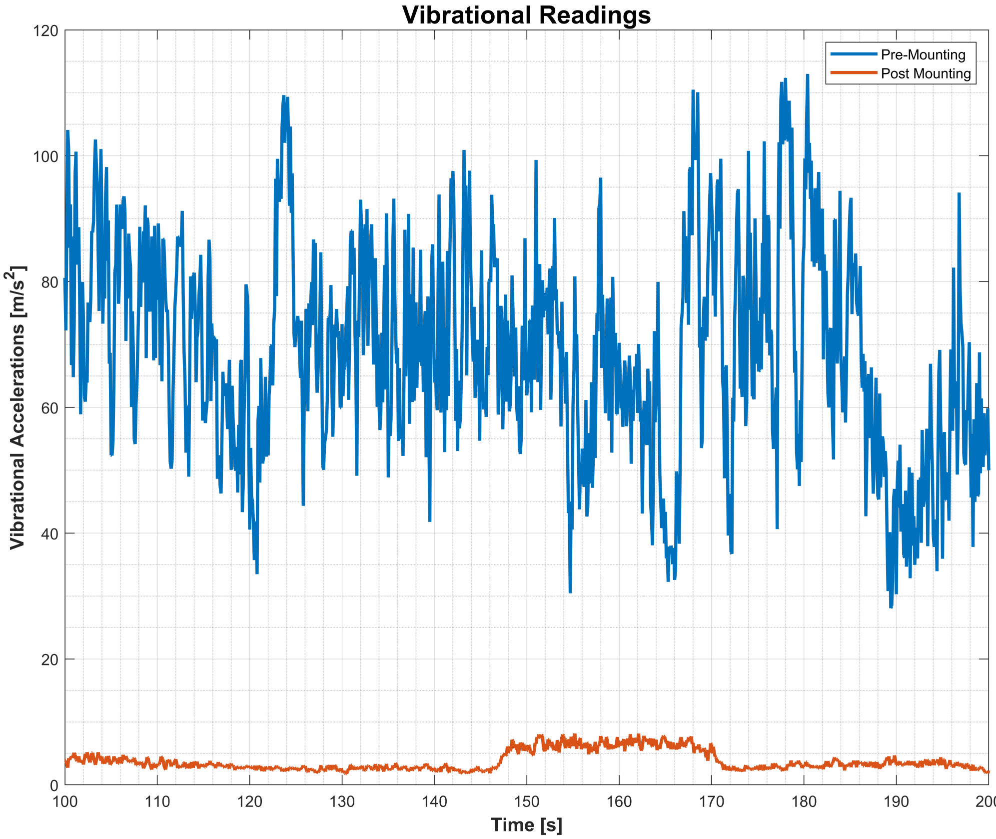


Figure 7: Vibrational Readings Pre and Post Mounting

The original camera mount, which consisted of a horizontal beam supported between the rails of the battery tray, had the unintended side effect of causing the camera to vibrate, significantly worsening the video quality. A replacement CAD model was designed, which is a simple plate mounted on the side of the payload container. This greatly reduced the vibration experienced by the camera.

1.2.4 Subsystem Design

Toward the beginning of the design phase of the project, four major proposed methods for payload delivery were considered and are detailed in Fig. 8, which was AI-generated. A trade study between the methods was conducted, ruling out the pressurized container due to the difficulty of its reloading, the large reaction force that creates instability in flight, and the potential difficulties in verifying the complete deployment. Pump-based release was similarly eliminated due to poor deployment speed, complete deployment verification, and large power requirements. Ballistic release was judged to have superior deployment speed, weather resistance, and ease of reload, but gravity release was selected for its ease of repeated testability, as it does not require a new container for each test.

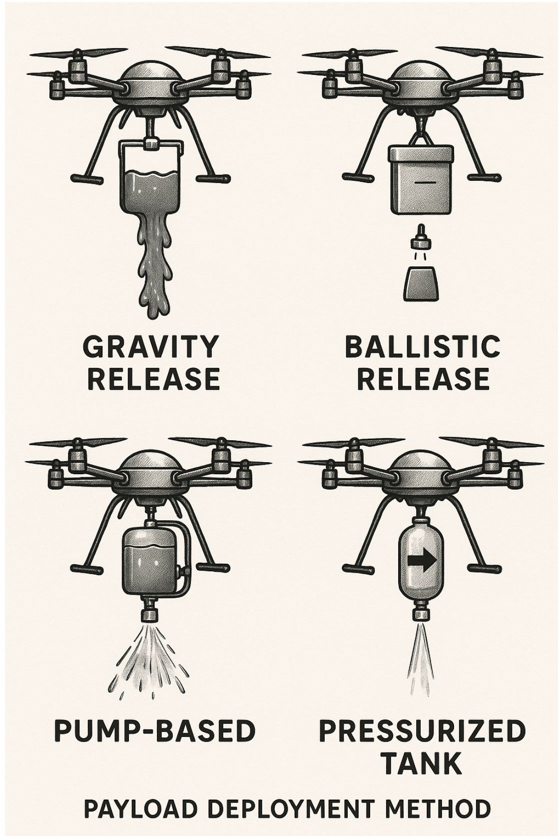


Figure 8: Payload Deployment Methods

The 3D-printed payload container, as detailed in Section 1.2.3 , is reloaded via one of two vinyl tubes. When the release is triggered by the remote switch, the solenoid valve, which is attached to the bottom of the container, toggles open, allowing the water to flow out downward, as detailed in the Functional Block Diagram section.

2 Testing, Verification, and Validation

Authors: Maximillian Brown, Joshua Geeting, Donovan Gavito, Drew Kane, Braden Nelson, Brady Sivey, Jared Steffen

2.1 Thrust Evaluation Test

2.1.1 Test Setup and Methodology

The Thrust Evaluation Test was run with the intent of gaining a better understanding of the motors being used on the drone. MAD Components, the manufacturers of the 5015 IPE 380KV V3.0 brushless motors, provides a data sheet with theoretical values of throttle percentage, voltage, current, input power, output power, torque, RPM, thrust, and efficiency. The purpose of the thrust evaluation test was to measure and record the data values to match the data sheet values and ultimately validate the motor output data. This data verifies functional requirements relating to the propulsion and power subsystems, as well as validates the mission profile power model.

The test setup was located in the basement of the engineering center. The test stand is a Series 1580 Test Stand belonging to the Design, Build, Fly club at CU. As shown in Fig. 9, after the motor and electronic speed controller were properly attached to the test stand, a computer with RCBenchmark was connected to the output of the test stand through a USB connection. The test stand featured a circuit board with four pancake load cells. These load cells were the main devices that measured the force. The datasheet for the test stand does not specifically detail what sensor measures the voltage and current, but it is known that they are taken from the output of the circuit board. Once the connection and calibration on the computer had been verified, the propeller was installed on the motor, followed by connecting the battery. Once all connections were secure, the test procedure began.

After preliminary testing to determine hardware and software tuning, the test setup was ready for full-scale thrust tests. Using the RCBenchmark script capabilities, the test was designed to increase the throttle by five percent, holding that percentage for five seconds before increasing to the next throttle percent. Once the software hits 100% throttle, the throttle is then reduced by 20% per increment until it is at rest, and the test is complete. Repeating the test three times allowed for a sizable data set for analysis and use in the power consumption model.

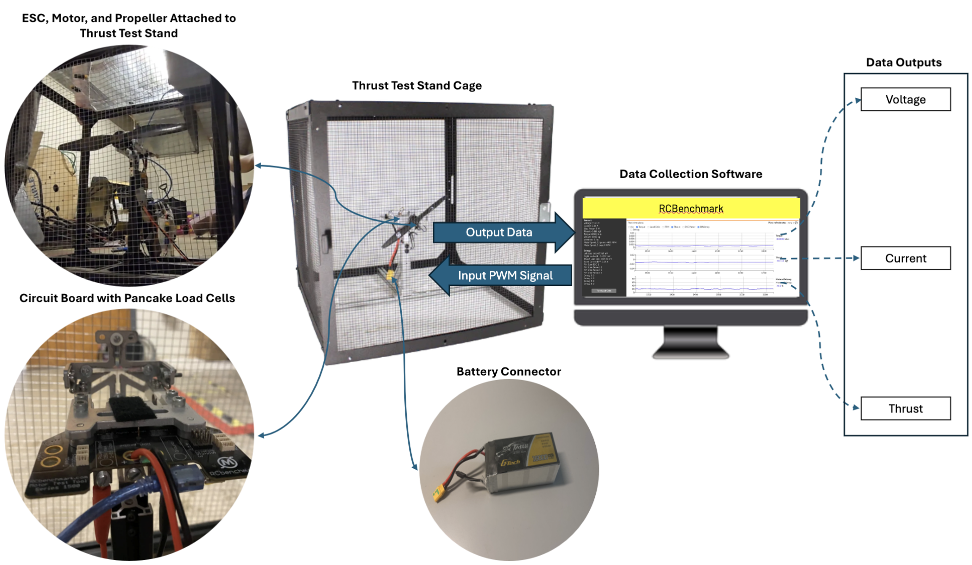


Figure 9: Thrust Test Setup Diagram

The current, voltage, force, and sampling rate details directly from the Series 1580 Thrust Test Stand datasheet provided by Tyto Robotics are detailed below, in Table 2. The maximum voltage is dictated by the 6S battery, which is fully charged to 25.2 V without voltage sag. The motor datasheet, when paired with an 18” propeller, stays under 5 kgf. The current is limited by the selected ESCs, which can handle a constant current of 50 A and a burst current of 60 A for less than 10 seconds. During the initial test, a safety current cutoff was hit at 90% throttle. For this reason, when the subsequent test was run, the throttle was not pushed past 90% to avoid overloading the ESC. This is the reasoning behind the test data presented in Fig. 10 going only to 90%. Since the expected throttle values of the motor were not expected to be near this value, the test was still deemed successful.

Table 2: Test Thrust Stand Sensor Specs

	Range	Tolerance
Voltage Supplied	0-50 V	0.05% ± 0.05 V
Current Supplied	0-55 A	1.0% ± 0.1 A
Force	-5 - +5 kgf	0.5% ± 0.005 kgf
Sampling	50 Hz	N/A

2.1.2 Test Results and Data

After completing the thrust tests, the data was loaded into MATLAB and divided to reflect the separate tests. RCBenchmark collects about a dozen different data types, with current, voltage, thrust, and time being the most useful to verify the model. By using the data and inputting it into a function that creates a best-fit line, a continuous curve is made for each data type. The thrust and current graphs are below, both plotted against throttle percentage.

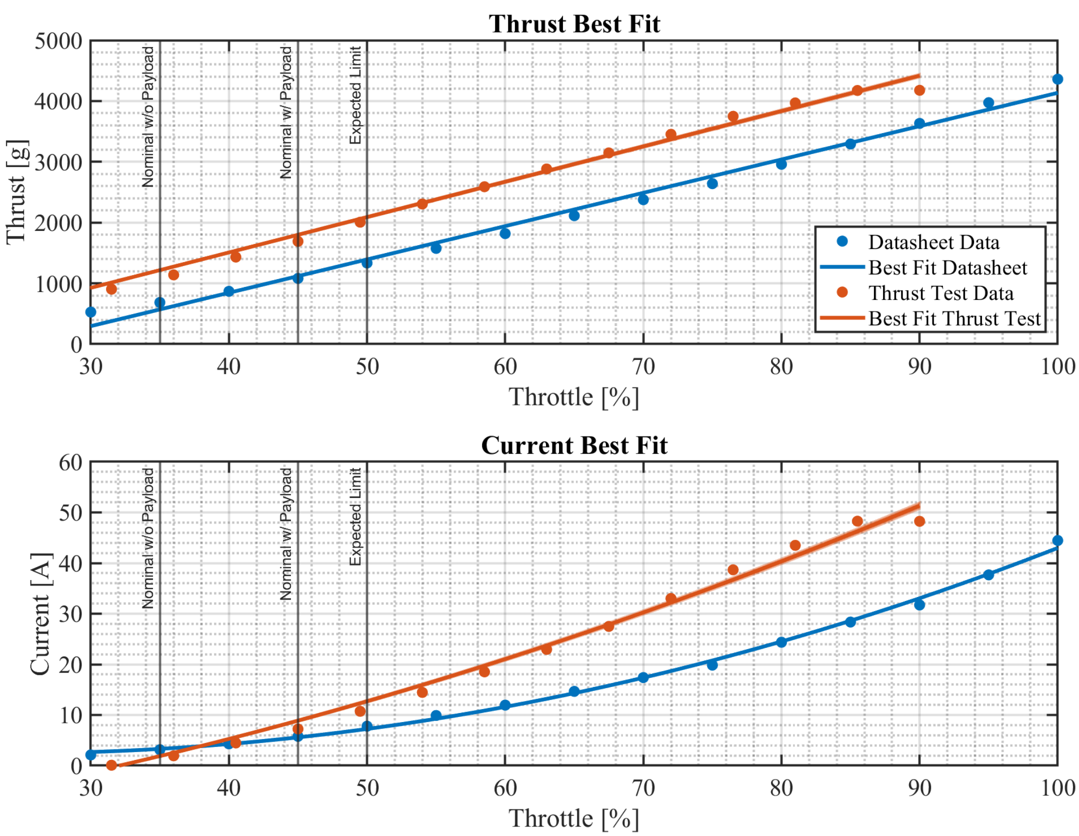


Figure 10: Thrust Test Evaluation Datasheet Comparison

The results of the thrust test can be seen in Fig. 10. Despite testing at altitude, which normally reduces thrust values, the motor was able to produce more thrust than the manufacturer's predicted data sheet.

This makes sense, as the current used throughout the test is higher than the data sheet's current usage. This includes the higher throttle percentages, and the UAS throttle should never exceed 60% according to the model. For the energy consumption model, the high thrust values at slightly higher current draw mean lower energy consumption for a single mission leg.

Fig. 11 shows the average throttle throughout the single-mission leg test discussed in Section 2.2. The plot demonstrates that the vehicle throttle remains consistently below 50% throughout all mission phases. This is a critical result, as it confirms that the propulsion system operates well within its throttle margin, leaving significant headroom under the 60% throttle cap. Maintaining this margin is essential to achieving the design goal of completing up to four mission legs on a single battery charge. One thing to note from Fig. 10 and Fig. 11 is that the payload-free flight segments operate at higher throttle than expected. The reasoning behind this is covered in depth in Section 2.2.3.

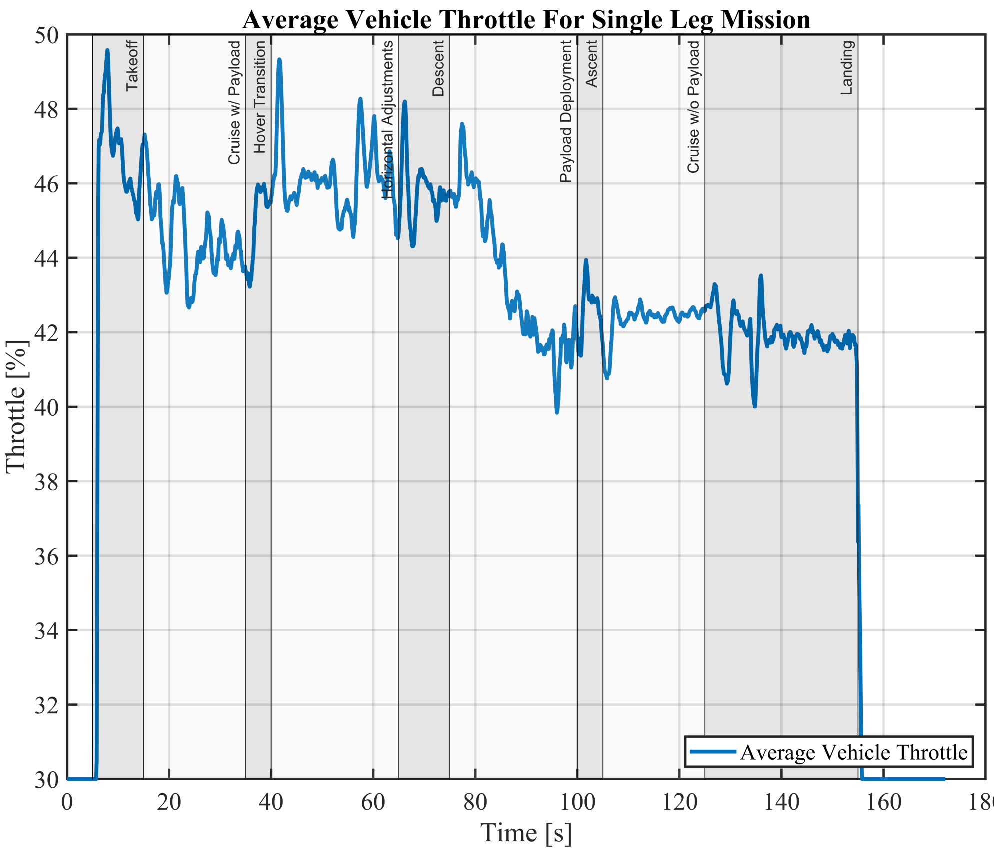


Figure 11: Mission Throttle Percents

Applying this data to the power consumption model by updating the current, voltage, and thrust values in MATLAB, the initial accuracy of the model improves. Fig. 12 shows the two curves, one using the datasheet values and the other using the thrust test values. Overall, the single leg power consumption goes down approximately 20 watt-hours over the entire flight leg's time, allowing for the UAS to make four mission legs on one battery.

2.1.3 Requirements Verification and Validation

Evaluating the thrust output of the selected motor and propeller pairs was critical to ensuring confidence in our engineering decisions to meet critical mission requirements. FR 2.2 and DR 2.1.1, 3.1.1, and 3.1.4, requirements outlining required power and propulsion for mission success, were the main requirements intended to be verified during this test. The results showing the ability of the aerial system to acquire a thrust-to-weight ratio of 2.33 verify both DR 3.1.1 and 3.1.4. FR 2.2 and DR 2.1.1 were unable to be verified through this test, but the power-to-thrust results gave confidence in the systems' ability to verify these requirements through subsequent tests. These requirements are outlined in Table 3.

The uncertainty in the results can be attributed to the difference between the modeled thrust and the power draw at specific throttle values versus the actual data collected through testing. The datasheet given from the motor manufacturers, MAD Components, contains theoretical thrust outputs and power draws at specific throttle values, while the data gathered from this test are a true representation of the combination of the selected motor, propeller, and ESC.

Table 3: Thrust Test Related Requirements

FR 2.2	The system shall be able to take off, fly, and land with a minimum fire retardant weight of 5 pounds
DR 2.1.1	The power subsystem shall provide power to support operation of all systems for the mission duration
DR 3.1.1	The propulsion subsystem shall provide thrust sufficient to lift the aircraft and its payload during takeoff and landing
DR 3.1.4	The propulsion subsystem shall provide sufficient thrust to reach the 30-foot hard deck

2.2 Single Mission Leg

2.2.1 Test Setup and Methodology

The test was carried out in the designated flight area of CU South using the fully assembled drone. A complete flight plan was developed and uploaded to Mission Planner prior to testing. To simulate the target delivery scenario, a pool was placed at the correct GPS coordinates, and the payload container was filled with 5.5 lbs of water. The test followed established flight readiness procedures and served multiple purposes: to verify the safety and stability of flight, ensure the current gain settings were acceptable, demonstrate accurate GPS-based navigation, and evaluate the precision of the payload delivery. In addition, flight data was recorded to support the validation of both the drone energy consumption and dynamic behavior models. This single mission leg test was also designed to verify high-level functional requirements that are critical to completing a full mission profile, while simultaneously validating the flight dynamics model and the mission-specific power consumption predictions.

2.2.2 Test Results and Data: Energy Analysis

One of the main models that was worked on in the fall semester and presented at the CDR was an energy consumption model. This model was used to test different pairs of motors and propellers and predict battery size. The PM03D power module is a digital power module that utilizes I2C communication to communicate current and voltage to the Pixhawk 6X. These components allow for the logging of the parameters that needed to be compared with the energy model for later analysis.

The parameters used to determine the total energy consumption for a mission leg were current and voltage, measured at each time step. The first step is to obtain the power used using $P = IV$. To obtain the energy used, the next step is to utilize $E = \int_0^t P dt$ and integrate it over the mission time and convert it into watt-hours. Fig. 12 presents the total energy consumed during a single 100-meter mission leg. This value for energy consumed was for the best-case scenario flight, which was also the flight with the most accurate payload deployment of 4.6/5 lbs. The total energy consumed for this leg was 49.2 watt-hours.

2.2.3 Comparison to Modeling: Energy Consumption

Fig. 12 and Fig. 13 below show the comparison of single-leg mission data. The model has inputs of mission segment times and thrust-to-weight ratios from a kinematics analysis, vehicle mass, and motor operational voltage and current draw. The model included the following assumptions:

1. Vertical adjustments were not made during horizontal translation.
2. The thrust-to-weight ratios were constant for each mission segment.
3. The missions segment speeds were constant.

Fig. 12 shows data from the original CDR model using motor manufacturer voltage and current, the adjusted model using static thrust test data, and the results from a single mission leg test for comparison. Fig. 13 shows the absolute value of the error between the two models and the flight data in watt-hours with respect to the duration of the mission.

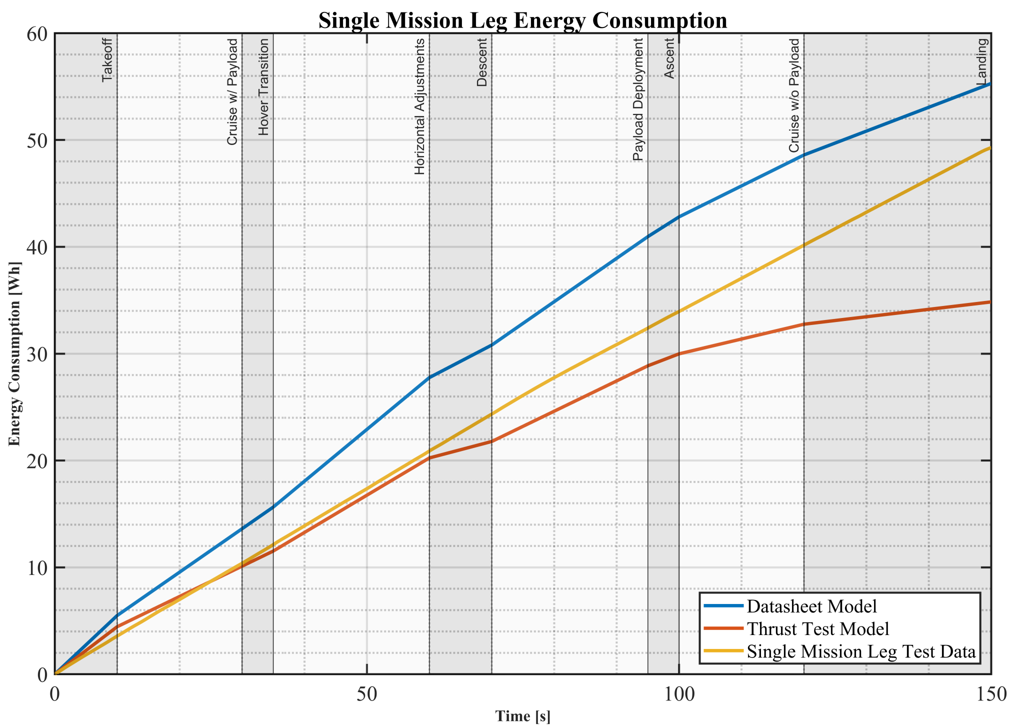


Figure 12: Single Mission Leg Data Compared to Energy Models

Fig. 12 shows that the thrust test model consistently aligns more with the test data during most of the mission phases, especially during take-off, payload cruise, and ascent. These are periods where the vehicle is heavier and demands more thrust, making the benefits of a more accurate thrust-to-current relationship from the test-derived model more evident. In contrast, the datasheet model systematically overestimates energy use, likely due to conservative assumptions on thrust generation and the absence of voltage sag effects.

During hover transitions, descent, and deployment, both models exhibit similar performance, reflecting lower power demands where differences in thrust modeling have less impact. Interestingly, the datasheet model, although generally less accurate, is closer to the cumulative test data at the end of the mission. This convergence is not due to improved modeling, but results from the compounding nature of cumulative plots masking the errors that occur in individual mission phases.

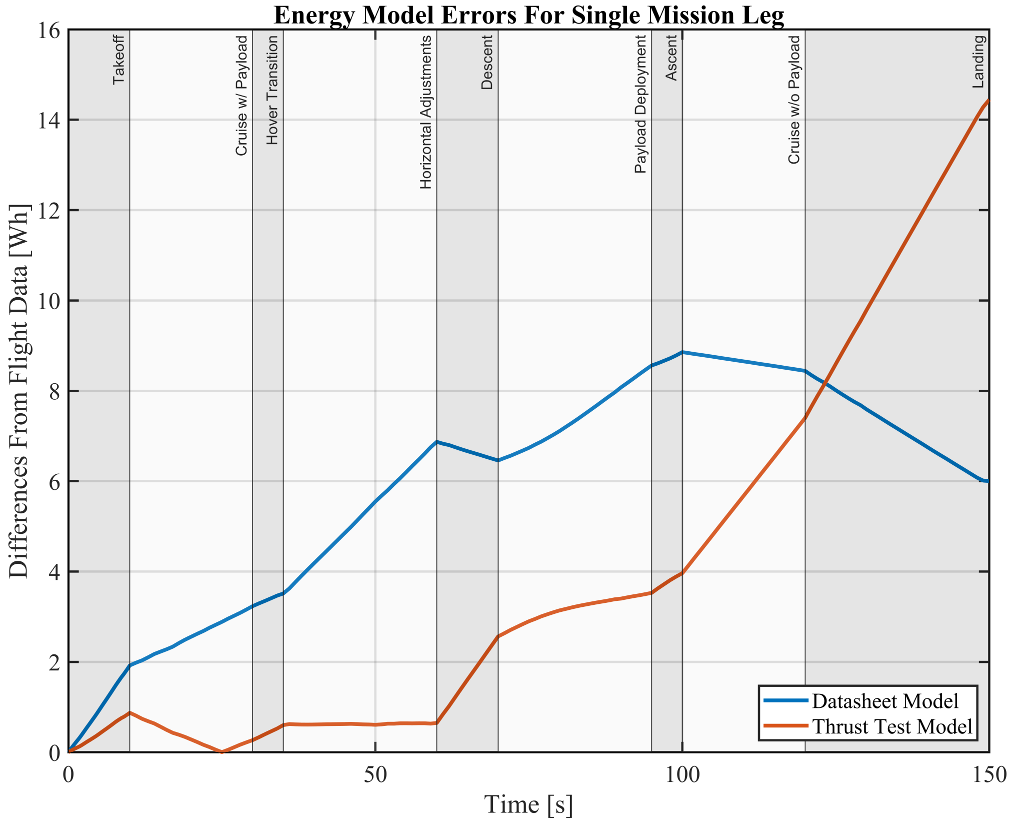


Figure 13: Energy Model Absolute Value Errors in Watt-Hours

Fig. 13 shows the absolute error between two energy consumption models, one based on manufacturer datasheet values and the other adjusted using static thrust test data, compared with actual flight data for a single mission leg. For the following discussion, it is important to note that on April 14th, during the test time, 10-meter wind conditions in Boulder, CO were averaged to be 15 miles per hour, which coincidentally verifies DR 7.2.2, DR 5.1.6, DR 1.9, and FR 1.7.

As stated above, in the early phase of the flight, when the drone is heavier due to the full payload, the thrust test model is considerably more accurate. This is because the increased weight of the drone reduces its susceptibility to wind disturbances, resulting in less trimming effort. Despite a higher overall throttle, less energy is wasted in maintaining stability. Additionally, the thrust test model better captures the drone's true thrust capability under load, since the test-derived thrust values are higher than those predicted by the datasheet while drawing a similar current. In contrast, the datasheet model overestimates energy usage early on because it under-predicts thrust and does not account for voltage sag under heavy load, leading to inflated power calculations.

In the later phase of the mission, after payload delivery, the drone is lighter and thus more affected by wind. Here, the thrust test model underestimates energy consumption more than the datasheet model, as shown by the increasing error. Although the motors require less throttle for flight, more energy is spent on trimming to maintain course because of the drone's lower mass and increased susceptibility to wind. The datasheet model performs relatively better in this phase because its higher baseline energy estimate compensates somewhat for this unmodeled trimming behavior. However, it still lacks voltage sag modeling, which becomes more prominent in longer flights as battery performance degrades. In general, Fig. 13 illustrates that the thrust test model offers improved accuracy during high-thrust, high-weight flight segments, while the datasheet model aligns better during low-thrust, high-variability conditions after payload release.

2.2.4 Test Results & Comparison to Modeling: Mass Flow

As the drone developed, it became necessary to understand the rate at which the mass is expelled from the payload deployment system. This model started with the general equation for mass flow $\frac{dm}{dt} = \rho A V$, where each term on the right side could be expanded based on the geometry of the container or first principles, such as energy conservation. The model breaks up the payload deployment to give an accurate idea of the mass flow rate depending on the height of the water in the container. There are two pieces to the overall container's shape: the cone section and the cylinder section. Saving a bit of the derivation, the respective equations can be seen below.

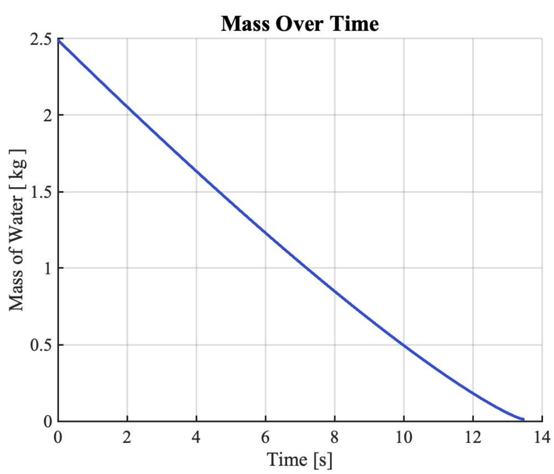
$$\frac{dm_{cyl}}{dt} = \rho \frac{\pi}{4} \cdot D_{cyl}^2 \cdot \frac{dh_{cyl}}{dt} \tag{1}$$

$$\frac{dh_{cyl}}{dt} = \sqrt{2gh} = V \tag{2}$$

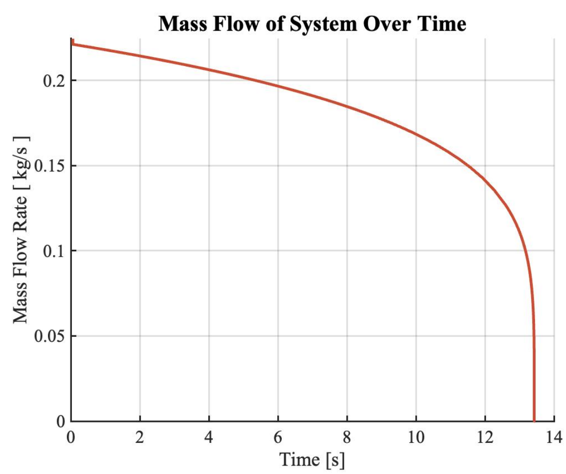
$$\frac{dm_{cone}}{dt} = \rho \pi h^2 \cdot \frac{dh_{cone}}{dt} \cdot \tan^2(\beta) \tag{3}$$

$$\frac{dh_{cone}}{dt} = \frac{\sqrt{2 \cdot g} \cdot D_{exit}^2}{4 \cdot h^{3/2} \cdot \tan^2(\beta)} \tag{4}$$

Using the above equations, one can integrate both height and mass to help produce the results in Fig. 14a and Fig. 14b below.



(a) Cylinder



(b) Cone

Figure 14: Modeling Payload Deployment

The model predicts that the container will empty in 13.46 seconds. The physical drop test resulted in the container emptying in 13.72 seconds when averaging the tests. These results gave the team confidence in the model while also fitting within previously held expectations of the payload deployment time (referring to estimates made by the team during conceptual development stages). Furthermore, the model revealed that the mass did not exit the container too quickly, allowing the drone to maintain its stability during deployment. However, this needed to be confirmed with the in-flight deployment test scheduled later in the semester. Overall, this model allowed for confidence in the container design and allowed the team to proceed with flight testing.

2.2.5 Test Results & Comparison to Modeling: Dynamics Model

The dynamics model is built on the equations seen in Fig. 15, which are numerically integrated to result in the standard twelve state variables: Position (x, y, z), Euler angles (ϕ, θ, ψ), velocity (u, v, w), and body angular rates (p, q, r). The general purpose of the model is to accurately model the flight of the drone (particularly distance and time) and ensure that the mission can be conducted within the system requirements.

$$\begin{pmatrix} \dot{x}_E \\ \dot{y}_E \\ \dot{z}_E \end{pmatrix} = \begin{pmatrix} c_\theta c_\psi & s_\phi s_\theta c_\psi - c_\phi s_\psi & c_\phi s_\theta c_\psi + s_\phi s_\psi \\ c_\theta s_\psi & s_\phi s_\theta s_\psi + c_\phi c_\psi & c_\phi s_\theta s_\psi - s_\phi c_\psi \\ -s_\theta & s_\phi c_\theta & c_\phi c_\theta \end{pmatrix} \begin{pmatrix} u^E \\ v^E \\ w^E \end{pmatrix}$$

$$\begin{pmatrix} \dot{\phi} \\ \dot{\theta} \\ \dot{\psi} \end{pmatrix} = \begin{pmatrix} 1 & \sin\phi\tan\theta & \cos\phi\tan\theta \\ 0 & \cos\phi & -\sin\phi \\ 0 & \sin\phi\sec\theta & \cos\phi\sec\theta \end{pmatrix} \begin{pmatrix} p \\ q \\ r \end{pmatrix}$$

$$\begin{pmatrix} \dot{u}^E \\ \dot{v}^E \\ \dot{w}^E \end{pmatrix} = \begin{pmatrix} rv^E - qw^E \\ pw^E - ru^E \\ qu^E - pv^E \end{pmatrix} + g \begin{pmatrix} -\sin\theta \\ \cos\theta\sin\phi \\ \cos\theta\cos\phi \end{pmatrix} + \frac{1}{m} \begin{pmatrix} X \\ Y \\ Z \end{pmatrix} + \frac{1}{m} \begin{pmatrix} 0 \\ 0 \\ Z_c \end{pmatrix}$$

$$\begin{pmatrix} \dot{p} \\ \dot{q} \\ \dot{r} \end{pmatrix} = \begin{pmatrix} \frac{I_y - I_z}{I_x} qr \\ \frac{I_z - I_x}{I_y} pr \\ \frac{I_x - I_y}{I_z} pq \end{pmatrix} + \begin{pmatrix} \frac{1}{I_x} L \\ \frac{1}{I_y} M \\ \frac{1}{I_z} N \end{pmatrix} + \begin{pmatrix} \frac{1}{I_x} L_c \\ \frac{1}{I_y} M_c \\ \frac{1}{I_z} N_c \end{pmatrix}$$

Figure 15: Quadrotor Equations of Motion

After creating the general structure of the dynamics model, the team had to address the difference in control laws. The ArduPilot controller uses a generally complex PID algorithm with an EKF to determine its motion. Due to the complex nature of designing these controls, as well as the challenges in implementing an EKF, the team decided to format the existing controls developed in class. The idea was that modeling a single mission leg totaling a maximum of 4 minutes would have small but noticeable errors from the actual flight of the drone. The team assumed that these errors were small enough to justify using controls different from those of the actual drone. The validity of this assumption can be discussed by comparing the data seen in Fig. 16, Fig. 17 and Fig. 18.

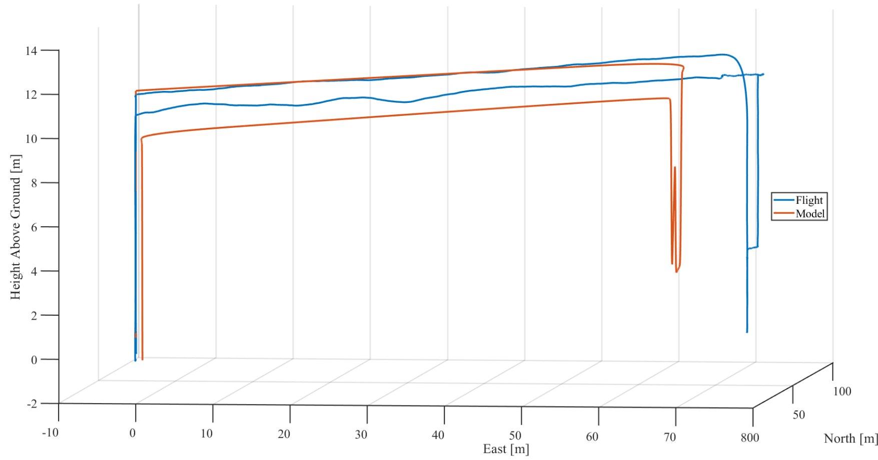


Figure 16: 3D Path of Model and Flight

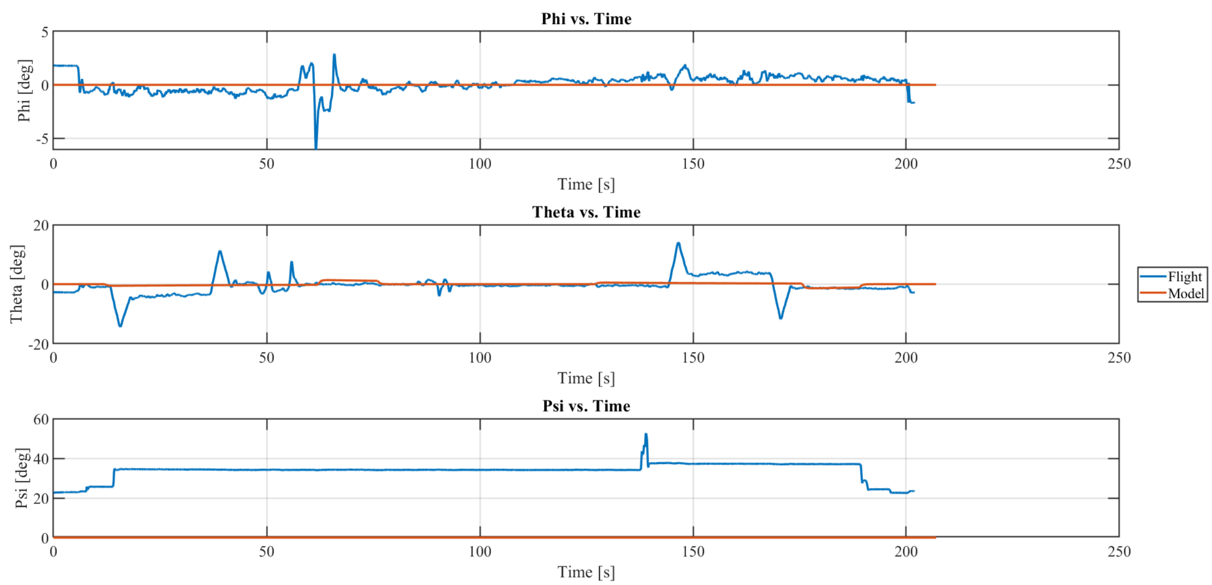


Figure 17: Euler Angles of Model and Flight

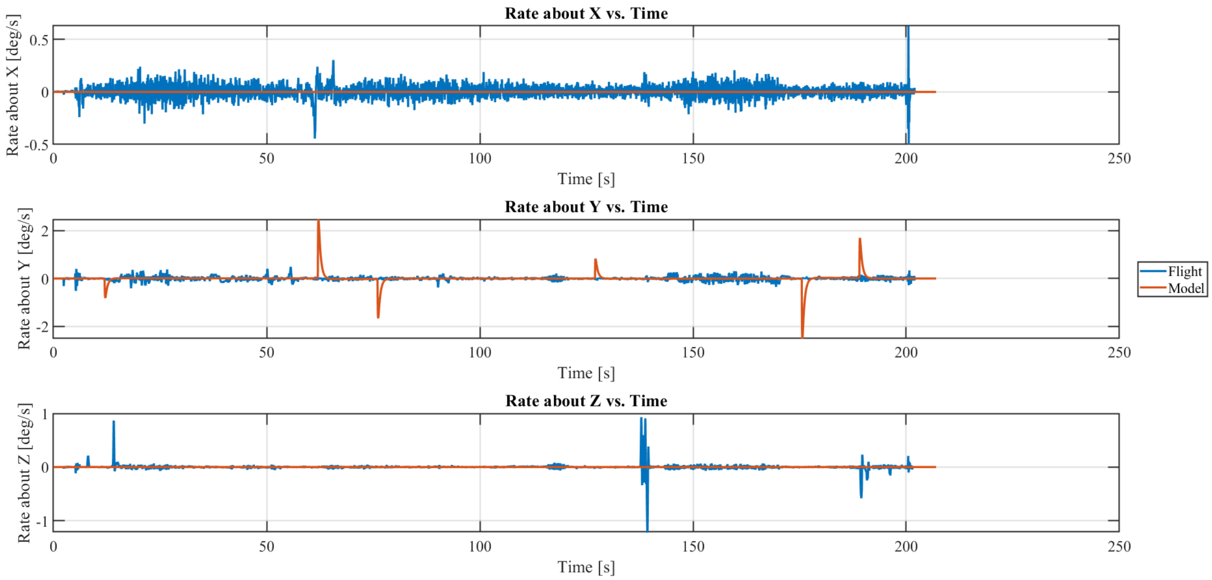


Figure 18: Angular Rates of Model and Flight

The first step to any model analysis is to check some standard metrics of the model and flight data and ensure that they are similar. Taking an initial dive into Fig. 16 above, it can be seen that the modeled flight and the test flights are very similar in shape. However, the simulated model has a slightly lower altitude when flying back (around the 30-foot hard deck versus the flight data 36 feet) and does not travel as far. The model has a total lateral distance of 101.1 meters, while the flight has a distance of 106.8 meters. To classify this error, one can look at the percent difference between the two distances, which is 5.483%. It can be shown that the modeled flight time is similar to that of the test data. The model results in a time of 204.5 seconds, and the test took 201.2 seconds when the stationary motion was cut off, where the percent difference in the two values is 1.627%. Looking into these two metrics, an initial discrepancy arises: the actual flight travels farther than the model in less time. The errors in flight time and distance are small in relation to the total duration of the mission, but are not negligible. This analysis gives an idea of what to look for in the other plots to determine why the discrepancy exists. It is difficult to compare the model and test when looking at the data from the point of view of the other two figures. The modeled results either have unnoticeably small changes relative to the flight data or are simply zero. To better characterize the success or failure of the model, Fig. 19 below "zooms in" on the relevant Euler angle, θ.

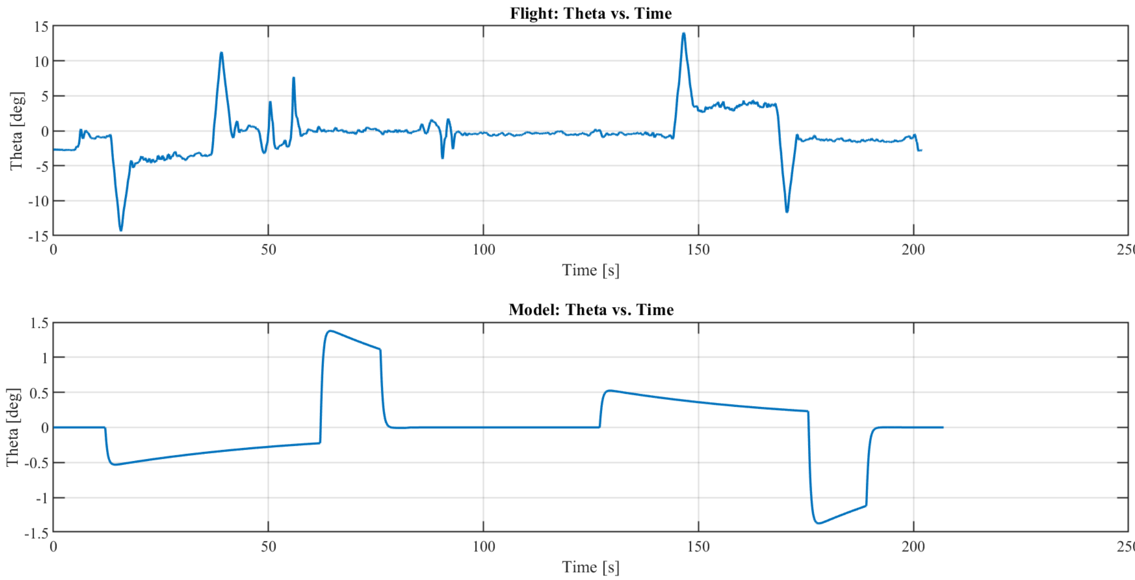


Figure 19: Visualizing Theta

The pitch angle (θ) is important to visualize because it is in the lateral direction toward and away from the target, which means its value dictates many characteristics of the drone's flight (speed, time of travel, motor forces, power consumption, etc.). An initial takeaway from this plot is to look at the time stamps of each motion. The drone accelerates toward the target, pitching down in the negative θ direction, then slowing down its motion sharply by increasing its pitch. The first long, flat section is where the drone is adjusted to its target, and the payload is dropped. The flight back repeats the same steps but with flipped signs, because the drone is traveling backward. It is evident that the shape of θ between the two data sets is relatively the same, but there are a few key differences that need to be analyzed further. First, θ in the model is almost an order of magnitude smaller than in the drone flight. The maximum θ seen in the flight data is 14.69 degrees, whereas the model hardly reaches 1.45 degrees. This reveals that the model has a very slow horizontal translation section. This is also seen in the upper plot of Fig. 19 where the translation section happens between 5 seconds and 54 seconds (for a total of 49 seconds) and the lower plot in Fig. **??** where the translation happens between 10 seconds and 78 seconds (for a total of 68 seconds). Note that this is only for the first translation section toward the target, so in total, a nearly 40-second difference between translation sections in the model and test showcases the difference in flight profile. Despite the time of both tests being nearly identical, the process by which they get there is quite different. Another way we can showcase the flight profile using the pitch angle plot is by looking at when θ is zero. Both the model and test data have a section in which it is zero, which corresponds to the drone hovering over the target and deploying the payload. It can be seen in the test data that 92 seconds are spent on hovering during payload deployment (54 seconds to 146 seconds) while the model only spends a brief 45 seconds (78 seconds to 123 seconds). The test spends over double the time that the model does during the payload deployment section, which corresponds with the overall narrative seen so far: Despite the time and distance being similar between the model and flight data, there are some key differences between the two in terms of the overall flight profile of the drone. However, there has only been a brief analysis between the two, and several other important claims need to be made to properly compare the drone data to the model and prove whether the previous assumption (modeling with a different control law than flight) is valid. To better understand the error in the model, the plots below reveal the difference between the test data and the model at each second of flight.

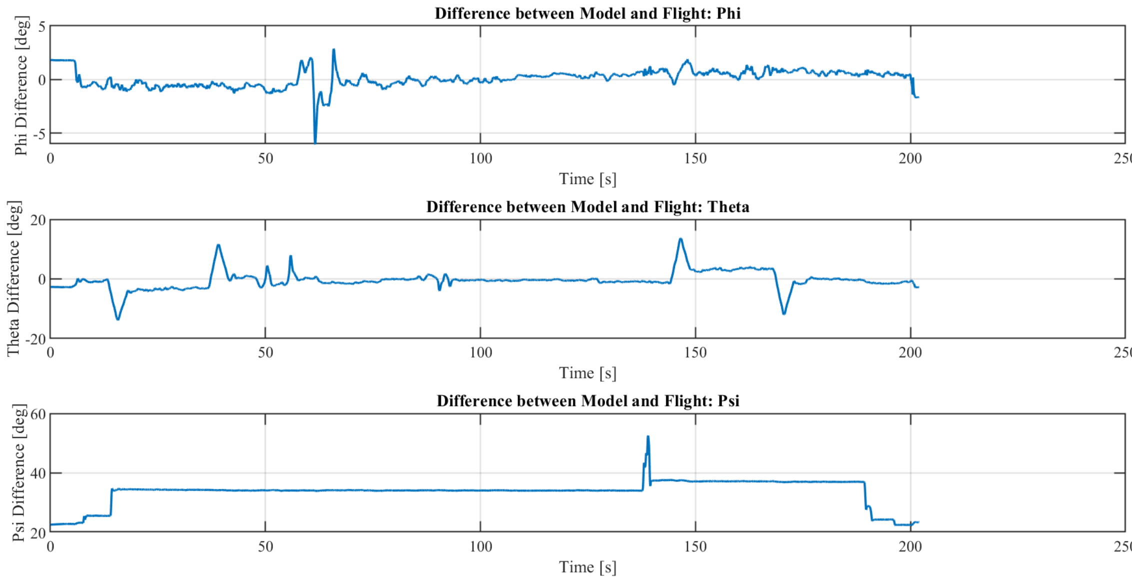


Figure 20: Euler Angle Errors

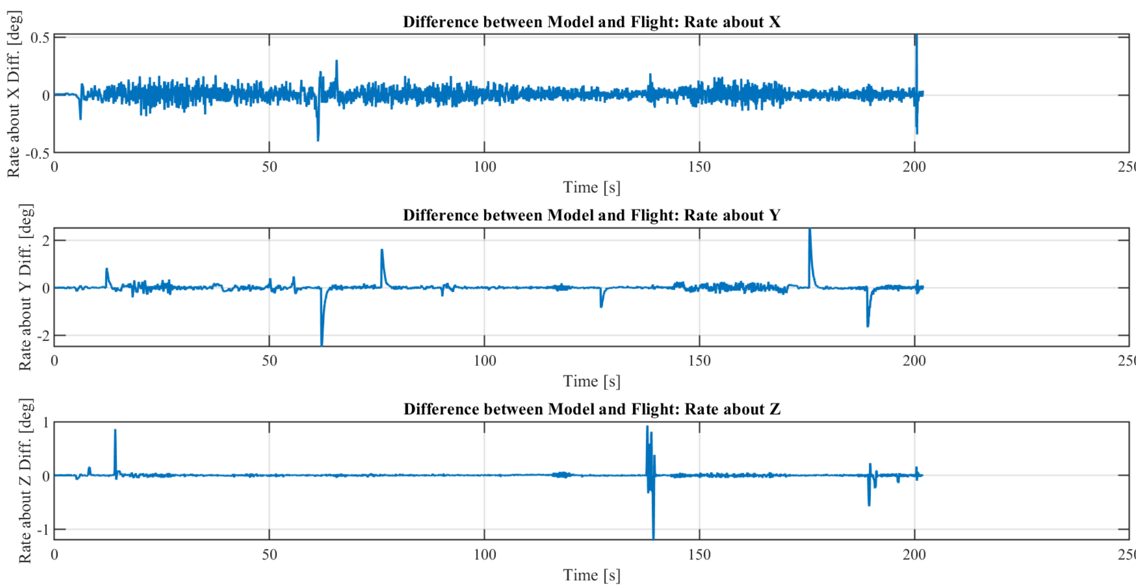


Figure 21: Angular Rate Errors

Ideally, Figures 20 and 21 will yield values near zero, showcasing that the model and the flight are nearly identical. In the plots above, their values are still very much non-zero, but this can be explained by the logic in the previous analysis. The time spent in each section (horizontal translation, payload deployment, etc.) differs, so when analyzing them using a difference depending on the time step, there will not be an exact cancellation. However, it is important to look at the magnitude of these plots and to check to make sure that the values are generally lower than or the same as in Figs. 17 and 18. It can be seen, then, that the pitch angle, θ decreases. The maximum has been reduced from 14.69 to 14.05, which, based on the magnitude difference talked about previously, reveals a generally positive result (in terms of validating the model). As for the other plots, their values are generally the same, but for another explainable reason. Referring to Figs. 17 and 18 to look at ϕ, ψ, p, q, and r, it is evident that the model values are zero throughout, whereas the flight values are non-zero and can be somewhat noisy. This is due to the model simulating an ideal flight, one in which the drone has no wasted movements. On the other hand, the drone uses GPS to navigate, which inherently sees non-ideal movements that must be corrected by the flight controls on board. To better compare the flight data with the model for these given values, we can take an overall average of the data and ensure that they are similar. Table 4 presents the results:

Table 4: Model and Test Differences

Variable	Model Avg.	Test Avg.
ϕ	0 deg	0.0792 deg
ψ	0 deg	36.4 deg
p	0 deg/s	0.0058 deg/s
q	3.63×10^{-4} deg/s	-0.0038 deg/s
r	0 deg/s	0.081 deg/s

After averaging the model and test data for the given variables above, one can see that the two sets result in nearly the same value, zero. To be more specific, on average, the extraneous movements due to GPS navigation cancel each other out to more accurately resemble the ideal flight values showcased in the model. After comparing the shape of the flight path, flight time, flight distance, the magnitude and shape of θ, and the overall average value of the other important flight variables, the model can finally be justified as a success or failure. Despite the relatively small differences in flight time and distance, to the more pronounced errors in the flight profile and θ magnitude, the model accomplished exactly what it needed to. Returning to the original purpose of the model, the team needed to show that the drone could perform the mission within the given requirements. It did not need to accurately model each mission segment or the non-ideal motions of GPS navigation; it simply needed to showcase exactly what is presented above, a model that is reasonably similar to the actual flight in terms of time and distance that illustrates a successful mission. However, the original assumption that the control law used in the model would properly resemble that on flight is not valid. Due to the large discrepancy seen with θ, one of the more important variables when determining values for speed, motor forces, and time of flight, it is evident that the control law in the model does not accurately correspond to ArduPilot's much more complex control law.

2.2.6 Requirements Verification and Validation

To ensure that the system met high-level requirements, the single leg mission test was conducted and compared to predicted models developed in the design phase. This test verified the most critical subsystems in ensuring the success of the system.

General Mission

The single mission legs test verified requirements related to the general mission architecture. DR 7.2.1, DR 7.2.5, and DR 7.2.6 ensured that the aircraft could support the full weight of the system at takeoff and withstand repeated forces from controlled takeoffs and landings at each drop site. These structural capabilities enabled safe execution of the mission profile within the designated 100-meter operational area. DR 2.1.1 was met through the power subsystem, which delivered 49.2 watt-hours of energy for a single leg of the mission. This was well within the capacity of the fully charged flight battery, which provides 355.2 watt hours. The mission was pre-programmed in Mission Planner in accordance with DR 3.1, allowing the UAS to autonomously navigate to each drop location. The corresponding flight plan is shown in Fig. 22.

Throughout the entire mission, the pilot maintained a visual line of sight with the UAS, satisfying DR 3.3 and fulfilling the overarching system requirement outlined in FR 3.3. These requirements are outlined in Table 5.

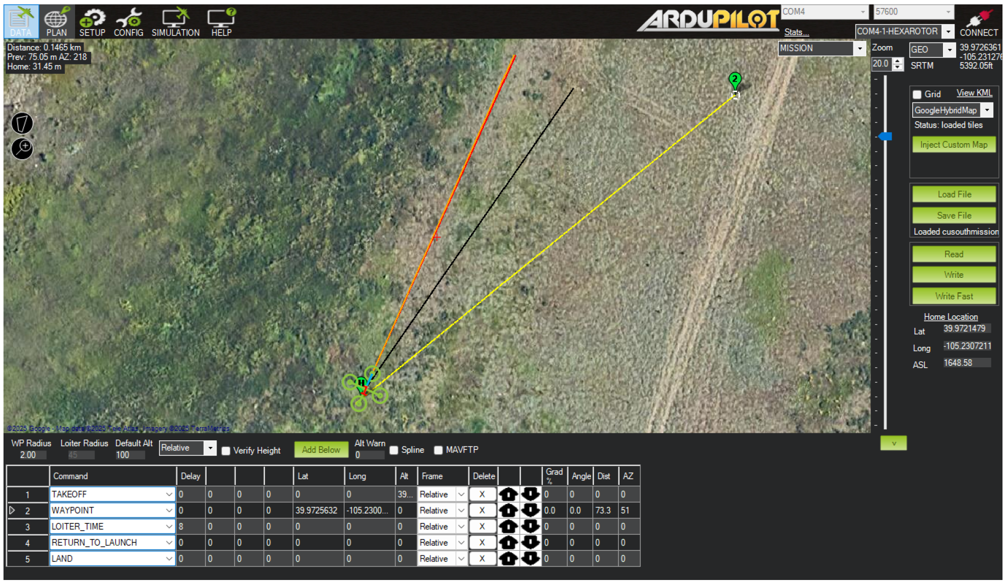


Figure 22: Mission Planner Flight Plan

Table 5: General Mission Requirements

FR 3.3	The system components shall maintain visual line of sight from pilot
DR 3.1	The ground system shall allow for pre-programming of target locations for fire retardant deployment
DR 3.3	The ground control system shall allow the operator to maintain line of sight with the UAS
DR 2.1.1	The power subsystem shall provide power to support operation of all systems for the mission duration
DR 7.2.1	The airframe subsystem shall be able to support the total weight of the UAS
DR 7.2.5	The airframe subsystem shall be constructed to withstand the forces associated with controlled takeoffs, descents and landings at locations of interest
DR 7.2.6	The airframe subsystem shall be designed to support operational ranges within a 100 meter diameter

Guidance, Navigation, and Control

The single mission leg test also successfully verified several design and functional requirements associated with the Guidance, Navigation, and Control subsystem, as seen in Table 6. Specifically, DR 5.1.4 and DR 5.1.5, which require the control subsystem to support autonomous takeoff and landing, as well as DR 5.1.1 and DR 5.1.2 which require stable flight above a 30-foot hard deck, were demonstrated during the test. These requirements, along with DR 5.3.1 (autonomous movement between locations of interest) and DR 6.1.3 and DR 5.1.3 (human-in-the-loop control), support broader subsystem-level goals such as DR 1.2, which can be visually verified by Fig. 23, as well as DR 1.4 and DR 1.5, which define autonomous and human-in-the-loop operation with specified payload and precision navigation capability. Furthermore, autonomous, stable flight to and from the target location, as well as autonomous take off and landing. In turn, these mid-level requirements trace to high-level functional requirements such as FR 1.2 and FR 1.3 for autonomous takeoff and landing, and FR 1.5 and FR 1.6, which can be visually verified through Fig. 24 require horizontal operation above the 30-foot hard deck and only allow vertical operation at specific mission segments, and FR 3.1 which requires human-in-the-loop navigation.

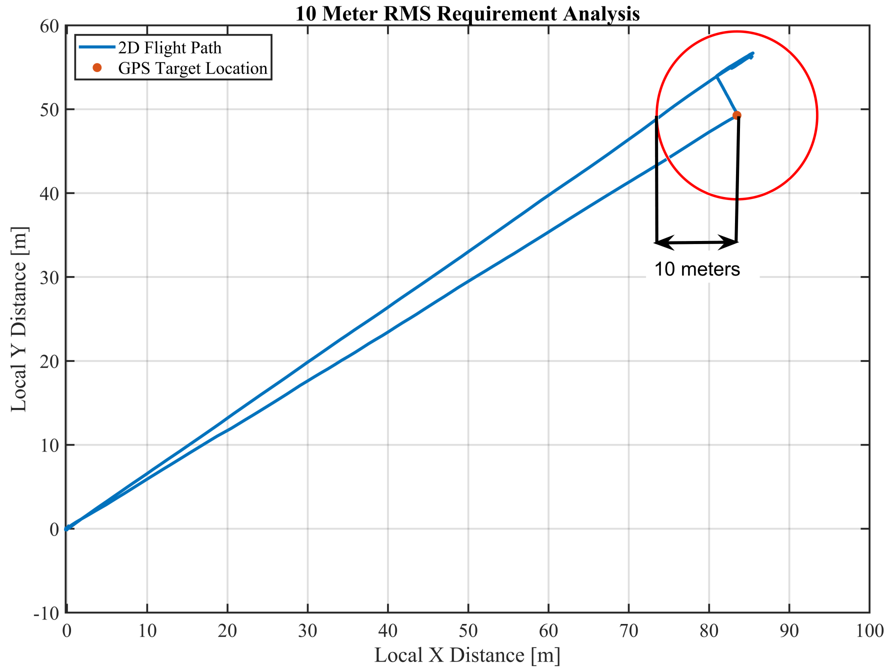


Figure 23: 10 Meter RMS Verification

Table 6: Guidance, Navigation, and Control Requirements

FR 1.2	The system shall be able to take off autonomously
FR 1.3	The system shall be able to land autonomously
FR 1.5	The system shall reach target location above the 30 foot hard deck
FR 1.6	The system shall only vertically translate below the 30 foot hard deck, at the location of interest and take-off/landing position
FR 3.1	The system shall utilize human-in-the-loop to navigate from 10 meters RMS accuracy to 1 meter from location of interest
DR 1.2	The avionics system shall ensure autonomous navigation within 10 meters RMS accuracy
DR 1.4	The avionics system shall support autonomous takeoff and landing with a specified payload of 5 lbs
DR 1.5	The avionics system shall allow human-in-the-loop input for precision navigation within 1 meter of the target using visual markers and payload deployment
DR 5.1.1	The control subsystem shall ensure stability and responsiveness of the aircraft during operation
DR 5.1.2	The control subsystem shall maintain flight above the 30 foot hard deck during horizontal translation
DR 5.1.3	The control subsystem shall allow for human-in-the-loop control
DR 5.1.4	The control subsystem shall support autonomous take-off
DR 5.1.5	The control subsystem shall support autonomous landing
DR 5.3.1	The control subsystem shall enable autonomous movement between locations of interest
DR 6.1.3	The payload deployment subsystem shall support human-in-the-loop deployment capabilities

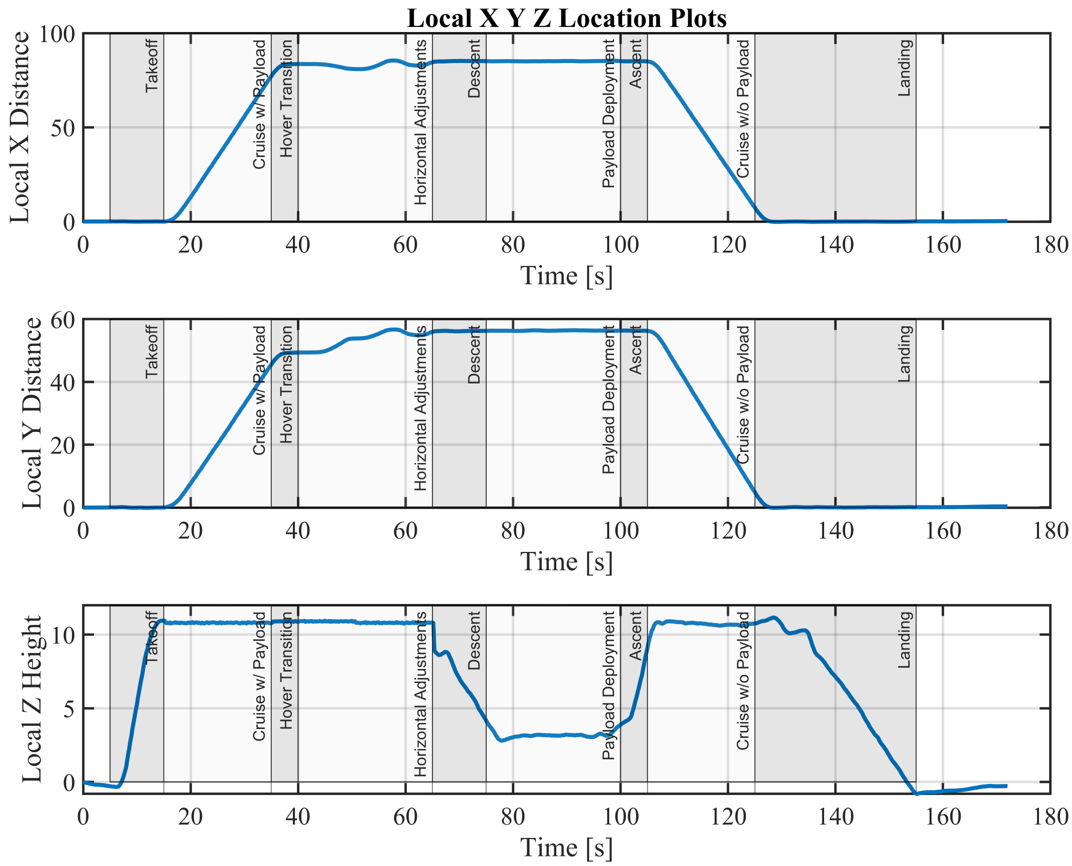


Figure 24: Local X Y Z Locations

Payload Containment and Deployment

The single leg mission test demonstrated the ability to deploy the payload, as well as allowed the verification of several requirements involving payload deployment and structural design. In terms of containment, DR 7.2.4, DR 6.2.1, and DR 2.5 (hold the payload) were verified with the assembly and leak-proofing of the payload container. Fig. 25 is a CAD model of the isolated payload deployment system. The design includes two upper mounting arms for attachment to the underside of the UAS and a container capable of holding up to 5.5 lbs of liquid, addressing requirements DR 7.2.3, DR 7.2.4 and DR / FR 2.1 (carry and deploy payload). By visual inspection, DR 6.2.1 was met by using rubber-based sealant at the connection points between the components. Furthermore, PLA plastic construction effectively prevented leakage during flight operations.

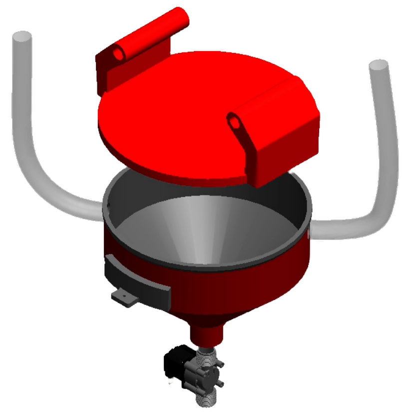


Figure 25: Exploded View of Payload Deployment System

During the single leg mission test, the UAS successfully deployed the payload over the target area, validating several deployment requirements. The payloads were loaded into a gallon jug and weighed by placing them on a scale. The scale was zeroed with the container placed on it and payloads were weighed until the scale said 5.6 lbs to account for the rounding of the scale. Fig. 26 shows a top-down view of the payload release, captured by a 120-degree FPV camera mounted on the payload system and aligned with the target. Based on this visual evidence, the requirements DR 6.2.2 and DR 6.1.2 were verified, confirming the accurate deployment at the intended location. Additionally, requirements DR 6.1.1 and FR 2.1 were met, as the payload landed within the designated 1 meter diameter pool using a solenoid valve release mechanism. However, post-test measurements indicated that only 4.6 lbs of the payload reached the target area, resulting in a 9.2% loss in total payload weight and preventing full compliance with requirement FR 3.5 (payload must land in $\pm$ 1 meter accuracy). This loss is likely due to the limited operator experience in precisely positioning the drone over the target, which could be improved with additional testing and practice.

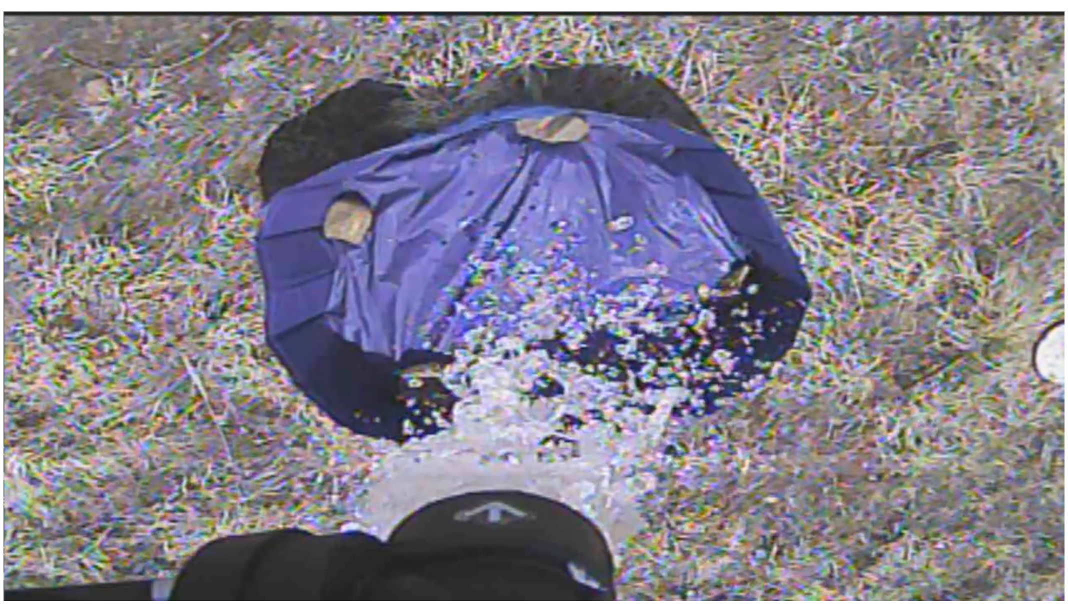


Figure 26: Top View of Payload Deployment Sequence

The single leg mission test was recorded to take approximately 3.5 minutes per run. This account for reloading between runs as well as pilot adjustment. In the end, a full mission was completed in a recorded time of ~11 minutes. This verifies DR 1.7 and FR 1.4 in completing three payload drops in a 20-minute time interval. These requirements are laid out in Table 7. Although there is no direct requirement for the deployment time, it is worth noting that the modeled deployment duration was not entirely accurate. As referenced in Section 2.2.4, the system was modeled on the basis of mass flow over time. During testing, the actual discharge time from full to empty was measured at approximately 12 seconds, compared to the modeled value of 13.46 seconds, a deviation of 8.9%. This discrepancy is attributed to uncertainty in the internal geometry of the solenoid valve. Because the exact internal structure of the valve during magnetic actuation was unknown, the outflow was modeled using a simplified design, leading to the observed variation.

Table 7: Payload Containment and Deployment Requirements

FR 1.4	The system shall be capable of making 3 drops in a 20-minute span
FR 2.1	The system shall carry and deploy 5 lbs of fire retardant per target location
FR 3.5	The system shall deploy the payload and have it land within a $\pm$ 1 meter accuracy of target location
DR 1.7	The avionics system shall enable at least 3, 5 lb fire retardant drops in 20 minutes
DR 1.10	The avionics system shall be capable of deploying a dyed water solution that mimics the viscosity and density of a PHOS-CHEK mixture
DR 2.1	The structure system shall support a payload capacity of at least 5 lbs for carrying fire retardant materials
DR 2.5	The structure system shall accommodate a payload designed to hold the dyed water solution without leakage
DR 6.1.1	The payload deployment subsystem shall include a payload deployment mechanism that allows for release of fire retardant with a precision of $\pm$ 1 meter
DR 6.1.2	The payload deployment subsystem shall be capable of deploying the dyed water solution at the locations of interest
DR 6.2.1	The payload deployment subsystem shall hold the payload of a dyed water solution while preventing leakage during flight
DR 6.2.2	The payload deployment subsystem shall be capable of deploying a 5 lb payload per target location
DR 7.2.3	The airframe subsystem shall include payload mounting and storage capabilities of an at least 5 lb payload
DR 7.2.4	The airframe subsystem shall include appropriate storage of the payload to prevent spillage

Communications

Data communications between the ground station and the UAS occurred via two 915 MHz telemetry radios, one connected to the ground station computer and one connected to the flight controller. This radio telemetry verified the requirements DR 1.3, DR 1.1.2, DR 4.2.1, DR 4.3.2, DR 4.3.3, DR 4.1.1, DR 4.1.2, DR 4.1.3 and DR 4.1.4, which are described in Table 8. The radio transmitted real-time accelerometer, gyro, barometer and GPS readings to the ground station as seen in Fig. 27. It also monitored connection strength and sent audio queries to the pilot if the connection was lost. This would allow the pilot to flip a switch on the transmitter and switch to manual control.

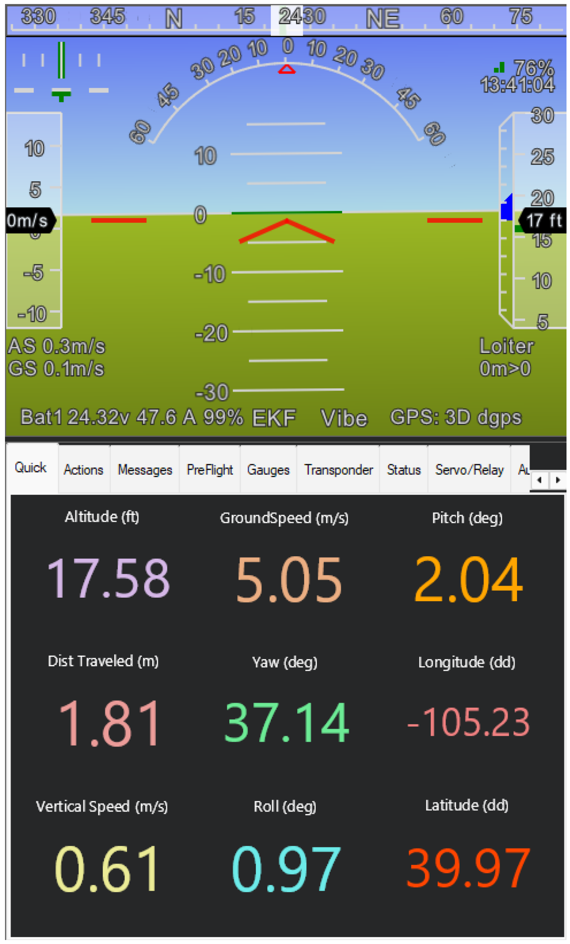


Figure 27: Mission Planner Telemetry

Data was logged onto a micro SD card on the flight controller. After flights, the SD card was read into Mission Planner where the data was viewed, which was necessary to verify the model and accuracy requirements.

Table 8: Communications Requirements

DR 1.3	The avionics system shall be capable of maintaining operational control and communication within a 100 meter diameter
DR 1.1.2	The sensing subsystem shall continuously monitor communication signals to detect loss of connectivity with ground control
DR 4.1.1	The communication subsystem shall transmit the status of the payload deployment to the ground control system in real-time
DR 4.1.2	The communication subsystem shall be capable of transmitting environmental data
DR 4.1.3	The communication subsystem shall be capable of detecting communication loss
DR 4.1.4	The communication subsystem shall comply with FAA and CU Flight Op guidelines
DR 4.2.1	The communication subsystem shall be capable of data transmission between the avionics and ground systems during the mission with a range of a 100 meter diameter
DR 4.3.2	The communication subsystem shall update the ground system with telemetry data
DR 4.3.3	The communication subsystem shall update the avionics system with telemetry data

CU Flight Ops and FAA

All requirements regarding CU Flight Ops and FAA requirements were verified through the airworthiness inspection conducted by CU Flight Ops. This included the certification of pilots, the registration of the vehicle with the FAA, and the logging of flights through Alaris Pro.

3 Lessons Learned and Future Work

Authors: Christian Bowman and Ethan Davis

3.1 Lessons Learned

3.1.1 Vibrational Damping and PID tuning

The greatest challenge for the UAS when it came to testing was finding the correct set of gains that the system could fly with that would be reliable for both empty and full payload flights. One major obstacle when tuning for gains was the vibrational issues caused by the propellers and engines when the payload was empty, and the full payload of water when it wasn't empty. If we had used a flight controller system equipped with damping or were aware of this potential issue and used a dampener earlier, it would have saved us valuable flight time to improve accuracy and given us more chances to obtain more data on test flights to validate our system.

3.1.2 Payload Deployment Accuracy

When testing the payload deployment system, we realized that the valve chosen for testing was spraying water in an inconsistent way, which decreased overall accuracy. This is likely due to the solenoid valve's geometry changing the flow of water as it travels through the valve, causing heavier dispersion than desired. Changing the type of valve for deployment can create a more desired stream for accuracy, and adding a more complex pump or pressurized system may be more desired in higher-level real-world applications. Unfortunately, due to time constraints, we did not conduct a trade study on multiple valve systems prior to major assembly, which would have allowed us to tailor the valve for greater accuracy.

In addition to the payload valve, the camera system impacted the accuracy of the system. In early testing, we noticed that the original camera rail system generated camera wobbling, making an accurate deployment almost unobtainable. When the camera was later moved down to the payload tank, the accuracy increased dramatically, but was still not as accurate as desired. Having a targeting crosshair and a better camera system could increase deployment accuracy and allow visual confirmation of target hits without an observer present.

3.2 Future Work

3.2.1 Short-term Improvements

Weatherproofing / Hardening

Currently, the airframe has very little weather protection, such that the flight controller is completely exposed on the top of the airframe. Electronic components should be encased in a waterproof container to avoid the risk of damaging electronics during filling operations and to allow operations during rain or other inclement weather. In addition, deploying a payload over an actual fire would expose the airframe to high temperatures and particulates. Leaving aside the numerous other problems with deployment during an actual fire, the high temperatures and particulates present a high risk of damaging sensitive electronics and wires. Implementing a rigorously validated thermal protection system would require much more data on what exact conditions the aircraft would be exposed to during a deployment over a fire, but isolating the sensitive electronics from the ambient atmosphere to prevent particulate intrusion would be a step in the right direction.

RTK

Replacement of the GPS system with a more expensive RTK GPS system would allow the UAS to be fully autonomous with minimal pilot supervision. The addition of RTK would also allow for more precise payload deployment, taking an accuracy of around 1-2 meters to centimeter accuracy.

Servo Switch Mechanism

In order to have a fully autonomous mission, it would be critical to have the water release mechanism triggered by a servo switch that is programmed in with the mission planner mission scheme. In the future this is possible with a voltage splitter to power the solenoid valve instead of the separate battery, and a servo signal to open the valve.

Improved Sensing

Adding a more complex sensor suite would allow for more hostile environments. The addition of a Forward Looking Infrared (FLIR) system could expand the mission role of the UAS system for scouting potential drop zones, and would allow for more feedback for future firefighting capabilities. Adding a gimbal to the standard camera system would improve visibility and accuracy of deployment.

Terrain Awareness

A realization from modeling experimental data was that the UAS had difficulty maintaining the assigned altitude during flight. A more robust LIDAR system or a terrain scan of the assigned operational area would allow for improved terrain following. The implementation of additional programming for collision avoidance using horizontally mounted LIDAR equipment could expand the system's operational range to include more lightly forested areas, and possibly urban areas. However, additional LIDAR systems would dramatically increase the price of the UAS.

3.2.2 Commercialization

Increased Scale

A commercial system would almost certainly need to be of much larger scale to provide useful firefighting capability. By the time a fire in a remote area is detected, located, and reached by the system, it will likely grow to a size such that hundreds of kilograms of retardant would be necessary for complete containment. Thus, to be useful, a commercial system would likely need to carry more than 100 kg of retardant payload along with the significantly higher airframe mass necessary to carry that payload. Furthermore, a commercial system would almost certainly need to operate at much longer ranges to reach remote areas, necessitating more weight in energy storage, further increasing the total system mass. The exact specifications of a commercial system would vary wildly depending on the functional requirements of that system, but the total airframe mass of a commercial system would likely range between 500 and 5,000 kg.

Hybrid Airframe

A commercial system intended to fight fires in remote areas would probably be required to operate at ranges of 10-30 km, or even higher. Operating in those ranges would result in the commercial system spending significant time and energy cruising between the ground station and the fire. The low speeds and ranges outlined by our functional requirements caused the team to forgo the inclusion of fixed lifting surfaces due to providing negligible efficiency gains at low airspeeds while incurring significantly increased cost and complexity. However, in the longer ranges over which a commercial system would operate, using fixed airfoils to generate lift in cruise could easily improve aerodynamic efficiency enough to justify the increased mass and complexity. Furthermore, increased aerodynamic efficiency could reduce energy consumption so that energy storage mass could be reduced, resulting in an overall reduction of system mass and cost. Developing a commercial system would warrant investigating at what ranges and flight profiles the efficiency benefits of a hybrid airframe outweigh the simplicity and reduced weight of a pure multi-rotor.

Pressurized Payload Deployment

A large-scale commercial system deploying a payload directly above a fire per our system poses numerous problems. Some include impaired visibility due to smoke, damage to the airframe due to particulates and high temperatures, and increased oxygen flow to the fire via rotor wash. One solution to these problems would be to pressure feed the payload into a nozzle and fire the payload at the fire from a significant distance, similar to a fire truck. Not only would doing so solve the above problems, but it also creates the opportunity to aim the payload with very high accuracy, such that firefighting effectiveness could increase significantly while maintaining the same payload mass. Such a system would incur the additional cost and complexity of incorporating the nozzle, pumps, and target system, which is why this project did not utilize a pressure-fed system. However, on a commercial scale, the potential benefits more than warrant investigating the feasibility of a more complex deployment system.

4 Individual Report Contributions

Christian Bowman: In the report, I worked on the Lessons Learned section with Ethan, placing particular focus on the "Future Work" section. During the semester I worked primarily on fabricating and assembling the Payload Deployment system. I tested various adhesives which were used to assemble the payload deployment system, worked with Alex to finalize the design of the payload tank, was responsible for fabricating the tank, attached all components to the payload tank, and mounted it to the airframe. I also participated in multiple flight tests as a visual observer, briefly piloted the drone twice, and presented on the payload deployment system during the final presentation.

Maximillian Brown: In the final report I contributed by creating all the dynamics comparison plots, adding the thrust test thrust and current plots, and analyzing the thrust test results. Throughout the semester, I presented on Safety and Software IDRs, TRR and SFR. I was a primary contributor to the static thrust test, being present at every test for the setup, testing, and breakdown. I helped create the safety procedures and checklists according to Flight Ops' advice. As a pilot, I was present at most flight tests, from tuning to autonomous flights. I was the pilot for our final full mission test. I worked heavily with MATLAB and our model to improve the model, analyze flight data and create useful resources.

Ethan Davis: For the final report, I worked on the lessons learned and future work section with Christian Bowman and assisted with editing for the section. During the semester, I worked as part of the electrical team. I participated in the electronics build-up with the rest of the electronics team. I also worked on the static thrust test with Maximillian Brown and Jack Pearse as a safety observer. I continued this work by acting as both a safety observer and one of the primary reload team members for all active drone testing.

Donovan Gavito: For the Final Report, I worked on many different sections. I mainly worked in the project purpose section. I also went through and edited many sections for consistency and proper grammar. During the semester I worked between the sub-teams to assemble and program the hexacopter. I was also the main pilot for the hexacopter, so I was at all but 1 flight test. I also made safety procedures and flight checklists to ensure safety during our flight tests.

Joshua Geeting: In the Project Final Report, I worked directly on Verification and Validation for the Single Leg Mission Test. Throughout the semester, I was directly involved in the physical assembly of the drone and actively participated in approximately 90% of the flight testing. This hands-on experience was essential for verifying our Single Leg Mission requirements and provided me with a comprehensive understanding of the system's challenges. As a result, I was able to contribute meaningfully to the Verification and Validation section of the final report.

Drew Kane: Within the Final Report, the majority of my contributions were regarding the static Thrust Evaluation Test. I also added minor contributions within Section 1 and 2, as well as checking for grammatical errors and edits to sentence structure throughout the report. During the semester, I helped facilitate and contributed to multiple aspects of the project in some form or another. This work, through the help of the entire team, resulted in the completion of all major milestones being completed before the end of the testing phase. This allowed ample time for results and data to be collected, considered, and compared to requirements and modeled data.

Ian McCarty: For the PFR, I helped with the project purpose section. I also helped edit for length and readability. To be more specific, I helped translate large chunks of information/writing from the fall final report to be consistent with our progress made during the last semester. This mainly included the project purpose, design, and concept of operations subsections. In terms of the content within the report, I was present at many flight tests to help ensure our UAS hardware was working as intended, and to be the designated driver in case of any emergency.

Braden Nelson: For the PFR, I did the test results and comparison to modeling sections for the mass flow and dynamics models. During the semester, I worked on the safety TRR and created a plan with the team to ensure safety during flight. I also created the mass flow models as well as edited the dynamics model to better match the flight tests. I was present during a few flight tests to oversee the drone's success and ensure safety protocols discussed in TRR.

Jack Pearse: For the Final Report I worked on the project purpose section. We all worked on the editing throughout the process. During the semester, I was the systems engineer for team FLAME. My primary goal was to make sure that all of our functional requirements were verified and validated in some capacity. During the semester I also ran the static thrust tests with Max at Design Build Fly. I was also the fire department contact for our team and was a visual observer during the single mission leg.

Alex Putnam: The majority of my contribution to the Final Report was in the Design section. I wrote about the payload container, its design process, and the other miscellaneous CAD models. I also wrote the section about the final payload release mechanism and the decision-making process behind it. Additionally, I helped edit the entire paper for grammar and spelling. During the semester, I was present as a visual observer on many days of testing, and I was the designated fire-safety officer on those days. I was also responsible for a large portion of the airframe's assembly, and I had a minor role in the Electrical/Software IDR.

Brady Sivey: For the PFR, I was one of the primary authors of the Thrust Test Section of the report. I also spent time going through the paper looking for proper grammar, terminology, and content. Throughout the semester, I was the CFO for this project, so I was responsible for planning and maintaining of our budget as well as making all purchases for the team. As the Flight Operations Coordinator, I was one of the primary authors of our flight test procedure and supported with many of our preliminary flight tests and tuning. Finally, I was one of the main contributors to the content used in our safety IDR including diagrams, slides, and text.

Jared Steffen: I was the primary author for the energy analysis and energy model comparison sections of the report. I also read the entire paper and edited it for consistency. For design and development, I determined vehicle parameters in Mission Planner, including PID tuning methods and the need for a vibration damping mount. I developed MATLAB scripts to process flight data, analyze energy consumption, and validate CDR model assumptions. Additionally, I contributed to the thrust evaluation test by setting up an automated test script for controlled throttle testing and processed the data to compare against the manufacturer's datasheet.

5 Individual Project Goals and Self-Assessment

Table 9: Team Roles and Goals

Team Name:		
Student Name	**Team Role(s)**	**Original Goals (100 words or less) and Self Assessment (100 words or less)**
Christian Bowman	Hardware Team Lead Visual Observer	**Original Goals:** Since I'm now the hardware lead, my hope is that I can successfully lead the hardware team, and effectively apply my previous experience in 3D printing while doing so. I'm also hoping to assist with flight operations and safety if/when I'm not focusing on hardware. I will probably end up working with others on system integration later in the semester. **Self Assessment:** Overall I feel satisfied in achieving my goals and with how the semester went in general. I think I successfully did exactly what I set out to do with designing and integrating the payload deployment system. I was not particularly involved with flight operations, but I was a visual observer for a few flight tests. I wish to a small degree I could have been more involved with flight testing, but am also not particularly perturbed by how things turned out.
Maximillian Brown	Software Team Member Pilot	**Original Goals:** This semester I will help setup and execute the software side of the hexacopter testing and operations. To this end, I will help with the Design, Build, Fly thrust test stand test, aid in the firmware setup and execution of mission planner to control the drone, and be an active participant for most critical tests of the drone. As a pilot, I will be an active participant in the critical aspects of testing the drone, including passing the ground school tests and helping the team command the drone safely while in flight both autonomously and manually. **Self Assessment:** I feel very satisfied with this semester's work. I achieved my goals, helping with some parameters of the drone's software, and being a lead member for the static thrust test execution and analysis. I passed my piloting tests, helping fly the drone safely manually and autonomously. I was present for nearly every single flight test, being either a crew member to aid in the test's execution or the primary pilot. I contributed a lot through my code in MATLAB, aiding in creating plots and analyzing data. I also presented a lot, being a professional and vocal member of the team.

Student Name	Team Role(s)	Original Goals (100 words or less) and Self Assessment (100 words or less)
Ethan Davis	Electronics Team Member	**Original Goals:** This semester I will work with establishing a safe testing environment and project with safety team members and safety lead Donovan Gavito. I will also continue to work with the electronics team to help assembly of the drone. I will also understand test and inspection procedures to make sure the aircraft is flight and mission ready, and safe for operation. I will likely assist other teams and personnel when needed for reports, safety, hardware, and software. **Self Assessment:** I believe I fulfilled these goals. I oversaw the drafting of our safety procedures, as well as worked on the UAS assembly although not as much as I would have liked. I also tried to help other teams when possible. I wish there was more time within the project to improve the system more or improve/learn more new skills. I regret not being able to participate in more of the software team's work. However, I am proud of what the team and it's individual members accomplished.
Donovan Gavito	Hardware Team Member Lead Pilot	**Original Goals:** This semester I will continue to apply my knowledge of building a drone to create a functional hexacopter that executes our mission. I also plan to successfully pilot the hexacopter in the "human in the loop" portion of our mission, working closely with CU Flight Ops and FAA guidelines to ensure a safe flight, mitigating risk of our crew and vehicle. These goals will be considered successful if our hexacopter verifies our mission requirements and safety procedures are met, even if there is a failure. **Self Assessment:** Starting the semester as the hardware lead, I took the lead in assembly of the hexacopter. I soldered the motors and escs and mounted them to the frame. I also installed the communication components on the frame. I also worked closely with Jared to configure the parameters on our flight controller and conduct successful test flights. I also created safety procedures and checklists to ensure safety during testing. I also successfully piloted our drone countless times, saving it at certain times when it was going to crash due to vibration.

Student Name	Team Role(s)	Original Goals (100 words or less) and Self Assessment (100 words or less)
Joshua Geeting	Electrical Team Lead	**Original Goals:** This semester, as electrical lead, I will oversee the testing of each electrical device separately to ensure it operates properly. I will work directly with Jared to test the communication devices between the software and the drone, then with Alex to test the other electrical components. After testing is complete, I will collaborate with Christian on the placement of each part within the airframe. This includes wiring, control, and waterproofing vital connectors and components. In addition, I will lead the electronics efforts related to soldering, wiring, and communication to ensure the project's success. **Self Assessment:** As the Electrical Team Lead, I took the lead in diagnosing and resolving electrical issues encountered during both development and testing phases. I collaborated closely with Jared to validate communication protocols and ensure proper integration of electrical components during system assembly. In parallel, I worked with Donovan to complete component installation, which included precision soldering, wire harnessing, and battery management to ensure safe and reliable power distribution. Throughout the testing phase, I applied hands-on problem-solving strategies to address critical system issues, including vibration mitigation, improving camera feed quality, sealing payload compartments to prevent leakage, and reinforcing structural integrity.

Student Name	Team Role(s)	Original Goals (100 words or less) and Self Assessment (100 words or less)
Drew Kane	Project Manager	**Original Goals:** My goal this semester is to oversee and facilitate the successful fabrication, integration, and testing of all systems critical to mission success. I will achieve this goal by working closely with each individual team member, our team leads, and our SE, Jack. A main focus of mine will be making sure every one of our members achieves each of their personal goals while ensuring mission success by dividing the necessary work equally between team members based on their personal skills and goals. I will follow up with each teammate through personal communication and weekly check-ups, making sure they are satisfied both with the work they are doing, as well as getting feedback on what they would like to see more from their leadership. **Self Assessment:** Overall the mission objective was satisfactorily fulfilled due to successful fabrication, integration, and testing from both myself and the team. I'm beyond happy with our results as a team, and am satisfied with my leadership and direction duties. I followed up on personal connection and weekly check-ups, but those weekly check-ups became biweekly check-ups at some point. I definitely could have done more for more consistent check ins regarding personal goals and work satisfaction. That is not to say that I am disappointed with my work, but that there is always room for improvement. I am nonetheless extremely proud of the engineering work our team completed this semester.

Student Name	Team Role(s)	Original Goals (100 words or less) and Self Assessment (100 words or less)
Ian McCarty	Hardware Team Member	**Original Goals:** As we move into the build/testing phase of our project, I hope to ensure the mechanical components of our airframe, especially our propulsion system, can fully validate our mission objectives. To do this, I will work with the hardware and electrical team to make sure that our UAS is conforming to our model and delivering our payload successfully. This goal will also include troubleshooting any physical problems as we build up our drone and continue testing. If needed, I will also work with Christian on 3D printing various components and learning more about that process. **Self Assessment:** Overall, I do not think I fully accomplished my original goals. While involved in the hardware component side of the project, I did not reach the level of engagement I initially expected. However, I still think I learned many valuable lessons and contributed in other meaningful ways. I experienced the complexity associated with working in a large team with many different tasks, and how to overcome obstacles as a group. I also enjoyed being exposed to technical reviews and learning the best ways to approach these presentations. Working with other team members to ensure we presented our project effectively was a challenging yet rewarding task.

Student Name	Team Role(s)	Original Goals (100 words or less) and Self Assessment (100 words or less)
Braden Nelson	Software Team Member	**Original Goals:** As a software team member, I would love to validate the model I created last semester. I want to continue to build on the model after we do some preliminary tests in order to refine it as best we can. I also want to get familiar with Ardupilot and understand the actual flight controls a bit more. I will be tuning gains for the ardupilot and I want to ensure that the drone is as safe as possible, even if the payload were to fall off the drone unexpectedly. **Self Assessment:** This semester I accomplished most of what I set out to do. I was the primary member working on validating the dynamics model I created. I unfortunately learned that some of my assumptions were not valid, but I properly showcased the results in SFR and PFR. I also was able to learn a bit about Ardupilot, but I did not get into the gain tuning as much as I wanted. My time during the test period was mostly focused on the payload deployment model required for the stability assessment. Even though I did not ensure drone safety through much gain tuning, I did so through modeling the payload deployment and integrating that into the dynamics model.
Jack Pearse	Systems Engineer	**Original Goals:** As systems engineer my main goal is to verify and validate all of our requirements through their verification methods. This will be from the SRR requirements table and will be comparing our tests and validation to that. Another goal I have for this semester is to make sure all of our subsystems and systems interface well together. Obviously, the overall goal of me and the team is to make sure that we can complete our mission objectives. **Self Assessment:** For the spring semester, I was in charge of making sure that our high level functional requirements were being met during testing. Our systems interfaced very well together since there was plenty of documentation of our parts. I helped with all of our internal design reviews and presented at every one of our Reviews. Most of the requirements were verified perfectly, and the system worked well with itself so overall I would say that I met my goals for the semester.

Student Name	Team Role(s)	Original Goals (100 words or less) and Self Assessment (100 words or less)
Alex Putnam	Electrical Team Member	**Original Goals:** This semester, I will continue to work towards the drone's assembly and the installation of its electrical systems. I hope to be heavily involved in the wiring and the interconnection of the sensors and systems. During the assembly process, I will likely be working alongside Josh and Jared. After the drone is completed, I have been assigned the role of handling fire safety, so I will likely be present at many of the flight tests, standing by in case of failure. **Self Assessment:** I believe that I partially fulfilled my goals, but many of my contributions ended up being in different areas than I originally planned. I did spend a lot of time assembling the drone, but not with a focus on the electronics as I had thought. I also ended up spending more time than was expected on the payload, as it required many more revisions after what was thought to be a near-final design. I was at many of the flight tests, as planned, and I was constantly on alert, with my fire-safety equipment within arms reach.
Brady Sivey	CFO Software Team Member	**Original Goals:** This Spring, I will be working as both the CFO and a Software Team Member. As CFO, I will be in charge of maintaining the budget, ordering necessary items, and planning for future orders. I have already ordered most items necessary for our project, but I may be needed to order additional items due to potential manufacturing and testing implications. As a software team member, I plan on becoming familiar with the ArduPilot/Mission Planner Software in order to be able to successfully tune our flight software. This will help ensure safe flight and proper validation of our models. **Self Assessment:** I feel that I definitely met my goals as CFO. I feel proud of the work I did in terms of planning, maintaining, and spending our budget. I am also thankful for the team with supporting myself with all of that. As far as my technical contributions, I was kind of all over the place, supporting wherever I was needed, so I wasn't necessarily only helping with the software team like I thought I would be. I did get familiar with Ardupilot, but I was relatively limited with it because my computer was unable to download the software.

Student Name	Team Role(s)	Original Goals (100 words or less) and Self Assessment (100 words or less)
Jared Steffen	Software Team Lead	**Original Goals:** For the spring semester of Senior Design I will be spearheading the integration of ArduPilot/Mission Planner with our design. This includes overseeing the tuning of our vehicle, planning the mission within Mission Planner, setting up methods for data logging/transmission to verify requirements or validate CDR models. I will also participate in the testing of our motor with Design-Build-Fly's thrust test stand. This effort will be pivotal in validating our CDR energy consumption model. These goals will be successful if the requirements DR 5.1.1 through DR 5.1.8, DR 2.1.1, and DR 3.1.1 through DR 3.2.1 are fulfilled. I will work closely with Jack (SE), Drew (PM), other software sub-team members, and our pilots to ensure the completion of these goals. **Self Assessment:** For the spring semester, I was the primary team member that was responsible for setting vehicle parameters, determining and implementing methods of tuning through QuikTune and AutoTune features. I set up all of our missions and ran the ground control station PC monitoring vehicle status through flight. I was at the final test for the thrust evaluation and made scripts to read in and graph data. All of the requirements mentioned in the"original goals" section have been successfully verified.